Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

IEEE 802.11 user fingerprinting and its applications for intrusion detection

Daisuke Takahashi [a], Yang Xiao [a,*], Yan Zhang [b], Periklis Chatzimisios [c], Hsiao-Hwa Chen [d]

[a] University of Alabama, USA
[b] Simula Research Laboratory, Norway
[c] University of Macedonia, Greece
[d] Department of Engineering Science, National Cheng Kung University, Taiwan

ARTICLE INFO

Keywords:
802.11
Fingerprinting
Intrusion detection

ABSTRACT

Easy associations with wireless access points (APs) give users temporal and quick access to the Internet. It needs only a few seconds to take their machines to hotspots and do a little configuration in order to have Internet access. However, this portability becomes a double-edged sword for ignorant network users. Network protocol analyzers are typically developed for network performance analysis. Nonetheless, they can also be used to reveal user's privacy by classifying network traffic. Some characteristics in IEEE 802.11 traffic particularly help identify users. Like actual human fingerprints, there are also unique traffic characteristics for each network user. They are called network user fingerprints, by tracking which more than half of network users can be connected to their traffic even with medium access control (MAC) layer pseudonyms. On the other hand, the concept of network user fingerprint is likely to be a powerful tool for intrusion detection and computer/digital forensics. As with actual criminal investigations, comparison of sampling data to training data may increase confidence in criminal specification. This article focuses on a survey on a user fingerprinting technique of IEEE 802.11 wireless LAN traffic. We also summarize some of the researches on IEEE 802.11 network characteristic analysis to figure out rogue APs and MAC protocol misbehaviors.

1. Introduction

IEEE 802.11 Wireless LAN (Wi-Fi) enables us easy Internet connections. A fundamental unit of a Wi-Fi network is called a Basic Service Set (BSS) and consists of a wireless AP and several wireless stations, such as laptop PCs or PDAs [1]. Although a wireless LAN provides many benefits such as easy deployment of wireless stations, nevertheless, it is easily exposed to many security problems. For example, due to the nature of the wireless medium, radio signals are vulnerable to interception of eavesdroppers. Apparently, some information, such as Medium Access Control (MAC) address, helps adversaries to identify network devices. More precisely, because MAC addresses are uniquely assigned to network interface cards (NICs) by manufacturers, if adversaries could associate them with individuals, identifying users from the MAC addresses is not a difficult task [1]. Basically, the MAC address is persistent for the entire life of a device, which is just like a unique social security number for an individual [1]. Usually, the MAC address is clear during a data communication.

Although IEEE 802.11 provides security protection mechanisms, such as Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access (WPA), the protection is not extended to the MAC address anonymity [2]. Basically, user and service

[*] Corresponding address: Department of Computer Science, The University of Alabama, 101 Houser Hall, Box 870290, Tuscaloosa, AL 35487-0290, USA.
E-mail addresses: yangxiao@ieee.org (Y. Xiao), yanzhang@ieee.org (Y. Zhang), hshwchen@ieee.org (H.-H. Chen).

0898-1221/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2010.01.002

authentication, as well as data confidentiality and integrity, are only covered by WEP (which is already deprecated by IEEE for vulnerability issues) and WPA protocols. Thus, in order to embody the MAC address anonymity, several works suggested virtual MAC address utilization. That is, to conceal the MAC address during a data communication transaction, each wireless station should have a different MAC address at a different session, rather than persisting only one MAC address for its whole life [2].

However, work in [2] demonstrated that merely employment of pseudonyms for MAC address is insufficient to hide user's identity from adversaries in a Wi-Fi network. Although pseudonyms can help preventing explicit associations of the MAC address with individuals, network packets sent by wireless stations still contain information that can be used to expose their location privacy. According to [2], these characteristics are called implicit identifiers. Examples of the implicit identifiers include the IP address of a service that a user frequently accesses, clock skew exposed by TCP timestamps and clickprints by users when they browse the Web [2]. For instance, unlike the MAC address, the IP address is not persistent for host devices. However, the IP requests sent by wireless stations may indicate some user's preferences although their own IP addresses cannot tell much about the senders. Network users may have their bookmarked websites and preferred e-mail servers. By use of implicit identifiers, the majority of users could be tracked by adversaries with nearly 90% accuracy [2]. Moreover, these implicit identifiers can be easily captured by network protocol analyzers, such as Wireshark and tcpdump, and for the worse, this kind of software can be easily obtained commercially. Additionally, wireless geographic logging engines, such as WiGLE.net, help retrieve the geographical information of wireless access points (APs) [3]. Thus, it is relatively easy to reveal the geographical location of each network device from that of APs [2].

On the other hand, an analysis of network traffic is also used to detect rogue APs or unauthorized APs as well as MAC protocol misbehavior. Rogue APs are usually installed by ignorant employees in order to extend their company benefits but with the minimum security settings. Accordingly, their actions often generate security vulnerabilities allowing hackers to compromise the company's systems. Although rogue APs are currently detected mostly by rudimentary approaches, authors in [4] propose a new approach utilizing a network traffic analysis.

This article provides a comprehensive survey on IEEE 802.11 user fingerprinting and its applications. The rest of the article is outlined as follows. We first review the Ethernet and Wi-Fi MAC layer protocols, and then we go through network traffic characteristics peculiar to the Wi-Fi networks and some issues particularly created by utilization of radio signals as virtual links. We then investigate network monitoring schemes that usually are employed for assessing network performance. Subsequently, we present certain techniques for analyzing network traffic that could be applied to detect rogue APs. We further discuss issues related to MAC protocol misbehavior and then study DOMINNO in [12], a scheme developed for detection of IEEE 802.11 greedy users, followed by the conclusion of the article.

2. Ethernet and IEEE 802.11 Wireless LAN

This section explores both IEEE 802.11 wireless LAN (Wi-Fi) MAC and Ethernet's multiple access protocols to figure out what differences potentially rise in use of these two different network technologies.

In the mid-1970, Bob Metcalfe and David Boggs developed the original Ethernet architecture, which was the first ever high speed LAN system [1]. One advantage of Ethernet compared to Wi-Fi is that a network adapter is able to sense signal energy from other adapters which are sending frames [1]. This feature enables collision detection, and thus Ethernet employs the carrier sense multiple access with collision detection (CSMA/CD) protocol. In the CSMA/CD protocol, when channel is free, a network adapter can immediately start sending frames. On the other hand, if it is busy, an adapter just waits until no more signal energy can be sensed on the channel and then starts transmitting. However, if a network adapter, during its frame transmission, also senses signal energy from another adapter, it then stops transmitting, sends a jam signal into the channel and gets into an exponential backoff phase [1].

