RL-002

Practical DHCP DNS Attack on WPA-TKIP
–Breaking WPA-TKIP in realistic environment–

小澤 勇騎 †　藤堂 洋介 †　大東 俊博 ‡　森井 昌克 †
Yuki Ozawa †　Yosuke Todo †　Toshihiro Ohigashi ‡　Masakatu Morii †

1 Introduction

WPA (Wi-Fi Protected Access)[1] is a security protocol for wireless LAN communication, and it provides confidentiality and integrity. WPA has been designed in order to fix weaknesses[2, 3, 4] of WEP (Wired Equivalent Privacy)[5], which is a past security protocol used in many wireless LAN products. WPA-TKIP is more secure than WEP, and the realistic attack except for off-line dictionary attack[6] has not been proposed yet. However, it is said that WPA-TKIP has some vulnerabilities to a falsification attacks. The falsification attacks which bring threats to message integrity have been discussed. The attack was proposed by Beck and Tews in 2008 (Beck-Tews attack)[7]. Their attack recovers a message integrity check key (MIC key) and a plaintext from an encrypted short packet like an ARP packet in the IEEE802.11e[8] network, and it falsifies the ARP packet in 12–15 minutes.

We have proposed a man-in-the-middle (MITM) attack[9, 10] since 2009, a reverse chopchop attack[11] and a QoS forgery attack[12] since 2010. We showed that the execution time on the falsification attack was much shorter and the attack can be executed in the general network which does not support IEEE802.11e. In 2009, F.M. Halvorsen showed that in the IEEE802.11e network they could falsify a DHCP ACK packet which was much larger than an ARP packet[13].

The some methods that the attacker can do a harm with the falsification attack in the real environment have been discussed. Their methods include ARP cache poisoning, DHCP DNS attack and NAT Traversal attack etc[13]. However, almost attacks are not developed from theoretical considerations and a part of the experimentation. In this paper, we implement ARP cache poisoning and DHCP DNS attack. In addition we discuss realistic damages caused by these attacks in the real environment and a variety of problems in implementing these attacks. Moreover, we demonstrate that the execution time on DHCP DNS attack can be reduced in half by the reverse chopchop attack and the QoS forgery attack in realistic environment. As an experiment result, in almost all networks the attacker can halt the client's use of the network services for over 30 minutes and force the client to set an IP address of the false DNS server within 18 minutes.

2 WPA-TKIP

WPA uses two kinds of keys, which are a MIC key and a encryption key. The former is used to detect the message forgery/falsification, and the latter is used to encrypt/decrypt packets. These keys are generated from a shared master key.

We describe a process of the sender. A MIC is generated from the MIC key and a data, and it is added to the data. If the data is very large, the data is fragmented into small data. In this paper, we describe about a small data is not fragmented an ICV is calculated from the data with the trailing MIC by using CRC32, and it is added to the data. The data with the trailing the MIC and the ICV is a plaintext packet. A pseudo-random sequence (called a keystream) is generated from an initialization vector (IV) and an encryption key by using RC4[14]. The plaintext packet is encrypted by XORing the plaintext packet with the keystream. The sender sends this encrypted packet to a receiver.

We describe a process of the receiver. The receiver receives the encrypted packet and an IV. The IV is compared with the TSC counter, which is a value of the IV corresponding to an encrypted packet accepted most recently. If the received IV is less than or equal to the TSC counter, the received encrypted packet is discarded. The same keystream as the one of the sender is generated from an IV and an encryption key. The encrypted packet is decrypted by XORing the keystream with the packet. The receiver calculates an ICV from the received packet, and the ICV is compared with the received ICV. If these ICVs differ, the received packet is discarded. The receiver calculates a MIC from the received packet, and the MIC is compared with the received MIC. If these MICs differ, the received packet is discarded and the receiver sends the error message of MIC (a MIC failure report frame) to the sender. In WPA, the MIC key is changed if more than two error messages of MIC are sent to the sender in less than a minute. When the received packet is accepted, the TSC counter is updated to the value of IV which the received packet included.

