Computers and Electrical Engineering 59 (2017) 76–85

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A client-side detection mechanism for evil twins☆

Fu-Hau Hsu, Chuan-Sheng Wang*, Yu-Liang Hsu, Yung-Pin Cheng, Yu-Hsiang Hsneh

Department of Computer Science and Information Engineering, National Central University, Taiwan

ARTICLE INFO

Article history:
Received 1 September 2014
Revised 25 October 2015
Accepted 26 October 2015
Available online 17 November 2015

Keywords:
Wireless
Evil twin
Rogue AP
Wi-Fi

ABSTRACT

In this paper, we propose a client-based solution to detect "evil twin" attacks in wireless local area networks (WLANs). An evil twin is a kind of rogue Wi-Fi access point (AP) which has the same SSID name as a legitimate one and is set up by an attacker. After a victim associates his device with an evil twin, an attacker can eavesdrop sensitive data forwarded through the evil twin. Most existing detection solutions are administrator-based, which are used by wireless network administrators to verify whether a given AP is in an authorized list or not. Such administrator-based solutions are limited, hardly maintained, and difficult to protect users 24–7. Hence, we propose a client-based detection mechanism, called evil twin detector, to detect this type of attacks. An evil twin detector changes its wireless network interface card (WNIC) to monitor mode to capture wireless TCP/IP packets. Through analyzing captured packets, our detector allows client users to easily and precisely detect an evil twin, thus avoids threats created by evil twins. Our method does not need to know any authorized AP list, and does not rely on data training or machine learning technique. Finally, we implement a detecting system on Windows 7.

1. Introduction

Nowadays, wireless local area networks (WLANs) are widely used in many public places, such as airports, schools, hotels, or cafés. Using the Wi-Fi access points (APs) installed at these places, users can use their mobile devices or laptop computers to connect to the Internet. Although wireless networking is more convenient than wired networking, it faces much more security threats. According to many Wi-Fi security reports, such as [1,2], rogue (phishing) AP is always among the top three wireless threats. "Evil twin" is a kind of rogue APs. According to the definition of [3], a rogue access point is a wireless access point that is installed on a network without explicit authorization from the administrator of the network. When a user connects to the Internet through a rogue AP, the creator of the rogue AP can sniff the data sent by the user. If the data are not encrypted, the attacker can obtain sensitive information contained in the data. Because a rogue AP is easy to setup, it raises serious menace to wireless users.

A rogue AP itself could connect to the Internet either through a wired network or a wireless network. A wired rogue AP is also called as a "wired rogue" and an "evil twin" is a wireless rogue AP. Usually an evil twin is set up with the same AP name (SSID) as a legitimate AP, called good twin, to attract normal users to connect to it, because normal users cannot distinguish an evil twin from the related good twin. Most modern operating systems connect to the AP with the best Received Signal Strength Indication

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. W-H Hsieh.

* Corresponding author. Tel.: +886-3-4227151.
E-mail address: 995402007@cc.ncu.edu.tw, luckyqqmaster@yahoo.com.tw (C.-S. Wang).

http://dx.doi.org/10.1016/j.compeleceng.2015.10.010
0045-7906/

(RSSI) [4] when they find multiple APs with the same SSID. Hence, an evil twin usually tries various approaches to generate strong RSSI to its target machines. However, in order to expand the signal coverage range of a WLAN, many organizations also assign the same SSID to multiple APs, which makes it more difficult to detect an evil twin.

To launch an evil twin attack, an attacker can configure a laptop to be an evil twin first. To enhance an evil twin's RSSI to its targets, the attacker either could deploy his evil twin at a location that is close to its targets or use a directional antenna. Then the attacker uses the SSID of related good twin to set the SSID of the evil twin. As a result, if a user tries to connect the Internet through the good twin, he may be cheated to use the evil twin, instead of the good twin. Besides, the attacker can launch a de-authentication attack [5] to force a victim to connect to the evil twin. After the above steps, the attacker can sniffer data forwarded through the evil twin.

An attacker typically launches an evil twin attack at public places, such as airports, schools, hotels, or cafés. Through setting up an evil twin, an attacker is able to obtain sensitive data of various users, such as passwords, web sessions, or credit card information, if these data are transmitted in plain text form in wireless packets. Besides, an attacker can also use an evil twin to launch a man-in-the-middle attack. Due to the serious threats created by evil twins, it becomes a critical issue to develop an evil twin detection mechanism.

