802.11 User Anonymization

Dingbang Xu* Yu Wang† Xinghua Shi‡ Xiaohang Yin*
*Department of Computer Science, Governors State University, University Park, Illinois, USA
†Department of Computer Science, University of North Carolina at Charlotte, Charlotte, North Carolina, USA
‡Harvard Medical School, Harvard University, Boston, Massachusetts, USA

***Abstract*—Privacy issues have been a serious concern for 802.11 Wireless LAN users. As demonstrated by Pang et al. [1] and Xu et al. [2], applying pseudonym techniques does not completely protect users' privacy. In particular, users' identities can be disclosed through *implicit identifiers* such as the IP addresses and port numbers users often access. In this paper, we study how to improve user anonymity even if implicit identifiers based identification is applied. The basic idea of our approach is to artificially generate bogus data and inject them into original traffic, and thus users' behavior patterns are *disturbed*. Specifically, we propose eight different methods to generate bogus data, where each of them applies different algorithms and metrics to generate bogus packets. Our simulation results with SIGCOMM 2004 wireless trace demonstrate that our anonymization methods can decrease user identification rates, and hence improve user anonymity.**

I. Introduction

Wireless local area networks (Wireless LANs) have been increasingly deployed and used in recent years. Though they provide lots of conveniences to many users, the broadcast nature of wireless signals has brought challenges to protect users' privacy. Users' locations and certain types of identities (e.g., the MAC addresses of the laptops) may be disclosed through using wireless networks.

To address these privacy challenges, various approaches have been proposed. A large proportion of these proposed approaches is pseudonym techniques, where a user's identity may be protected through a series of frequently changing pseudonyms. For example, Beresford and Stajano [3] propose to apply pseudonyms and mixed zones to protect a user from being identified. Gruteser and Grunwald [4] propose to frequently dispose MAC addresses (these MAC addresses are "short-lived" and "disposable") to protect user privacy.

Pseudonym techniques provide the solutions to anonymize *explicit identifiers* such as MAC addresses, however, recent research demonstrates that users' identities may also be disclosed through *implicit identifiers*, where implicit identifiers may be related to the observations of a user's behavior pattern. For example, the IP addresses and port numbers frequently accessed by a user, and the SSIDs (Service Set Identifiers) embedded in probe frames in Wireless LANs. In [1], based on four implicit identifiers (*Network Destination*, *SSID*, *Broadcast Packet Size* and *MAC Protocol Fields*), a user's behavior pattern can be constructed, and then data classification techniques are applied to identify users. In a follow-up work [2], an enhanced feature selection is proposed using the same implicit identifiers. One result in [2] shows 80.6% identification rate, which clearly demonstrates that implicit identifiers are capable of identifying users even under pseudonyms.

This paper is a step further to the work of [1] and [2]. We aim to study how to improve the user's anonymity even if implicit identifiers based user identification is applied. In particular, assuming that Pang et al.'s method [1] or Xu et al.'s method [2] is employed by the adversary, we investigate new techniques that can decrease its user identification rates, and thus improve user anonymity. To achieve this goal, we adopt a natural idea: *injecting bogus traffic* into the network, so that users' activities may be "disturbed" and users cannot be identified as they originally are. Carefully designing injection patterns of bogus traffic is critical to achieve user anonymity against 802.11 user identification methods. In this paper, we propose eight different methods (*Methods* $\mathcal{M}_1$ to $\mathcal{M}_8$) to generate bogus traffic for 802.11 user anonymization, where each of them applies different algorithms and metrics to generate different patterns of bogus packets.

Via extensive simulations with SIGCOMM 2004 data sets [5], we evaluate both the effectiveness of each proposed method and how the amount of bogus traffic affects identification rates. We observe that generating the bogus traffic pure-randomly (*Method* $\mathcal{M}_1$) does not help with anonymization. However, using our proposed advanced methods, the user identification rates can be effectively reduced, even with a small percentage of bogus packets. For example, in one set of simulations with 5% bogus packets, the average user classification rate has decreased from 0.528 to 0.420 (under *Method* $\mathcal{M}_7$) in Pang et al.'s method, and from 0.904 to 0.791 (under *Method* $\mathcal{M}_6$) in Xu et al.'s method.

