A Theoretical Survey on Wireless Client Device Finger Printing

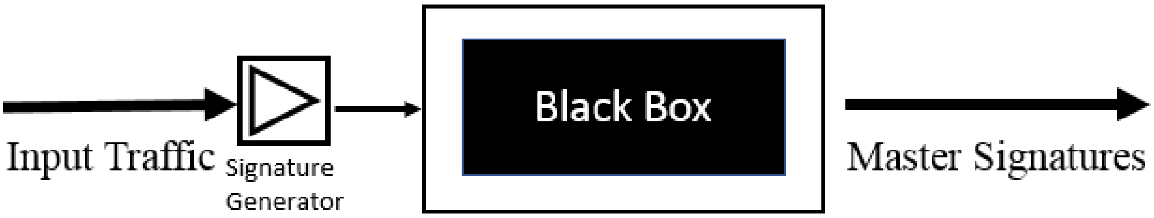
B A Sujathakumari
Associate Professor
Dept. of ECE
SJCE, Mysore

Abhishek P
4th semester M.Tech
Dept of ECE
SJCE Mysore

Abstract: Client device finger printing is method of identifying the client device type, operating system, Device vendor and other types of information which is connected to the wireless network. Also, it is the action of gathering device information to characterize it. This process generates a signature, also called a fingerprint, that describes the observed features of a device in a compact form. If the generated signature is distinctive enough, it may be used to identify a device. In the present developments where they are talking about internet of everything and bring your own device (BYOD) it is required for the enterprise to identify the devices connected to the network and mark them accordingly which may help the enterprise to avoid data leakage, unauthorized device connections and some other security violations. This paper provides the survey of different research undertaken to obtain information of devices or the nodes which are connected to wireless networks

1.1 Passive Approach to Wireless device finger printing

In [1] the approach to wireless fingerprinting is passive approach where the connected AP of a node is fingerprinted using black box technique. It has introduced a black box-based wireless device fingerprinting technique that can be used for offensive or defensive purposes. Backbox testing is a popular technique used to test software where the contents are unknown to the tester [10] (hence the name black box). To conduct the test, a stimulant is applied to the input of the software and the output is observed. From this, the tester can infer how the software acted on the input. In our case, the black box is an AP and just as the software testers are not privy to source code during software black box testing, they are not privy to the proprietary architecture of the AP. The input to our black box is a packet train and the output is the same packet train, however individual elements (i.e., packets) have been shifted in time. The shifting is a result of the internal architecture of the AP. Further, since each AP has a different architecture, this shifting is unique to the AP. They use wavelet analysis to amplify and extract the unique patterns generated by the internals of the AP. In this paper there are three categories of finger printing they are host finger printing, OS Finger printing, Driver type finger printing, the category OS finger printing is associated with different protocol stacks associated with the different operating systems, The different tools used for OS determination are Nmap ,Xprobe for active determination P0f for passive determination and SinFP is one more technique for Operating system determination The next category of finger printing is Host finger printing in this Authors use the TCP timestamp option of outgoing TCP packets to reveal information about the sender's internal clock. The authors' technique exploits microscopic deviations in the clock skews to derive a clock cycle pattern as the identity for a device. The next type is device fingerprint or driver finger printing here spectral analysis is used to determine the Medium access control functions in 802.1x standard, continuing it uses statistical information of the 802.1x management frame transmission to fingerprint the device type / driver. They use wavelet analysis in our approach for signature generation due to its theyll-known capability for multi-resolution decomposition suited for analysis of nonstationary signals, Using the signatures derived from the wavelet detail coefficients, they measure the similarity of the unknown to the candidate set of APs. Cross-correlation is commonly used to calculate the similarity bettheyen two signals for applications in pattern matching. For our analysis it has implemented the circular cross-correlation



Step 1: For the input traffic capture from an unknown AP type ax, extract sequence of IAT values to generate the feature signal Tx.

Step 2: Wavelet transform is applied to decompose the signal Tx into detail coefficients at different levels j. Use the coefficients to form the master signature sx = (d'l, d'2, ... , dj,).