3 ARP Cache Poisoning

3.1 Previous Attack

Beck and Tews showed that the attacker can recover MIC key is used in WPA-TKIP in 9–13 minutes, and can execute an ARP cache poisoning. The ARP cache poisoning is an attack that an attacker can confuse a network which a client belongs to with poisoning the client of ARP

† 神戸大学大学院工学研究科, Graduate School of Engineering Kobe University
‡ 広島大学情報メディア教育研究センター, Information Media Center, Hiroshima University

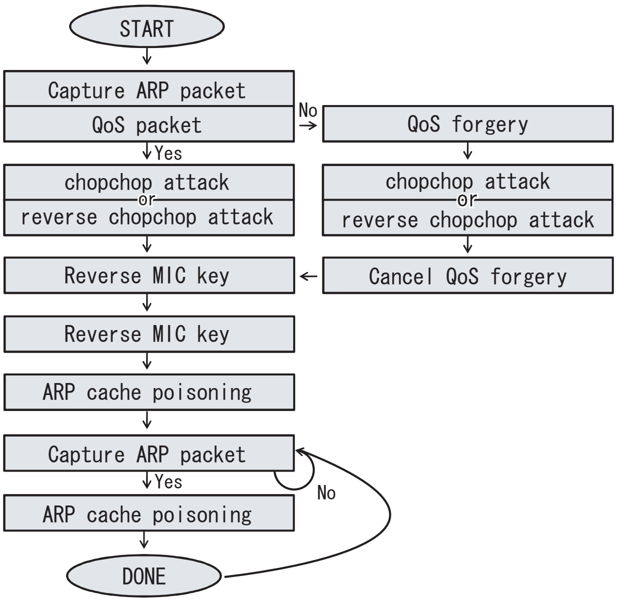

Fig.1 DoS attack using ARP cache poisoning

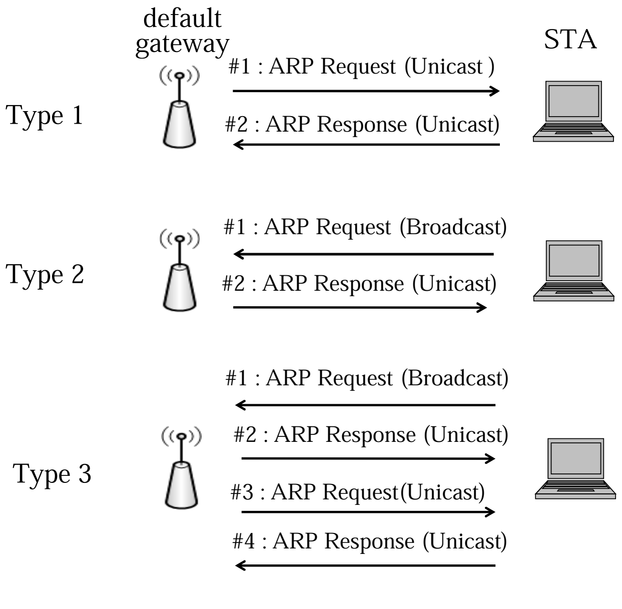
default gateway
STA
Type 1
#1 : ARP Request (Unicast)
#2 : ARP Response (Unicast)
Type 2
#1 : ARP Request (Broadcast)
#2 : ARP Response (Unicast)
Type 3
#1 : ARP Request (Broadcast)
#2 : ARP Response (Unicast)
#3 : ARP Request(Unicast)
#4 : ARP Response (Unicast)

Fig.2 3 types of ARP packets

cache. The attack which takes advantage of the weakness of the ARP protocol is not prevented easily and is not noticed to be attacked by the attacker. Though Beck and Tews showed that the the attacker can execute the ARP cache poisoning, they have not proposed an attack model which can damage in a real environment. A specific attack model with the ARP cache poisoning has been proposed by F.M. Halvorsen. They showed that it was the most effective to poison a MAC address corresponding to IP address of a default gateway, and the attacker could execute a DoS attack. This reason is that the packets pass the default gateway certainly when the attacker send the packets from a local network to outside. The client who was poisoned the MAC address of the default gateway cannot send packets to the outside of network, and a variety of network services like net-surfing and e-mail are not available. However, the ARP cache is refreshed automatically after the specific time. The client can recover from the DoS, because their attack cannot send the only seven falsified packets.

