SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2014; **7**:1510–1516
Published online 3 April 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.756

SPECIAL ISSUE PAPER

A novel approach to detection of mobile rogue access points

Iluk Kim[1], Jungtaek Seo[2], Taeshik Shon[3]* and Jongsub Moon[1]

[1] Graduate School of Information Security, Korea University, Seoul, Korea
[2] The Attached Institute of ETRI, Daejeon, Korea
[3] Division of Information and Computer Engineering, College of Information Technology, Ajou University, Suwon, Korea

ABSTRACT

Rogue access points (APs) have been used in several attacks such as packet sniffing and man-in-the-middle attacks. It is becoming a serious security threat to users in public and enterprise networks. Moreover, it is easy to install malicious APs using mobile devices and networks, and existing solutions do not effectively detect these rogue APs. In this paper, we propose a method to detect rogue APs over mobile networks using round-trip time measurements, without relying on information from authorized lists of APs or users. Through experiments, we proved that our proposed method could detect rogue APs successfully. Copyright © 2013 John Wiley & Sons, Ltd.

KEYWORDS
rogue AP detection; mobile AP; network security

***Correspondence**
Taeshik Shon, Ajou University, Division of Information and Computer Engineering, College of Information Technology, Suwon, Korea.
E-mail: tsshon@ajou.ac.kr

1. INTRODUCTION

Most company provides wireless networks to help increase efficiency and convenience in the workplace. Because anyone can collect packets in a wireless network, wireless systems use protocols that are designed to increase security, such as IEEE 802.11i or wireless protected access. The IEEE 802.11i protocol offers an encryption and authentication process to protect users from unauthorized access to wireless data [1]. However, these mechanisms cannot protect users from attacks that use rogue access points (APs) because attackers can install unauthorized APs that get around such security measures and which ordinary users are unaware of.

Rogue APs have been identified as one of the critical vulnerabilities of wireless networks [2]. Attackers can obtain personal information by collecting packets sent from users' devices, and attacks can be performed through man-in-the-middle attacks, such as web page tampering, session hijacking, and certification stealing.

With the spread of smartphones and tablet PCs, APs can be easily installed over 3G or Long-Term Evolution (LTE) networks. Unlike traditional APs, mobile APs make it easy to connect to external networks. Even if security policies exist to prevent leakage of data, attackers can avoid those security measures using mobile APs. Furthermore, user-created unauthorized APs, even if they are not malicious, have potential security vulnerabilities and could be targets of outside attacks. If attackers succeed in finding these APs, they can attack internal networks without passing firewalls or intrusion detection systems.

In this paper, we propose a client-side rogue AP detection method over mobile networks through round-trip time (RTT) measurement. Several solutions detecting rogue APs have been proposed; however, most of them are admin-side approaches. These solutions need network administrators who have to check entire networks and control policies for APs and users. Moreover, they require databases, such as the authorization lists of APs and users, which have to be maintained regularly—a type of security service that is difficult to provide in public networks. In fact, these approaches are difficult to apply in mobile networks, in general. Therefore, mobile networks need client-side solutions that do not require administrators or any information about APs and users.

The rest of the paper is organized as follows. In Section 2, we discuss the related works. In Section 3, we describe the proposed method. In Section 4, we present our experiments and evaluation results. Finally, we conclude the paper in Section 5.

2. RELATED WORK

Prior studies on detecting rogue APs can be classified based on two approaches: the first approach uses information about the AP, such as the media access control (MAC) address, and the second one takes advantage of characteristics of wired and wireless networks.

2.1. Information about the AP

One approach to detect rogue APs uses the hardware information of the AP, such as the MAC address, the service set identifier (SSID), and the vendor's name. This method stores information about authorized APs and scans them regularly. In [3], the Dense Array of Inexpensive Radios (DAIR) system, a framework for monitoring wireless network, collects information about the AP through USB sensors called Air Monitor. The information includes the AP's MAC address, which is stored in the DataBase (DB) server. In [4], the authors proposed another method to detect unauthorized APs by using the MAC address, the SSID, the channel, and the received signal strength indicator.

