INVITED PAPER *Special Section on Information and Communication System Security*

Cryptanalysis for RC4 and Breaking WEP/WPA-TKIP

Masakatu MORII[†a)], ***Senior Member and*** **Yosuke TODO**[†], ***Student Member***

SUMMARY In recent years, wireless LAN systems are widely used in campuses, offices, homes and so on. It is important to discuss the security aspect of wireless LAN networks in order to protect data confidentiality and integrity. The IEEE Standards Association formulated some security protocols, for example, Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access Temporal Key Integrity Protocol (WPA-TKIP). However, these protocols have vulnerability for secure communication. In 2008, we proposed an efffective key recovery attack against WEP and it is called the TeAM-OK attack. In this paper, first, we present a different interpretation and the relation between other attacks and the TeAM-OK attack against WEP. Second, we present some existing attacks against WPA-TKIP and these attacks are not executable in a realistic environment. Then we propose an attack that is executable in a realistic environment against WPA-TKIP. This attack exploits the vulnerability implementation in the QoS packet processing feature of IEEE 802.11e. The receiver receives a falsification packet constructed as part of attack regardless of the setting of IEEE 802.11e. This vulnerability removes the attacker's condition that access points support IEEE 802.11e. We confirm that almost all wireless LAN implementations have this vulnerability. Therefore, almost all WPA-TKIP implementations cannot protect a system against the falsification attack in a realistic environment.

key words: *wireless LAN network, RC4, WEP, WPA-TKIP, cryptoanalysis*

1. Introduction

Nowadays wireless LAN systems are widely used in campuses, offices, homes and so on. It is important that we have the secure communication, especially protecting transmission data and authentication. Wired Equivalent Privacy (WEP) [1] and Wi-Fi Protected Access Temporal Key Integrity Protocol (WPA-TKIP) [2] are security protocols that protect data confidentiality and integrity in IEEE 802.11 wireless standard. It is widely known that these protocols have vulnerability for secure communication. The first key recovery attack against WEP has been proposed in 2001 by Fluhrer, Mantin, and Shamir (called the FMS attack [3]). The FMS attack can know partial information on secret key from the first byte of the keystream when a specific initialization vector (weak IV) is used. We can read it is a chosen plain text attack. The FMS attack has been extended to use more weak IV classes by several researchers [4]–[7]. Tews, Weinmann, and Pyshkin have proposed a key recovery attack against WEP without using any weak IV in 2007 (called the PTW attack [8]). Their attack is a known plain text attack. However, if the PTW attack recover a 104-bit secret key only by observing encryption packets, it takes long time to collect the packets. The use of active attacks may collect packets faster. But such an attack is detected by Intrusion Detection System (IDS). In 2008, Teramura, Asakura, Ohigashi, Kuwakado and Morii have proposed a known plain text attack against WEP (called the TeAM-OK attack [9]), with collecting a small quantity of any IP packets. Since their attack is passive and much effective for the key recovery, it has been made known that WEP has serious vulnerability.

WPA-TKIP was introduced in order to prevent the vulnerability of WEP. Several researchers have deliberated about the security aspects of WPA-TKIP. However, thus far, no realistic attack against WPA-TKIP, except for the dictionary attack, had been known. In 2008, Beck and Tews proposed a falsification attack (called the Beck-Tews attack [10]) on WPA-TKIP. Their attack succeeds only in the case of networks that support IEEE 802.11e features. Thereafter, Ohigashi and Morii proposed a falsification attack (called the Ohigashi-Morii attack [11]) that is based on the man-in-the-middle attack against WPA-TKIP. This attack expands its targets to other products that do not support IEEE 802.11e. However, it is necessary to interrupt communication between an access point (AP) and a client for executing the man-in-the-middle attack. Hence, it is not easy to execute the Ohigashi-Morii attack in a realistic environment.

In this paper we present a different interpretation and relation between other attacks and the TeAM-OK attack against WEP. Furthermore we propose an attack (called the QoS forgery attack [12]) that is executable in a realistic environment and which is not based on the man-in-the-middle attack. This attack exploits the vulnerability implementation in the QoS packet processing feature of IEEE 802.11e. The receiver receives a falsification packet constructed as part of attack regardless of the setting of IEEE 802.11e. This vulnerability removes the condition that APs support IEEE 802.11e. We confirm that almost all wireless LAN implementations have this vulnerability. Therefore, almost all WPA-TKIP implementations cannot protect a system against the falsification attack in a realistic environment.

This paper is organized as follows. RC4 which is the cipher used in WEP/WPA-TKIP is discussed in Sect. 2. In Sect. 3, the TeAM-OK attack and the relation between other attacks against WEP are presented. Furthermore Sect. 4 gives the QoS forgery attack for WPA-TKIP. Finally, we conclude this paper in Sect. 5.

Manuscript received July 20, 2011.