We develop the existing attack and propose a concrete attack model with ARP cache poisoning. With our attack, the attacker can keep executing the DoS attack unless the client disconnects the network or the MIC key is renewal.

3.2 An Improved DoS Attack

In this section, we propose that the attacker can execute an additional attack in client's recovering from the DoS, after the attacker can execute the ARP cache poisoning. A flowchart of our attack shows Fig. 1. We show 3 type of sending and receiving ARP packets in Fig. 2 to propose the additional attack. Type 1 shows that the default gateway sends ARP Request packets to unicast for the client, and the client sends ARP Response packets to unicast. Type 2 shows that the client sends ARP Request packets to broadcast for the default gateway, and the default gateway sends ARP Response packets to unicast. Type 3 shows that after Type 2, the default gateway sends ARP Request packets to unicast for the client, and the client sends ARP Response packets to unicast. The attacker must confirm which types of ARP packet were sent when he/she could execute a DoS attack with ARP cache poisoning. This reason is that there are two types of the ARP packets are sent between the default gateway and the client. They are ARP Request and ARP Response. And they are different in construction. If the attacker mistakes the type of an ARP packet, he/she fails to execute the ARP cache poisoning.

The attacker can recover an IP address of the sender or the receiver in 10 seconds with a reverse chopchop attack. The attacker can guess easily the IP address of the default gateway, because the default gateway must communicate with all clients connected with a network. Next, the attacker observes the ARP packets are sent from the default gateway to a target client and confirms which types of the ARP packets in Fig. 2. As a result, the attacker can guess the construction of an ARP packet which will be sent next. The attacker captures the ARP packet which is sent in the client's recovering to the real ARP cache, and can execute the additional attack by specifying a keystream in a moment.

3.3 Evaluation Experiment

We make evaluation experiments of the DoS attack with the ARP cache poisoning. We observe the network well and evaluate whether the attacker can halt the client's use of the network services for over 30 minutes. We make the evaluation experiments in two kinds of network. One is Type 1, the first ARP packet is sent from the default gateway to the client, the other is Type 2 and Type 3, the first ARP packet is sent from the client to the default gateway. In the former case, a first ARP Response packet is sent and the attacker sends the falsified packet which was en-

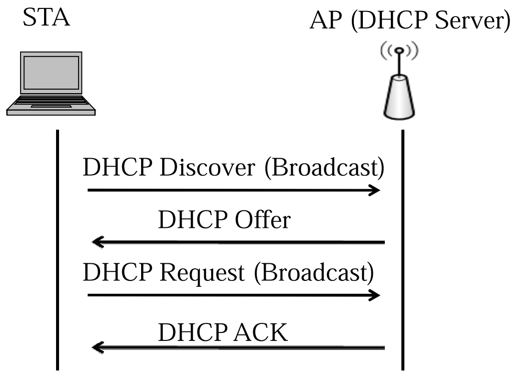

Fig.3　DHCP protocol sequence

crypted with a keystream of an ARP Request packet. In the latter case, the first ARP Response packet was sent 10 seconds after the attacker sends the falsified packet which was encrypted with a keystream of the first ARP Response packet. Here, the attacker waits for 10 seconds because he/she can execute the ARP cache poisoning in Type 2 and Type 3. The evaluation experiment showed that the attacker can execute the DoS attack for over 30 minutes in the both type networks.

We explain what the damages will be arisen by our attack. First, the client cannot use any web services, the web surfing and web mail etc. And even if the client is damaged the ARP cache poisoning, he/she cannot identify the cause. The client must reconnect the network or refresh the ARP cache himself/herself to recover from the DoS, because the client who was damaged our attack cannot recover automatically.

