Poster: Crowdsourced Location Aware Wi-Fi Access Control

Bingxian Lu, Zhicheng Zeng, Lei Wang
Dalian University of Technology
{bingxian.lu, z.alex.zeng}@gmail.com
lei.wang@dlut.edu.cn

Brian Peck, Daji Qiao
Iowa State University
{bpeck, daji}@iastate.edu

ABSTRACT

In recent years, Wi-Fi has seen extraordinary growth; however, due to the cost, performance and security issues, many Wi-Fi hotspot owners would like to restrict the network access only to individuals inside the physical property. Unfortunately, due to the nature of wireless, this is difficult to accomplish, especially with the off-the-shelf omni-antenna devices. In this work, we develop and implement CLaWa, a Crowdsourced Location Aware Wi-Fi Access Control scheme to address this challenge. Our system is based on observations of differing characteristics of physical layer information across physical boundaries such as walls and corners. CLaWa crowdsources both channel state information (CSI) and received signal strength (RSS) of already validated users to classify future users. We have also selected an appropriate machine learning algorithm for CLaWa. Evaluation results show that CLaWa can identify the boundary around a given area precisely, thus granting network access only to users inside the area while not validating users outside the boundary. Compared to indoor localization schemes, CLaWa is a lightweight solution which does not require expensive localization operations.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication

Keywords

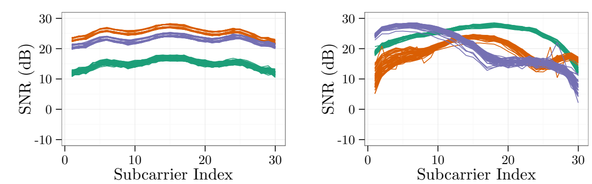
Crowd Sourcing, User Validation, IEEE 802.11 WLAN

1. INTRODUCTION

Currently, many businesses and public areas seek to attract customers and patrons by offering Internet access through Wi-Fi Access Points (APs). However, as providing this access is not cheap, a business may wish to restrict Wi-Fi access only to paying customers, or restrict access to customers inside the physical space.

This work is supported by the Haitian Scholar Grant of Dalian University of Technology, and NSF China grant under No. 61272524. Lei Wang is the corresponding author.

MobiCom'15, September 7–11, 2015, Paris, France.
ACM 978-1-4503-3619-2/15/09.
DOI: http://dx.doi.org/10.1145/2789168.2795183.



(a) Close to AP (b) Far from AP

Figure 1: CSI Amplitude values at the AP. We observe that close, stationary users result in high correlation coefficients.

According to our observations, CSI variance correlates with multipath effect and movement status. When a user is stationary and close enough to an AP, the corresponding CSI values on different antennae will be similar and stable between frames, as in Figure 1a, where each line in a given color (antenna) represents a single frame.

CLaWa first uses CSI data to determine if a user is in direct proximity of an AP; if so, the user is classified as a valid user. We also collect RSS data from each user to form a fingerprint. As valid users move around the area, we feed their RSS data into a machine learning algorithm, which quickly populates a training set. This crowdsourcing allows CLaWa to quickly identify future valid users. Finally, users that are incorrectly classified as being outside the area (false negatives) could then obtain an access code which would add their device to a whitelist, granting them accesses on future validation attempts.

Different from indoor localization solutions [4] and some works relying on special hardware [3], CLaWa attempts to address a distinct but related problem, while not requiring accurate localization of each user.

Therefore, compared with indoor localization schemes, CLaWa is a lightweight solution which does not require expensive localization operations or supporting devices.

2. SYSTEM DESIGN

2.1 System Overview

The design of CLaWa is outlined in Figure 2. The user's RSS and CSI data is collected and fed into a model which outputs a probability P that the user should be classified

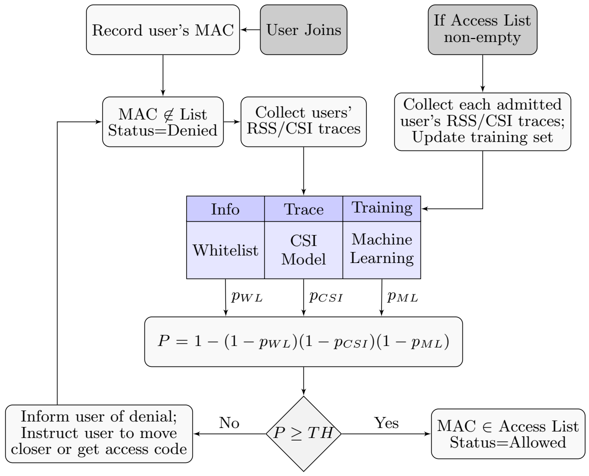


Figure 2: Design Overview

as a valid user. We then compare P to a threshold TH to determine if this user should gain access.