Obviously, in a Wi-Fi network, wireless stations and an AP (AP) cannot be interconnected with cables. Wireless signals basically propagate through the air between terminals and an AP making virtual links while in turn suffering interference or jamming which causes more difficult issues to be handled, such as higher bit-error rates, than Ethernet [1]. Typically, two disadvantages compared to Ethernet must be taken into account to implement an efficient multiple access protocol in Wi-Fi [1]: (1) it is too costly for adapters to have the functionality to send and receive signals at the same time; (2) it is impossible for adapters to detect signals from hidden terminals.

Because of these two difficulties, Wi-Fi cannot employ a collision detection (CD) scheme, but instead it employs a collision avoidance (CA) scheme. Compared to CSMA/CD, one noticeable difference of CSMA/CA is that a station gets into a random backoff phase whenever its network adapter senses channel busy or does not receive any acknowledgement in response to its sending frame [1]. On the other hand, in Ethernet, a station only suffers random backoffs when its network adapter experiences signal energy during its frame transmission (collision) [1].

Furthermore, in order to avoid high bit-errors, Wi-Fi also employs link layer acknowledgement to check whether each frame can reach the destination and remain intact. Hence, a sender that does not receive an ACK from the destination in response to its transmitting frame will start over the same transmission again based on CSMA/CA.

Moreover, to cope with the hidden terminal problem, Wi-Fi utilizes auxiliary frames, called Request to Send (RTS) and Clear to Send (CTS) control frames. In general, a RTS frame is sent to an access point when a wireless station wishes to transmit a DATA frame to the destination. In response to a RTS frame, when the requested channel is not busy, a CTS frame is broadcasted by the AP as an acknowledgement to the sender. A CTS frame additionally plays another important role by restricting data transmissions from other stations [1]. This collision avoidance mechanism along with CSMA/CA forms an important part of the IEEE Distributed Coordination Function (DCF) method.

However, this link layer acknowledgement and RTS/CTS frame exchange potentially delays network traffic, where it takes longer time to send entire data to destination. Authors in [4] focused on these delays to distinct network traffic out of rogue APs from wired terminals.

3. More about IEEE 802.11 wireless LANs

Accordingly, it is obvious that traffic in a computer network consisting of only wired links (e.g., Ethernet) and that including wireless connectivity (e.g., Wi-Fi) have different characteristics. By examining inter-packet spacing of network traffic, for example, work in [4] successfully discriminates networks that include wireless links and those not including them in order to detect rogue APs. Thus, in the next two subsections, we summarize peculiar network traffic characteristics and problems of Wi-Fi in comparison to Ethernet.

3.1. Problems in IEEE 802.11

Apparently, it is harder to maintain consistent, robust and secure communications in an IEEE 802.11 wireless LAN (Wi-Fi) than in Ethernet because of the nature of the wireless medium. In other words, several problems exist peculiarly in use of Wi-Fi: connectivity problems, performance problems and network security [5].

In utilization of radio signals, Wi-Fi cannot provide network users with consistent access to the Internet. For example, even in a small building that can be covered by an AP, such as a network cafe, some spaces are hidden from radio signals by obstacles. Such spaces are called “dead spots” or “RF holes”, and typically this problem is resolved by deploying more APs [5].

Accordingly, since associations between APs and wireless stations are made by virtual wires in Wi-Fi, they are physically inconsistent and vulnerable to interference in surrounding devices, such as a microwave oven and cordless phones as well as other APs operating nearby. Additionally, these virtual wires easily allow radio signals to take different paths to a destination, causing multipath propagation problem [6]. More precisely, while some radio signals head directly to the destination, others may bounce back from obstacles and suffer longer delays to the destination. The previously described characteristics of IEEE 802.11 apparently increase retransmissions, resulting in degradation of communication performance.

With respect to network security, one of the problems to be considered is installation of unauthorized APs or rogue APs [5]. Unsecured rogue APs are likely to be security holes for enterprise networks and could provide access to unauthorized users. Recently, problems caused by rogue APs have been very common, and installations of them are typically done by ignorant enterprise employees. In practice, detecting rogue APs, though more efficient ways have been developed, still counts on a very rudimentary way, where equipped IT personnel walk around the buildings catching beacons from wireless APs and identifying their MAC addresses.

On the other hand, for wireless network users, accessing unsecured APs that are not protected by passwords may raise a threat, nicknamed as “evil twins”. Work in [7] reports that evil twins are dummy APs that are intentionally deployed by adversaries targeting to intercept sensitive data from associating wireless stations. Thus, during associations with evil twins, all transmitting packets could be logged and the ones without encryption could be analyzed thereafter. Furthermore, evil twins often send stronger signals than legitimate APs in order to grab wireless connections from the legitimate ones [7]. However, even with encryption and pseudonyms, attackers with expertise can read user information and distinguish them from others by utilization of fingerprinting techniques (i.e., identify the sender of the packets) [2].

3.2. IEEE 802.11 traffic characterization

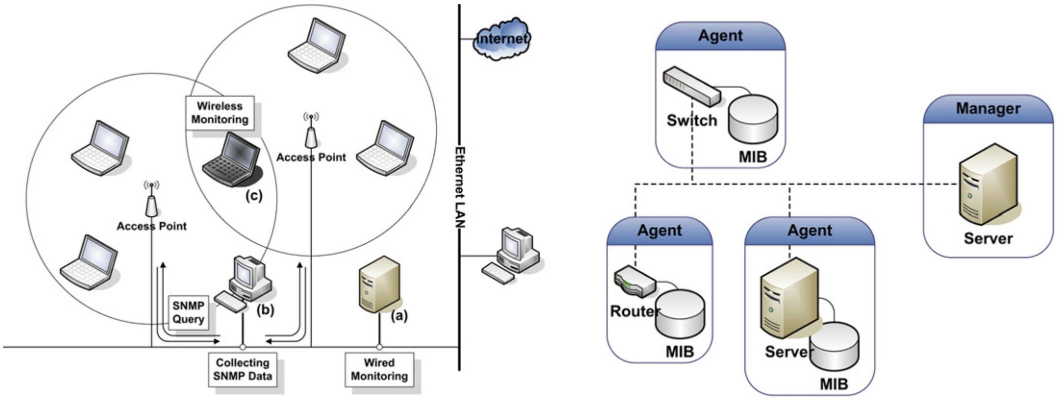
Especially in a hotspot setting, IEEE 802.11 traffic may suffer plenty of retransmissions that can be used as a metric to distinguish Wi-Fi from a wired network. Authors in [8] experimentally demonstrated that Wi-Fi suffers high rate of retransmissions due to channel contention and interferences. In general, retransmission time of Wi-Fi in a hotspot setting occupies 28% of all data transmissions and 46% of data transmission time [8]. In addition to retransmissions, there is extra overhead originated from PHY and MAC headers as well as control frame transmissions, such as RTS and CTS. Thus, in practice, only 40% of time is available for transmitting data packets and the remaining time is consumed in retransmission, acknowledgement and management traffic (as indicated in Table 1 [8]).