[†]The authors are with the Graduate School of Engineering, Kobe University, Kobe-shi, 657–8501 Japan.

a) E-mail: mmorii@kobe-u.ac.jp

DOI: 10.1587/transinf.E94.D.2087

2. RC4 and Wireless LAN Security

In this section, we describe a stream cipher RC4. Next, we describe wireless LAN security protocols, Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access-Temporal Key Integrity Protocol (WPA-TKIP). These protocols use a stream cipher RC4.

2.1 RC4

The RC4 is a stream cipher designed by R. Rivest in 1987. This cipher is used in many protocols; for example, SSL, WEP and WPA-TKIP. The RC4 consists of two algorithms: the key-scheduling algorithm (KSA) and the pseudo-random generation algorithm (PRGA). The KSA initializes an internal state from a secret key K. Next, the PRGA generates the keystream. RC4's internal state consists of a 256 bytes permutation array S and two indices i and j. Figure 1 shows the RC4 algorithm.

2.2 Wired Equivalent Privacy (WEP)

WEP is a security protocol for IEEE 802.11, and it uses the stream cipher RC4 for encryption. First, WEP generates a packet key K as follow:

$$K = IV \,||\, R_k, \tag{1}$$

where IV is a 24-bit initialization vector, R_k is a fixed secret key and || is concatenation. WEP uses a different packet key by changing IV for each packet. Second, WEP generates a keystream $Z = (Z_1, Z_2, \ldots, Z_L)$ from a packet key K and RC4; here, Z_i is one-byte variable and L is a length of a plaintext. The keystream is XOR-ed with a plaintext $P = (P_1, P_2, \ldots, P_L)$ to obtain a ciphertext $C = (C_1, C_2, \ldots, C_L)$ as follows:

$$C_i = P_i \oplus Z_i \quad (i = 1, 2, \ldots, L), \tag{2}$$

RC4 KSA:
1: for $\forall\, x \in \{0, 1, \cdots, \text{N-1}\}$
2: $S[x] \leftarrow x$
3: end for
4: j $\leftarrow$ 0
5: for $\forall\, i \in \{0, 1, \cdots, \text{N-1}\}$
6: $j \leftarrow j + S[i] + K[i \bmod 16] \bmod 256$
7: Swap($S[i]$,$S[j]$)
8: end for

RC4 PRGA:
1: i $\leftarrow$ 0
2: j $\leftarrow$ 0
3: loop
4: i $\leftarrow$ i + 1
5: j $\leftarrow$ j + S[i] mod 256
6: Swap($S[i]$,$S[j]$)
7: Output z $\leftarrow$ S[($S[i]$+$S[j]$)]
8: end loop

Fig. 1 The RC4 algorithm.

where both C_i and P_i are one-byte variable. We show the description of WEP in Fig. 2.

The WEP was expected to provide data confidentiality comparable to that of a wired network when this protocol was standardized initially. However, several researchers proposed fatal vulnerability of WEP. Now, WEP is not secure, but many people are still using this protocol.

2.3 Wi-Fi Protected Access-Temporal Key Integrity Protocol (WPA-TKIP)

After the vulnerability of WEP was reported, the IEEE Standards Association formulated a new security protocol WPA-TKIP. This protocol has an integrity check function by TKIP, and uses the stream cipher RC4 for encryption but can prevent many attacks against WEP.

In WPA-TKIP, a 512-bit master key is shared between an AP and a client. This master key generates a 64-bit MIC key K^* and a 128-bit encryption key K. The MIC key K^* is used to generate a MIC, and the encryption key K is used to encrypt packets.

The processing for the sender

First we describe the processing for a sender. The sender generates a MIC from the MIC key and a MAC Service Data Unit (MSDU) by using a message integrity check function MICHAEL. The MIC is added to the MSDU as follows:

$$MSDU||michael(K^*, MSDU),$$

where $michael(K^*, MSDU)$ is a 64-bit MIC and || denotes concatenation. The MSDU with the MIC is fragmented into MAC Protocol Data Units (MPDUs). A 32-bit checksum is calculated from each MPDU by using CRC32 and is added to the MPDU as follows:

$$MPDU||CRC32(MPDU),$$

where $CRC32(MPDU)$ is the 32-bit checksum.

Encryption of WPA is executed for each MPDU with

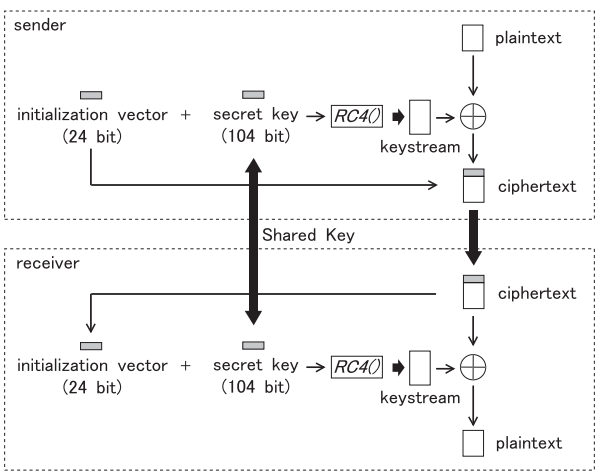

Fig. 2 The description of WEP.

the checksum. A packet key PK is generated from a 48-bit IV, an encryption key K, and a MAC address by using a specific hash function for WPA, $hash()$. Each MPDU has a different IV and the value of the IV is incremented by 1 each time a new IV is generated. In WPA-TKIP, the IV is called the TKIP sequence counter (TSC). The sender generates keystream from a packet key PK by using RC4 and encrypt MPDU with the checksum as well as WEP.