The core of CLaWa is a set of three parallel prediction components, which each determines a probability that a particular user is valid. The results of each component are then combined as:

$$P = 1 - (1 - p_{WL})(1 - p_{CSI})(1 - p_{ML}) \tag{1}$$

where each p is the probability of positive classification for each component.

2.2 Whitelist Component

The Whitelist component simply outputs a binary probability p_{WL} to signify whether or not a user has been added to a whitelist. If a user enters an access code, we know that they are a valid user of the system and should be given access, and thus they are added to a whitelist.

2.3 Channel State Information Component

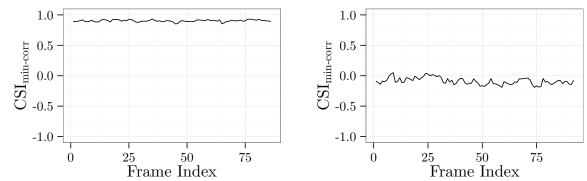
The CSI component outputs the probability p_{CSI} that a user is in the direct proximity of an AP, and thus in target area. To do this, we use a simple CSI model based on data collected at the AP. Each data point is the received SNR value and is recorded as $H_j^{i,k}$ which represents the j-th frame collected by the i-th antenna on the k-th subcarrier. A group of each SNR value across all subcarriers for one frame j on a single antenna i can be represented as CSI_j^i.

2.3.1 *CSI Stability Measurements*

Since we wish to capture both the stability of CSI values over time, as well as the similarity of CSI values between antennae, we propose a two step process to determine a CSI stability metric. We first measure the correlation of CSI values between antennae for successive frames using the Pearson correlation coefficient, as in:

$$CSI_{corr}^{a,b} = \frac{\text{cov}(CSI_j^a, CSI_{j+1}^b)}{\sigma_{CSI_j^a} \cdot \sigma_{CSI_{j+1}^b}}, \forall a, b \in \{1, 2, 3\}, \tag{2}$$

where cov is the covariance between successive CSI_j^i readings, and σ is the standard deviation of the respective values. This yields 9 total correlation coefficients, one for every combination of antenna pair. We then use the minimum of these values as our final CSI stability metric for a given frame, as



(a) Close to AP (b) Far from AP

Figure 3: $CSI_{min-corr}$ at the AP.

in:

$$CSI_{min-corr} = \min_{\forall a,b \in \{1,2,3\}} (CSI_{corr}^{a,b}). \tag{3}$$

2.3.2 *User Recognition*

When a user connects to an AP, it will measure its CSI and give a $CSI_{min-corr}$ for each frame according to the above equations. Intuitively, we expect $CSI_{min-corr}$ to be large (close to 1) when the user is near the AP. We count the number of frames with $CSI_{min-corr}$ above the threshold 0.5 and use the ratio of these frames to the total number of frames in a measurement period to determine the likelihood that a user is close to the AP, as in:

$$p_{CSI} = \frac{\sum_{j=1}^{N} [CSI_{min-corr,j} > 0.5]}{N}. \tag{4}$$

We continue the example from Figure 1a, and look at all frames as shown in Figure 3a. In this case, every $CSI_{min-corr,j}$ for every frame is above the threshold, and thus $p_{CSI} = 1$. As seen from Figure 1b, the shapes of the lines are not similar, and thus the $CSI_{min-corr,j}$ values in Figure 3b are low.

2.4 Machine Learning Component

The Machine Learning component outputs a probability p_{ML} that a user is in the target area according to calculations by a machine learning algorithm. We use the One-Class Support Vector Machine (OSVM) [2] algorithm, where our one class is valid users.

CLaWa uses RSS data from valid users to build a training set in real time. When a user device sends out N_{total} frames in a measuring period, and by OSVM there are $N_{outliers}$ frames which are classified into the negative class, then the machine learning component outputs a probability given by

$$p_{ML} = \frac{N_{total} - N_{outliers}}{N_{total}}. \tag{5}$$

3. IMPLEMENTATION

We provide a working implementation of CLaWa in order to fully evaluate our scheme. The CLaWa implementation consists of two complementary components: *CLaWa Access Point and Monitor*, and *CLaWa Server*, as depicted in Figure 4. The AP, along with passive CLaWa monitors, collects RSS and CSI information using a modified `iwlwifi` driver by employing the Linux 802.11n CSI Tool [1]. The server primarily uses `iptables` to control access for connected users.