II. Related Work

In recent years, the privacy concern in wireless networks has drawn lots of attention. Due to the broadcast nature of wireless signals, wireless users may be easily identified and tracked through certain explicit identifiers such as MAC addresses. To address these privacy challenges, pseudonym based techniques such as [3], [4] have been proposed, where the general idea among them is to anonymize those explicit identifiers through frequently changing them. Pseudonym techniques solve the problems of identifying users through those explicit identifiers. However, recent research also demonstrates that users/devices may be identified based on *implicit identifiers*. There are two types of implicit identifiers: *device fingerprint based* or *user pattern based*. Device fingerprint based methods either use

The work of Y. Wang is supported in part by the US National Science Foundation (NSF) under Grant No. CNS-0721666 and No. CNS-0915331.

hardware-based fingerprinting [6] or software-based fingerprinting [7] of individual devices to distinguish network nodes. On the other hand, user pattern based methods [1], [2] use patterns of wireless traffic such as network destinations a user frequently visits to identify users.

Both user identification methods in [1] and [2] adopt classic classification techniques to identify 802.11 users through *implicit identifiers*. Four implicit identifiers have been mentioned by [1] such as *Network Destination* (the IP addresses and port numbers a user frequently accesses) and *SSID*. The major difference between [1] and [2] lies in how to select and generate features for classification models.

User Identification Method by Pang et al. [1]. Assume that there are an implicit identifier $\mathcal{I}$ and a user $\mathcal{U}$, and the set of all attribute values corresponding to $\mathcal{I}$ in the training data set is S. Given a sample data set S', the feature f of this implicit identifier is defined as

$$f = \frac{\sum_{v \in (S \cap S')} weight(v)}{\sum_{v \in (S \cup S')} weight(v)},$$

where v is an attribute value, and $weight(v)$ is defined as the inverse of the number of users in S having value v.

User Identification Method by Xu et al. [2]. In [2], we enhance Pang et al.'s method by proposing new approach to select and generate features for classification models. Feature selection examines the training data sets and decides how many features are selected. Specifically, given an implicit identifier $\mathcal{I}$ in a training data set S, all different attribute values for $\mathcal{I}$ in S are enumerated, and then an index value related to each attribute value v is computed as the ratio between the number of packets having value v and the number of users whose packets have value v. Next, certain number of attribute values are selected based on these index values and some other constraints. Each attribute is then transformed into a feature. Multiple features are grouped and used for building user profiles and identifying users.

The basic idea of our proposed anonymization methods in this paper will be based on bogus packet injection. This is not a completely new idea, and has been studied for source location privacy issues in wireless sensor networks [8], [9], privacy protection in wireless networks through dynamic MAC address exchanging [10], and anonymous services via designing of mix networks [11], [12]. However, how to generate bogus packets (in what patterns) is completely different among these applications, depending on the adversaries' threat models. We believe that we are the first one to use this idea and provide detailed methods in 802.11 user anonymization against *user pattern based* identification methods.

III. 802.11 User Anonymization

Before we present the details of our anonymization techniques for 802.11 users, let us first summarize all assumptions.

- Pseudonym techniques are applied to wireless devices.
- Adversaries can deploy monitoring devices (eavesdroppers) to observe 802.11 traffic from nearby users. Then existing user identification methods ([1] or [2]) based on implicit identifiers, such as *Network Destination* and *SSID*, may be applied, aiming at identifying users.
- The traffic between 802.11 users and access points are protected by encryption. Thus, eavesdroppers will not be able to view the payloads of packets/frames. However, they may still derive some information through implicit identifiers from the headers of packets/frames.

The general idea of our techniques is to generate bogus packets (or frames) and inject them into the original traffic with certain patterns. We assume that each 802.11 device can generate bogus packets which are marked inside payload data. The authorized users can tell what packets are bogus packets when receiving them, and thus ignore them. However, the eavesdroppers cannot tell the difference between real packets and bogus ones by examining the header information.