4　DHCP DNS Attack

4.1　Previous Attack

4.1.1　DHCP

DHCP is used for the dynamically configure IP network parameters of a client in a local network. DHCP is based on a server-client model, when the client requests network parameters from a DHCP server. Figure 3 shows a DHCP protocol sequence. The DHCP server typically provides the client with IP address, Subnet Mask, Gateway IP, DNS Server and other parameters required for the client in the network. The DHCP consists of four basic phases, DHCP -Discover, -Offer, -Request and -ACK. The client connects to a network, then the DHCP server sends DHCP Discover packet to the broadcast. The DHCP server which received the DHCP Discover packet sends clients DHCP Offer packet. The DHCP Offer packet includes IP address offered by the DHCP server and other network parameters. The DHCP server preserves the network parameters, IP address etc. The DHCP Request packet includes IP address of the client and Transaction ID included in the DHCP Offer packet. If the value of Transaction ID which the client sent is not equals to the value of Transaction ID which the DHCP server sent, the DHCP server eliminates the preserved network parameters in sending the DHCP Offer packet. The DHCP sever does not respond with the DHCP Request packet. If the value of Transaction ID which the client sent is equal to the value of Transaction ID which the DHCP server sent, the DHCP server sends DHCP ACK packet. The client receives the DHCP ACK packet and sets IP address and other network parameters included in the DHCP ACK packet. The DHCP procedures are finished.

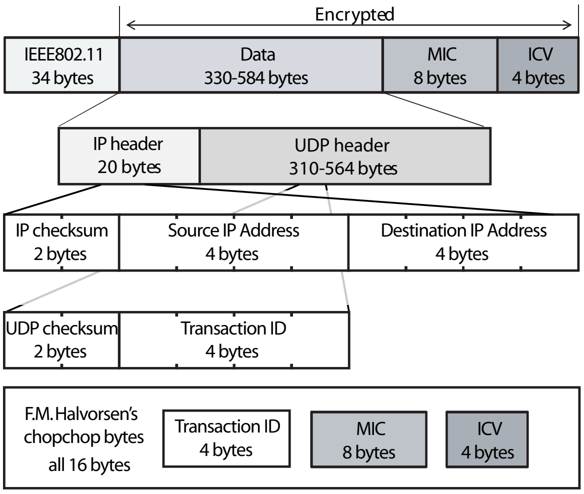

Fig.4　Format of DHCP ACK packet and Halvorsen's attack

4.1.2　DHCP ACK packet

Figure 4 shows a DHCP ACK packet format. The DHCP ACK packet is very large size packet which is over 300 bytes. They are always sent by the same size in the same network. However the packet format is a little different depending on the vender, we can guess the packet format form BSSID of the AP.

The DHCP ACK packet includes 34 bytes of the IEEE802.11 part, 330–584 bytes of the Data part, 8 bytes of the MIC part and 4 bytes of the ICV part, and it is encrypted except for the IEEE802.11 part. The Data part includes 20 bytes of the IP header part and 310–564 bytes of the UDP header part. The IP header part includes 2 bytes of the IP checksum, IP addresses of the sender and the receiver and the fixed value which shows the kind of packet. The IP checksum is the value to check whether IP header part is broken and is calculated with the only 20 bytes of the IP header part. The UDP header part includes 2 bytes of UDP checksum, 4 bytes of the Transaction ID, a various of network parameters, for example IP address of DNS server etc. The UDP checksum is also calculated with the only UDP header part. The MIC is used to detect whether the packet is falsified, and is calculated with the IEEE802.11 part and the Data part. The ICV is used to detect whether the packet is broken, and is calculated with the Data part and the MIC part. If a network is established by a router like a standard home, we can guess the value of network parameters included the UDP header part. If we know the IP addresses of the sender and the receiver, unknown bytes of the DHCP ACK packet are only 16 bytes, which are 4 bytes of Transaction ID, 8 bytes of MIC and 4 bytes of ICV. The DHCP ACK packet has a feature that there are few unknown bytes, because zeros make up of a majority of the Data part.