According to Table 1, more than half of all transmitted frames are control and management frames, whereas they usually consist of fewer bits, such that in terms of transferred bits they occupy merely less than 10% of the total bits. However, since 802.11 sends control and management frames in a low transmission rate (usually 1 Mbps), utilization of the medium by these frames rises more than the occupancy of the frames.

Moreover, the 802.11 retransmissions severely affect its performance. From Table 1, 28% of the data frames are retransmissions. As a result, nearly half of the data frame airtime is spent by these frames. Authors in [8] mentioned that basically two factors increase a retransmission rate: signal strength and contention level. According to [8], retransmissions in Wi-Fi are proportional to the contention level and inversely related to the signal strength.

Table 1
Network traffic characteristics of IEEE 802.11 wireless LAN (Wi-Fi).

Frame type and subtype	Airtime (s)	Bits (MB)	Frames (1000s)	Avg. rate (Mbps)
Data	6802	1884	5 540	6.46
Originals	3616	1276	3 988	7.30
Retransmits	3185	608	1 552	4.31
Control	1418	74	5 442	1.89
Ack.	1332	69	5 135	1.90
RTS	42	3	142	1.69
CTS	40	2	155	1.75
PS poll	2	0	10	1.60
Management	878	82	1 098	1.12
Assoc. req.	1	0	2	1.42
Assoc. res.	1	0	3	1.08
Authentication	6	0	13	1.13
Beacon frame	412	39	428	1.00
Deauth.	0	0	0	1.30
Dissassoc.	6	0.40	13 794	1.00
Probe req.	177	16.07	333 707	1.35
Probe res.	270	25.44	296 250	1.00
Reassoc. req.	0	0.03	2 727	1.00
Reassoc. res.	0	0.03	621	1.00
Totals	9098	2040	12 080	3.92



(a) Wireless network traffic monitoring methodologies.

(b) SNMP architecture model.

Fig. 1. Wireless monitoring and SNMP.

4. Monitoring methodology

In this section, we investigate wireless network traffic monitoring methodologies. Wireless network traffic measurements are typically conducted for wireless network diagnosis and to assess the network performance. Traditionally, wireless network traffic is either monitored from wired vantage points (i.e., wired monitoring) or by utilizing Simple Network Management Protocol (SNMP) statistics [9]. However, these two schemes are not sufficient for detection of instantaneous wireless medium characteristics, such as PHY/MAC in Wi-Fi [9]. These instantaneous characteristics usually contain rich information of network traffic. Thus, instead, work in [9] employs wireless monitoring, in which sniffers are connected at wireless vantage points measuring wireless network traffic in order to identify the characteristics that the previous two monitoring schemes may miss. Work in [30,31] also considers wireless monitoring. In the next two subsections, we summarize three methodologies for evaluating wireless network traffic.

4.1. Wired monitoring and SNMP statistics

In the wired monitoring, sniffers are connected directly to the wired portion of a network as shown in Fig. 1(a) [9]. Thus, sniffers capture data and control packets that flow through network links and usually analyze them from the header information. Under wired monitoring, Fig. 1(a) illustrates a sniffer that is connected directly to a wired portion of a LAN (a). For SNMP querying, the figure depicts a management machine that is connected to a wired portion of a LAN (b), and periodically sends SNMP queries, collects responses from APs and obtains statistics. On the other hand, in wireless monitoring, a sniffer is deployed in a range of radio signals from APs (c), and collects PHY/MAC information directly from them.

The SNMP protocol could be used to monitor and maintain the performance of all network devices so that a network comprising the devices can perform efficiently. A simple SNMP architecture is represented in Fig. 1(b) and consists of a manager application running on a management server as well as agent applications running on managed devices [9]. Typically, managed devices include host machines, wireless APs, various network equipment (such as routers, switches, hubs), printers and UPS. The manager application periodically requests status information from each agent by checking their Management Information Base (MIB) in order to make sure that network devices work well. Furthermore, whenever conditions in MIB change or an important event takes place, agents notify the manager by sending a trap message.

However, since SNMP is subject to periodical polls of device's state information by a local manager, the results are not sensitive to instantaneous conditional changes. Typically, this interval takes one to five minutes. Thus, SNMP summaries hardly reflect the instantaneous wireless medium characteristics, such as PHY/MAC in IEEE 802.11. Moreover, the current MIBs of SNMP for wireless APs rarely support device's MAC level behaviors, and they are very limited.

4.2. Wireless monitoring

In the wireless monitoring, sniffers are deployed in a Wi-Fi jungle or within radio range of multiple wireless APs. In general, network traffic captured in this way discloses rich information about the physical and MAC layers [9].

One advantage of the wireless monitoring is easy installation of devices into networks. Let us consider IEEE 802.11 wireless LAN (Wi-Fi). Wireless devices only require to be associated with one of surrounding APs via a particular channel before starting data communication with subnets. This creates virtual links between wireless devices and APs. Thus, devices of the wireless monitoring can work independently of target networks.

Another advantage of the wireless monitoring is that unlike the wired monitoring, the wireless monitoring is quite sensitive to physical information, such as wireless medium itself [9]. It also helps analyze physical and link layer header information. Accordingly, it reveals signal strength, noise level and data rate for each packet as well as IEEE 802.11 type and control fields that are derived by these headers [9,30]. Moreover, error rates and throughput performance can be also determined from physical layer information [9].

On the other hand, sniffers of wireless monitoring should be portable to move smoothly, and thus they have limitations in their disk sizes and processing power [9]. Furthermore, there may be dead spots or RF holes to prevent consistent monitoring within a target site. For this reason, the efficient placement of sniffers in a Wi-Fi jungle is another issue [9]. Finally, optimization of the placement of sniffers should target to reduce cost of a monitoring system.

5. Unauthorized wireless AP detection

As we already mentioned, security vulnerabilities caused by rogue APs are very common in almost every enterprise. Although rogue APs usually are incautiously installed by their own employees, there are many cases that are intentionally deployed by malicious hackers trying to collect confidential and critical data from enterprises. This section briefly reviews a mechanism of the rogue AP problem and presents the main detection schemes described in the literature. Moreover, a classification of rogue APs is also presented.

5.1. Rogue AP classification

According to the work reported in [10], rogue APs are classified into four categories: improperly configured, unauthorized, phishing and compromised APs.

