This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE CCNC 2006 proceedings.

Active Fingerprinting of 802.11 Devices by Timing Analysis

Bartlomiej Sieka
Computer Science Department
University of Illinois at Chicago
bsieka@cs.uic.edu

***Abstract*— This paper presents a method for identifying 802.11 nodes. It utilizes precise timing measurements together with statistical methods and the Support Vector Machine (SVM) classifier to classify radio communication as coming from a given transceiver. It does not require any specialized hardware, and the identification is involuntary. The paper describes the experiment performed and presents some initial classification results. Five different 802.11 devices are fingerprinted with the success rate of 86%.**

I. Introduction

Wireless networks are becoming increasingly widespread, in both business and private environments. It is clear then that the security of wireless networks is quickly becoming a vital issue. There are inherent differences between wireless and wire-line networks. For this reason, there exist security issues in the former for which solutions found in the latter cannot be applied. One such issue is node identification. In particular, it is a challenging problem to determine whether a given wireless communication originated from some particular physical node.

A network node can be thought of as a hardware device, for example, a Network Interface Card (NIC). Any two hardware devices do exhibit physical differences that are bound to manifest themselves in the way the devices communicate. This paper seeks to explore the existence and nature of such differences in order to uniquely characterize wireless network nodes. The proposed method is to perform precise measurements of the time it takes for a node to perform various communication functions. In particular, this paper has two objectives. The first is to present experimental evidence that the differences between 802.11 devices (i.e., devices conforming to the IEEE 802.11 standard, [1]) are observable by precise timing measurements of their communication. The second objective is to show how such a timing approach can be used to reliably identify 802.11 devices.

The paper is organized as follows. Section II presents the prior art and describes the motivation for this work. Section III contains the description of the experiment performed. Section IV presents the classification method and its results. Section V discusses the results and the approach taken. Conclusions are found in Section VI.

II. Motivation and Prior Art

Wireless transmitter fingerprinting has many potential security applications. For one, the military is certainly interested in identifying wireless nodes that are transmitting. However, a reliable method of identifying an 802.11 node can also be valuable in civilian applications. Having a robust, low-cost method for associating a hard-to-repudiate identity with a wireless transmission can be used, for example, to devise an authentication system for a wireless network [2]. It can also be applied to build an intrusion detection system [3] or a mechanism to prevent impersonation attacks (spoofing). An important property of fingerprinting is that it is involuntary, meaning that the node being fingerprinted cannot affect the outcome. This is due to the fact that the physical characteristics of the device are used, hence modifying the fingerprint would require a physical modification to the device. This in turn would produce a different physical object with, obviously, a different fingerprint.

Fingerprinting techniques were first developed by the army. The method called Radio Frequency Fingerprinting (RFF) is believed to have existed during the Cold War and has been de-classified only in the mid-1990s [4]. It uses the physical properties of the signal (amplitude, phase) during the turn-on transient of the transmitter. It was successfully used by cellular companies to detect fraud (cell-phone cloning) [4]. It is also successful with other wireless technologies, for example, Bluetooth [5] or 802.11 networks [3].

RFF methods do however have a serious drawback. They require a dedicated hardware to measure the physical properties of the radio signal. The additional cost of such equipment may be prohibitive for many applications. Hence this paper proposes a method that does *not* require any additional hardware beyond the NIC that is already in place. It focuses on one specific family of wireless networks, namely, the ones implementing the IEEE 802.11 protocol suite. For simplicity of presentation, the terms *802.11 node* and *wireless node* are used interchangeably.

The next section describes the experiment performed and also presents the analysis of the gathered data.

III. Experiment

The experiment focuses on the MAC sub-layer of the data link layer of the 802.11 protocol. The goal is to identify a particular 802.11 transmitter by precisely measuring timings for the MAC sub-layer communications. The timings related to the authentication procedure described in [1] are measured.

1-4244-0086-4/06/$20.00 ©2006 IEEE.

Refer to Figure 1 for the exact radio frames exchanges involved.