4.1.3 DHCP DNS Attack

As we described in Sect. 4.1.1, when the client resolves a domain name, the client inquires DNS server for the first time. The client must set IP address of the DNS server himself/herself. In case of the setting IP address of the DNS server with DHCP, the client sets the IP address included in the DHCP ACK packet which is sent from DHCP servers. The DHCP DNS attack is an attack that an attacker forces the client to set the forged IP address of the DNS server and other network parameters. Some OSs[*1] increase the value of Transaction ID simply each new packet. If the attacker can recover the value of Transaction ID of packet sent from the OS, he/she can guess the value of Transaction ID of a DHCP ACK packet which the client will receive next. Even if the attacker sends the falsified DHCP ACK packet with the guessed Transaction ID simply, the client doesn't receive the packet. The attacker must force the client to send the DHCP Request packet to force the client to receive the falsified DHCP ACK packet. It becomes possible for the attacker to force the client to occur an IP conflict. If the some OSs occur the IP conflict, DHCP is automatically updated. The attacker sends the falsified DHCP ACK packet with the valid Transaction ID, after the client sent the DHCP Request packet. If the falsified DHCP ACK packet reaches the client before the legitimate packet reaches, the client receives that falsified packet without a doubt. Then, the client does not receive the legitimate DHCP ACK packet.

According to the F.M. Halvorsen's experiments, the attacker must send four ARP packets which include the same IP address as the client to force the client to occur the IP conflict. The target OS has four priorities[*2]. The priorities refer to the passage of the packets. The attacker can use only the three priorities to attack, because one priority of four is used in regular communication. However, the attacker cannot force the client to occur IP conflict, because he/she creates only three ARP packets from a keystream. The attacker must get two keystreams from two packets. Specifically, the attacker sends the two falsified ARP packets which were encrypted with each different keystream to one priority, and does the same to another priority.

We show a flowchart to execute DHCP DNS attack in Fig. 5. The attacker creates De-authentication packet which is used a disconnection and sends it to AP. The client was forced to disconnect by the attacker starts to reconnect to the AP. Then, ARP packet with IV=X and DHCP ACK packet with IV=Y are sent in the network, where Y>X. The attacker gets these packets. The attacker executes chopchop attacks for the ARP packet and recovers IP addresses of the client and the AP, MIC key and a keystream of IV=X. The attacker finished the chopchop attacks for the ARP packet, he/she executes the chopchop attacks for the DHCP ACK packet and recovers the value of Transaction ID and a keystream of IV=Y. The attacker creates four ARP packets, two ARP packets have different priorities respectively and they were encrypted using the keystream of IV=X, and the other two ARP packets have different priorities respectively and they were encrypted using the keystream of IV=Y. These four ARP packets include the same IP address as the client. The client misunderstands that the IP conflict occured and starts to update DHCP, if the attacker executes the ARP cache poisoning with these four ARP packets. The attacker sends the falsified DHCP ACK packet is encrypted using the keystream of IV=X to the client immediately after the client sent the DHCP Request packet to the AP. If the falsified DHCP ACK packet reaches the client before the legitimate DHCP ACK packet reaches, the attacker can force the client to set the forged IP address of the DNS server.

