A Correlation Attack Against User Mobility Privacy in a Large-scale WLAN Network

Keren Tan
Dept. of Computer Science
ISTS
Dartmouth College
keren@cs.dartmouth.edu

Guanhua Yan
Information Sciences (CCS-3)
Los Alamos National Laboratory
ghyan@lanl.gov

Jihwang Yeo, David Kotz
Dept. of Computer Science
ISTS
Dartmouth College
jyeo@cs.dartmouth.edu
kotz@cs.dartmouth.edu

ABSTRACT

User association logs collected from real-world wireless LANs have facilitated wireless network research greatly. To protect user privacy, the common practice in sanitizing these data before releasing them to the public is to anonymize users' sensitive information such as the MAC addresses of their devices and their exact association locations. In this work, we demonstrate that these sanitization measures are insufficient in protecting user privacy from a novel type of correlation attack that is based on CRF (Conditional Random Field). In such a correlation attack, the adversary observes the victim's AP (Access Point) association activities for a short period of time and then infers her corresponding identity in a released user association dataset. Using a user association log that contains more than three thousand users and millions of AP association records, we demonstrate that the CRF-based technique is able to pinpoint the victim's identity exactly with a probability as high as 70%.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Network Operations—*network monitoring*; I.5 [**Pattern Recognition**]: Design Methodology—*pattern analysis*

General Terms

Algorithms, Measurement, Security

Keywords

privacy, mobility, wireless network, sanitization, anonymization, Conditional Random Field

1. INTRODUCTION

Network traffic traces collected from production networks have played a critical role in understanding user activity patterns, analyzing network protocol dynamics, and evaluating the performance, reliability, and security of new network designs. We, at Dartmouth College, have monitored a campus-wide wireless local-area network (WLAN) for almost one decade and some of our datasets have been made public through our CRAWDAD archive [1]. By their nature, these traces captured from production networks inevitably contain private or proprietary information about the target networks and their users. To preserve users' privacy, we need to *sanitize* the network traces before sharing them with the public. Many sanitization techniques have been proposed and developed to make the traces resistant against host or user re-identification attacks [3,4]. These techniques mainly focus on truncating or randomizing sensitive fields in the datasets that may be explicitly linked to user identity, such as IP/MAC addresses, or contain sensitive information that users may not be willing to share with others, such as TCP/UDP payloads.

In this work, we demonstrate that a new type of privacy attack, which we call *CRF-based correlation attack*, can be launched to compromise a victim's mobility privacy from user association logs. In such a correlation attack, the adversary first downloads a sanitized user association log released in the past. All users' MAC addresses in this log are sanitized and thus unknown to the adversary. Provided that the adversary observes a sequence of AP association records of the target victim for a short period of time,[1] his goal is to infer the MAC address from the released dataset that is associated with the victim. If done correctly, the attacker obtains broader knowledge of the victim's mobility history from the released dataset, which leads to an infringement on the privacy of the user.

2. WLAN USER ASSOCIATION LOGS

WLAN user association logs record where and when a user has connected to the network. Each row of the user association log has four comma-separated fields: the MAC address of the wireless card, the name of the AP that the wireless card has connected with, and the start and end POSIX timestamp. The following is a snippet of the user association log we extract from the SNMP information:[2]

S3'10, September 20, 2010, Chicago, Illinois, USA.

[1] Such knowledge can be obtained either through a trojan installed on the victim's machine, or by following the victim and observing her activities (e.g., when she opens or closes her laptop lid), or from a separately released non-sanitized dataset.

[2] The displayed MAC addresses have been anonymized to protect user privacy.

```
001d4f3bc496, 14.5.1,  1251690285, 1251691544
002608e4cdf7, 80.3.2,  1251690458, 1251691544
0021e9082bfd, 142.6.1, 1251689384, 1251691544
```

Two things are worth noting. First, because the Aruba Mobility Controller only generates the start timestamp for each connection and we poll the controller every 5 minutes, the connection's end timestamp is only an estimated value, whose error is therefore bounded by 5 minutes. Second, we use a hierarchical naming scheme for APs in the dataset. For an AP named $x.y.z$, x is its building number, y is its floor number, and z is its serial number within the floor.

Sanitization. When sanitizing the user association logs, we use a one-to-one mapping function to rename the MAC addresses in the original dataset. Hence, the anonymized MAC addresses in the sanitized dataset do not have any physical meaning and are thus only symbolic names. By taking advantage of its hierarchical naming scheme, we truncate an AP's name according to different sanitization levels. For example, if we want to only keep building and floor information, we truncate the AP's name from $x.y.z$ to $x.y$.

3. THREAT MODEL

Given the simple sanitization scheme discussed in the previous section, there is one key question: is it possible to obtain private information from the user association log if it is released to the public? To answer this question, we first define the three assumptions in our threat model, which presents the capabilities and knowledge required by the attacker to succeed.