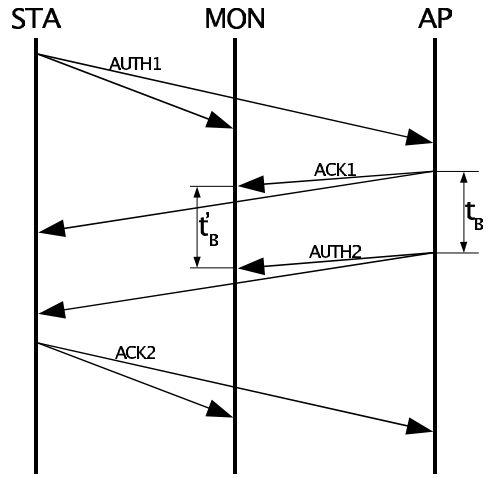

Fig. 1. Radio frames exchange during the authentication procedure

The focus is on t_B, i.e., the time that elapses between the moment the first acknowledgment is sent (ACK1) and the moment the authentication response in sent (AUTH2). However, in principle it is not possible to measure time t_B, as it would require access to the node being fingerprinted. What is measured instead, is time $t_B^{'}$. The relationship between t_B and $t_B^{'}$ is further discussed in Section V.

The wireless network used for the experiment is depicted in Figure 2. The devices are set up to operate in the infrastructure mode, and the access point (AP) is the target of the fingerprinting. Both the station node (STA) and the monitor node (MON) are under the control of the entity performing the fingerprinting. The MON device is in monitor mode and records all 802.11 frames that it receives, together with a timestamp. The STA device initiates the authentication procedure by sending the first authentication frame (AUTH1). STA does this repeatedly, once about every 2ms (depending on the operating system scheduling on the STA).

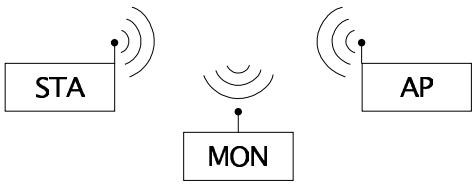

Fig. 2. Network setup

The experiment is performed with five different wireless NICs for the AP, denoted as AP1, AP2, AP3, AP4, and AP5. Please refer to Table I for hardware details of the NICs used as APs.

The first phase of the experiment is to perform a large number of authentication procedures, and for each one, to extract the time $t_B^{'}$. Such an extracted time is referred to as *primary sample*, and there were 6,700,000 primary samples gathered for every AP. They were recorded only for valid frame sequences exchanged between STA and AP, thus filtering out other radio traffic from the data. The next step is to split the set of all primary samples into *chunks*. In the current setup, each chunk has 50,000 primary samples. The intention is to transform the data in a chunk into a fixed-size vector that can be used as a *sample* in the classification process. So for every chunk, the mean of primary samples is computed, and then the difference between each sample and the mean is calculated. Refer to the first two graphs of Figure 3 for an example of the resulting "deviations". Visually, the samples appear to be different, but it is not straightforward to characterize the differences. It should be noted that it is very important to transform the chunk deviations into a vector in a way that captures statistical properties that are specific to a given AP. The approach taken is to compute a relative histogram of the deviations. More specifically, a histogram of 135 equally sized bins is used. The resulting vector is called a *sample*. The third graph on Figure 3 shows samples corresponding to the first two graphs, plotted as histograms. It is apparent that the differences in the deviations are indeed captured by the histograms, as the shapes of the two histograms are different. The purpose of these graphs is to help the reader with the intuitions about the statistical properties of the primary samples, and about how they are converted to the samples that are classified.

The next section describes the classification process.

IV. Classification

The goal of this paper is to demonstrate that the physical differences between 802.11 nodes can be observed in the timings of the radio frames. Such a claim can be supported by showing a formal method that, given a sample, can determine which AP that sample comes from. In other words, if there exists a classifier with sufficient accuracy, then one can assume that the claim is true.

The Support Vector Machine (SVM: [6], [7]) approach was employed to classify the samples. In particular, the LIBSVM software library ([8]) was used.

Due to space limitation, the details of the SVM classification process are omitted, and the focus is on the classification results. The accuracy of the classifier can be evaluated by computing the *estimated classification rate* (ECR) for test samples, that is, the samples not used in the training. ECR is the proportion of correctly classified test samples. It gives the estimate of the probability of correct classification. The higher the value of ECR, the better the classifier. ECR however does not convey information about how the classification works with respect to different classes. This information can be obtained by computing the *estimated classification rate matrix* (ECRM). ECRM is a matrix that contains ECRs computed per class. In the ideal case, ECRM has the form of the identity matrix. The rows of a ECRM correspond to classes, i.e., to APs. For example, values in the second row denote, in the order they appear in the row, the probability of a sample