The processing for the receiver

Next we describe the processing for a receiver. The receiver receives an encrypted MPDU and an IV, and the received IV is compared with the TSC counter which is an IV value corresponding to the encrypted MPDU accepted most recently. If the received IV is less than or equal to the TSC counter, the received encrypted MPDU is discarded. This method is effective against the replay attack which misapplies the packet used before. We call this method by which the packet is discarded the *TKIP-IV check*.

In the decryption of WPA, the receiver decrypts the MPDU with the checksum by using a shared packet key PK. The receiver calculates a checksum from the received MPDU, and this calculated checksum is compared with the received checksum. If their values are different, the received MPDU is discarded. We call this method by which the packet is discarded the *checksum check*.

When all MPDUs are obtained, they are reassembled to form the MSDU. The receiver calculates a MIC from the received MSDU and the MIC key by using the function Michael, and then, this calculated MIC is compared with the received MIC. If their values are different, all the received MPDUs corresponding to the MSDU are discarded and the receiver sends an error message of MIC (MIC failure report frame) to the sender. In WPA, the MIC key is changed if more than two MIC failure report frames are sent to the sender in less than 1 min. When the MSDU is accepted, the TSC counter is updated to the largest value of the IVs corresponding to all the MPDUs. We call this method by which the packet is discarded the *MIC check*.

3. Attacks against WEP

In this section, we describe various key recovering attacks against WEP. Moreover, we consider a technique for preventing the abovementioned attacks.

3.1 Attacks by Using Weak IVs

In 2001, Fluhrer et al. proposed a key recovering attack against WEP, and we call this attack the FMS attack. If WEP uses specific IVs, an attacker can know partial information on secret key from the first byte of the keystream. We call these IVs weak IVs. This attack can recover a 104-bit secret key by observing about 4,000,000 to 6,000,000 encryption packets. As a result, original WEP was broken by this attack. However, if WEP removes weak IVs, the improved WEP can prevent the FMS attack.

After the FMS attack was proposed, many attacks against WEP by using weak IVs were proposed; for example the KoreK attack. This attack expands a concept of weak IVs. Weak IVs of the FMS attack do not depend on values of the secret key, however weak IVs of the KoreK attack depend on values of the secret key. This attack can recover a 104-bit secret key by observing about 500,000 to 1,000,000 encryption packets.

Quantity of weak IVs which abovementioned attacks use exists only a little. Then if WEP removes weak IVs cautiously, the improved WEP can prevent the FMS attack and the KoreK attack. However, it is more difficult to prevent the KoreK attack than to prevent the FMS attack, because weak IVs of the KoreK attack depend on values of the secret key. Namely, we have to change the method to prevent the KoreK attack for each secret key.

In 2005 Ohigashi et al. proposed an attack against WEP by using new weak IVs. Information that this attack obtains in using a weak IV is less than that of the FMS attack or the KoreK attack. However, quantity of weak IVs which this attack can use is numerous, and this attack can consider almost all IVs weak IVs. Namely, WEP can not remove weak IVs and prevent this attack.

3.2 The Klein's Attack

In 2006, Klein proposed an improved way of attacking WEP using related keys that does not need any weak IV [13]. We call this attack the Klein's attack, and this attack can recover a secret key by fewer packets.

First, from the step 6 of the KSA in Fig. 1,

$$K[i \bmod 16] = j_{i+1} - j_i - S_i[i]. \tag{3}$$

Now if an attacker wants to know $K[3]$, one has to know $K[0]$, $K[1]$ and $K[2]$. However, in WEP, $K[0]$, $K[1]$ and $K[2]$ are IV and they are known. Then, j_3 and $S_3[3]$ are known with a probability 1. Therefore, if the attacker can know j_4, one can know a key byte $K[3]$. Similarly if the attacker wants to know a key byte $K[i]$, one has to know $K[0], K[1], \ldots, K[i-1]$. Then, the attacker can know j_i and $S_i[i]$ with a probability 1. Therefore, if the attacker can know j_{i+1}, one can know a key byte $K[i]$. And from the step 7 of the KSA in Fig. 1,

$$S_{i+1}[i] = S_i[j_{i+1}]. \tag{4}$$

Klein showed that the value of $S_{i+1}[i]$ in the KSA is not changed to $S^*_{i-1}[i]$ in the PRGA with a probability about $1/e$, where S^*_i be the i-th step of the PRGA where the internal state consists of a random array. Then, if the attacker can know $S^*_{i-1}[i]$, one can recover a key byte $K[i]$ with a probability about $1/e$.