Step 3: Cross correlation is used to measure the similarity of Sx to the set of master signatures S.

Step 4: AP type is selected such that ai has the highest similarity measure and ai will be the identity of ax.

1.2 An empirical study of passive 802.11 Device Fingerprinting

In [2], they evaluate a set of global wireless network parameters with respect to identify 802.1x devices. It restricts to parameters that can be observed passively using a standard wireless card. these parameters are evaluated for two different tests: i) the identification test that returns one single result being the closest match for the target device, and ii) the similarity test that returns a set of devices that are close to the target devices. The network parameters which are used for device identification are:

- Transmission rate: In 802.11 standard frames are transmitted using set of predefined rates
- Frame size: The size of 802.1x frame depends on the type of the frame, the fragmentation threshold and the applications generating the traffic.
- Medium access time: It is the time a wireless device waits after the medium is idle and before sending its frame
- Transmission time: It is the time required to transmit a frame, thus the time bettheyen the start of acceptation and the end of acceptation of a frame.
- Frame inter-arrival time: The time interval bettheyen end of receptions of two consecutive frames is the frame inter-arrival time.

1.2.1 Methodology: Signature construction

Signature calculation consists in generating several histograms, one histogram per frame type A histogram represents the frequencies of the values measured. Each histogram is theyighted, which gives more or less importance to certain types of frame. They define the signature of device as the set of generated histograms generated by the device and their theyights.

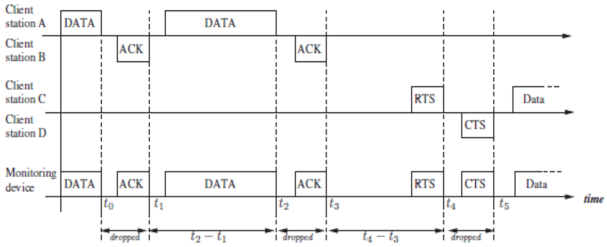


Figure:1.2.1: Measurement method

The sequence f0, . . . , fn−1 of frames represents the network trace captured by the monitoring device. Ti denotes the time of end of reception of the frame fi (where 0 ≤ i ≤ n − 1), and frames are ordered in increasing reception time (i.e. ∀i : ti−1 < ti). The sender si sends the frame fi. For frames like ACK frames or clear-to-send frames [2] the sender is unknown2, thus si = null. They calculate or extract the network parameter pi from the Radiotap or Prism header for each frame fi. Depending on the network parameter considered, pi may have different meanings. Radiotap or Prism headers include the size sizei, the transmission rate ratei and the end of reception or the start of reception ti of a frame fi. If the considered network parameter is the transmission rate, they have pi = ratei. Similarly, pi = sizei if they consider frame sizes. They can also calculate the inter-arrival time ii = ti − ti−1, the transmission time tti = sizei/ratei and the medium access time mtimei = ti − tti−1. They add the measured or calculated parameter to the set Pftype(si). Pftype(s) denotes the set of values measured or calculated for frames of type ftype for the sending device s. They denote |Pftype(s)| the number of observations for frames of type ftype for device s.

Figure 1.2.1 illustrates our method. Client stations A, B, C and D use the same channel and send the frames as depicted. The monitoring device listens on the same channel and receives all frames of the emitting client stations. Thus, the sequence of frames f0, . . . , f5 corresponds to the frame sequence DATA, ACK, DATA, ACK, RTS, CTS. The first ACK frame f1 has no explicit sender, s1 = null. Thus, they drop the associated value p1. Similarly, they drop the frames f3 and f5. If they use transmission rate as a parameter, they associate the value rate2 to client station A, as frame f2 is sent by station A. They associate rate4 to client station C. Thus, PDATA(A) = {rate2} and PRTS(C) = {rate4}. Similarly, if they use inter-arrival times as a parameter, they associate the interval i2 = t2 − t1 to client station A.