AP	Brand	Model	Chipset	Hardware/Firmware Version	Comments
AP1	D-Link	DWL-G520	Atheros 5212	H/W Ver.: B3, F/W Ver.: 4.10	visually identical to AP2 and AP4
AP2	D-Link	DWL-G520	Atheros 5212	H/W Ver.: B3, F/W Ver.: 4.10	visually identical to AP1 and AP4
AP3	D-Link	DWL-G520	Atheros 5212	H/W Ver.: B3, F/W Ver.: 4.10	markings different than AP1, AP2, AP4
AP4	D-Link	DWL-G520	Atheros 5212	H/W Ver.: B3, F/W Ver.: 4.10	visually identical to AP1 and AP2
AP5	Netgear	WG311T	Atheros 5212	Rev. A2	visually different than others

TABLE I

HARDWARE DETAILS OF THE APS.

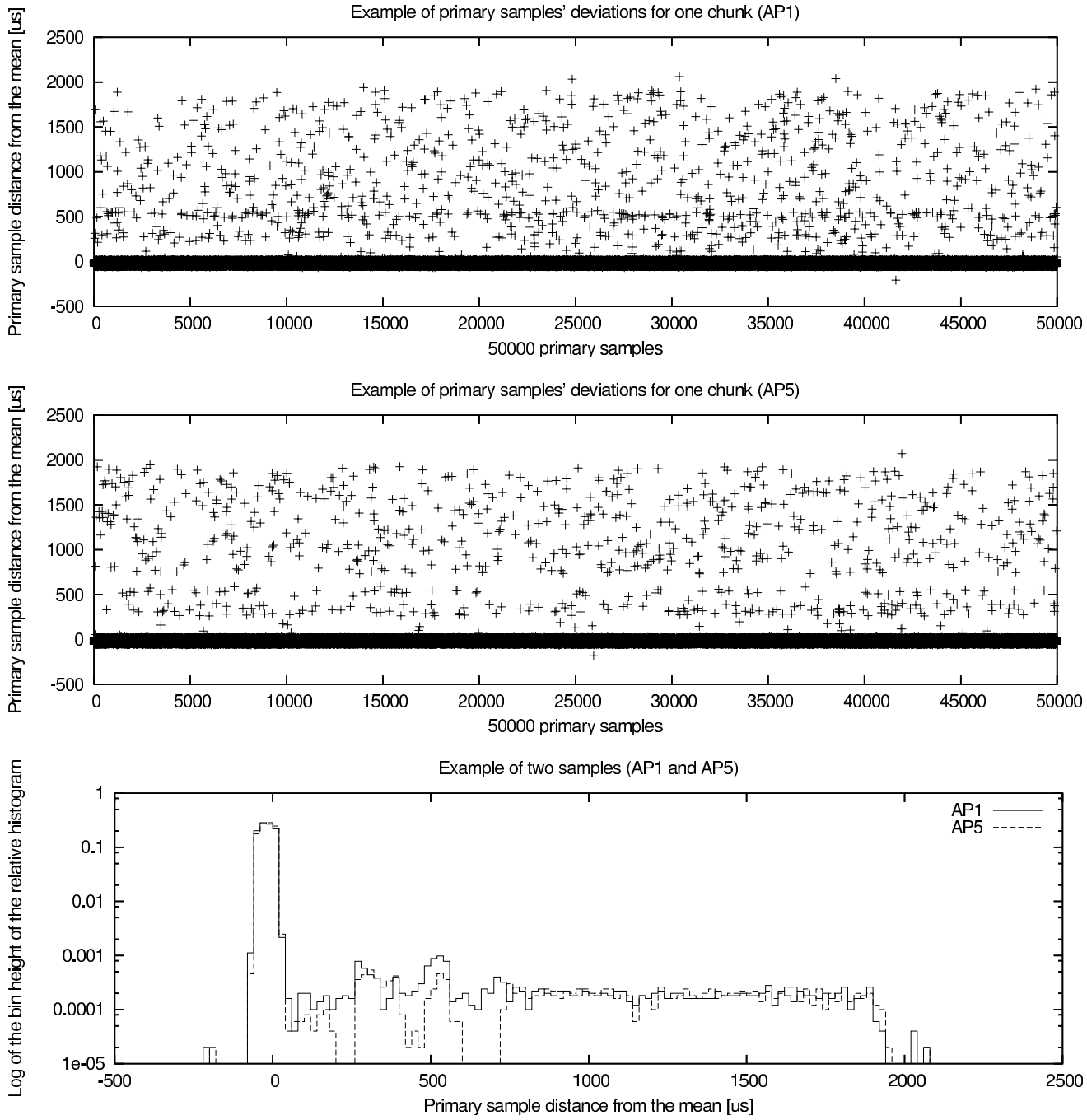

Fig. 3. Two first graphs show the deviations of primary samples. The third graph shows the resulting samples (relative histograms). The logarithmic scale is used for the histogram.

coming from AP2 being classified as coming from AP1, from AP2, from AP3, from AP4, and from AP5, respectively.