Assumption 1: The adversary has access to a sanitized WLAN user association log $\mathcal{L}_s$, which is shared to the public by a trace publisher. There are N_s users in this log. All users' real MAC addresses are anonymized in $\mathcal{L}_s$ using a one-to-one one-way mapping function: each real MAC address has been replaced with a new identifier ID_i ($1 \leq i \leq N_s$). Hence, given an anonymized MAC address ID_i, the adversary cannot find the real MAC address that is mapped to ID_i. The AP's name can be either preserved or truncated. The rest of the fields, such as the start and end timestamp of each connection, are preserved during the sanitization process.

Assumption 2: The adversary knows a sequence of association records about a victim user's device, denoted by $\mathcal{Q}$. This sequence of records, $\mathcal{Q}$, need not be collected during the same time period as $\mathcal{L}_s$ (otherwise the problem will be trivial). The information provided in $\mathcal{Q}$ can be rather coarse. For example, the adversary may only need to know which building the victim has visited instead of which exact AP the victim has associated with.

Assumption 3: The adversary knows that the sanitized dataset $\mathcal{L}_s$ must contain the victim's AP association records. In many cases, $\mathcal{L}_s$ is published at an organization level (e.g., by a university) and thus contains complete AP association logs of the organization's wireless users. Hence, if the adversary knows that the victim was a member of the organization when $\mathcal{L}_s$ was collected, it is easy for him to know that $\mathcal{L}_s$ should contain the victim's AP association records.

Given the three assumptions in the adversarial model, the *(exact) correlation attack problem* is then formulated as follows: *given $\mathcal{L}_s$ and $\mathcal{Q}$, which anonymized identity ID_i ($1 \leq i \leq N_s$) in $\mathcal{L}_s$ has also generated $\mathcal{Q}$?* In practice, however, due to incomplete data for training or inference, or some intra- and inter-user association activity variations, finding an algorithm to solve the exact correlation attack problem is difficult or even infeasible. In this work, we consider a relaxed version of this problem. The *(relaxed) correlation attack problem* is formulated as follows: *given $\mathcal{L}_s$ and $\mathcal{Q}$, which subset of anonymized identities would contain the one that generated $\mathcal{Q}$ with high probability?*

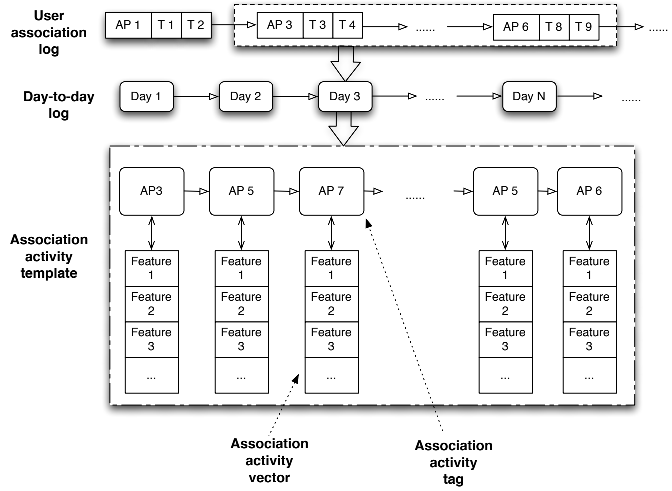

Figure 1: Represent an user's association log using association activity template

4. ALGORITHM DESCRIPTION

The intuition behind the proposed algorithm is that different users may have different association behaviors, and the differences between these behaviors are distinguishable and can be modeled from the temporal and spatial information contained in the user association logs. For example, Tom often stays in the library in the morning for 2 hours and then goes to the classroom around 3pm in the afternoon, while Jack spends most of day in the lab and goes back to his dormitory at late night. We aim to build a model that not only can characterize such inter-user differences but also is robust to intra-user variations.

4.1 Data Representation

We propose a new approach that uses *association activity templates* to represent user association logs. In this method, we first split the user's association log into day-to-day pieces and then for each day build an individual association activity template. An association activity template is a collection of association activity tags and their corresponding association activity vectors. As shown in Figure 1, the association activity tag is the name of the visited AP. Each element in an association activity vector is called a *feature*. In the current implementation, we let an activity vector have six features: *duration*, *day of week*, *starting time*, *previous AP*, *next to previous AP*, and *next AP*.

4.2 Algorithm Procedure

We now discuss the steps in launching correlation attacks against released a dataset $\mathcal{L}_s$ using an observed sequence $\mathcal{Q}$. The procedure is summarized as follows:

Step 1. For each user in $\mathcal{L}_s$, split his/her association log into day-to-day pieces and represent each day's log using an association activity template as described in Section 4.1.

Step 2. Feed each user's association activity templates into a linear-chain CRF [2] to model this user's association behavior. As there are N_s users in $\mathcal{L}_s$, we build N_s CRF models. The input fed to a CRF model is a sequence of association activity vectors (Figure 1) and the output is a sequence of association activity tags, which are actually AP names. Let $CRF_i(\mathcal{V})$ denote the output from the i-th user's CRF model, where $1 \leq i \leq N_s$ and $\mathcal{V}$ denotes the sequence of association activity vectors fed to the CRF model.