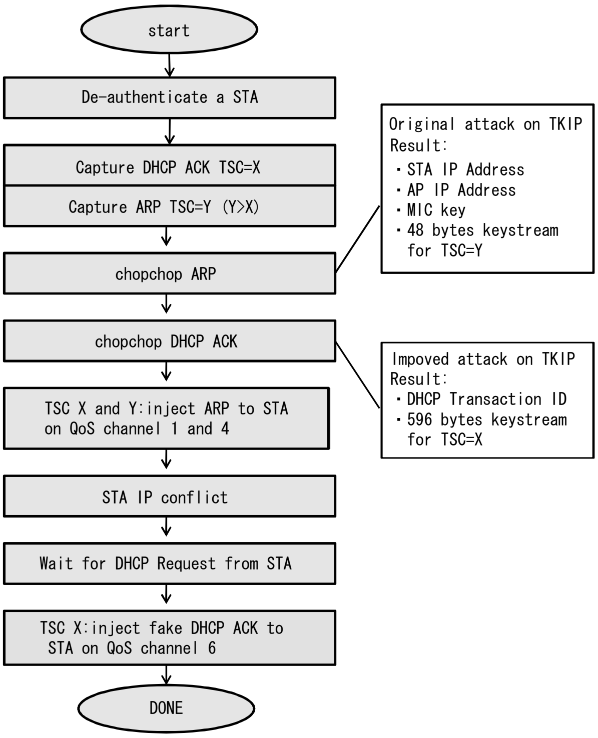

Fig.5 Flowchart of DHCP DNS attack

4.1.4 Execution Time on Attack

In 2008, Beck and Tews showed that they could falsify ARP packet on WPA supports IEEE802.11e QoS features in about 12 minutes. Their attack is implemented on tkiptun-ng[*3]. Moreover, F.M. Halvorsen developed the tkiptun-ng and showed that they could falsify DHCP ACK packet on WPA supports IEEE802.11e QoS features. Their attack can recover the total 16 bytes including 8 bytes of MIC, 4 bytes of ICV and 4 bytes of Transaction ID. The execution time is about 18 minutes and 30 seconds if the attacker receives all MIC error messages correctly. The execution time is about 20-25 minutes in the

[*1] Mac OS X 10.5.

[*2] Mac OS X 10.5.

[*3] http://www.aircrack-ng.org/

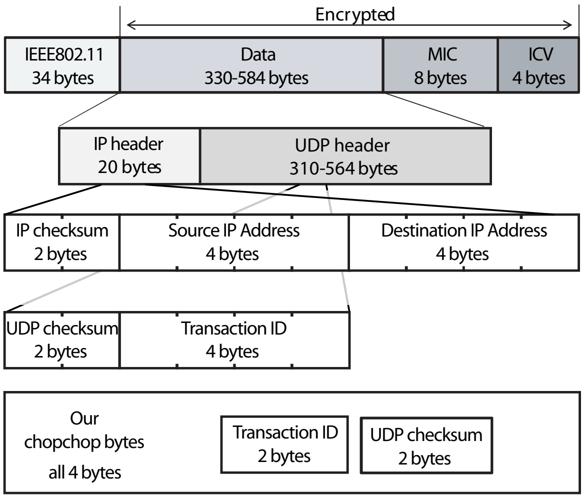

Fig.6 Our attack on the DHCP ACK packet

real environment because the attacker may not receive all MIC error messages correctly. However, the attacker must get IP addresses of the sender and the receiver in advance. The attacker must recover these IP addresses from the ARP packet if the attacker doesn't know them. The execution time for the ARP packet and the DHCP ACK packet is about 40 minutes.

4.2 An Improved Method for Falsifying Packet

If we know the IP addresses of the sender and the receiver, unknown bytes of the DHCP ACK packet are only 16 bytes, which are 4 bytes of Transaction ID, 8 bytes of MIC and 4 bytes of ICV. The DHCP ACK packet has a feature that there are few unknown bytes, because the Data part is almost made up of zeros.

We can estimate the execution time of the existing attack at about 40 minutes, and the target network must support IEEE802.11e. We are difficult to say that this attack is a realistic attack in the real environment. The reason is that the default update interval of the MIC key is an hour and the attacker must keep attacking for about 40 minutes without the client's noticing the attack.

In this paper, we propose the attack method which reduces the execution time greatly and enables us to execute almost all networks using reverse chopchop attack and QoS forgery attack.

4.2.1 Reverse Chopchop Attack

The Beck-Tews attack enables to recover from the least significant byte of unknown part, ICV in most cases, by 1 byte. The reverse chopchop attack[11] made a presentation in 2010 enables to recover from the most significant byte of unknown part by 1byte. The attacker must guess the higher part he/she will recover correctly to execute the reverse chopchop attack.