This paper proposes a client side solution, ET detector, which can securely and reliably detect evil twins without any data training. It is difficult for an adversary to evade the detection of ET detector, even though the attacker knows the detail of ET detector. ET detector changes the wireless network interface controller (WNIC) of a laptop to monitor mode so that the laptop can capture all packets that are transmitted through the channel monitored by the WNIC and are transmitted through IEEE 802.11 protocols[1]. Via analyzing sniffed packets, ET detector can determine whether an AP forwards wireless packets to another AP. Because packet forwarding is a key property of evil twins, ET detector uses this property to accurately detect evil twins. ET detector has the following advantages. (i) ET detector does not require any authorized list. (ii) By switching a WNIC to monitor mode, a user does not need to associate his laptop with any AP while making detection. Thus, background processes, such as auto-login processes, that send sensitive data automatically to remote servers when connecting to the Internet will not unwittingly leak the information to the owners of evil twins. (iii) When making detection, ET detector does not need to pass web authentication, because it is a passive solution. However, many former solutions need to associate their machines with an AP first to connect to the Internet and make their detection. Hence, if the related AP uses web authentication to make access control and a tester does not have a related account, the tester cannot use the solutions to detect an evil twin. (iv) ET detector does not require any parameter training and is more reliable than others in a complicated condition. (v) An attacker is hardly to evade the detection of ET detector, because he cannot remove the packet forwarding feature of evil twins.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 describes the principle and the detection algorithm of our solution. Section 4 discusses various experimental results to evaluate the effectiveness and efficiency of our solution. Section 5 gives the conclusion.

2. Related work

There are two categories of solutions to detect evil twin attacks. The first one is administrator-based solutions. This kind of solutions usually perform RF signal monitoring. It may be implemented on the core network, such as switches or routers or special devices. Besides, these solutions usually verify specific "fingerprints" of an AP based on a pre-defined authorized list. [6–14] belong to this kind of solutions. An administrator-based solution is usually used by network administrators. However, when an attacker launches an evil twin attack, it is difficult for these solutions to provide a real-time protection. [15–21] monitor traffic at a traffic aggregation point of the wired side, such as gateway, to determine whether a machine uses wired or wireless connections. These solutions also compare collected information with an authorization list to determine whether the associated AP is a rogue one or not.

The other type of solutions is client-based solutions, which is usually deployed on users devices to detect evil twins. The advantage of this solution is that when users are not sure if a wireless network is secure, they can make the detection themselves. Therefore, they can protect their information more timely. Song et al. [22] proposed a client-side system to detect evil twin attacks. They proposed two detection algorithms to detect evil twins, which are called Train Mean Matching (TMM) and Hop Differentiating Technique (HDT). The TMM requires training of one-hop and two-hop wireless connection features. The HDT does not need training of wireless connection features. Panch and Singh [23] proposed a key exchange method after handshaking to detect evil twins. Nikbakhsh et al. [24] detects evil twins by checking IP addresses from APs.

To detect evil twin attacks on client side, the above solutions need to associate a computer with an AP first, then either connects to the Internet or not. Due to this property, some of them needs to specify the AP to associate with. It is impossible to detect an evil twin when having a low RSSI or be attacked by a de-authentication attack. Moreover, some of them do not work, if the AP under tests requires a user to provide account information to log in and the user does not provide it before the tests.

3. Principle and detection algorithm

This section describes the monitor mode of a Wireless Network Interface Controller (WNIC), fundamental phenomenon of evil twin attacks, and the detection algorithm of ET Detector.

[1] WLAN frequency band standards described in IEEE 802.11, for instance, 802.11g.

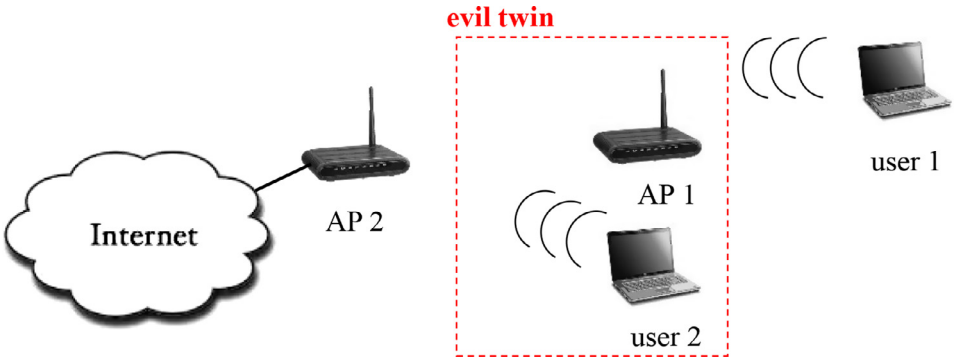

Fig. 1. WNIC' s view of an evil twin.

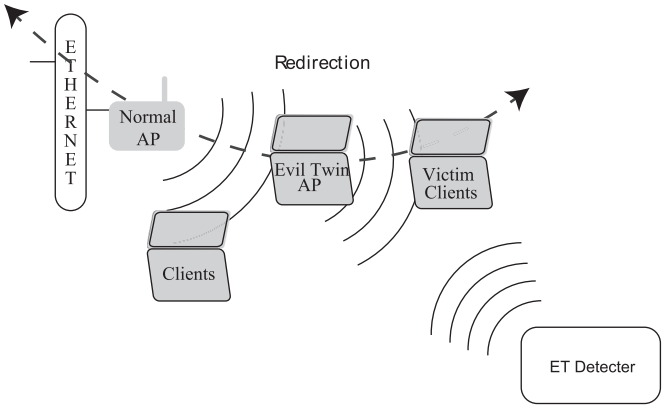

Fig. 2. Using ET detector to monitor the packet forwarding behavior of an evil twin.