Step 3. For the observed AP association sequence $\mathcal{Q}$, we preprocess it as described in Section 4.1 to obtain an association activity template $\mathcal{T}$. Let $\mathcal{V}_\mathcal{T}$ and $\mathcal{G}_\mathcal{T}$ denote the sequence of association activity vectors and the sequence of association activity tags in template $\mathcal{T}$, respectively.

Step 4. We feed $\mathcal{V}_\mathcal{T}$ to all CRF models trained in Step 2 and count the number of tags that overlap between $\mathcal{G}_\mathcal{T}$ and $CRF_i(\mathcal{V}_\mathcal{T})$ $(1 \leq i \leq N_s)$, a score we denote w_i. The intuition applied here is that the victim's CRF model is more likely to produce correct activity association tags from her observed activity association vectors in $\mathcal{Q}$, and therefore score w_i is higher than the others if ID_i is the victim's identifier in the released AP association dataset.

Step 5. We sort all the users based on score w_i in non-increasing order and the algorithm outputs this sorted list.

Ideally, the top identifier on the sorted list should be treated as the sole candidate that generated the observed AP association sequence $\mathcal{Q}$. As mentioned in Section 3, due to various limitations in practice, the top identifier may not correspond to the victim who produced $\mathcal{Q}$. Thus, we tackle the relaxed correlation attack problem instead and consider a small number of top identifiers on the sorted list as possible candidates. Clearly, from the attacker's perspective, the smaller the number of top identifiers needed to include the victim's, the more successful his attack is.

5. EXPERIMENTAL EVALUATION

We used a 62-day user association log collected between January 4, 2010 and March 6, 2010, corresponding to one academic term, in this evaluation. Because Dartmouth's wireless network is open to anyone, we eliminated those *transient* users who were active fewer than 50 days or who connected with fewer than 10 unique APs, and the resulting dataset contained 2,179,671 association records, 3,313 distinct users, and 1,364 distinct APs. We partition this into 10 bins of approximately the same length for each user. We perform 10 rounds of experiments. In the j-th round $(1 \leq j \leq 10)$, we use the j-th bin of each user's AP association records as the testing dataset $(\mathcal{L}_u)$ and the remaining nine as the training dataset $(\mathcal{L}_s)$ to build the CRF models. The results shown here are the 10-round averages. Under this configuration, the length of a victim's AP association sequence observed by the adversary, $\mathcal{Q}$ $(\mathcal{Q} \in \mathcal{L}_u)$, is 5-6 days. This short trace is all the knowledge needed by the adversary about the victim. We use the *minimum size of candidate identifier set* (MSCIS) as the metric to measure the attack efficiency. Consider the relaxed correlation attack problem with a sanitized user association dataset $\mathcal{L}_s$ and an observed sequence of AP association records $\mathcal{Q}$. For each ID_i where $1 \leq i \leq N_s$ in $\mathcal{L}_s$, we compute score w_i according to Step 4 in the CRF-based method. Suppose that ID_j is the user ID of the victim who generated $\mathcal{Q}$. The MSCIS is defined as the number of user IDs whose scores are no smaller than w_j. MSCIS establishes an upper bound on how many candidate user IDs need be considered in order to contain the victim's user ID in the sanitized dataset.

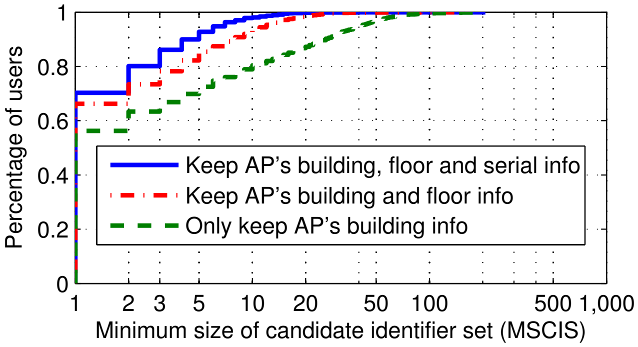

Figure 2: Evaluation of the CRF-based method

We apply the proposed method to three datasets with different sanitization strengths. Figure 2 shows that when AP's building, floor and serial information are preserved, 70.28% of 3,313 users could be pinpointed exactly from $\mathcal{L}_s$ (MSCIS = 1). In the worst case that only the building information is preserved, 56.25% users are still pinpoint-able.

Due to lack of space here, further details about the proposed approach and more comprehensive evaluation results will be presented in a subsequent paper.

6. ACKNOWLEDGMENTS

This paper results from a research program in the Institute for Security, Technology, and Society (ISTS), supported by the U.S. Department of Homeland Security under Grant Award Number 2006-CS-001-00000, and by the NetSANI project at Dartmouth College, funded by Award CNS-0831409 from the National Science Foundation. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security or the National Science Foundation.

7. REFERENCES

[1] Community Resource for Archiving Wireless Data At Dartmouth (CRAWDAD). `http://www.crawdad.org/`.

[2] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2001.

[3] J. Pang, B. Greenstein, R. Gummadi, S. Seshan, and D. Wetherall. 802.11 user fingerprinting. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2007.

[4] R. Pang, M. Allman, V. Paxson, and J. Lee. The devil and packet trace anonymization. *ACM SIGCOMM Computer Communication Review*, 36(1):29–38, 2006.