Next, Klein showed a method of knowing $S^*_{i-1}[i]$. Let c be an integer in $\{0, 1, \ldots, N-1\}$. Then, the probability $P[\left(Z_i + S^*_{i-1}[i] \bmod N\right) = c]$ is given as

$$P[\left(Z_i + S^*_{i-1}[i] \bmod N\right) = c] = \begin{cases} \frac{2}{N} & \text{if } c = i, \\ \frac{N-2}{N(N-1)} & \text{if } c \neq i. \end{cases} \tag{5}$$

From Eq. (5), we derive

$$S^*_{i-1}[i] = i - Z_i \text{ (with a probability } 2/N). \quad (6)$$

By summarizing the Eqs. (3)–(6), the Klein's attack is derived following function,

$$K[x] = f_{Klein}(K[0], \ldots, K[x-1], Z_x), \quad (7)$$

which holds with a probability P_{Klein} described as

$$P_{Klein} \approx \frac{1}{e} \cdot \frac{2}{N} + \left(1 - \frac{1}{e}\right) \cdot \frac{N-2}{N(N-1)} \approx \frac{1.36}{N}. \quad (8)$$

Klein has remarked that this attack can recover a key byte $K[x]$ to apply this function to keystreams iteratively. This iterative approach has a significant disadvantage on the correction of falsely guessed key byte. If a secret key byte has not been correctly recovered, the whole key will be probably mis-recovered due to the key byte dependency. Thus, the recovery of all key bytes following the incorrect key byte must be repeated. The key byte dependency of the Klein's attack degrades the performance of this attack.

3.3 The PTW Attack

Tews et al. have pointed out the fault of the Klein's attack, and proposed an attack that overcome this fault. We call this attack the PTW attack, and this attack independently recovers the sum of the secret key bytes.

First, from the step 6 of the KSA in Fig. 1,

$$j_{i+1} = j_i + S_i[i] + K[i \bmod 16].$$

However, an attacker can not know key bytes excluding $IV(K[0], K[1], K[2])$. Then, one uses a following approximation,

$$j_{i+1} = j_3 + \sum_{l=3}^{i} S_3[l] + \sigma_i, \quad (9)$$

where $\sigma_i = \sum_{l=3}^{i} K[l \bmod 16]$. And one uses a following approximation,

$$S_{i+1}[i] = S_3[j_{i+1}]. \quad (10)$$

By summarizing the Eqs. (9)–(10), (5)–(6), the PTW attack is derived a following function,

$$\sigma_x = f_{PTW}(K[0], K[1], K[2], Z_x), \quad (11)$$

which holds with a probability P_{PTW} described as

$$P_{PTW} \approx q_x \frac{1}{e} \cdot \frac{2}{N} + \left(1 - q_x \cdot \frac{1}{e}\right) \cdot \frac{N-2}{N(N-1)}, \quad (12)$$

where

$$q_x = \left(1 - \frac{1}{N}\right)^{x-3} \cdot \left(1 - \frac{x-3}{N}\right) \cdot \prod_{k=1}^{x-3} \left(1 - \frac{k}{N}\right). \quad (13)$$

The PTW attack can independently recover each sum of the key bytes because only IVs are required. However the success probability decreases even if guessed previous key bytes are correct. The PTW attack can recover a 104-bit secret key by observing about 40,000 encryption ARP packets.

If the PTW attack is executed by using ARP packets, the attacker have to execute the reinjection attack against the target AP. However if the PTW attack is executed by using IP packets, it takes much time to obtain enough IP packets for the PTW attack only by the observation of radio channel. So, the collection of packets is the most time consuming part of the attack process.

3.4 The TeAM-OK Attack

In 2008, Teramura et al. proposed a new attack against WEP, and we call this attack the TeAM-OK attack. This attack can recover a secret key by observing encryption IP packets, so an attacker can get the secret key without being discovered.

First, the TeAM-OK attack recovers σ_{15} with a high probability as

$$\sigma_{15} = f_{PTW}(IV, Z_{15}), \quad (14)$$
$$\sigma_{15} = f_{PTW}(IV, Z_{16}) - IV[0], \quad (15)$$
$$\sigma_{15} = f_{PTW}(IV, Z_{17}) - IV[0] - IV[1], \quad (16)$$
$$\sigma_{15} = f_{PTW}(IV, Z_{18}) - IV[0] - IV[1] - IV[2]. \quad (17)$$

Abovementioned process increases the votes cast of σ_{15}.

Next, the TeAM-OK attack uses the Klein function not the PTW function, and creates the votes table. Moreover, the TeAM-OK attack uses the OKM function as

$$K[x-16] = f_{OKM}(K[0], \ldots, K[x-17], \sigma_{15}, Z_x) \quad (18)$$
$$(x \geq 19).$$

The OKM function uses the keystream bytes $(Z_{19}, \ldots, Z_{30})$. This function can recover a key byte $K[i]$ with a higher probability than the PTW function under the condition that σ_{15} has been correctly recovered.