3.1. Monitor mode

The monitor mode of a WNIC allows it to monitor all nearby wireless traffic. Unlike promiscuous mode, the monitor mode enables a WNIC to capture wireless packets without the need to associate it with an AP. Most of the WNICs and modern operating systems support monitor mode. In Windows, monitor mode is supported after Windows Vista and is controlled by NDIS (Microsoft Windows Network Driver Interface Specification) API. Microsoft provides a network monitor program called Microsoft Network Monitor to users to operate WNICs in a convenient way. Activating monitor mode in Unix OS family is simpler. There are some built-in network-related commands for this purpose.

Because an evil twin needs to use a good twin (legitimate AP) to connect to the Internet, an evil twin needs two wireless adaptors. One adaptor imitates a legal AP to induce users. This adaptor has an SSID so that a WNIC deems it as an AP. The other one is used to connect to the good twin; thus, it does not have an SSID and behaves like a laptop. Even though both an evil twin and its corresponding good twin have the same SSID, they have different BSSIDs (Basic Service Set ID), i.e. MAC addresses. As a result, as shown in Fig. 1, when an evil twin is forwarding packets, from the information contained in the wireless packets that a WNCI collects, an evil twin will be deemed as an AP and a laptop. In the following sections, we use an AP and a laptop to represent an evil twin.

3.2. ET detector

For a TCP connection between a victim and a server, an evil twin needs to forward corresponding packets between the victim and the good twin, because the evil twin uses the good twin to connect to the server. Therefore, packet forwarding is a fundamental phenomenon of an evil twin attack. ET detector was designed based on this unchangeable property to make it difficult for evil twins to evade ET detector' s detection.

As illustrated in Fig. 2, ET detector passively observes wireless packets. It monitors the network to detect whether an AP forwards wireless packets to detect whether an AP is an evil twin. Section 3.3 provides more detailed information. The layout of ET detector is shown in Fig 3. The system contains four components which are the progress controller, packet monitor, redirection detector, and AP record. The progress controller is the main component to coordinate other components in the system. Besides, it also controls the WNIC and shows the detection results to users. The packet monitor gets packets from a WNIC. After extracting useful information from the packets, the packet monitor stores the information in the AP record component. In the later stage, the redirection detector reads the records in AP record component and produces the detection results. The detection results are sent to the progress controller to determine whether the related AP is an evil twin.

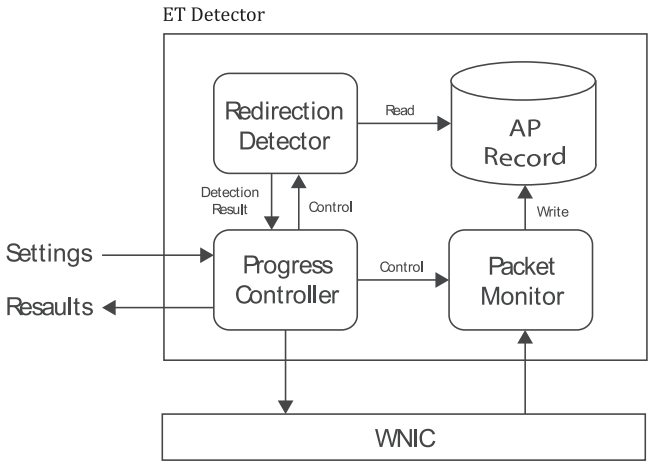

Fig. 3. Layout of ET detector.

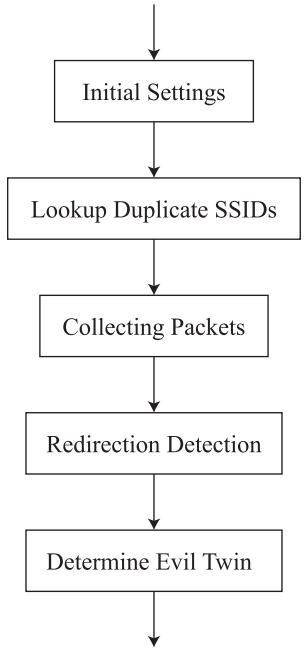

Fig. 4. Logical flow of ET detector.

ET detector has two detection mechanisms that run at the same time. It is the **default testing** and **secondary-device testing**. In the normal case, we use the default testing to discover evil twins. But in some particular circumstances, we may use an extra device to perform the secondary-device testing . Any device that has the Wi-Fi capacity can be the extra device. To use the secondary-device testing, an extra device is required to associate it with the AP shown by its system menu, and establishes a TCP connection to a specified IP address provided by ET detector. By doing so, ET detector is able to catch the evil twin even its user is the only user in a WLAN that has an evil twin.