Assume that the *original traffic* is $\mathcal{S}_o$ with n packets. Our anonymization methods artificially generate the *bogus traffic* $\mathcal{S}_b$ with $p\% \cdot n$ packets, where $p\%$ is an adjustable parameter controlling the amount of bogus packets. The *mixed traffic* $\mathcal{S}_m$ is generated through combing $\mathcal{S}_o$ and $\mathcal{S}_b$. We will study how this mixed traffic ($\mathcal{S}_m$) affects the user identification rate compared with $\mathcal{S}_o$. To be more specific, we will compute the user classification rates applying both identification methods ([1] or [2]) on $\mathcal{S}_o$ and $\mathcal{S}_m$, and observe the difference between them if any. A good anonymization technique will significantly reduce the user classification rate by injecting bogus traffic.

The key question of designing our anonymization method is *how to generate the bogus traffic $\mathcal{S}_b$ given $\mathcal{S}_o$*. The pattern of bogus traffic will clearly affect the user classification rate of 802.11 user identification methods. Here we focus on the discussion of how to generate bogus attribute values for two specific implicit identifiers: *Network Destination* and *SSID*. These proposed techniques can be easily extended to other attributes or other implicit identifiers. For *Network Destination*, we need to generate artificial pairs of IP addresses and port numbers for bogus packets. For *SSID*, we just need to generate artificial SSIDs for bogus frames. For other attribute values in packets/frames, they can be generated either randomly or by some existing techniques. Notice that those values will not affect user identification rates. In the following, we propose eight different methods to generate artificial values for implicit identifiers based on different levels of knowledge.

Pure Random *Method* $\mathcal{M}_1$. The first method generates attribute values randomly in the whole domain without using any additional knowledge or statistics. Given an attribute domain (i.e., the possible values for this attribute), we randomly generate attribute values with equal probabilities in this domain. For example, for IPv4 addresses in the format of *A.B.C.D* and port numbers, we randomly generate A, B, C and D from 0 to 255 and port number from 0 to 65535. We may further restrict the attribute domain if necessary.

Uniformly from Global List *Method* $\mathcal{M}_2$. The second method uses a "global" statistical list, which includes all popular attribute values via statistical information in a global or wide-area setting. The attribute values are then generated randomly and uniformly from top k elements in this list. For example, for the implicit identifier *Network Destination*, we first obtain a top $k = 2000$ list of Web sites based on the statistical information (Web traffic ranking) provided by *Alexa* (http://www.alexa.com/). Then we perform nslookup to find the sites' corresponding IP addresses. When generating the bogus packets, we randomly select IP addresses from this list. For simplicity, we set corresponding port numbers to 80 (i.e., only consider http traffic). For the implicit identifier *SSID*, we obtain the statistical information from WiGLE (http://wigle.net/), where top $k = 1000$ SSIDs are provided.

Uniformly from Training-Set List *Method* $\mathcal{M}_3$. The third method is very simple. The attribute values are uniformly generated from all attribute values appeared in training data sets. Given a training data set, we can first enumerate a list of all different attribute values, then bogus attribute values can be randomly chosen from this list.

Distribution from Training-Set List *Method* $\mathcal{M}_4$. The forth method is similar to $\mathcal{M}_3$. The only difference is that, instead of uniformly selecting the attribute values from the training-set list, $\mathcal{M}_4$ randomly generates the bogus attribute values following distribution of attribute values in the training data set. Thus, it may create more packets with attribute values more frequently observed in the training data set.

The first four methods share the same philosophy, randomly generating the bogus traffic, so that it is hard to distinguish users from each other. In an extreme case, bogus traffic dominates the traffic pattern, then perfect anonymization is achieved. However, that requires large amount of bogus traffic which is not feasible in practice. Therefore, we further refine our anonymization techniques by proposing four more methods, which carefully pick the bogus traffic against the user identification methods ([1] or [2]).