Many commercial tools detecting rogue APs are available. Airdefense [5], Netstumbler [6], and Roguescanner [7] use a combination of various pieces of information to provide users with lists of APs that might be a threat. The advantages of these methods are that they can detect rogue APs in a short time and that they have a low false detection rate. However, their disadvantage is that network managers have to check information about authorized APs periodically. In addition, they can be easily bypassed by malicious attackers because hardware information about APs could be fabricated.

Another approach uses the location information of the AP [8–10], which is determined through sensor devices. If the AP is found at an unexpected location, it is classified as a rogue AP. Currently, the wireless AP location detection technology offers a 3- to 5-m degree of accuracy [11]. However, additional equipment for the measurement is required, and it is difficult to apply this strategy in large-scale networks or organizations where equipment changes frequently. Furthermore, it is difficult to detect mobile rogue APs using only location information because mobile APs are not constrained to a single installation location.

2.2. Characteristics of wired and wireless networks

Another approach to detect rogue APs checks whether an additional wireless hop exists because most rogue APs are connected to an existing AP. To find an additional wireless hop in [12], authors use the time interval of two packets that are captured in real networks. Similarly [13,14], use the time interval of TCP-ACK pairs. To increase the accuracy of the detection rate in [15], authors calculate RTT by sending randomly generated packets to a Domain Name System DNS server. This method obtains values through measurements of RTT from the AP to the DNS server. Then, if it exceeds the threshold value, the AP would be classified as a rogue AP. These methods do not need to maintain a DB and are more difficult to manipulate. However, it is becoming increasingly difficult to discriminate using a certain threshold because of the emergence of the high-speed protocol 802.11n and the dual antenna. In addition, some methods require internal servers or DNS servers; therefore, it is difficult to apply these methods in mobile APs, which are connected to external networks over mobile networks.

3. PROPOSED METHOD

3.1. Detection scheme

In this paper, we use the difference in RTT values between mobile and wired networks to detect rogue APs. Figure 1 illustrates the network structures of wired APs and mobile APs. All of the APs are connected by Wi-Fi in Section (A); however, in Section (B), the wired AP (①) is connected to the next node (commonly a router) via a wired network.

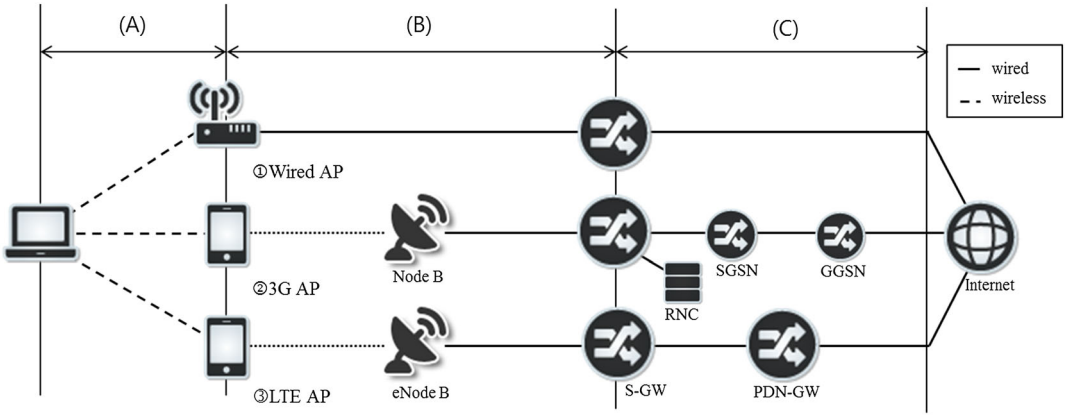

Figure 1. Network structure of wired and mobile access points (APs). LTE, long-term evolution. SGSN, Serving GPRS Support Node. GGSN, Gateway GPRS Support Node (GPRS, General Packet Radio Service). S-GW, Serving Gateway. PDN-GW, Packet Data Network Gateway. RNC, radio network controller.