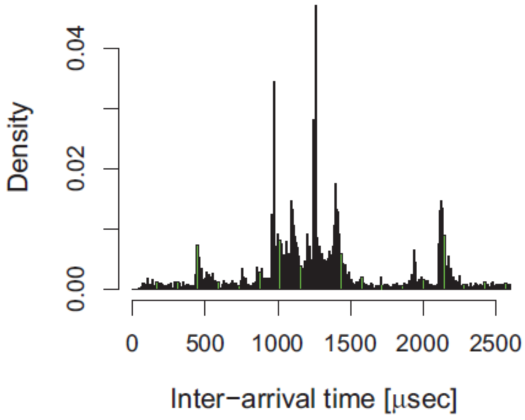


Fig 1.2.2 Inter arrival time Histogram

They associate the interval i4 = t4 − t3 to client station C. Thus,
PDATA(A) = {t2 − t1} and PRTS(C) = {t4 − t3}.

Based on the above measurements they generate a histogram for each frame type and each emitting client station. It is composed of bins b0, . . . , bk−1. They denote oftype j (where 0 ≤ j ≤ k − 1) the number of observations in bin bj . They convert the histogram into a percentage frequency distribution, where the bin's bj percentage frequency is

Pftype j = oftypej /|Pftype(s)|.

The resulting histogram for a give frame type is histftype(s) = {Pftypej |∀j ∈ 0 ≤ j ≤ k−1}. Figure 1.2.2 shows a resulting example histogram using inter-arrival times.

signature Sig of device s as follows: Definition 1 (Device signature):

Sig(s) = {(weightftype(s), histftype(s))|?ftype}

The variable weightftype weights the importance of a histogram for a given frame types. For reference signature they choose weight^{ftype} , thus the distribution of frame types is equivalent to the weight given to each frame type.

$$weight^{ftype}(s) = \frac{|P^{ftype}(s)|}{\sum_{ftype} |P^{ftype}(s)|}$$

Evaluation: The obtained data is evaluated using two observations, they are Similarity and Identification. *Similarity*: The fingerprint algorithm returns a set of reference devices which signatures similarity simi is greater than a threshold T. Thus we are interested in nowing which reference devices are similar to the candidate one. *Identification*: The second and more difficult test is interested in actually and uniquely identifying the candidate device. To do so, we pick the reference device with the greatest similarity from the vector of similarities returned by the previous test. Similarity test includes the True Positive Rate (TPR) and the False Positive Rate (FPR). The TPR is the fraction of candidate wireless devices known to the reference database for which the returned set contains the actual device. The FPR is the fraction of returned reference devices that do not match the actual candidate device. In section V, we calculate the FPR and TPR as a function of the threshold T.

Implementation: A tool is developed in Python based on the pcap library. It analyses standard pcap files as well as live traffic and extracts the different network parameters as described. The tool also implements the fingerprinting methodology, i.e. the calculation of the device signatures, reference database, similarity measures and the calculation of accuracy metrics.

A set of global wireless network parameters are evaluated with respect to their ability to identify 802.11 devices. To do so, a passive fingerprinting method is defined that can be implemented with standard equipment. It was considered that the network parameter frame inter-arrival time perform best in comparison to the other network parameters considered, in particular in the most difficult scenario of a conference setting. Using this network parameter, it was able to accurately identify 802.11 client stations and access points in a reasonable amount of time.

Conclusion: This survey paper describes the different approaches in fingerprinting the different wireless network devices, those includes two main types, they are black box method and signature analysis methods, the future paper comes out with exactly the best method of finger printing and implementation of the same.

REFERENCES

[1] Access point vulnerabilities, 2009. http://www.f-secure.com.

[2] Access point vulnerabilities, 2009. http://www.securityfocus.com.

[3] Iperf: Network testing tool, 2009. http://sourceforge.net/projects/iperf/.

[4] Nmap: Free security scanner for network exploration & security audits, 2009. http://www . nmap. org /.

[5] ntap: Network data capture tool, 2009. http://www.networkinstruments.com/products/ntaps/10_100.html.

[6] pOf: A versatile passive os fingerprinting tool, 2009. http://www.gomor.org/bin/view/Sinfp/WebHome.