Several reasons are considered for legitimate APs to turn into improperly configured APs. For example, in an enterprise, a network administrator due to the pressure for early installation of wireless LANs is hardly given sufficient time for careful implementation. Consequently, although he/she finally finishes deploying all APs in a facility, their security settings appear to be poor and rarely integrate similar security protocols to wired networks, such as wired equivalency protocol (WEP) [4]. Therefore, these unauthorized APs apparently form a security hole for the whole enterprise. It is also possible that a network administrator does not have enough knowledge of network security to set up APs properly [10]. Moreover, driver software malfunctions or updates may also cause security vulnerabilities [10]. Furthermore, ad hoc mode of a wireless station without strong security measures may enable attackers to intrude a connected network via this station when it utilizes both wired and wireless interfaces [10].

In most cases, unauthorized APs are installed without malicious intent and usually appear in large organizations with many employees [10]. For instance, in an enterprise that hardly employs a wireless intrusion system because of a monetary limitation, employees easily and incautiously connect cheap APs to the network [4,10]. However, these employees are usually not network administration or security gurus, and therefore unsecured APs consequently form a large security hole to malicious hackers. Accordingly, the security hole potentially exposes confidential data and gives hackers the opportunity to compromise enterprise assets via the network [10].

On the contrary, phishing APs are deployed outside of an enterprise by attackers who try to steal employee's usernames and passwords [10]. Phishing APs, also called "evil twins", always pretend to be legitimate APs in the enterprise and, thus,

Table 2
Classification of rogue APs and possible scenarios.

AP class	Possible scenario
Improper configured	Insufficient security knowledge; faulty driver; physically defective; multiple network cards
Unauthorized	Connected to internal LAN without permission; External neighborhood AP
Phishing	Fabricated by adversary
Compromised	Disclosure of security credentials

they broadcast beacon frames which overhear from legitimate APs. From phishing APs, attackers typically initiate man-in-the-middle (MITM) attacks [10]. Therefore, once an employee tries to connect to a company network via a phishing AP, all communication traffic is logged and this authenticating information could be disclosed by the attacker.

Finally, legitimate APs are often cracked by WEP/WPA-PSK key cracking tools, such as Aircrack-ng, resulting in compromised APs [10]. By using these broadly available key cracking tools, even a network novice can break encryption of WEP/WPA-PSK in Wi-Fi [10]. With stolen secret keys, legitimate APs with the same credentials are not obstacles for attackers anymore. Table 2 summarizes a classification of rogue APs [10].

5.2. Current rogue AP detection approaches

Currently, detection of rogue APs counts on very rudimental ways. A reasonable and also robust approach is to employ equipped IT personnel walking around through entire buildings to search for signals or beacons of rogue APs [4]. Obviously, this approach consumes a great deal of time as well as money and cannot be conducted on a regular basis. In some cases by following similar approaches we can enhance the method. More specifically, instead of allowing IT personnel roaming entire buildings, they deploy specified sensors that periodically send back unauthorized signals or beacons of wireless APs to a central location. Although this is a more sophisticated version of the previous approach, it is still costly and inefficient when wireless APs employ multiple frequency bands, i.e., 802.11a (in 5 GHz) and 802.11b (in 2.4 GHz).

On the other hand, some vendors, such as Cisco, developed wired approaches to detect rogue APs [4]. These approaches check the MAC addresses of organizational routers and switches whether they belong to an organization or not. The rationale of this approach is that rogue APs typically cannot show the affiliation of an organization, that is, they are not on the organizational system management database. Instead, they may show the affiliation of their manufactures, such as Linksys. Accordingly, routers and switches that are not owned by an organization might be deployed outside of the organization. However, considering easy alteration of MAC addresses, it is not hard for hackers to deceive the technique as well with cloned MAC addresses [4].

Although network administrators must have knowledge that a node might be an AP, sending a HTTP query to a residing Web server is a possible approach to detect a rogue AP [4]. However, a drawback of this approach is that it generates unnecessary packets in a network for scanning. Consequently, a sensitive hacker should know that rogue APs were scanned, giving them a chance to an evasion [4].

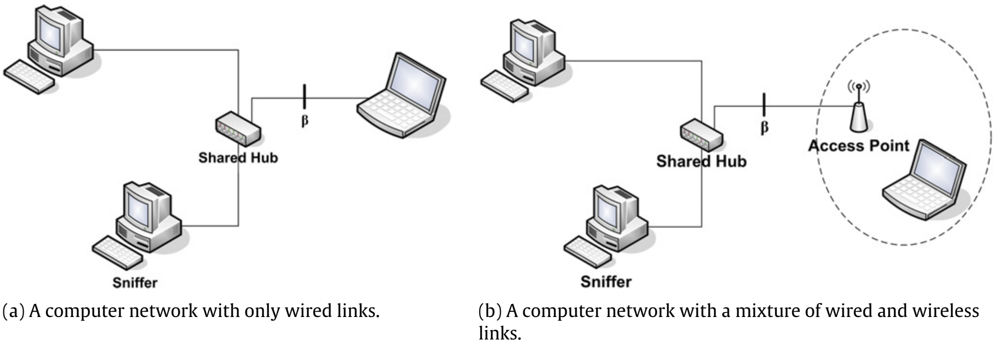
5.3. Rogue AP detection with Wi-Fi network characteristics

The authors in [4] designed a rogue AP detection scheme utilizing an analysis of the traffic differences in a LAN network. More precisely, their supposition is that network segments involving wireless APs should create more randomness in network traffic than those only with wired links. For example, from their hypothesis, two networks shown in Fig. 2 should have network traffic with different characteristics observed. In Fig. 2, another switch is inserted at a place with symbol β to examine the case where monitoring software sniffs network traffic one hop away from a target device.

There are a couple of factors to follow this hypothesis: (1) different wireless link capacities and random backoffs due to the congestion control make some changes in the network performance; (2) link speeds between wired and wireless networks reveal some characteristics in network traffic. In fact, their experiments showed that each network revealed particular characteristics in terms of the inter-packet spacing. For example, when scanning a switch residing a rogue AP, as shown in Fig. 2, while most (more than 80%) of the packets are transferred with less than 0.001 s of the inter-packet spacing in wired only links, in wireless links only nearly 10% of packets experience the same range of the inter-packet spacing, but experience a longer inter-packet spacing time [4]. More specifically, the deployment of a wireless AP one hop away from the switch (that was sniffed by network monitoring software), a network with only wired links experiences a shorter inter-packet spacing time than the one with a rogue AP [4].

5.4. Rogue AP detection with 802.11 user fingerprinting

From the aforementioned technique, a sniffer can only detect whether a LAN consists of wireless links and APs, but can hardly detect unauthorized APs. In order not only to detect existence of wireless links but also to identify rogue APs in a network, the work in [5] suggested an advanced scheme utilizing a fingerprinting technique, in which, a 4-tuple $\langle$MAC address, SSID, channel, RSSI$\rangle$ is used to test if APs belong to an organization or they are rogue APs. In the 4-tuple notation, channel represents which channel that an AP uses and RSSI stands for Received Signal Strength Indication. In their



(a) A computer network with only wired links.