Furthermore, the mobile APs (②, ③) are connected to subsequent nodes by a wireless mobile network. In general, IEEE 802.11 wireless networks (Wi-Fi) have an average latency of less than 1 ms, and wired networks have a similar latency [16]. On the other hand, 3G mobile networks supporting high-speed downlink packet access technology have an average latency of 60–100 ms [17]. The user-plane latency in LTE networks is around 10–20 ms [18]. These differences in latency denote the differences of RTT values between wired and mobile networks. Moreover, the base stations called Node B and eNode B over mobile networks increase the differences. In short, it is possible to classify APs into mobile AP or wired AP through RTT measurement because of the differences in response latency in Section (B), from the AP to the subsequent node that supports the Internet Protocol (IP).

Every mobile AP and most wired APs have IP addresses as gateways. Therefore, it is possible to measure RTT values through Internet Control Message Protocol (ICMP) packets. Although some wired APs do not work as gateways, measurement is still possible using another subsequent node that was assigned an IP address. Some small increases in RTT value can be negligible because the latency of a wired link is sufficiently shorter than those of mobile wireless links. At the client level, the APs that are installed over mobile networks in public places or in companies, especially with well-known SSIDs, could be malicious APs. Thus, in this paper, we consider mobile APs in public places or within company networks as suspect. This approach does not rely on information about APs; therefore, DB maintenance is not required continuously, and attackers cannot bypass these methods through manipulation.

Moreover, it does not require specific servers, such as a DNS server, and measurement errors of RTT values could be reduced because the methods measure the latency only between the AP and a subsequent node. Even though the traceroute process does not guarantee finding an appropriate subsequent node, it is more accurate than measuring RTT values between APs and end nodes or DNS servers.

Figure 2 shows the rogue AP detection phases. First, IP trace, ICMP packet send/receive, and RTT measurement are performed in the RTT Measurement Phase to obtain the RTT values. Next, the outlier removal and grouping processes are performed to obtain accurate data in the preprocessing phase. Finally, in the classification phase, the classification engine classifies the AP into wired AP or mobile AP, using the preprocessed RTT data. As we discussed earlier, that AP is suspected to be a rogue AP.

3.2. RTT measurement algorithm

Table I shows the RTT measurement algorithm. First, we determine the IP addresses of the AP (hop1) and the next node of the AP (hop2). To get IP_{hop1} and IP_{hop2}, two ICMP echo request packets with TTL values set to 1 and 2 are sent to a public domain. Then, ICMP echo request packets of k bytes are sent to IP_{hop1} and IP_{hop2}, and the RTT values of the hops (RTT_{hop1} and RTT_{hop2}) are calculated. Next, RTT_i are calculated using the RTT values. The values could be below zero, depending on the network situation; such values are discarded for efficiency of classification. Through these processes, a total of n RTT values are obtained.

3.3. Preprocessing

For an accurate classification result, raw data must be refined in the preprocessing phase using two processes: outlier removal and grouping.

- Outlier removal
  Because of varied network conditions, some values are extremely deviated from the others. These values make it difficult to obtain proper data sets and classification results, thereby increasing the error rate of rogue AP detection. Therefore, these values should be removed or replaced. One of the most efficient methods to manage this type of problem is an outlier removal algorithm. The classical definition of an

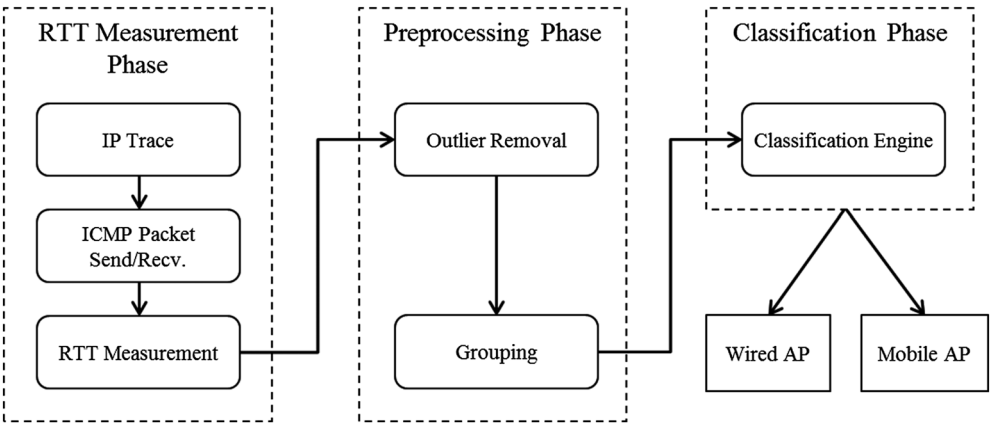

Figure 2. Rogue access point (AP) detection phases. RTT, round-trip time; ICMP, Internet Control Message Protocol.