3.3. Detection algorithm

The goal of ET detector is to detect the AP which shows packet forwarding behavior. An AP exhibiting packet forwarding behavior is deemed as an evil twin. A logical flow of our detection algorithm is given in Fig. 4. The algorithm contains five steps:

1. Initialize.
2. Lookup APs with duplicate SSIDs.
3. Monitor and capture packets.
4. Detect packet forwarding behavior.
5. Determine evil twin APs.

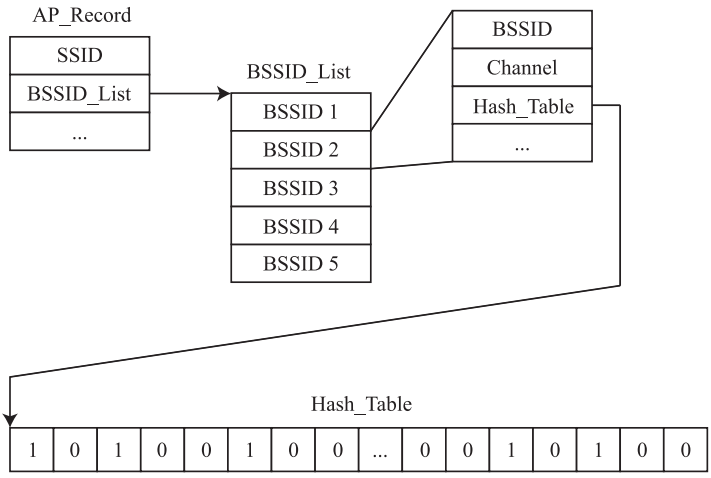

Fig. 5. Structure of AP_Record and Hash table.

Step 1: Initializing system. At the beginning, users will be prompted to select the WNIC driver which ET Detector will use. The selected WNIC will be requested to scan all available APs in current wireless network environment and records the scanning result. The scanning results include SSID, encryption type, BSSID[2], protocol, channel, etc. The BSSID is the MAC address of an AP and we can use it to distinguish APs. In this step, encrypted APs will be discarded from records, because it is illogically for an attacker to deploy an encrypted evil twin at a public place. After all, two APs with different encryption types will be shown as different APs even they have the same SSID under the RSSI policy of current operating systems. If an evil twin is encrypted, no user can associate his device with it, because the user does not have an account on the evil twin to log in. For a user who wants to use the secondary-device testing, ET detector provides him an IP address to use.

Step 2: Finding APs with duplicate SSIDs. By analyzing recoded AP information, ET detector further excludes APs which have unique SSIDs in the WLAN and lists the APs that have duplicate SSIDs to its users. Then users are prompted to choose an AP group with the same SSID for ET detector to analyze. The system discards un-chosen AP groups to save memory. The chosen APs are stored in *AP_Record* structures which contain the APs with the same SSID. Then, ET detector produces filter rules to sniff wireless TCP packets handled by these APs. The headers of TCP packet contain several fields that can provide reliable information which could be used to detect packet forwarding behavior. Besides, most sensitive user data are transmitted through TCP/IP connections.

Step 3: Capture packets via monitor node. In this step, ET detector starts the monitor mode of its WNIC to capture 802.11 WLAN packets in the WLAN. According to the filter rules produced in previous step, if the selected AP group contains various channels or protocols, the WNIC swaps between them at a fixed frequency. In order to improve performance, by default, our system configures this frequency as high as possible. Besides, a handle thread is executed simultaneously at this step to process packets captured by the WNIC. Each AP in the selected AP group has its own hash table and BSSID. Each hash table contains an array of buckets. Each hash bucket contains a value which is either 1 or 0. The initial value of each hash bucket is 0. Whenever ET detector receives a packet from the WNIC, it extracts the sequence number and the acknowledgment number from the packet. Then ET detector uses the extracted sequence number and acknowledgment number as a key to calculate a hash index. The hash index is used to choose a hash bucket inside the hash table of the AP that is the destination of the packet. The value stored in the chosen hash bucket is set to 1. The hash table structure is illustrated in Fig. 5. Inside Fig. 5, ***AP_Record*** is a structure contains common information about a group of APs that have the same SSID. An ***AP_Record*** contains a pointer which points to a ***BSSID_List*** structure. Each AP in the chosen AP group has an entry in *BSSID_List*. The *BSSID_List* entry for access point AP_i contains a pointer which points to the hash table of AP_i. In order to reduce memory space, a hash value can be calculated using modulo operation with a constant value.

Step 4: Detecting packet forwarding behavior. An evil twin forwards every packet it receives from a victim to the related good twin; hence, it does not change the sequence numbers and acknowledgment numbers in the TCP/IP headers of IP packets. ET detector unveils evil twins using this property. When ET detector sees a packet, it utilizes the sequence number and acknowledgment number of the packet as a hash key to calculate a hash index, h_i. Then it sets the value of the hash bucket with index h_i in the hash table of the destination AP to 1. After the above steps, ET detector checks whether the values of another hash buckets with index h_i in other APs' hash tables are also one. If such a hash bucket exists, it means that the AP whose hash table contains the hash bucket has received and forwarded the packet; hence, the AP is an evil twin. Fig. 6 gives an example. Assume the hash index

[2] Basic service set identification.