With our attack, the attacker recovers the total 10 bytes, each the least significant bytes of a sender and a receiver IP address and 8 bytes of MIC using the reverse chopchop attack for an ARP packet. And the attacker calculate ICV without executing the reverse chopchop attack which takes longer. The attacker takes about 11 minutes to falsify the ARP packet. Then, the attacker calculates a keystream and a MIC key from the recovered MIC.

The first unknown part is IP checksum when the attacker recovers a DHCP ACK packet using the reverse chopchop attack. The attacker doesn't recover the IP checksum because the IP checksum is calculated with the sender and receiver IP addresses recovered form the ARP packet as describe in Sect. 4.1.2. The parts which the attacker executes the reverse chopchop attack on the DHCP ACK packet first, are 2 bytes of UDP checksum and 2 bytes of Transaction ID. The attacker calculates the candidates of 2^{16} UDP checksums from the candidates of 2^{16} Transaction IDs and the other bytes of the UDP header to recover the other 2 bytes without executing the reverse chopchop attack. The attacker compares these candidates of 2^{16} UDP checksums with the UDP checksum which the attacker recovered. As a result, the candidates of the UDP checksum which is equal to the recovered UDP checksum is always only one. The attacker can guess all the other parts of the UDP checksum. The attacker calculates MIC from the Data part which he/she guessed and the MIC key which he/she recovered from the ARP packet. And the attacker calculates ICV. The attacker calculates a keystream from the DHCP ACK packet he/she recovered and the encrypted DHCP ACK packet. In the attack on the DHCP ACK packet with our method, the attacker must recover 4 bytes. The execution time is about 5 minutes in the real environment. Figure 6 shows that the attacker executes the reverse chopchop attack on the parts for the DHCP ACK packet. As described above, the attacker recovers only 14 bytes in all and the execution time is at most 20 minutes in the real environment. The execution time on our attack is less than half the execution time on the existing attack.

4.2.2 QoS Forgery Attack

If the network the client and AP belong to dose not support IEEE.11e on the recent implementation on the wireless LAN, QoS packets are not sent in the network. The existing attack with the Beck-Tews attack is effective for only QoS packets. In other words, the target network must support IEEE.11e to success the existing attack. We applied the QoS forgery attack[12] presented in 2010 to falsify the DHCP ACK packets. The attacker can execute this attack on condition that the client's WLAN chipsets support IEEE802.11e. Almost all the client's WLAN chipsets which were released recently support IEEE802.11e. As a result, almost all the client are the target of our attack.

4.3 Evaluation Experiment

We make evaluation experiments to evaluate how long we take to falsify an ARP packet and a DHCP ACK packet using the QoS forgery attack and the reverse chopchop attack in the real environment. The purpose of the evaluation experiments is that we measure the falsifying time for the two packets and evaluate whether our attack is useful in the real environment. Furthermore, we confirm whether we can execute our attack in the network which the existing attack cannot execute.

Table 1 shows a measurement environment. The ex-

Table 1 Measurement Environment

AP	WHR-HP-G
Client System	Mac OS X 10.5
Client WLAN Chipset	BCM4321(built-in iMac)
Connection	IEEE802.11g

isting attack cannot execute in this measurement environment because the network of this measurement environment doesn’t support IEEE802.11e. The evaluation experiments showed that the attacker could falsify the ARP packet and the DHCP ACK packet using the reverse chopchop attack and the QoS forgery attack in about 18 minutes. We confirmed that the execution time on our attack was much shorter than the one on the existing attack. As a result, the attacker will be able to execute a DHCP DNS attack even if you set shorter update interval of the MIC key. And, he/she will be able to execute the DHCP DNS attack in the network which doesn’t support IEEE802.11e.