(b) A computer network with a mixture of wired and wireless links.

Fig. 2. Illustration of a typical computer network.

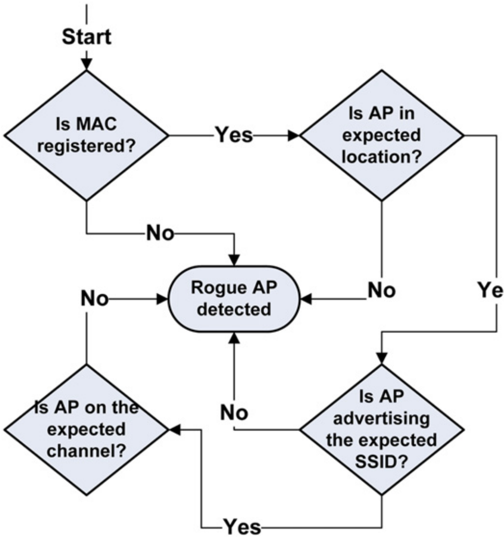

Start
Is MAC registered?
Yes
Is AP in expected location?
No
Rogue AP detected
No
No
Yes
Is AP on the expected channel?
No
Is AP advertising the expected SSID?
Yes

Fig. 3. A flowchart shows how to detect rogue APs from a 4-tuple ⟨ *MAC address, SSID, channel, RSSI*⟩.

assumption, malicious users can alter MAC addresses of their devices to match those of organizational machines (for the sake of intrusion). In this way, the organizational systems cannot distinguish these malicious nodes from their own property only by MAC addresses, even though they were duplicated.

After receiving the 4-tuples from broadcast beacons of each AP, monitoring machines, which in [5] are called diagnostic clients, periodically send them to diagnostic servers connecting to a local database. Diagnostic servers then compare the fingerprints to those in the database to check if they really belong to the organization. This comparison is made by testing each entity in the 4-tuple as shown in Fig. 3 [5]. Thus, organizations should maintain this information in their databases.

A disadvantage of this scheme is its scalability. Because diagnostic clients must cover the whole building to monitor beacons of all deployed APs, it is apparently very expensive.

6. MAC layer misbehavior detection

Selfish user behavior as well as non-cooperative networks have been studied in the literature. A typical selfish misbehavior may include terminals that refuse to forward packets on behalf of other hosts to conserve energy, or terminals that knowingly modify protocol parameters to gain unfair access to the channel [11–13]. The work in [11] proposed a modification to IEEE 802.11 MAC protocol in order to detect selfish misbehavior. The approach, however, assumed a trustworthy receiver, which represents its major drawback. This section provides a detailed investigation of detecting intentional misbehavior of IEEE 802.11 MAC protocol by using network traffic characteristics as an application of the IEEE 802.11 user fingerprinting.

Despite its prevalence, IEEE 802.11 standard still incurs several security holes to allow particular advanced users to have hidden benefits with respect to network traffic. For example, IEEE 802.11 MAC protocol employs RTS/CTS frames in order to avoid hidden terminals that cause collisions resulting in degraded network performance [1]. However, cunning IEEE 802.11 wireless network users can utilize this characteristic in their favor such that other stations can fail transmitting data for a long time, but instead they can utilize ample bandwidth on their purpose. More precisely, in the RTS/CTS exchanging phase, a cheater may intentionally hit a frame against a CTS control frame addressed to the data transmission requester in order for it not to reach him. Accordingly, the station prevented from receiving a CTS control frame has to suffer a longer backoff interval until it is again allowed to send a RTS control frame to the AP due to the utilization of distributed coordination function (DCF) [1,12].

Since the RTS/CTS control frames are usually combined with long DATA frames [1], cheaters may alternatively target DATA/ACK frames to gain their network bandwidth benefits [12]. As described earlier, a data transmitter losing an ACK frame in response to its DATA frame must get into a random backoff phase and suffer delay. As a result, cheaters can get more chances to occupy limited bandwidth.

Another way to allow the IEEE 802.11 MAC protocol to misbehave is to compromise the protocol parameters from which they can attain benefits of sharing more bandwidth on their purpose. For example, for senders, utilizing SIFS instead of DIFS before sending a RTS control frame can apparently reduce waiting time for data transmission, and thus capture the medium [12]. Moreover, intentionally utilizing lower contention window sizes also benefits cheaters to occupy network bandwidth [12].

The works in [12,13] presented a detection scheme for IEEE 802.11 MAC protocol misbehavior that relies partly on the measurement of a random backoff rather than throughput (although throughput seems to be the most intuitive metric). This is because, although it is natural to consider that a wireless station with larger throughput must occupy higher share of the bandwidth, the difference in throughputs, in fact, largely depends on what application wireless stations are running. For example, two stations running two different applications, such as Voice over Internet Protocol (VoIP) and video streams, apparently have different throughputs. In general, throughput is affected by too many factors to use as a metric of the 802.11 protocol misbehavior detection [12].

DOMINO, standing for System for Detection Of greedy behavior in the MAC layer of IEEE 802.11 public NetwOrk, was developed in [12,13] in order to detect IEEE 802.11's greedy behavior that is initiated by sly network users to gain their network bandwidths. The system components of DOMINO are shown in Fig. 4(a), consisting of six tests, of which the first test (Test 1) is designed to examine scrambled CTS/ACK/DATA frames, and the others (Tests 2 through 6) are designed to detect compromised protocol parameters. In DOMINO, a particular network characteristic in the IEEE 802.11 MAC protocol is examined in each test and whenever an anomaly is figured out, it judges whether the station is misbehaving and calls the punishing function. Each test measures particular wireless network characteristics and compares them to training data in order to detect anomalies.

For example, in order to capture scrambled frames in Test 1, DOMINO measures the number of repeated frames by examining repeated sequence numbers in the header of RTS and DATA frames by retransmissions [12,13]. In other words, stations that repeatedly retransmit their RTS and DATA frames with same sequence numbers can be suspected to be attacked by a sly network user, and, on the other hand, a station that successfully transmits frames during this period may be the sly network user. Thus, determining scrambled frames is an efficient way to detect the 802.11 misbehaviors and DOMINO puts this step at the beginning of the six tests.

In Test 2, the length of DIFS before a DATA frame is examined to figure out an anomaly. When DOMINO specifies a station that does not wait DIFS, but instead it waits a shorter amount of time repeatedly until it starts sending a DATA frame after DOMINO senses an ACK frame, it calls the punishing function [12,13].