Finally, Teramura et al. proposed a method of executing this attack by using encryption IP packets. The Internet Protocol (IP) is one of the widely-used network protocol to transmit information, and the traffic is mainly based on IP version 4 (IPv4). Figure 3 shows the first 30 bytes of an 802.11 frame containing an IPv4 packet and the attack function applied to the byte. We suppose the following three situations.

Situation 1: Only fixed values and the "0xXX" values in Fig. 3 can be guessed.

Situation 2: In addition to the situation 1, the "0xYY" values in Fig. 3 can be guessed.

Situation 3: All values except the "0x??" values in Fig. 3 can be guessed.

In a real environment, the situation 1 is usually applicable. If a network uses the TCP protocol and a private address, then the situation 2 is applicable. The environment

			An attack function applied to the byte
1	0xAA	LLC/SNAP header	———
	0xAA		———
	0x03		$K[3] = f_{Klein}(K[0],K[1],K[2],Z_3)$
	0x00		$K[4] = f_{Klein}(K[0],K[1],K[2],K[3],Z_4)$
	0x00		⋮
	0x00		⋮
	0x08		⋮
	0x00		⋮
	0x45	IP version etc.	⋮
	0x00	Type of Service	⋮
	0xXX	Total Length	⋮
	0xXX		$K[12] = f_{Klein}(K[0],K[1],...,K[11],Z_{12})$
	0x??	Identification	
	0x??		
15	0x40	Flags and Fragment Offset	$\sigma_{15} = f_{PTW}(K[0],K[1],K[2],Z_{15})$
	0x00		$\sigma_{16} = f_{PTW}(K[0],K[1],K[2],Z_{16})$
	0xZZ	Time to Live	$\sigma_{17} = f_{PTW}(K[0],K[1],K[2],Z_{17})$
	0xYY	IP version etc.	$\sigma_{18} = f_{PTW}(K[0],K[1],K[2],Z_{18})$
	0x??	Header Checksum	
	0x??		
	0xYY	Source IP Address	$K[5] = f_{OKM}(K[0],...,K[4],\sigma_{15},Z_{21})$
	0xYY		$K[6] = f_{OKM}(K[0],...,K[5],\sigma_{15},Z_{22})$
	0xZZ		⋮
	0xZZ		⋮
	0xYY	Destination IP Address	⋮
	0xYY		⋮
	0xZZ		⋮
	0xZZ		⋮
	0xZZ	Source Port Number	⋮
30	0xZZ		$K[14] = f_{OKM}(K[0],...,K[13],\sigma_{15},Z_{30})$

Fig. 3 Fast 30 bytes of a 802.11 frame containing an IPv4 packet and an attack function applied to the byte.

that the situation 3 is applicable is a special case, and is far from a common state. However, the existing attacks against WEP is executed against the situation 3.

As described in Sect. 3.2, the key byte dependency of the Klein's attack degrades the performance of this attack. Then, Teramura et al. proposed a method of correcting the votes table without revoting by the Klein's attack. This method uses difference between a correct key byte and an incorrect key byte. In this method, the minimum probability which this attack succeeds is equal to that of the PTW function and the maximum probability is equal to that of the Klein function. In addition, we need not revote when the restoration of a key byte is incorrect.

If we consider the same situation of the existing attacks against WEP (the situation 3), this attack can recover a 104-bit secret key by observing about 29,500 encryption packets. However this situation is far from a common state. Then if we consider the situation 1 and the situation 2, this attack can recover a 104-bit secret key by observing about 36,500 encryption packets and about 34,000 encryption packets, respectively.

3.5 Consideration about the Attack against WEP

We described several attacks against WEP. In this section, we consider about the abovementioned attacks. First, it is very difficult to prevent several attacks against WEP, because the abovementioned attack is easily executable for the attacker. Then, we strongly recommend the disuse of WEP. If we want to use WEP, we should update the secret key whenever it is communicated for 8,000 packets [14]. However, even if we use this technique, we cannot make the guarantee that WEP is safe. So we recommend the use of WPA.

4. Attacks against WPA-TKIP

As shown by the abovementioned, WEP has the fatal vulnerability. Then the IEEE Standards Association formulated a new security protocol WPA-TKIP. This protocol can prevent many attacks against WEP. However WPA-TKIP has the possibility to be attacked, too. In this section, we describe some attacks against WPA-TKIP.

As far as we know, "the key recovering attack against WPA-TKIP" can not be executed in a realistic environment. However, "the falsification attack against WPA-TKIP" can be executed in a realistic environment. In Sect. 2.3, we described that WPA-TKIP includes three methods that prevent the falsification attack (the TKIP-IV check, the checksum check, and the MIC check). If an attacker can break these three checks in a realistic environment, one can executes the falsification attack.