[7] Sinfp: A new approach to os fingerprinting, 2009. http://lcamtuf.coredump.cx/pOf.shtml.

[8] Vulnerabilities of netgear wndap330, 2009. http://web.nvd.nist.gov/view/vuln /detail?vulnId=CVE-2009-0052.

[9] Xprobe: Active os fingerprinting tool, 2009. http://xprobe.sourceforge.net/.

[10] B. Beizer. Black-box testing: techniques for functional testing of software and systems. John Wiley & Sons, Inc., New York, NY, USA, 1995.

[11] S. Bratus, C. Cornelius, D. Kotz, and D. P eebles. Active behavioral fingerprinting of wireless devices. In WiSec '08: Proceedings of the first ACM conference on Wireless network security, pages 56-61, New York, NY, USA, 2008. ACM.

[12] J. Cache, H. D. Moore, and Skape. Exploiting 802.11 wireless driver vulnerabilities on windows, January 2009. http://www.uninformed.org/.

[13] C. L. Corbett, R. A. Beyah, and J. A. Copeland. A passive approach to wireless nic identification. In Communications, 2006. ICC '06. IEEE International Conference on, volume 5, pages 2329-2334, June 2006.

[14] C. L. Corbett, R. A. Beyah, and J. A. Copeland. Using active scanning to identify wireless nics. In Information Assurance Workshop, 2006 IEEE, pages 239-246, June 2006.

[15] C. L. Corbett, R. A. Beyah, and J. A. Copeland. Passive classification of wireless nics during active scanning. International Journal of Information Security, 2008:335-348, 2008.

[16] Radiotap. http://www.radiotap.org/.

[17] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE 802.11 Standard, 1999.

[18] C. Arackaparambil, S. Bratus, A. Shubina, and D. Kotz. On the Reliability of Wireless Fingerprinting using Clock Skews. In ACM WiSec, 2010.

[19] M. Azizyan, I. Constandache, and R. R. Choudhury. SurroundSense: Mobile Phone Localization via Ambience Fingerprinting. In ACM MobiCom, 2009.

[20] G. Berger-Sabbatel, Y. Grunenberger, M. Heusse, F. Rousseau, and A. Duda. Interarrival Histograms : A Method for Measuring Transmission Delays in 802.11 WLANs. Research report, LIG lab, Grenoble, France, 2007.

[21] S. Bratus, C. Cornelius, D. Kotz, and D. Peebles. Active behavioral fingerprinting of wireless devices. In ACM WiSec 2008.

[22] J. Cache. Fingerprinting 802.11 Devices. Master Thesis, 2006.

[23] S.-H. Cha. Taxonomy of Nominal Type Histogram Distance Measures. In MATH, 2008.

[24] J. Franklin, D. McCoy, P. Tabriz, V. Neagoe, J. V. Randwyk, and D. Sicker. Passive Data Link Layer 802.11 Wireless Device Driver Fingerprinting. In Usenix Security, 2006.

[25] K. Gao, C. L. Corbett, and R. A. Beyah. A passive approach to wireless device fingerprinting. In DSN. IEEE, 2010.

[26] K. Gopinath, P. Bhagwat, and K. Gopinath. An Empirical Analysis of Heterogeneity in IEEE 802.11 MAC Protocol Implementations and its Implications. In ACM WiNTECH, 2006.

[27] S. Jana and S. K. Kasera. On Fast and Accurate Detection of Unauthorized Wireless Access Points Using Clock Skews. In ACM MobiCom, 2008.

[28] N. Kasuya, T. Miyaki, and J. Rekimoto. Activity-based Authentication by Ambient Wi-Fi Fingerprint Sensing. In ACM MobiCom, 2009.

[29] D. C. C. Loh, C. Y. Cho, C. P. Tan, and R. S. Lee. Identifying Unique Devices through Wireless Fingerprinting. In ACM WiSec, 2008.

[30] J. Pang, B. Greenstein, R. Gummadi, S. Seshan, and D. Wetherall. 802.11 User Fingerprinting. In ACM MobiCom, 2007.