Five different SVM classifiers were built using five random training sets of 20 samples. Their averaged ECRM and ECR are presented in what follows.

$$averageECR = 0.865970$$

$$averageECRM = \begin{pmatrix} 1.000000 & 0.000000 & 0.000000 & 0.000000 & 0.000000 \\ 0.000000 & 0.763158 & 0.042105 & 0.131579 & 0.063158 \\ 0.003509 & 0.094737 & 0.870176 & 0.008772 & 0.022807 \\ 0.000000 & 0.168421 & 0.010526 & 0.801754 & 0.019298 \\ 0.000000 & 0.078947 & 0.022807 & 0.003509 & 0.894737 \end{pmatrix}$$

V. Discussion

It should be noted that AP1, AP2, and AP4 are identical devices. Identical in the sense that the only visual differences between them are two labels, one with the MAC address and the other with the serial number. All other inscriptions and markings on both devices are the same. AP3 differs from that group only by certain inscriptions on the board. And yet the SVM is able to differentiate among the first four boards with average accuracy exceeding 85%. The average classification rate for all five boards tested is even higher, i.e., 86% (more sophisticated classifiers are currently under study). It is then reasonable to assume that the physical differences between devices are indeed observable in the timings of the authentication procedure. Had the differences been not observable, the classification rates in the ECRM would be close to 20% for every AP. Observe also that all the APs used in the experiment have the same chipset. It is reasonable to assume that the differences between APs equipped with NICs with different chipsets are also detectable using this approach.

An interesting question is what causes the timing differences. This issue requires more research and some initial candidate explanations are as follows. Physical differences in NICs board layout (paths dimensions, soldering, etc.) might lead to time variability when electrical signals are transmitted. Compounded time variability of the turn-on and turn-off transients of various elements of the board could also possibly affect the external communication timings. Another reason is that the operation of the NIC is driven by an on-board oscillator. The differences in oscillator frequencies across different NICs may also be a factor.

One could argue that the timestamps $t_B^{'}$ measured by MON can not be used to accurately measure t_B (refer to Figure 1). Two factors could contribute to this inaccuracy.

1) The radio frames have to reach MON, thus adding a possibly varying propagation time. The difference between t_B and $t_B^{'}$ is the difference between propagation times from AP to MON for ACK1 and AUTH2. Considering that it is the deviations of $t_B^{'}$ that are analyzed, and furthermore that the experiment setup is static, one can assume that the difference between t_B and $t_B^{'}$ is negligible. Here static experiment setup means that the distance between nodes (especially between AP and MON) is unchanged. It would be interesting to see how the classification method behaves when the MON and AP are mobile – this aspect is currently under research. Nevertheless, the static scenario is also of interest. For example, consider an office where desktop computers use a fixed access point and want to perform the fingerprinting in order to avoid and detect rogue APs.
2) Another potential cause of inaccurate measurement of t_B is the timing variability of the MON itself. It could be due either to MON's physical properties or to the operating system overhead while performing the timestamping of radio frames. This is addressed by carefully configuring the software on MON. It is set up so that there are no other processes running on the system. Any remaining variability will average out over the number of primary samples gathered.

To sum up, the experiment is unchanged across different APs, which are the only variable factor in the setup. Thus any time variability detected between APs is due to their differences.

Another issue that can be raised is the one of timing measurement resolution. Can the timing measurements be made with sufficient accuracy? The method is to use the CPU cycle counter available on the machine performing the timings (MON). The i386 architecture since the Pentium processor offers the `RDTSC` instruction, that allows to access the number of CPU cycles since the power-up. Assume that the timing machine has a Pentium 4 Mobile running at 1.6GHz. One CPU cycle then takes $6.25s \times 10^{-10}$ ($0.000625\mu s$). Consider the 802.11g standard, with data rate of $54Mbit/s$. A simplistic computation thus yields $1.852s \times 10^{-8}$, i.e., $0.01852\mu s$ per bit. Since the length of an 802.11 frame is in the order of tens, or even hundreds of bits, it is clearly seen that the timing resolution is three orders of magnitude smaller than the duration of events that are to be measured.