Security Comm. Networks (2014) © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

Table I. RTT measurement algorithm.

Algorithm 1 RTT measurement

```
IP_hop1, IP_hop2 = traceroute to public domain
for i = 1 to n do
  for j = 1 to 2 do
    send ICMP echo req. pkt to IP_hopj (k bytes size)
    while ICMP echo response pkt does not arrive
      wait for timeout
    end while
    calc. RTT_hopj
  end for
  RTT_i = RTT_hop2 − RTT_hop1
end for
```

outlier is an observation (or subset of observations) that appears to be inconsistent with the remainder of that set of data [18]. Several outlier detection algorithms such as those based on distribution, distance, and density are available. Each algorithm has advantages and disadvantages [19], and we chose to use the clustering-based *k*-means [20] algorithm for characteristics of RTT data and client-side detection. Some algorithms need probability models of data; however, it is difficult to determine the proper probability model of RTT data because of its extremely deviated values. The *k*-means algorithm does not rely on the probability models; therefore, it fits our method. Moreover, the *k*-means algorithm has low computation complexity, and, therefore, it is suitable for the client-side detection that is usually performed on mobile devices. The steps of the *k*-means outlier removal algorithm are as follows:

(1) Define the number of clusters $k = 2$.
(2) Initialize the k cluster centroids.
(3) Assign each object to the group that has the closest centroid.
(4) Recalculate the centroids of both modified clusters.
(5) Repeat steps (3) and (4) until the centroids do not change any more.
(6) Calculate the distances between two centroids and the mean of all data.
(7) Determine which group has the farthest centroid as an outlier.
(8) Replace outlier values with the mean of other groups.

- Grouping
  After the outlier removal, the grouping process is performed on the basis of the central limit theorem (CLT) for data classification. The CLT states that, given a distribution with a mean μ and variance σ^2, the sampling distribution of the mean approaches a normal distribution with a mean (μ) and a variance σ^2/N, as the sample size N increases.

$$\sqrt{n}\left(\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) - \mu\right) \xrightarrow{d} \mathrm{N}\left(0, \sigma^2\right) \qquad (1)$$

Generating data following a normal distribution through grouping has several advantages. First, it is possible to use classification algorithms that need a probability model, such as the Bayesian classifier. Second, it enables more accurate detection through repeated RTT measurements, instead of sending only one or two packets. Generally, the approximation in the CLT is accurate when $n \geq 30$ in most cases. Thus, we set the n value of the RTT measurement algorithm (Table I) to 30.

3.4. Data classification

The RTT data obtained by RTT measurement and preprocessing have these characteristics: it is one-dimensional and continuous, it follows a normal distribution, and it can be classified into three classes (wired, LTE, and 3G). We chose the Naïve Bayesian classifier algorithm because it is ideal for RTT data that have the characteristics listed earlier. Furthermore, it is lightweight, and it does not require training data after trained, unlike some algorithms, such as the *k*-NN classifier.

The Naïve Bayesian classification process is as follows [21]:

(1) Each data sample is represented by an n-dimensional feature vector $X = (x_1, x_2, \ldots, x_n)$, depicting n measurements made on the sample from n attributes $A_1, A_2, \ldots, A_n$, respectively.
(2) As $P(X)$ is constant for all classes, only $P(X|C_i)P(C)$ needs to be maximized. If the class's previous probabilities are not known, then it is commonly assumed that the classes are likely equal; that is, $P(C_1) = P(2) = \cdots = P(C_n)$. Therefore, $P(X|C_i)$ must be maximized.