4.1 The Beck-Tews Attack

Beck and Tews proposed methods which break three checks. For breaking the TKIP-IV check, they used a special feature of IEEE 802.11e [15]. For breaking the checksum check, they proposed a method in which the chopchop attack [16] on WEP is applied to WPA-TKIP. For breaking the MIC check, they proposed a reversible function of MICHAEL. We call this attack the Beck-Tews attack.

IEEE 802.11e is a technology that controls the QoS in a wireless LAN network. Communication using IEEE 802.11e has eight communication channels which are allocated priority. Moreover, the TSC counter is managed in each priority in IEEE 802.11e. Therefore, each priority has a different TSC counter. When an attacker captures the encryption packet of $IV = x$, one selects the priority that TSC counter is less than or equal to $x - 1$ and executes the replay attack. The Beck-Tews attack can break the TKIP-IV check by using abovementioned technique. Figure 4 is the model

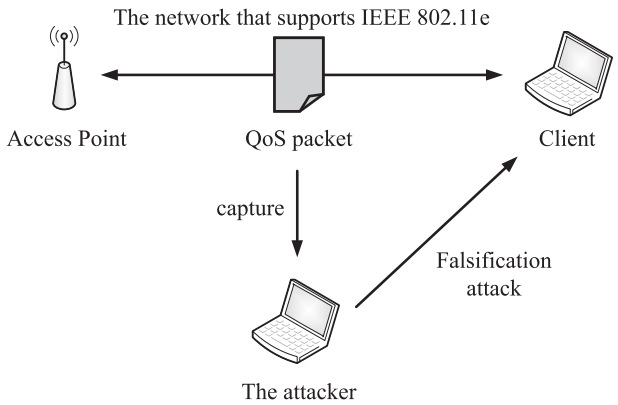

Fig. 4 The model of the Beck-Tews attack.

of the Beck-Tews attack.

Next, they proposed a method in which the chopchop attack on WEP is applied to WPA-TKIP. An attacker can obtain information about a plaintext from a given ciphertext by using the chopchop attack. It should be noted that this attack cannot acquire the encryption key. Usually, this attack can not be executed against WPA-TKIP. However, by breaking the TKIP-IV check, the attacker can execute the chopchop attack against WPA-TKIP too. This attack sequentially restores the unknown bytes of the ciphertext from the lower byte of the packet, and can decrypt all information about the ARP packet within 11–14 min.

Finally, they proposed a method for breaking the MIC check. In WPA-TKIP, the MIC is calculated by using the message integrity check function MICHAEL as follows:

$$MIC = michael(MICKey, DestinationMACAddress, SourceMACAddress, QoS\ priority, Data).$$

Beck and Tews proposed a reverse function of MICHAEL as follows:

$$MICKey = reverse\_michael(MIC, DestinationMACAddress, SourceMACAddress, QoS\ priority, Data).$$

If an attacker knows the MIC and the Data (all information about the ARP packet) by executing the chopchop attack, the MIC key is easily restorable. Once the attacker obtained the keystream corresponding to the MIC key and the IV, one can counterfeit the encryption packet, whose size is the same as that of the keystream. However, if the falsification packet is accepted, the TSC counter is updated to the largest value of the IVs corresponding to all the MPDUs. Then, the attacker can execute this attack only 7 times.

4.2 The Reverse Chopchop Attack

After Beck and Tews proposed the attack against WPA-TKIP, we proposed two new attacks against WPA-TKIP, the reverse chopchop attack and the QoS forgery attack. In this section, we describe the reverse chopchop attack. An attacker can use the reverse chopchop attack in place of the chopchop attack.

In this paper, we describe only the principle of this attack. The existing chopchop attack sequentially restores the unknown bytes of the ciphertext from the lower byte of the packet. However, if all the bytes of the packet, except for CRC32, are already known, the restoration of CRC32 is unnecessary. Then, we apply a technique for restoring the ciphertext using higher bytes of the packet to WPA-TKIP. This technique for WEP has been proposed [17], and we call this attack the reverse chopchop attack.

If an attacker uses this attack, one can achieve various effects. First, the reverse chopchop attack can reduce the execution time required to restore the MIC key. In this attack, the attacker can decrypt all information about the ARP packet within 7–8 min; this execute time is shortened about 4 min compared to the chopchop attack. Moreover, the attacker can restore the MIC key from all information about the ARP packet as well as the Beck-Tews attack. Next, the reverse chopchop attack can execute an information gathering attack. In the chopchop attack, the lower byte of the CRC32 is decrypted first of all. However, this information has no special significance. On the other hand, if the reverse chopchop attack is executed against an ARP packet, this attack can generally decrypt the IP address. This information is effective to execute this attack again after the MIC key is updated, because the attacker needs not decrypt the IP address. Finally, the reverse chopchop attack can falsify a variable-length packet. This attack can also restore a keystream with a length more than that of the keystream used for the chopchop attack. Therefore, this attack can falsify a variable-length packet, but a time of 1 min is required to enhance the keystream by 1 byte.