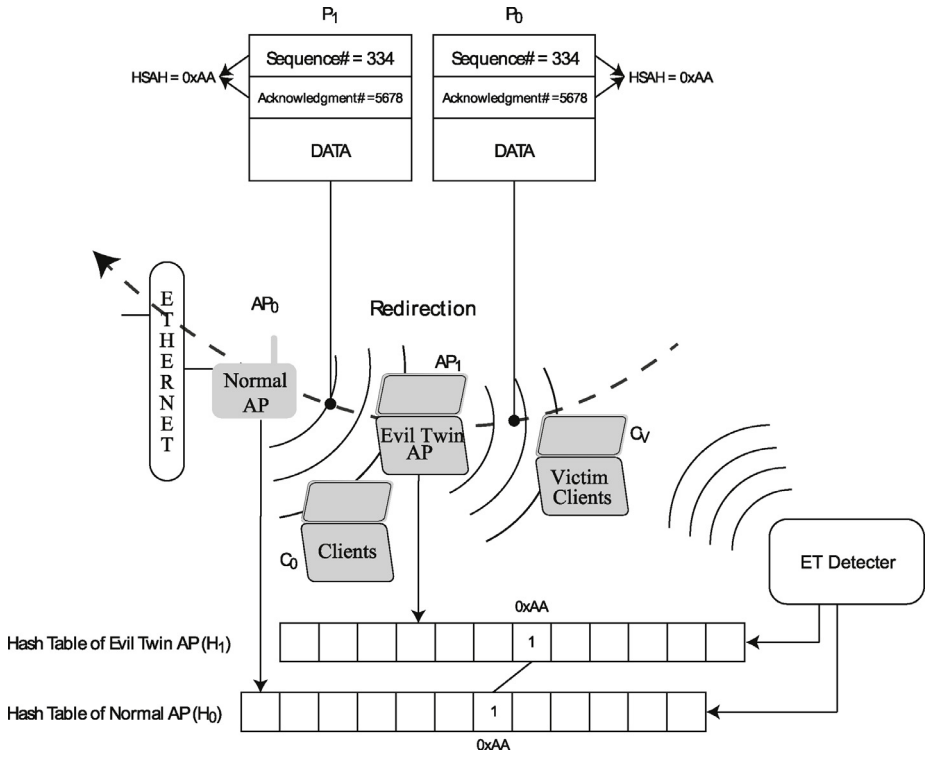

Fig. 6. Detection mechanism of ET detector.

of packet P_0 is 0xAA. The destination AP of the packet is AP_1. The packet was sent by victim client C_V and forwarded as packet P_1 by evil twin AP_1 to a good twin AP_0. When ET detector sees packet P_0, it sets the value of the hash bucket with index 0xAA as one. The hash bucket is inside the hash table H_1 of AP_1. Similarly, when ET detector sees packet P_1 whose destination AP is AP_0, it sets AP_0's hash bucket with index 0xAA to 1. AP_0's hash table is H_0. After checking H_1's hash bucket with index 0xAA, ET detector can infer that AP_1 has received the same packet and forwarded it to AP_0. Hence, it occurs a **hit condition** in hash table H_1 of AP_1 and AP_1 is forwarding packets to AP_0. As a result, AP_1 is an evil twin. In contrast, if ET detector sees a packet from client C_0 who is associated with AP_0, no hit condition will occur because AP_0 does not forward packets. While doing the above work, ET detector also checks the destination IP of each TCP packet to collect information for the secondary-device testing. Whenever a pre-defined destination IP, which is specified in Step 1, appears in two different packets that are sent to two different APs, ET detector also can conclude that there is a hit condition and the AP that receives the earlier shown packet is an evil twin.

Step 5: Determine evil twin and show up results. In the previous step, ET detector produces a detection result about packet forwarding behavior in a record. With this record, ET detector can determine which AP is an evil twin. Last ET detector prints out a warning message to users, if there is any AP inferred as an evil twin.

4. Evaluation

In this section, we utilize various experiments to evaluate the detection accuracy and efficiency of ET detector. We set up our experiments at a university campus with more than 10,000 students. We used a laptop equipped with two 802.11 network adapters to create an evil twin. The evil twin had the same SSID with its good twin which was a normal campus AP. The evil twin laptop had a 2.4 GHz Intel Core 2 Duo CPU and 4 GB memory. One of the laptop wireless cards was configured as the AP part of the evil twin. The evil twin connected to the Internet through the good twin. The evil twin was deployed right beside a user laptop; hence, it produced almost 100% RSSI. The network sharing option of the evil twin laptop was switched on in order to redirect packets. ET detector was installed on another laptop running Microsoft Windows 7 64-bit operating system with a wireless network card, a 2.4 GHz Intel Core 2 Duo CPU, and 4 GB memory. Besides, ET detector used Microsoft Network Monitor 3.4 to capture and analysis network traffic.