4. EXPERIMENTS AND EVALUATION

4.1. Training data

We configured three APs to collect RTT values. Table II shows the equipment used for the experiments. IPTIME, a wireless router, was connected to the external network by a 100-Mbps wired link. Mobile APs were also connected to the external network over mobile links, such as LTE and 3G. First, we changed the mobile phones to AP mode, hotspot or tethering, and then collected 9000 RTT values using a laptop. The wired AP was set to IEEE 802.11b, g, and n protocols, and then, 3000 RTT values were collected for each protocol. To improve detection rate, we increased the packet size. However, a packet over the maximum transmission unit is fragmented, and,

Table II. Equipment.

Type of AP	Device	Specs
Wired	IPTIME n604M	802.11 b/g/n
LTE	LG Optimus LTE	Android 2.3, 802.11 g
3G	iPhone 4	iOS4, 802.11b ad hoc

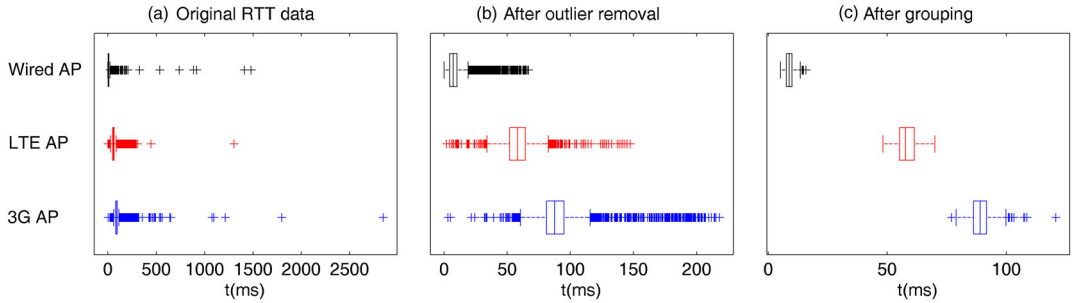

Figure 3. Round-trip time (RTT) data (original, after outlier removal, and after grouping). AP, access point; LTE, long-term evolution.

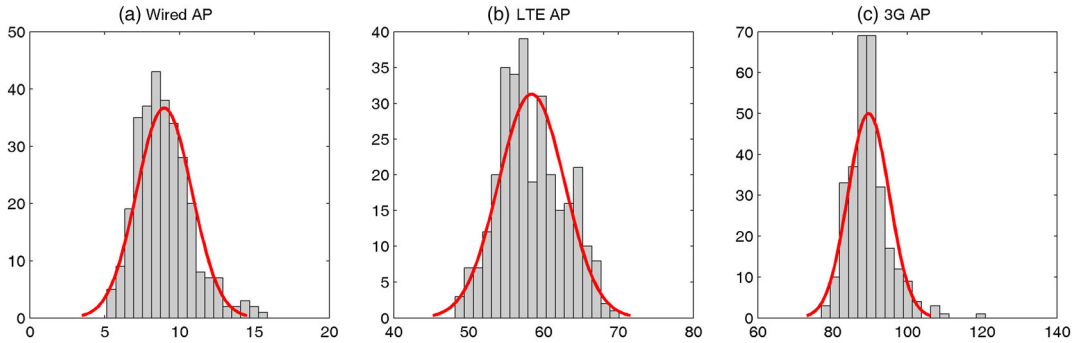

Figure 4. Histograms of the round-trip time data after preprocessing. AP, access point; LTE, long-term evolution.

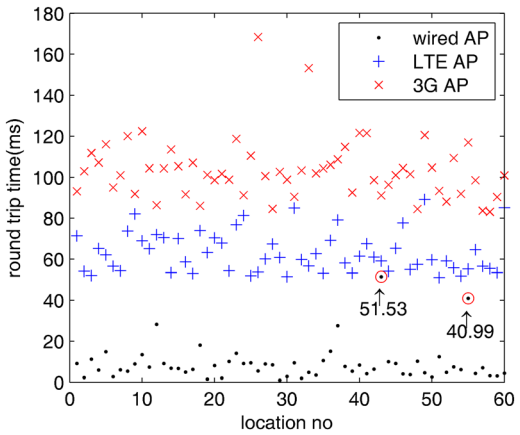

Figure 5. Classification result. AP, access point; LTE, long-term evolution.

therefore, it could affect the result. Therefore, we set the data size of the packets to 1400 bytes, which is below the maximum transmission unit threshold.