Uniformly from Top Packet-Count List *Method* $\mathcal{M}_5$. In this method, we try to inject bogus packets with attribute values more frequently observed in the training data set. $\mathcal{M}_5$ is similar to $\mathcal{M}_3$, except that the training-set list is sorted in decreasing order based on packet count $P(v)$ from the training data set, and we only take the top k values from the list. Here $P(v)$ is the number of packets with attribute value v in the training data set. For example, assume that we have three *SSID*s in a training data set: *linksys1*, *linksys2*, and *linksys3*, where the corresponding packet number is 100, 200, and 300, respectively. If we decide only generate bogus SSIDs from the top $k = 2$ list, then the SSID in each generated frame will be either *linksys3* or *linksys2*.

Uniformly from Top Packet/User-Ratio List *Method* $\mathcal{M}_6$. The basic procedure in $\mathcal{M}_6$ is similar to that of $\mathcal{M}_5$. The only difference is that $\mathcal{M}_6$ uses packet/user ratio $R(v)$ as the ranking metric. Here packet/user ratio is defined as $R(v) = \frac{P(v)}{U(v)}$, where $P(v)$ is the number of packets with attribute value v, and $U(v)$ is the the number of users having the packets with value v in the training data set. To continue our example in $\mathcal{M}_5$, further assume that the number of users accessing SSIDs *linksys1*, *linksys2*, and *linksys3* is 10, 5, and 40, respectively, then we can get $R(linksys1) = 10$, $R(linksys2) = 40$, and $R(linksys3) = 7.5$. If we still let $k = 2$, then the bogus SSID is randomly selected from $\{$*linksys2*, *linksys1*$\}$.

Uniformly from Top Reverse-User-Count List *Method* $\mathcal{M}_7$. This method is similar to $\mathcal{M}_5$ and $\mathcal{M}_6$, but its ranking metric is the inverse of user numbers, which is $\frac{1}{U(v)}$. Therefore, the attribute values where less number of users accessing them are on the top of the list, and will be selected as artificial attribute values for bogus packets. This is motivated by an observation that if an attribute value is unique and consistent for a user, then it can be a great indication for identification purpose of that user. In the above example, $1/U(linksys1) = 0.1$, $1/U(linksys2) = 0.2$, and $1/U(linksys3) = 0.025$. Thus, the bogus SSID is uniformly and randomly selected from $\{$*linksys2*, *linksys1*$\}$ if we still let $k = 2$ based on $\mathcal{M}_7$.

Sequentially from Top Reverse-User-Count List *Method* $\mathcal{M}_8$. $\mathcal{M}_8$ is a revision to $\mathcal{M}_7$. In $\mathcal{M}_8$, we use the same procedure to generate the top k list. However, when generating bogus packets, it picks the values from the list following a sequential order instead of random order.