4.1. TCP/IP connection establishment pattern

To evaluate the detection accuracy of ET detector, we used two dedicated hosts to simulate normal users' client hosts and used these hosts to create TCP/IP connections to remote servers. In order to make the TCP/IP connection establishment patterns created by a client host similar to the patterns created by a normal user, we first used a sniffer to observe the traffic created by a host when

Table 1
Experimental results of detection accuracy using 10 rounds of experiments.

#	Hit count of the normal AP	Hit count of the evil twin
1	0	128
2	0	121
3	0	120
4	1	112
5	1	111
6	0	113
7	0	126
8	0	118
9	0	118
10	1	114

Table 2
Experimental results of ET detector effectiveness in different RSSIs.

RSSI to good AP = 80%–100%								
RSSI to evil AP #	90% HNA[a]	HET[b]	80% HNA[a]	HET[b]	50% HNA[a]	HET[b]	25% HNA[a]	HET[b]
1	4	436	1	607	0	71	4	216
2	1	548	0	483	1	213	5	165
3	1	442	1	361	1	315	1	18
4	0	559	0	590	0	271	0	8
5	1	562	5	486	0	176	0	17
6	1	483	0	609	1	221	0	47
7	0	567	4	435	1	188	1	29
8	1	484	1	275	0	72	0	4
9	2	450	0	137	0	147	1	48
10	4	732	1	407	0	68	0	0

[a] HNA means the hit count of the normal AP.
[b] HET means the hit count of the evil twin.

a user surfs Gmail, Facebook, Google, Twitter, and other popular services. And then we used the observed patterns to make TCP/IP connections to the following two websites, www.google.com and tw.yahoo.com. The results of monitoring Facebook, Gmail, and Twitter show that there is at least one TCP/IP connection established in 30 s, and at least five connections are established in one minute when surfing these websites. In later section, we use the above results to make our experiments.

4.2. Evaluation of detection accuracy

We set up our experiments in a university campus with more than 10,000 students. According to the observations discussed in Section 4.1, we wrote a python program in our client hosts. The program made five TCP/IP connections to a remote website server specified by us per minute. The good twin and the evil twin which were used in our experiments utilized same channel to transmit packets.

In our evaluation, two notebooks were used to simulate two normal users. One notebook chose the good twin to connect to website www.google.com , the other chose the evil twin to connect to website tw.yahoo.com. These two client hosts continued connecting to the chosen websites five times per minute during the experiment.

We manually operated our system 10 rounds. Each round lasted for one minute. Table 1 shows the result of hit counts. The hit count of an AP is increased by one if there is a hit condition occurred in the hash table of that AP. Hence, a hit count of an AP represents the number of occurrences of hit conditions occurs in that AP, which also means the count of times that an AP has been found to forward packets. Table 1 shows that the variation of hit counts between two APs is significant enough to determine the evil twin.

We also evaluate the detection accuracy of ET detector with different RSSI values. The results are shown in Table 2. The detection accuracy decreases seriously if the RSSI value is lower than or equal to 25%. However, an AP with 25% RSSI value is too low to provide stable Wi-Fi service. A user connects to an AP with RSSI value that is lower than or equal to 25% would disconnect frequently, thus the service is unusable. The evaluation result shows that in a normal case ET detector could detect evil twins in different RSSI values.

Theoretically, the hit count of a normal AP should be zero. However, our experimental results show that sometimes the hit count of our normal AP is not 0. After further investigation, we found that this phenomenon is produced by the retransmission of Wi-Fi packets during our detection time. For instance, after a packet is forwarded to a normal AP by an evil twin, a retransmitted version of that packet may be sent to the evil twin later. When the above situation occurs, the his count of the normal AP will be increased by one. However, the above situation does not happen frequently. Hence, if ET detector just chooses APs with large hit counts, this noise could be filtered out.

Table 3
Experimental results of time efficiency when experiments are lasted for different lengths of duration.

Time	Hit count of the normal AP	Hit count of the evil twin
15 s	0	17
30 s	0	50
1 min	0	115
3 min	0	326
10 min	1	1055

4.3. Time efficiency

In this section, we evaluate the time efficiency of our algorithm. ET detector was measured with various lengths of detection duration, the result is shown in Table 3. The number of captured packets increases when the detection period increases. However, the hit count of an normal AP is always much smaller than the hit count of an evil twin. Therefore, a short detection period is enough for ET detector to make accurate detection.

5. Discussion

The section explains the limitation and weakness of our system and describes our future work.

5.1. Limitation

In our detection mechanism, there must be at least one client who has associates his device with either the normal AP or the evil twin. If the client is associated with an evil twin, ET detector can detect the packet forwarding behavior of the evil twin and identify it as an malicious one. In contrast, if the client associates his device with a normal AP, ET detector can find that the AP does not forward packets. Hence, it is safe to use that AP. However, if an ET detector user is the only user in a WLAN, the above mechanism is not able to detect evil twins. In spite of our belief that an attacker is not tend to launch evil twin attacks in a wireless network environment without users. We cannot rule out this situation. Under this circumstance, a user can use an **secondary device** which is a wireless device such as smart phone, tablet, or anything else to assist the detection. The user associates the secondary device with the target AP first. Then he can use ET detector to make his detection.