Let us now discuss two approaches for fingerprinting wireless nodes.

- *Active fingerprinting.* Here, specifically crafted 802.11 frames are sent to the device and precise timings of the responses are gathered. This approach allows for adaptive fingerprinting, where the types and contents of subsequent frames depend on the timings gathered so far. This is roughly similar to a chosen-plaintext attack against a cipher. This approach is potentially detectable by the node being fingerprinted, as 500 authentication requests per second are unlikely to be seen as legitimate traffic.
- *Passive fingerprinting.* Here the entity doing the fingerprinting listens only to transmitted 802.11 frames and measures their timings. The advantage of this approach is that the target is unaware of the fingerprinting taking place.

In this paper, the active approach has been used, but research is underway to devise a passive fingerprinting method.

Another important aspect of a fingerprinting method is how hard it is to circumvent. It is desirable that a fingerprinting

method be involuntary in the sense described in Section II. This rules out the possibility of the node taking any measures to evade detection. One of the reasons why a node would want to circumvent the identification method is to simply avoid being fingerprinted (anonymity). Another reason is to impersonate some other node, and this effectively requires emulating some other node's fingerprint. In any case – barring extremely well-funded adversaries – circumventing the method described here requires to compensate for the physical differences using software (firmware). For a given 802.11 NIC, this is likely to be impossible (e.g., differences measured require time variability that is beyond the time accuracy that the NIC hardware/software platform provides).

One final note is about where the approach described here lies in terms of the OSI network layers. This method operates on the MAC level of the data link layer. It is one layer above the RFF approach, which utilizes the properties of the physical layer. Being positioned in the MAC layer enables the timing approach to use the existing hardware, without incurring additional costs. It should be noted that there exist fingerprinting mechanisms [9], [10] that operate on higher layers as well (network layer and up). These include detecting some properties of the IP and TCP protocols, which are mainly due the the differences in software implementations of the network stack (OS-fingerprinting). Thus the fingerprints are OS-specific, and not hardware-device specific.

VI. Conclusions

A method for identifying 802.11 nodes is presented. It utilizes precise timing measurements together with statistical methods and Support Vector Machine (SVM) to classify radio communication as coming from a given transceiver. It does not require any specialized hardware, thus minimizing additional costs. The identification is also involuntary, meaning the method can not be circumvented, save by physical alteration of the network device.

The paper made the following contributions.

- It showed that it is indeed possible to detect physical differences between 802.11 nodes by analyzing the timings of the devices' communications. The SVM classifier was able to differentiate the nodes with the average success rate of 86%.
- The described method applies naturally to the task of identifying wireless nodes.

The following extensions to the fingerprinting identified in Section V are the focus of current research.

- A method that can identify a 802.11 node in a passive manner, only by listening to the ongoing network traffic.
- An adaptation of fingerprinting approach to a mobile scenario.
- A classifier with the classification rate close to the RFF approach (95%).

References

[1] *ANSI/IEEE Std 802.11, 1999 Edition (R2003). Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE, 1999, Reaffirmed 12 June 2003.

[2] D. Glynos, P. Kotzanikolaou, and C. Douligeris, "Preventing impersonation attacks in manet with multi-factor authentication," in *Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'05)*, April 2005.

[3] J. Hall, M. Barbeau, and E. Kranakis, "Enhancing intrusion detection in wireless networks using radio frequency fingerprinting," in *Proceeding of Communications, Internet, and Information Technology (CIIT)*, St. Thomas, US Virgin Islands, November 2004, pp. 46–56.

[4] R. Anderson, *Security Engineering: A Guide to Building Dependable Distributed Systems*. Wiley and Sons, 2001.

[5] J. Hall, M. Barbeau, and E. Kranakis, "Detection of transient in radio frequency fingerprinting using phase characteristics of signals," in *Proceedings of the 3rd IASTED International Conference on Wireless and Optical Communications (WOC)*, Banff, Alberta, Canada, 2003, pp. 13–18.

[6] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*. New York, NY, USA: ACM Press, 1992, pp. 144–152.

[7] C. Cortes and V. Vapnik, "Support-vector networks." *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[8] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available for download at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[9] "Nmap," `http://www.insecure.org/nmap/`.

[10] "p0f," `http://lcamtuf.coredump.cx/p0f.shtml`.