4.3 The QoS Forgery Attack

In this section, we describe the QoS forgery attack. This attack is based on the vulnerability of the QoS packet processing. The QoS forgery attack is used to break the TKIP-IV check and can expand its targets compared to the Beck-Tews attack. Table 1 lists the differences between the Beck-Tews attack and the QoS forgery attack.

The Beck-Tews attack can be executed against only a network that supports IEEE 802.11e features. However, IEEE 802.11e can be disabled by setting an AP appropriately, and a client connected to the AP that does not support IEEE 802.11e cannot be attacked. Because this network does not use the QoS packet, and an attacker can not break the TKIP-IV check. We call this network the standard network. On the other hand, the QoS forgery attack does not depend on whether the network supports IEEE 802.11e. In the QoS forgery attack, we proposed the method of rewriting the usual packet to the QoS packet. And we executed this attack against some clients chosen at random. As a result, our attack succeeded under the condition that the chipset of the client supported IEEE 802.11e even if the client disabled this standard through the OS. In recent years, many chipsets of clients are available in the market support IEEE

Table 1 Comparison of the Beck-Tews attack and the QoS forgery attack.

	Access Point	Client	Network
The Beck-Tews attack	QoS enabled	QoS enabled	QoS enabled
The QoS forgery attack	-	IEEE 802.11e function (chipset)	QoS disabled

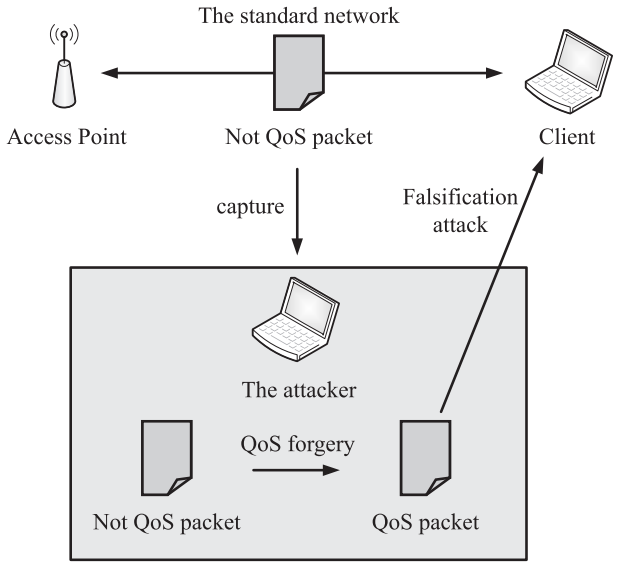

Fig. 5 The model of the QoS forgery attack.

802.11e. Therefore, if other clients have this vulnerability, almost all WPA-TKIP implementations would fail to protect a system against the falsification attack in a realistic environment. Figure 5 is the model of the QoS forgery attack.

4.4 Consideration about the Attack against WPA-TKIP

In this section, we describe realistic damage caused by the abovementioned attacks. Moreover, we propose some techniques for preventing the abovementioned attack.

First, we describe realistic damage. The attack against WPA-TKIP can not recover the encryption key. As a result, executable damage to WPA-TKIP is limited compared with damage to WEP. We made an abovementioned attack tool and executed a denial-of-service attack (DoS attack). As a result of the experiment, we were able to recover the MIC key by using the reverse chopchop attack and the QoS forgery attack against the standard network. In this time, we were able to execute the ARP cache poisoning attack. Namely, the ARP table of the client which is attacked is rewritten, and the client fell into the DoS.

In 2009, F.M. Halvorsen et al. proposed the DHCP DNS attack [18]. This attack can be executed only against the specific operation system (OS); for example MAC OS X. However, this attack can rewrite the DNS table of the target to the spoofed DNS table. As a result, an attacker can mislead the target to a malicious site.

Next, we propose some techniques for preventing the abovementioned attacks. The QoS forgery attack is based on the vulnerability of the QoS packet processing of the client. First, vendors should immediately take steps to overcome this vulnerability. If vendors implement a client that discards the QoS packet when it does not use IEEE 802.11e, the attacker cannot use the QoS forgery attack. However, it is important to propose some techniques for preventing this attack until the vulnerability is overcome. First, the abovementioned attacks can be executed against WPA-TKIP, but can not be executed against WPA-AES. Then, we strongly recommend the shift to WPA-AES. Second, there is a technique of reducing the key update interval. This technique was proposed for preventing the Beck-Tews attack, but this requires meticulous attention because it cannot prevent the information gathering attacks. However, we consider that this technique will be able to prevent the abovementioned realistic damage.