4. EVALUATION

In the experiment, we use three Lenovo ThinkPad T400 laptops as the AP and monitors shown in Figure 5. Each

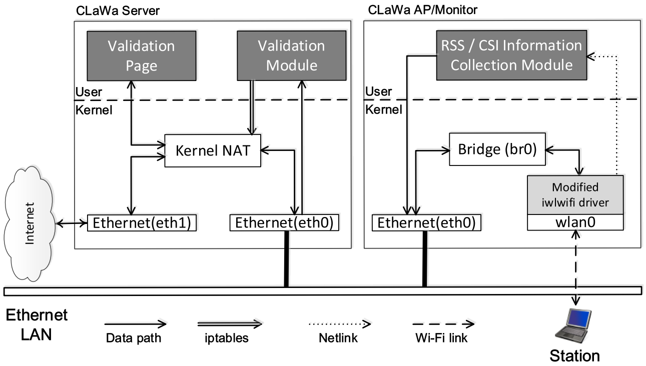
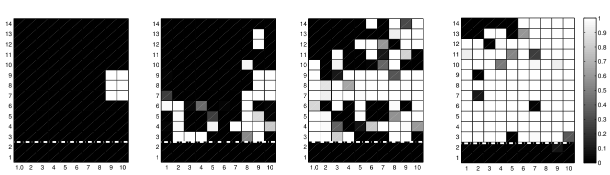


Figure 4: System Architecture

Figure 5: Thinkpad T400 with Intel 5300 NIC

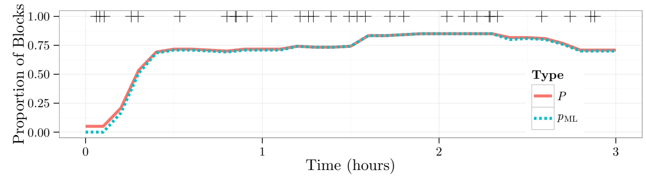
laptop runs a 32-bit Ubuntu Server 10.04 (LTS) operating system, and is configured with the Intel 5300 802.11n wireless network card. We set up a testbed in a $6.5m \times 8m$ laboratory with 8 work cubes and 8 tables. The laboratory has a concrete wall of 0.5 meters, and the door is open throughout the experiments.

We partition up the area into 140 $60cm \times 60cm$ blocks, as shown in Figure 6, with 120 blocks inside the room (top 12 rows above the dotted line), and 20 blocks outside the room (bottom two rows below the dotted line). We show the output of the entire scheme at each block over time in Figure 6. We show the probability of validation at each test block at 0, 10, 20, and 120 minutes. Initially, the CSI model outputs high validation probability near the AP, allowing the entire scheme to output high probability. This effectively bootstraps the system for the Machine Learning component to begin working, which learns of valid locations as users travel around the room over time. As a result, the Machine Learning algorithm, and thus the entire algorithm, output high validation probabilities for an increasing number of blocks. After two hours, nearly the entire room is correctly identified. Additionally, the 20 blocks outside the room always have a very low validation probability. This confirms that CLaWa is able to correctly identify the boundary precisely around the room.

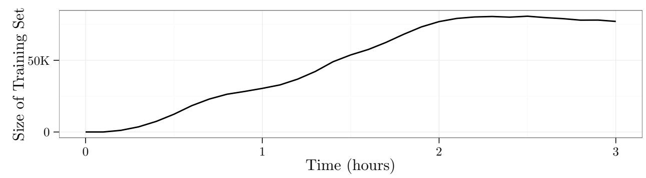


(a) 0 min (b) 10 mins (c) 20 mins (d) 120 mins

Figure 6: Probability P of validation for each test block. Lighter colors signify higher probabilities.



(a) Proportion for both P and p_{ML}. The black ticks represent user arrival times.



(b) Training Set Size

Figure 7: Result with 30 users in given order.

Given the blocks above, we show the proportion of blocks with $p_{ML} \geq TH$ and $P \geq TH$ in Figure 7a. We also show the size of the training set over time in Figure 7b. In both cases, we see that the blocks with high probability increases along with the size of the training set. Specifically, the proportion increases very rapidly at the beginning of the simulation, as many users are dwelling in new locations. After the initial period, the proportion remains relatively stable until new locations are visited around 1.5 hours. Finally, after about two hours, the proportion stabilizes since the training window W_T is two hours.

5. CONCLUSIONS

We present and implement CLaWa, which enables a location to offer Wi-Fi access to users inside a particular boundary, while preventing users outside the boundary to connect to the network. Specifically, CLaWa first recognizes valid users within direct proximity of the AP by leveraging the rapid de-correlation of CSI between multiple antennae as the distance between the user and the AP increases. Using this data, we then utilize crowdsourcing to build a training set from valid users, based on which we recognize additional valid users over time. As a result, CLaWa is able to successfully recognize valid users, while denying access to invalid users. Future work includes adding better boundary detection for varied settings and environments.

6. REFERENCES

[1] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Tool Release: Gathering 802.11n traces with channel state information. *ACM SIGCOMM CCR*, 41(1), 2011.

[2] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[3] A. Sheth, S. Seshan, and D. Wetherall. Geo-fencing: Confining Wi-Fi coverage to physical boundaries. In *Springer Pervasive, LNCS*, volume 5538, 2009.

[4] Z. Yang, Z. Zhou, and Y. Liu. From RSSI to CSI: Indoor localization via channel response. *ACM Computing Surveys (CSUR)*, 46(2), 2013.