ET detector is designed to detect evil twins; hence, it cannot be used to solve the *rouge AP problem* which is out the scope of the *evil twin problem*. In a rouge AP attack, an attacker may use a 3G/4G mobile network or other approaches to connect to the Internet. When an attacker launches a rogue AP attack, the forwarded packets are transmitted through a different channel; thus, ET detector cannot obtain the information it needs to make the detection. As a result, ET detector cannot detect such kinds of rouge APs.

Finally, ET detector requires the supporting of monitor mode. To run the system, monitor mode must be supported by WNIC in client's laptops. According to the list provided in [25], most common WNIC chipsets in the market support monitor mode for Linux. After our investigation, all of them support the monitor mode for Windows. So, there is a very low possibility that a laptop's WNIC does not support monitor mode.

5.2. Analysis

An attacker is not easy to evade our system in normal cases, because an evil twin creator usually only sets the evil twin sharing function or packet redirection function. As a result, most of the TCP/IP header fields of a forwarded TCP/IP packet will not be changed.

But an advanced attacker who knows our detection algorithm may re-generate a forwarded packet with a different sequence number and acknowledgment number by shifting the acknowledgment number and the sequence number with a fixed number at an evil twin. Under this situation, the default testing of ET detector is unable to detect the redirect behavior of the evil twin. To solve this problem, a user can use the secondary-device testing described in Section 3.2 to detect the evil twin.

5.3. Future work

Currently, our system is implemented on operating systems which are common used in laptop computers. Since the popularization of mobile devices such as smart phones and tablets, more and more users surf the Internet in public space without laptops. In our future work, we plan to port this system to popular mobile operating systems like Android or iOS to protect more users from evil twin attacks. Furthermore, we plan to extend this mechanism to detect not only evil twins but rogue APs which have different SSIDs.

6. Conclusion

An evil twin attack could be easily launched, especially in public spaces. Due to the RSSI policy of modern operating systems, users do not have enough information to protect themselves from evil twin attacks. In this paper, we propose a lightweight

client-based technique to detect evil twin attacks. We implemented a prototype system, ET detector, on Windows 7. ET detector could detect evil twin attacks accurately and efficiently, as shown by experiments. Through ET detector, users are able to uncover evil twins in public space and protect themselves timely.

This paper proposes an evil twin detecting solution on client side. It makes the following contributions:

- Proposes the first idea of detecting evil twin by operating the WNIC in monitor mode. It has many advantages as we described above.
- This concept has been implemented as a prototype system, ET detector (Lightweight Evil Twin Detector of Wireless Network), on the Windows 7 operating system. Windows 7 is widely used on laptop nowadays, and the system proves the feasibility and revealing the high compatible characteristic.