Figure 3 shows the results of the RTT data collection. We used a box-and-whisker plot to show deviation and median. A box contains 50% of the data, and a line in a box indicates the median value. A "+" mark on the *X*-axis represents the RTT value in milliseconds. The original data graph (a) in Figure 3 illustrates the RTT data immediately after collection. It shows that most of the values are densely concentrated; however, some are not. As we mentioned earlier, various network conditions can cause deviated values. In addition, this density makes it difficult to classify the APs into wired and mobile APs.

The (b) and (c) graphs describe the RTT data after the outlier removal and the grouping phases. The outlier values were replaced using the *k*-means outlier removal algorithm. The data points were highly concentrated near the median value after grouping. As we can see in the (c) graph, the RTT values of wired APs are clearly distinguishable from the RTT values of mobile APs after grouping.

Figure 4 shows the histograms of the RTT data after preprocessing. The solid line on each graph represents a normal distribution, and the gray bars show the distribution of RTT data after preprocessing. If the entire shape of the bars is similar to the solid line, we can say that it follows a normal distribution. As the three histograms indicate, each group approximately follows a normal distribution, and, therefore, the preprocessed RTT data is appropriate for use with the Naïve Bayesian classifier.

4.2. Evaluation

To evaluate the detection algorithm in real scenarios, we collected RTT values from public APs for testing data. Existing public wired APs installed in a total of 60 public places, such as cafés, subways, banks, and university campuses, were used for the experiment. Unfortunately, it was difficult to find public mobile APs. Therefore, we set up our own mobile

Security Comm. Networks (2014) © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

Table III. Comparison result.

	Proposed method			DNS-based (Server1)			DNS-based (Server2)		
	Wired	LTE	3G	Wired	LTE	3G	Wired	LTE	3G
Mean	8.98	58.39	89.63	23.21	65.64	546.78	259.49	271.89	1270.2
Std. dev.	1.82	4.38	5.49	8.35	19.68	47.45	21.56	28.6	84.46
Classification rate		100%			92.4%			78.66%	

APs near public wired APs and collected RTT values. According to the detection phases described in Figure 2, prepossessing and classification were also conducted.

Figure 5 describes the classification result. The *X*-axis indicates the locations where the testing data were collected, and the *Y*-axis represents the RTT value in milliseconds. As a result, only 2 of 180 APs were misclassified. The classification accuracy is 98.9%. This result shows that our detection method properly works in real scenarios and that false detection rate is low.

4.3. Comparison

To prove the effectiveness of the proposed method, we compared it with a DNS-based method, which is discussed in Section 2. The DNS-based method calculates the RTT value from the AP to the DNS server. To simulate an optimized network and a congested network, we conducted experiments where DNS Server1 had low latency and DNS Server2 had high latency. The proposed method uses a link that only goes from the AP to the subsequent node; therefore, we used the training data in Section 4.1 for comparison. The same RTT measurement method (packet type, packet size, and preprocessing) was used for data collection to remove factors that affect the result, except for the differences in RTT measurements. Finally, we classified the data collected in each network environment (wired, LTE, and 3G) using the Naïve Bayesian classifier.

Table III shows the detailed classification results of the comparison experiment. The proposed method successfully classified every RTT value, whereas the DNS-based method has a classification accuracy of only 92.4% in the experiment performed on DNS Server1 (low latency). Furthermore, the experiment with DNS Server2 (high latency) shows a significantly lower classification accuracy of 78.66%. The experiment result indicates that the DNS-based method is easily affected by network conditions. On the other hand, the proposed method is less affected by various conditions because it uses a shorter link from the AP to a subsequent node. The lower standard deviation of the proposed method also indicates that the RTT data collection is more stable.