We describe the necessary conditions for the attack in the real environment. The attacker must figure out the composition of the target network in advance. The attacker can guess the IP addresses of the DHCP server and the DNS server etc from one ARP packet, if one router makes up the network like a standard home. However, the more complex the target network is, the more difficult for the attacker to guess these IP addresses from the ARP packets. The attacker must recover these IP addresses using the extra reverse chopchop attacks, if he/she cannot guess them. In the some environments, the client and the AP don’t initiate a DHCP renewal, even if the attacker sends De-authentication packet which forces the client and the AP to disconnect. The attacker cannot initiate the attack at his/her convenience in those environments. The attacker must capture the DHCP ACK packet which is sent in the client’s reconnecting and initiate the same attack. The client can prevent this attack partially by setting the each different IP address of DHCP server, DNS server and AP, because the attacker requires much more time to complete the attack. It is also important that the client sets up not to connect to the AP automatically when the client receives the De-authentication packet.

5 Conclusion

In this paper, we showed that the attacker could halt the client’s use of the network services for over 30 minutes in almost all networks using the reverse chopchop attack and the QoS forgery attack for the ARP packets. We also showed that we could falsify an ARP packet and a DHCP ACK packet in about 18 minutes in almost all networks using the reverse chopchop attack and the QoS forgery attack. Therefore, we improved DHCP DNS attack into a more practical attack.

ACKNOWLEDGEMENTS

This work was supported in part by Gtant-in-Aid for Scientific Research (KAKENHI 21700018) of JSPS.

Reference

[1] Wi-Fi Alliance, “Wi-Fi protected access,” available at http://www.weca.net/opensection/protected_access.asp

[2] R. Teramura, Y. Asakura, T. Ohigashi, H. Kuwakado, and M. Morii, “Breaking WEP with Any 104-bit Keys –All WEP Keys Can Be Recovered Using IP Packets Only–,” Proc. of SCIS2009, CDROM, 1A2-6, Jan. 2009.

[3] KoreK, “chopchop (Experimental WEP attacks),” 2004, available at http://www.netstumbler.org/showthread.php?t=12489

[4] W. A. Arbaugh, “An Inductive Chosen Plaintext Attack against WEP/WEP2,” available at http://www.cs.umd.edu/~waa/attack/frame.htm

[5] IEEE Computer Society, “Wireless lan medium access control (MAC) and physical layer (PHY) specifications,” IEEE Std 802.11, 1999.

[6] R. Moskowitz, “ Weakness in Passphrase Choice in WPA Interface, ” 2003, avail- able at http://wifinetnews.com/archives/2003/11/weakness_in_passphrase_choice_in_wpa_interface.html

[7] M. Beck and E. Tews, “Practical Attacks Against WEP and WPA,” PacSec’08, pp.79–85, 2008.

[8] 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_WMM20QoS%20In%20Wi-Fi_9-1-04.pdf

[9] T. Ohigashi and M. Morii, “A practical message falsification attack on WPA,” Proc. JWIS 2009, CDROM, 5A-4, 2009.

[10] Y. Ozawa, T. Ohigashi, and M. Morii, “Weaknesses on WPA-TKIP and Application to the Message Falsification Attack,” Computer Security Symposium2009 (CSS2009), vol.2009, no.11, pp.805–810, Oct. 2009. (in Japanese)

[11] Y. Todo, Y. Ozawa, T. Ohigashi, and M. Morii, “A Note on Realistic Damages Caused by a Message Falsification Attack on WPA-TKIP,” IEICE Tech. Rep., vol. 109, no. 445, ISEC2009-115, pp. 233-240, March 2010. (in Japanese)

[12] Y. Todo, Y. Ozawa, T. Ohigashi, and M. Morii, “A Forgery Attack for WPA-TKIP by Modifying Any Packet to the QoS Packet, – Almost All Implementations of WPA-TKIP Can Be Attacked –,” IEICE Tech. Rep., vol. 109, no. 445, ISEC2009-114, pp. 225-232, March 2010. (in Japanese)

[13] F.M. Halvorsen, O. Haugen, M. Eian, and S. F. Mjϕlsnes , “An Improved Attack on TKIP,” Proc. NordSec2009, Lecture Notes in Computer Science, vol.5838, pp.120–132, Springer-Verlag, 2009.

[14] B. Schneier, Applied Cryptography, Wiley, New York, 1996.