References

[1] GopinathKN.. Be aware of 5 Wi-Fi security threats. 2010. http://fanaticmedia.com/infosecurity/archive/Feb10/5-Wireless-Security-Threats_v2.htm.
[2] PhiferL.. Top ten Wi-Fi security threats. 2010. http://www.esecurityplanet.com/views/article.php/3869221/Top-Ten-WiFi-Security-Threats.htm.
[3] Wikipedia. Rogue access point. 2011. http://en.wikipedia.org/wiki/Rogue_access_point.
[4] Wikipedia. Received signal strength indication. 2012. http://en.wikipedia.org/wiki/Received_Signal_Strength_Indication.
[5] Bellardo J, Savage S. 802.11 denial-of-service attacks: Real vulnerabilities and practical solutions. In: Proceedings of the Twelfth Conference on USENIX Security Symposium; 2003.
[6] Cisco. Rogue detection under unified wireless networks. 2007. http://www.cisco.com/en/US/tech/tk722/tk809/technologies_white_paper09186a0080722d8c.shtml.
[7] AirMagnet. (2007). Description of AIRMagnet. http://www.airmagnet.com.
[8] NetworkStumbler. (2005). http://www.netstumbler.com.
[9] Wisentry. (2003). Wireless access point detection system. http://www.wimetrics.com/Products/WAPD.htm.
[10] The inSSIDer software. (2005). http://www.metageek.net/products/inssider.
[11] AirWave. (n.d.). AirWave Management Platform. http://www.airwave.com.
[12] Wavelink. (n.d.). Avalanche MC. http://www.wavelink.com.
[13] Proxim Wireless. (2004). Rogue access point detection: Automatically detect and manage wireless threats to your network.http://www.proxim.com.
[14] Bahl P, Chandra R, Padhye J, Ravindranath L, Singh M, Wolman A, et al. Enhancing the security of corporate Wi-Fi networks using DAIR. In: Proceedings of the Fourth International Conference on Mobile Systems, Applications and Services, MobiSys '06. New York, NY, USA: ACM; 2006. p. 1–14. ISBN 1-59593-195-3. doi:10.1145/1134680.1134682. http://doi.acm.org/10.1145/1134680.1134682
[15] Wei W, Suh K, Wang B, Gu Y, Kurose J, Towsley D. Passive online rogue access point detection using sequential hypothesis testing with TCP ACK-Pairs. In: Proceedings of the Seventh ACM SIGCOMM Conference on Internet Measurement, IMC '07. New York, NY, USA: ACM; 2007. p. 365–78. ISBN 978-1-59593-908-1. doi:10.1145/1298306.1298357. http://doi.acm.org/10.1145/1298306.1298357
[16] Yin H, Chen G, Wang J. Detecting protected layer-3 rogue APs. In: Proceedings of the Fourth International Conference on Broadband Communications, Networks and Systems, 2007. BROADNETS 2007.; 2007. p. 449–58. doi:10.1109/BROADNETS.2007.4550468.
[17] Wei W, Wang B, Zhang C, Kurose J, Towsley D. Classification of access network types: Ethernet, wireless LAN, ADSL, cable modem or dialup? Comput Netw 2008;52(17):3205–17. doi:10.1016/j.comnet.2008.08.018. http://dx.doi.org/10.1016/j.comnet.2008.08.018
[18] Shetty S, Song M, Ma L. Rogue access point detection by analyzing network traffic characteristics. In: Proceedings of the 2007 IEEE Military Communications Conference; 2007. p. 1–7. doi:10.1109/MILCOM.2007.4455018.
[19] Wei W, Jaiswal S, Kurose J, Towsley D. Identifying 802.11 traffic from passive measurements using iterative Bayesian inference. In: Proceedings of the Twenty-Fifth IEEE International Conference on Computer Communications; 2006. p. 1–12. doi:10.1109/INFOCOM.2006.291.
[20] Baiamonte V, Papagiannaki K, Iannaccone G. Detecting 802.11 wireless hosts from remote passive observations. In: Proceedings of the Sixth International IFIP-TC6 Conference on AD HOC and sensor networks, wireless networks, next generation internet, NETWORKING'07. Berlin, Heidelberg: Springer-Verlag; 2007. p. 356–67. ISBN 978-3-540-72605-0. http://dl.acm.org/citation.cfm?id=1772322.1772361
[21] Watkins L, Beyah R, Corbett C. A passive approach to rogue access point detection. In: Proceedings of the 2007 IEEE Global Telecommunications Conference; 2007. p. 355–60. doi:10.1109/GLOCOM.2007.73.
[22] Song Y, Yang C, Gu G. Who is peeping at your passwords at Starbucks? – To catch an evil twin access point. In: Proceedings of the 2010 IEEE/IFIP International Conference on Dependable Systems and Networks (DSN); 2010. p. 323–32. doi:10.1109/DSN.2010.5544302.
[23] Panch A, Singh SK. A novel approach for evil twin or rogue ap mitigation in wireless environment. Int J Secur Appl 2010;4(4):33–8.
[24] Nikbakhsh S, Manaf A, Zamani M, Janbeglou M. A novel approach for rogue access point detection on the client-side . In: Proceedings of the Twenty-Sixth International Conference on Advanced Information Networking and Applications Workshops (WAINA); 2012. p. 684–7. doi:10.1109/WAINA.2012.108.
[25] Aircrack-ng. Determine the driver. 2011. http://www.aircrack-ng.org/doku.php?id=compatibility_drivers&DokuWiki=01c89160d88dc29d91c1546165ca8089#determine_the_driver.

Fu-Hau Hsu received his Ph.D. degree in the Department of Computer Science from Stony Brook University, New York, USA in 2004. He is an associate professor at National Central University and has had an appointment in the Department of Computer Science and Information Engineering since August 2005. He is affiliated with the Advanced Defense Lab and the Wireless Network and Multimedia Lab.

Chuan-Sheng Wang is a Ph.D. student in the Department of Computer Science and Information Engineering of National Central University. He received his M.S. degree in Computer Science and Information engineering from National Central University, Taoyuan, Taiwan, in 2010. His research areas include network security, operating system and malware analysis.

Yu-Liang Hsu is a Ph.D. student in the Department of Computer Science and Information Engineering of National Central University. He received the M.S. degree in Computer Information Science from Soochow University, Taipei, Taiwan,in 2007. His research interests include network security, wireless security, and system security.

Yung-Pin Cheng received his Ph.D. in Computer Science from Purdue University, West Lafayette, U.S. 2000. His major research interest is in software engineering. He has published his papers in some important software engineering conference and journals, including SIGSOFT FSE, SIGSOFT ISSTA. His research topics include automatic software verification, software visualization, object-oriented design and analysis.

Yu-Hsiang Hsneh received his M.S. degree in Computer Science and Information Engineering from National Central University, Taoyuan, Taiwan, in 2012. His research areas include operating system and wireless network security.