5. CONCLUSION AND DISCUSSION

We proposed a client-side method to detect rogue APs over mobile networks. We described the detailed process of the detection method through three phases: RTT measurement, preprocessing, and classification. To evaluate the method, we collected training data in a lab environment and test data from public places, and we also showed the classification results. The results indicate that our detection method successfully classifies wired and mobile APs using RTT measurements.

Mobile network environments have difficulty predicting network status and identifying network entities because of several factors such as locations, physical obstacles, or even weather conditions. Because we collected training data from an optimized network environment, we need to conduct more experiments in various conditions. However, the proposed method showed good performance in evaluation tests conducted in public places. Moreover, the detection rate could be enhanced by other existing methods that are discussed in Section 2. We are investigating several methods that work well with our method. For instance, it is possible to use the MAC address from specific vendors, such as mobile carriers or smartphone makers, or the probe packet [15] with the ICMP packet to decrease measurement error rate. Finally, we plan to validate these methods on various network environments.

ACKNOWLEDGEMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (no. 2012R1A1A1010667).

REFERENCES

1. He V, Mitchell JC. Security analysis and improvements for IEEE 802.11 i. *The 12th Network and Distributed System Security Symposium*, San Diego, Feb. 2005; 90–110.
2. Henry P, Luo H. WiFi: what's next? *IEEE Communications Magazine* 2002; **40**:66–72.
3. Bahl P, Chandra R, Padhye J, *et al*. Enhancing the security of corporate Wi-Fi networks using DAIR. *MobiSys* 2006:1–14.
4. IEEE 802.11 user fingerprinting and its applications for intrusion detection.

5. Airdefense. Available from: http://www.airdefense.net/whitepapers/roguewatch_request2.php
6. NetStumbler. Available from: http://www.netstumbler.com/
7. RogueScanner. Available from: http://www.paglo.com
8. Schweitzer D, Brown W, Boleng J. Using visualization to locate rogue access points. *Journal of Computing Sciences in Colleges* 2007; **23**:134–140.
9. Solution to the Wireless Evil-Twin Transmitter Attack.
10. A Location-aware Rogue AP Detection System Based on Wireless Packet Sniffing of Sensor APs.
11. Bahl P, Padmanabhan VN. RADAR: an in-building RF-based user location and tracking system. *The 19th Annual Joint Conference of the IEEE Computer and Communications Societies*, Tel Aviv, March 2000; 775–784.
12. Beyah R, Kangude S, Yu G, Strickland B, Copeland J. Rogue access point detection using temporal traffic characteristics. *GLOBECOM* 2004; **4**:2271–2275.
13. 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. *The Seventh ACM Internet Measurement Conference*, San Diego, Oct. 2007; 365–378.
14. Watkins L, Beyah R, Corbeet C. A passive approach to rogue access point detection. *IEEE GLOBECOM* 2007:355–360.
15. Han H, Sheng B, Tan CC, Li Q, Lu S. A measurement based rogue AP detection scheme. *Infocom* 2009; 1593–1601.
16. Lacage M, Manshaei MH, Turletti T. IEEE 802.11 rate adaptation: a practical approach. *The 7th ACM International Symposium on Modeling*, *Analysis and Simulation of Wireless and Mobile Systems*, Oct. 2004; 126–134.
17. Holma H, Toskala A. 3GPP release 5 HSDPA measurements. *17th Annual IEEE International Symposium on Personal*, *Indoor and Mobile Radio Communications (PIMRC)*, Sept. 2006; 1–5.
18. Holma H, Toskala A. *LTE for UMTS : OFDMA and SC-FDMA Based Radio Access*. Wiley, 2009; 244–245.
19. Hodge V, Austin J. A survey of outlier detection methodologies. *Artificial Intelligence Review* 2004; **22**:85–126.
20. MacQueen JB. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press: Berkeley, 1967; 281–297.
21. Sumathi S, Esakkirajan S. *Fundamentals of Relational Database Management Systems*. Berlin Heidelberg: Springer, February 2007; 439.

Security Comm. Networks (2014) © 2013 John Wiley & Sons, Ltd.
DOI: 10.1002/sec