Moreover, setting the network allocation vector (NAV) value longer keeps a particular wireless station occupying an AP unnecessarily without interruption during this time. Thus, comparison of the actual transmission time to the NAV value in the header of RTS or DATA frame can be used for detection of the 802.11 misbehaviors, and Test 3 does this measurement [12,13].

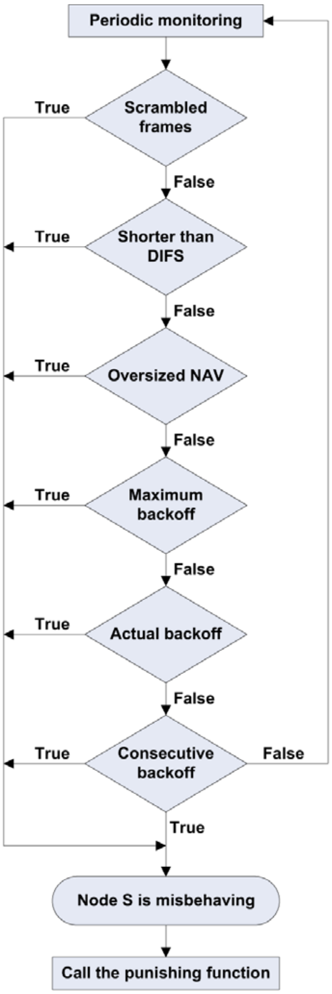
Another characteristic of the 802.11's greedy behavior is shorter maximum random backoffs. Basically, the duration of random backoffs is determined in relation to the number of retransmissions. That is, the more a station retransmits frames the longer random backoff time it has. Regarding the maximum random backoffs, the work in [12] makes an assumption that they should eventually become close to $CW_{\text{min}} - 1$ from large samples. Thus, in Test 4, by comparing them to a predefined threshold ($\text{threshold}_{\text{maxbkf}}$), DOMINO can detect stations that have shorter backoff times.

In addition to the maximum backoff time, the paper [12] also suggested to detect shorter actual backoffs in order to determine the 802.11's greedy behavior. The actual backoff duration is determined as shown in Fig. 4(b) [12]. In general, stations having shorter actual backoff times can be suspected as cheaters. In the previous case, DOMINO compares them to the nominal backoff value (B_{acnom}) and suspects stations that have shorter backoff time than the threshold.

At last, DOMINO examines consecutive backoffs caused by the MAC layer queuing in Test 6, as shown in Fig. 4(c) [12,13]. When frames are not interleaved by other frames, duration between two frames typically consists of DIFS and a random backoff due to the channel contention. Measuring this duration can help detecting greedy behaviors from TCP sources. This is an actual backoff metric in Test 5 and it does not work well for TCP sources because for those sources delays are mainly caused by TCP congestion control [12,13]. Hence, examining duration of consecutive backoffs in TCP sources in Test 6 can help concluding indecisive results from Test 5.

7. User detection with IEEE 802.11 user fingerprints

Like a human fingerprint, network traffic has unique characteristics that can be used to identify a sender device. People usually called these characteristics a network user fingerprint. Some characteristics can perfectly help to associate packets



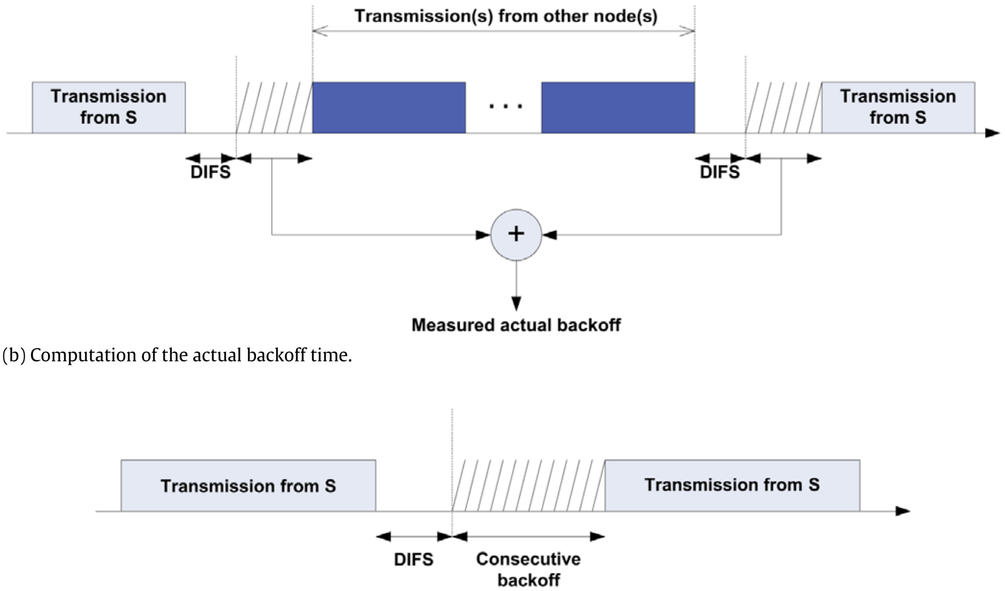
(a) DOMINO: The 802.11 misbehavior detection scheme.

Fig. 4. DOMINO architecture.

with a sender, whereas some can only conduct nearly perfect association. The former characteristics are called explicit identifiers, which include the MAC address [2]. In the latter case, characteristics of some traffic samples may overlap, or in the other words these traffic characteristics are not necessarily unique to each sender, but the combination of these characteristics can sufficiently differentiates senders. Thus, these characteristics are called implicit identifiers [2]. In this section, we investigate these two network traffic identifiers and propose a technique that can successfully associate implicit identifiers with network users.

7.1. Explicit identifiers

Explicit identifiers can uniquely differentiate network devices. Traffic samples containing these identifiers must be associated with their sender devices. For example, the MAC address of a network adapter is used to distinguish each adapter. Basically, the assignment of the MAC address is controlled by IEEE, and in production of each network adapter manufacturers require to purchase a set of free address space. Unique MAC addresses are assigned to unique network adapters by the manufacturers, such that they can peculiarly identify them [1]. Unlike the IP address, which is dynamically assigned in each session and sensitive to a device's location, the MAC address is static and persistent during the adapter's entire life. In the link layer protocol, corresponding destination and sender's MAC addresses are basically put in the packet headers. For instance, the MAC address of a device is analogy to the social security number of a person regardless of its movement, and the social security number of a person must be kept the same, no matter where he/she moves. On the other hand, an IP address is considered as a postal address of a person because it is sensitive to a human location where he/she lives. Thus, when a



(b) Computation of the actual backoff time.

(c) A figure represents two consecutive frames from source S.

Fig. 4. (*continued*)

wireless network does not employ the link layer cryptography, a sender of packets is easily disclosed to attackers via some of explicit identifiers.