IV. Simulations

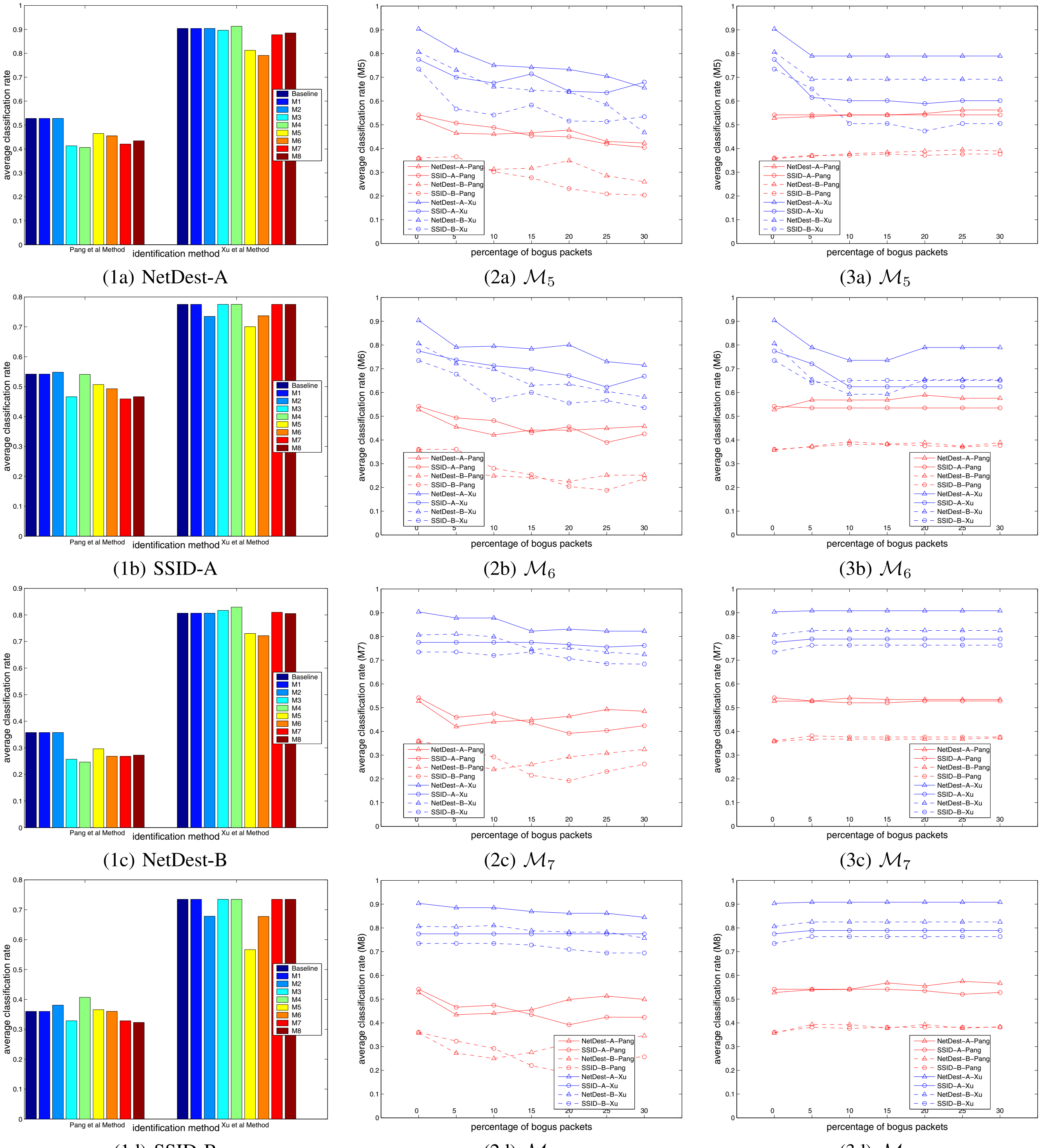
Simulation Setup and Implementation. To evaluate the effectiveness of our techniques, an extensive number of simulations has been performed. As in [1] and [2], we use the SIGCOMM 2004 802.11 network data sets [5]. Details about data sets are given in Table I. There are $49,830,432$ packets being processed via *WireShark*, to extract important attribute values such as IP addresses. We have implemented both user identification methods in [1] and [2]. In our implementation, we use Naive Bayes classifier provided by WEKA (http://www.cs.waikato.ac.nz/ml/weka/) for all classification tasks.

TABLE I
Simulation data set description

Data source	Wireless traces tcpdumped in SIGCOMM 2004
Data size	9.27GB
# Packets processed	$49,830,432$
Implicit identifiers	Network Destination & SSID
Training time period	9AM-7PM 08/31/2004 & 9AM-7PM 09/01/2004
Testing time period	9AM-7PM 09/01/2004
# Training/Testing users	10 most active users

In our simulation, we train/test the classifier over a group of users. Assume that the group size is g. We differentiate g labels: $\mathcal{U}_1, \cdots, \mathcal{U}_g$, where label $\mathcal{U}_i$ represents the instances are from ith user in the group. We define two types of user identification tasks: *Type A* classification from two users ($g = 2$) and *Type B* classification from three users ($g = 3$). The *classification rate of a user* $\mathcal{U}$ is defined as $\frac{\#\text{correctly classified instances}}{\#\text{total instances classified}}$. For each type of tasks, 10 groups of users are used and the average classification rates are reported.