5. Conclusion

In this paper we discussed the vulnerability of WEP and WPA-TKIP. WEP is widely used in a wireless LAN security even now. However very serious vulnerability has been discovered in WEP as we show. Consequently we strongly recommend the disuse of WEP. Otherwise, we should update the secret key whenever it is communicated for 8,000 packets whatever we want to use WEP. In WPA-TKIP, we proposed the QoS forgery attack, which is a falsification attack based on the vulnerability of QoS packet processing. In this attack, a condition of the Beck-Tews attack that APs support IEEE 802.11e is negated. In addition, we discovered that almost all clients support IEEE 802.11e with a chipset and cannot disable the IEEE 802.11e function. In other words, if an attacker uses the proposed attacks, almost all wireless LAN implementations can be attacked. Therefore, WPA-TKIP is not secure in a realistic environment.

References

[1] IEEE Computer Society, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," IEEE Std 802.11, 1999.
[2] IEEE Std 802.11i-2004, "Part11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 6: Medium access control (MAC) security enhancements," IEEE, July 2004.
[3] S. Fluhrer, I. Mantin, and A. Shamir, "Weaknesses in the key scheduling algorithm of RC4," SAC2001, Lecture Notes in Computer Science, vol.2259, pp.1–24, Springer-Verlag, 2001.
[4] Orinoco, "WEPplus white paper," Oct. 2001.
[5] The Aircrack-NG Team, "Aircrack-NG." http://www.aircrack-ng.org/
[6] T. Ohigashi, Y. Shiraishi, and M. Morii, "FMS attack-resistant WEP implementation is still broken -Most IVs leak a part of key information-," LNCS, vol.3802, pp 17–26, 2005.

[7] T. Ohigashi, Y. Shiraishi, and M. Morii, "New weakness in the key-scheduling algorithm of RC4," IEICE Trans. Fundamentals, vol.E91-A, no.1, pp.3–11, Jan. 2008.
[8] E. Tews, R. Weinmann, and A. Pyshkin, "Breaking 104 bit WEP in less than 60 seconds," Cryptology ePrint, 2007, available at http://eprint.iacr.org/2007/120.pdf
[9] R. Teramura, Y. Asakura, T. Ohigashi, H. Kuwakado, and M. Morii, "Fast WEP-Key recovery attack using only encrypted IP packets," IEICE Trans. Fundamentals, vol.E93-A, no.1, pp.164–171, Jan. 2010.
[10] M. Beck and E. Tews, "Practical attacks against WEP and WPA," Proc. PacSec'08, pp.79–85, 2008.
[11] T. Ohigashi and M. Morii, "A practical message falsification attack on WPA," Proc. JWIS 2009, CDROM, 5A-4, 2009.
[12] Y. Todo, T. Ohigashi, and M. Morii, "Effective falsification attack on WPA-TKIP by modifying any packet to QoS packet," Proc. JWIS 2010, CDROM, 1B-3, 2010.
[13] A. Klein, "Attacks on the RC4 stream cipher," Designs, Codes and Cryptography, vol.48, no.3, pp.269–286, Sept. 2008.
[14] M. Vuagnoux, "News key recovery attacks on RC4/WEP," 27th Chaos Communication Congress, 2010.
[15] Wi-Fi Alliance, "Wi-Fi CERTIFIED™ for WMM™ - Support for multimedia applications with quality of service in Wi-Fi® networks," available at http://www.wi-fi.org/files/wp_1_WMM%20QoS%20In%20Wi-Fi_9-1-04.pdf
[16] KoreK, "Chopchop (Experimental WEP attacks)," 2004, available at http://www.netstumbler.org/showthread.php?t=12489
[17] W.A. Arbaugh, "An inductive chosen plaintext attack against WEP/WEP2," available at http://www.cs.umd.edu/˜waa/attack/frame.htm
[18] F.M. Halvorsen, O. Haugen, M. Eian, and S.F. Mjølsnes, "An improved attack on TKIP," Proc. NordSec2009, LNCS, vol.5838, pp.120–132, 2009.

Masakatu Morii received the B.E. degree in electrical engineering and the M.E. degree in electronics engineering from Saga University, Saga, Japan, and the D.E. degree in communication engineering from Osaka University, Osaka, Japan, in 1983, 1985, and 1989, respectively. From 1989 to 1990 he was an Instructor in the Department of Electronics and Information Science, Kyoto Institute of Technology, Japan. From 1990 to 1995 he was an Associate Professor at the Department of Computer Science, Faculty of Engineering at Ehime University, Japan. From 1995 to 2005 he was a Professor at the Department of Intelligent Systems and Information Science, Faculty of Engineering at the University of Tokushima, Japan. Since 2005, he has been a Professor at the Department of Electrical and Electronics Engineering, Faculty of Engineering at the Kobe University, Japan. His research interests are in error correcting codes, cryptography, discrete mathematics and computer networks and information security. Dr. Morii is a member of the IEEE and the Information Processing Society of Japan.

Yosuke Todo received the B.E. degree from Kobe University, Japan, in 2010. Since 2010, he has been a master's student in Graduate School of Engineering, Kobe University. His current research intersts are in information security and cryptography.