To conceal MAC address information accompanied with the packet header, several papers proposed MAC address pseudonymous techniques [2]. For example, by periodically changing a MAC address to a temporary unlinked name, a user prevents adversaries from associating the MAC address with his/her device. However, the MAC address pseudonym cannot be applied frequently. One reason is that a wireless station suffers difficulties to reassociate with an AP from frequent MAC address changes. Also, it requires reauthentication with some web pages. These situations may apparently generate overhead for wireless network communications.

7.2. *Implicit identifiers*

Another type of network traffic identifier is called implicit identifiers. The work done in [2] provided four major implicit identifiers to experimentally measure the accuracy of using individual and combination of them. A list of these implicit identifiers is given as follows: (1) Service Set Identifiers (SSIDs), (2) Network destinations, (3) Broadcast packet sizes, and (4) MAC protocol fields.

Unlike explicit identifiers, such as the MAC address, implicit identifiers cannot be associated directly with senders but they may have unique characteristics to be distinguished from other traffic. For example, when a Windows machine initiates a wireless connection, it tries to find out user's preferred networks by sending requests with network names, or Service Set Identifiers (SSIDs) (active probing) [2]. Since these requests cannot be encrypted before association, they are easily overheard by any user with a network protocol analyzer [2]. Thus, by eavesdropping on SSID requests, it is possible for other people (no matter good or bad) in the same network to distinguish network traffic and finally link them to each user [2]. Although SSIDs derived from traffic samples are not necessarily unique to users and wireless networks (because it is possible that several users share a single network (AP), and also more than one network may have the same SSID), a set of SSIDs may produce some dissimilarity to distinguish the traffic samples. Because a Windows machine creates a user original SSID list and probe SSIDs for it periodically (regarding their combination here), authors in [2] demonstrated its uniqueness from their experimental trace of IEEE 802.11 traffic. Moreover, they also showed that it is possible not only to identify traffic samples but also figure out geographical information for users from the IEEE 802.11 probes. This can be achieved when an eavesdropper obtains the location of wireless APs supplied by WiGLE.net [2,3]. WiGLE.net [3] basically provides geographical information for each IEEE 802.11 AP by name (SSID), such as which AP is deployed where. For example, from the 2004 SIGCOMM trace given in [2], one user broadcasted his SSID probes, such as "University of Washington" and "dwj". Thus, from a hint of "University of Washington", they searched for SSID "dwj" around the Seattle area and eventually found out only one SSID having such a name. In this example, WiGLE.net [3] detected the location of this person's home within 192 ft.

Another example of the implicit identifier is a network destination which is a pair of an IP address and a port number [2]. Usually, user-visiting websites are distributed in a form of Zipf distribution. That is, a few major web sites are called at by the majority of users, whereas a lot of other sites are called at nearly individually, which was at most 1.2 users for each website from the 2004 SIGCOMM trace [2]. This comes from the fact that users usually have their preferred web e-mail services, and also keep bookmarks for their preferred web sites in the web browsers. Thus by tracking network destinations, an observer can form a URL preference of each user.

Broadcast packets are sent by applications requiring constantly receiving updates from their vender's servers. In [2], a packet size of each advertisement also forms a unique characteristic, and thus it can make a difference as an application and a packet size pairs do. Work carried out in [2] actually showed a list of applications and its port numbers as well as how many times packet sizes appear at the 2004 SIGCMOMM conference. Although the size of the most applications in the list exhibit the uniqueness, some wireless driver or OS and DHCP come out many times in the trace. This is because these applications are common to every user to manage an IP address and advertise its low power mode, respectively. Thus these applications should be ruled out for fingerprinting network traffic [2].

Usually, MAC header information alone tells less about their users than the other three implicit identifiers, but it can be used together with other identifiers to enhance the implicit identifier discriminability. A combination of bits in the MAC header, such as, "more fragments", "retry", "power management" and "order", as well as an authentication type and a transmission rate can nearly specify user's identity and it is considered to be persistent. This is because different NICs are likely to have different configurations of these fields [2].

7.3. Classifier

In order to make the implicit identifiers useful, a classifier must be designed with a threshold. Basically, a classifier is used to check if some condition (threshold) is satisfied by existing evidence (combination of implicit identifiers) to tell where traffic samples come from the target. For example, authors in [2] prepared a simple classifier C_U and a threshold T so as to answer a question like: "Did this sample come from user U?". The classifier answers this question with "Yes" if a computed value exceeds the threshold T, and "No" otherwise. This classifier is computed by a combination of conditional probabilities using Bayes' theorem and probabilities generated by quantifying each implicit identifier, and they also call it a feature. In other words, this classifier will find out a probability, such as "With a given sample with features $f_1, f_2, \ldots, f_m$, a probability that the sample came from user U" and this is computed by a formula as

$$\Pr[s \text{ is from } U | s \text{ has } f_1, f_2, \ldots, f_m] = \frac{\prod_i^m (\Pr[s \text{ has } f_i | s \text{ is from } U]) \cdot \Pr[s \text{ is from } U]}{\prod_i^m (\Pr[s \text{ has } f_i])} \tag{1}$$

where $\Pr[s \text{ has } f_i | s \text{ is from } U]$ and $\Pr[s \text{ has } f_i]$ are previously calculated from training data and quantified features. On the other hand, since $\Pr[s \text{ is from } f_i]$ depends largely on the belief of the observers about how frequently the target is supposed to appear in the network [2] although it is calculated from training data, it may be biased by an individual observation.

Additionally, the calculation of the probability above counts on quantified features. For example, in the field of implicit identifier, which involves several field values in the MAC header, an authentication type and a transmission rate, each combination of matching field values must generate a different value [2]. More precisely, matching only a bit in "more fragments" field and bits in all the three fields (i.e., "retry", "power management" and "order") with training data must generate different values, and more precisely to say, the latter must indicate being more identifiable.

On the other hand, when converting a set of discrete values, such as *netdests*, into some quantity, the authors in [2] applied a weighted version of the Jaccard similarity index as

$$\text{feature}_U(s) = \frac{\sum_{e \in \text{Profile}_U \cap Set_s} w(e)}{\sum_{e \in \text{Profile}_U \cup Set_s} w(e)} \tag{2}$$

where Profile_U is a union of elements of any feature obtained from every training sample from user U, Set_s is the set of elements from a sample s, and $w(e)$ is the weight of a element e. For example, the *netdests* identifier comprises a number of destinations addressed by the target. Then, the ratio of the sum of weighted destinations captured during the sampling period to the sum of those captured during the training period makes a quantified feature of sample s.

A major difference between explicit and implicit identifiers is, while the explicit identifiers can be connected directly to each network device, the implicit identifiers cannot be always done in the same way. Although recent researches suggested the necessity of a pseudonym for explicit identifiers, even with implicit identifiers more than 60% of users are identified with 90% of accuracy [2].