Simulation Results. To evaluate the effectiveness of our proposed techniques, we did three sets of simulations. In these simulations, we generate different percentages of bogus packets, and inject them into original traffic. When we perform training, either original or mixed traffic is used. The testing is performed on mixed traffic.



(1a) NetDest-A

(2a) $\mathcal{M}_5$

(3a) $\mathcal{M}_5$

(1b) SSID-A

(2b) $\mathcal{M}_6$

(3b) $\mathcal{M}_6$

(1c) NetDest-B

(2c) $\mathcal{M}_7$

(3c) $\mathcal{M}_7$

(1d) SSID-B

(2d) $\mathcal{M}_8$

(3d) $\mathcal{M}_8$

Fig. 1. Performance results: **Left column (1a-1d)**: average classification rates of two identification methods on traffic data generated from eight proposed anonymization methods with 5% bogus packets. **Middle column (2a-2d)**: average classification rates when $\mathcal{M}_5$-$\mathcal{M}_8$ are applied with different amount of bogus packets. Here the training sets are original traffic. **Right column (3a-3d)**: average classification rates when $\mathcal{M}_5$-$\mathcal{M}_8$ are applied with different amount of bogus packets. Here, the training sets are mixed traffic.

In the first set of simulations, we study if injecting a small amount of bogus traffic can decrease the user classification rates, and thus improve the user anonymity. We first train the classifiers using the original traffic data. We then set the size of the bogus traffic to be 5% of the original traffic, and generate eight bogus traffic sets using proposed methods (*Methods* $\mathcal{M}_1$ to $\mathcal{M}_8$). We then perform the testing on the original and eight mixed traffic sets. Figures 1(a), 1(b), 1(c) and 1(d) show the average classification rates of both identification methods on these traffic sets. For comparison purpose, we mark the testing results from the original traffic sets with *baseline*, and all the other 8 testing results are marked with $\mathcal{M}_1$, $\mathcal{M}_2$, $\cdots$, $\mathcal{M}_8$, respectively. For example, Figure 1(a) shows the average classification rates for implicit identifier *Network Destination* on Type A classification task. Based on all four figures, we observe the following: (1) Xu et al.'s method [2] outperforms Pang et al.'s approach [1]. Notice that Xu et al.'s method uses more features to achieve better classification rates. (2) Pure random anonymization method ($\mathcal{M}_1$) cannot reduce the classification rates at all. Such pure random bogus traffic can be easily treated as background noise. (3) Other proposed anonymization techniques can decrease user classification rates in some cases, and hence improve user anonymity. Especially, Methods $\mathcal{M}_5$ to $\mathcal{M}_8$ can clearly reduce classification rates in most cases. For example, in Figure 1(a), the average classification rate has decreased from 0.528 (baseline) to 0.420 ($\mathcal{M}_7$) in Pang et al.'s method, and from 0.904 to 0.791 ($\mathcal{M}_6$) in Xu et al.'s method. (4) Different anonymization methods have different impacts on user anonymity for different user identification methods. For example, we observe that when Pang et al.'s method is applied for user identification, *Methods* $\mathcal{M}_7$ and $\mathcal{M}_8$ provide better user anonymity, while when Xu et al.'s method is applied, *Methods* $\mathcal{M}_5$ and $\mathcal{M}_6$ provide better anonymity. This is due to different features are used by these user identification methods.

In the second set of simulations, we study the impact of injecting different amount of bogus packets. Based on aforementioned observations, we only focus on *Methods* $\mathcal{M}_5$ to $\mathcal{M}_8$. We test the injected amount of bogus packets from 5% to 30% of the original traffic size. The simulation results are shown in Figures 2(a), 2(b), 2(c), and 2(d). In these four figures, we observe that in general, injecting more bogus data results in decreased average classification rates, which means that the user anonymity is improved. However, we also observe some exceptions, for example, in Figure 2(c), the line marked with *NetDest-A-Pang*. In addition, comparing two user identification methods, the general trend that more bogus packets result in more decreased classification rates is more consistent in Xu et al.'s method [2] than that in Pang et al.'s method [1]. This may be because the classification rates of Pang et al.'s method are almost lower than 50% for Type A and 30% for Type B. Furthermore, even in Pang et al.'s method [1], the results from implicit identifier *SSID* are less irregular than those from *Network Destination*. These observations largely depend on the data sets.