8. Conclusion

In this article, we summarized current researches on wireless network traffic analysis and a user finger printing technique. Currently, network traffic analyses are mainly used for security enhancement and performance maintenance. However, these noble techniques are also expected to expand to other research areas in computer science. From our observation, these techniques can be of practical use in digital criminal investigations, digital forensics, and intrusion detection.

In fact, there are plenty more papers related to fingerprint and signatures, and the readers can refer to [14–29].

Acknowledgement

This work was supported in part by the US National Science Foundation (NSF) under grants CNS-0716211, CNS-0737325, and CCF-0829827.

References

[1] J.F. Kurose, K.W. Ross, Computer Networking: A Top–down Approach Featuring the Internet, 3rd ed., Addison Wesley, 2006.
[2] J. Pang, B. Greenstein, R. Gummadi, S. Seshan, D. Wetherall, 802.11 User fingerprinting, in: Proceedings of Mobicom 2007, pp. 99–110.
[3] WiGLE.net, http://www.wigle.net/gps/gps/main.
[4] R. Beyah, S. Kangude, G. Yu, B. Strickland, J. Copeland, Rogue access point detection using temporal traffic characteristics, in: Proceedings of the IEEE Global Telecommunications Conference, Globecom 2004, vol. 4, 29 Nov. 3, Dec. 2004, pp. 2271–2275.
[5] A. Adya, P. Bahl, R. Chandra, L. Qiu, Architecture and techniques for diagnosing faults in IEEE 802.11 infrastructure networks, in: Proceedings of the International Conference on Mobile Computing and Networking, MobiCom 2004, pp. 30–44.
[6] J. Geier, Multipath a potential WLAN problem, Tutorial, Wi-Fi Planet, May 14, 2002, http://www.wi-fiplanet.com/tutorials/article.php/1121691.
[7] 'Evil Twin' fear for wireless net, BBC News, Jan 20, 2005, http://news.bbc.co.uk/2/hi/technology/4190607.stm.
[8] M. Rodrig, C. Reis, R. Mahajan, D. Wetherall, J. Zahorjan, Measurement-based characterization of 802.11 in a hotspot setting, in: Proceedings of the ACM SIGCOMM Workshop on Experimental Approaches to Wireless Network Design and Analysis, pp. 5–10.
[9] J. Yao, M. Youssef, A. Agrawala, A framework for wireless LAN monitoring and its applications, in: Proceedings of the ACM Workshop on Wireless Security, WiSe 2004, pp. 70–79.
[10] L. Ma, A.Y. Teymorian, X. Cheng, A hybrid rogue access point protection framework for commodity Wi-Fi networks, in: IEEE INFOCOM 2008, pp. 1220–1228.
[11] P. Kyasanur, N.H. Vaidya, Selfish MAC layer misbehavior in wireless networks, IEEE Transactions on Mobile Computing 4 (5) (2005) 502–516.
[12] M. Raya, I. Aad, J.P. Hubaux, A.E. Fawal, DOMINO: Detecting MAC layer greedy behavior in IEEE 802.11 hotspots, IEEE Transactions on Mobile Computing, 5 (12), 1691–1705.
[13] M. Raya, J.P. Hubaux, I. Aad, DOMINO: A system to detect greedy behavior in IEEE 802.11 hotspots, in: Proceedings of the ACM MobiSys 2004, pp. 84–97.
[14] A. Kiayias, M. Yung, Secure scalable group signature with dynamic joins and separable authorities, International Journal of Security and Networks 1 (1–2) (2006) 24–45.
[15] Y.C. Cheng, J. Bellardo, P. Benkö, A.C. Snoeren, G.M. Voelker, S. Savage, Jigsaw: Solving the puzzle of enterprise 802.11 analysis, SIGCOMM'06, Pisa, Italy, 2006.
[16] I. Hamadeh, G. Kesidis, A taxonomy of internet traceback, International Journal of Security and Networks 1 (1–2) (2006) 54–61.
[17] H. Englund, T. Johansson, Three ways to mount distinguishing attacks on irregularly clocked stream ciphers, International Journal of Security and Networks 1 (1–2) (2006) 95–102.
[18] J. Deng, R. Han, S. Mishra, Limiting DoS attacks during multihop data delivery in wireless sensor networks, International Journal of Security and Networks 1 (3–4) (2006) 167–178.
[19] X. Wang, The loop fallacy and deterministic serialisation in tracing intrusion connections through stepping stones, International Journal of Security and Networks 1 (3–4) (2006) 184–197.
[20] S.F. Owens, R.R. Levary, An adaptive expert system approach for intrusion detection, International Journal of Security and Networks 1 (3–4) (2006) 206–217.
[21] Y. Chen, W. Susilo, Y. Mu, Convertible identity-based anonymous designated ring signatures, International Journal of Security and Networks 1 (3–4) (2006) 218–225.
[22] C.H. Tan, A new signature scheme without random oracles, International Journal of Security and Networks 1 (3–4) (2006) 237–242.
[23] O. Erdogan, P. Cao, Hash-AV: Fast virus signature scanning by cache-resident filters, International Journal of Security and Networks 2 (1–2) (2007) 50–59.
[24] N.S. Artan, H.J. Chao, Design and analysis of a multipacket signature detection system, International Journal of Security and Networks 2 (1–2) (2007) 122–136.
[25] R. Bhaskar, J. Herranz, F. Laguillaumie, Aggregate designated verifier signatures and application to secure routing, International Journal of Security and Networks 2 (3–4) (2007) 192–201.
[26] I. Ray, N. Poolsappasit, Using mobile ad hoc networks to acquire digital evidence from remote autonomous agents, International Journal of Security and Networks 3 (2) (2008) 80–94.
[27] T. Kilpatrick, J. Gonzalez, R. Chandia, M. Papa, S. Shenoi, Forensic analysis of SCADA systems and networks, International Journal of Security and Networks 3 (2) (2008) 95–102.
[28] E. Cronin, M. Sherr, M. Blaze, On the (un)reliability of eavesdropping, International Journal of Security and Networks 3 (2) (2008) 103–113.
[29] J.S. Okolica, G.L. Peterson, R.F. Mills, Using PLSI-U to detect insider threats by datamining e-mail, International Journal of Security and Networks 3 (2) (2008) 114–121.
[30] K. Meng, Y. Xiao, S.V. Vrbsky, Building a wireless capturing tool for WiFi, (Wiley Journal of) Security and Communication Networks 2 (6) (2009) 654–668.
[31] Y. Xiao, Flow-net methodology for accountability in wireless networks, IEEE Network 23 (5) (2009) 30–37.