In the third set of simulations, we perform both training and testing using mixed traffic. The simulation results are shown in Figures 3(a), 3(b), 3(c), and 3(d). We have multiple observations from these four figures. (1) When applying *Methods* $\mathcal{M}_5$ and $\mathcal{M}_6$, Xu et al.'s method generally follows the trend that more bogus packets result in more decreased classification rates (note the exceptions exist). (2) When applying *Methods* $\mathcal{M}_7$ and $\mathcal{M}_8$, injecting more bogus packets almost does not affect the results from Xu et al.'s method. This is because in Xu et al.'s method, the bogus packets generated from *Methods* $\mathcal{M}_7$ and $\mathcal{M}_8$ may not change the top list of features selected from the data sets. (3) The lines related to Pang et al.'s method are quite flat, which means that there are no significant changes in classification rates. Note that these results may be largely data set dependent.

V. Conclusions and Future Work

Based on the concept of implicit identifiers, both Pang et al. [1] and Xu et al. [2] propose techniques to identify the users in Wireless LANs. Assuming that implicit identifier based user identification is applied, this paper focuses on the techniques that can decrease user identification rates, and thus improve the user anonymity. More specifically, we propose eight different anonymization methods to generate bogus packets, and then inject them into original traffic to decrease the possibility of users being identified. Our simulation results with SIGCOMM 2004 wireless data sets demonstrate that several of our anonymization methods are effective. There are several directions that we will explore in our future research: (1) discover and study more implicit identifiers, and (2) investigate more anonymization methods to help improve user anonymity from various user identification methods.

References

[1] J. Pang, B. Greenstein, R. Gummadi, S. Seshan, and D. Wetherall, "802.11 user fingerprinting," in *Proc. of ACM MobiCom 07*, 2007.

[2] D. Xu, Y. Wang, and X. Shi, "Enhanced feature selection and generation for 802.11 user identification," in *Proc. of IEEE ICCCN 09*, 2009.

[3] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 46–55, 2003.

[4] M. Gruteser and D. Grunwald, "Enhancing location privacy in wireless LAN through disposable interface identifiers: a quantitative analysis," *Mob. Netw. Appl.*, vol. 10, no. 3, pp. 315–325, 2005.

[5] M. Rodrig, C. Reis, R. Mahajan, D. Wetherall, J. Zahorjan, and E. Lazowska, "CRAWDAD data set uw/sigcomm2004 (v. 2006-10-17)," Downloaded from http://crawdad.cs.dartmouth.edu/uw/sigcomm2004.

[6] D. B. Faria and D. R. Cheriton, "Detecting identity-based attacks in wireless networks using signalprints," in *Proc. of ACM Workshop on Wireless Security*, 2006.

[7] T. Kohno, A. Broido, and K. C. Claffy, "Remote physical device fingerprinting," in *Proc. of IEEE Symp. on Security and Privacy*, 2005.

[8] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," in *Proc. of ICDCS*, 2005.

[9] M. Shao, Y. Yang, S. Zhu, and G. Cao, "Towards statistically strong source anonymity for sensor networks," in *Proc. of INFOCOM*, 2007.

[10] M. Lei, Z. Qi, X. Hong, and S. Vrbsky, "Protecting location privacy with dynamic mac address exchanging in wireless networks," in *Proc. of IEEE Conf. on Intelligence and Security Informatics*, 2007.

[11] W. Shbair, A. Bashandy, and S. Shaheen, "A new security mechanism to perform traffic anonymity with dummy traffic synthesis," in *Proc. of Int'l Conf. on Computational Science and Engineering (CSE 09)*, 2009.

[12] O. Berthold and H. Langos, "Dummy traffic against long term intersection attacks," in *Proc. of Privacy Enhancing Technologies workshop (PET)*, 2002.