Proceedings of the Federated Conference on Computer Science and Information Systems pp. 623–627
DOI: 10.15439/2018F290
ISSN 2300-5963 ACSIS, Vol. 15

Raspberry Pi as an Inexpensive Platform for Real-Time Traffic Jam Analysis on the Road

Robert Baumgartl, Dirk Müller
Faculty of Informatics/Mathematics
Dresden University of Applied Sciences
Friedrich-List-Platz 1, D-01069 Dresden, Germany
Email: [firstname].[secondname]@htw-dresden.de

***Abstract*—Using mobile phones for accessing the Internet has become a standard use case of such devices, nowadays even more important than the good old phone call. WiFi at home or public ones allow for a low-cost or even unpaid access to the virtual world of the Internet. But, as we will show, this is only true to some degree in terms of monetary cost. One thing we're paying a lot with is the loss of our privacy. In this paper, we will show how easily and cheap potential attackers can track your mobile phone and, thus, you via data it sends all the time, so-called probe requests. Additionally we show by experimental data how this tracking can be used for traffic jam analysis on roads.**

***Index Terms*—Raspberry Pi Real-Time Traffic Security Privacy**

I. Introduction

DURING the last years, the use of mobile devices like mobile phones and tablets for accessing the Internet has celebrated a breakthrough due to technological advances and social changes. Mobile access has overtaken stationary one from desktop computers and statically used laptop computers.

As a result of this development, end users can access the huge library of the Internet from many places all over the world including situations when traveling by car, by train or even by airplane. From one point of view, this is great news since data can be retrieved easily, and this new level of information can be used for the good. On the other hand, data transmitted via the Internet often turns out to be spam, noise or just jokes. But at least we get the potential to do more useful things with the almost ubiquitous mobile Internet.

A matter of particular interest for huge market penetration is the pricing of the goods and services. A cost-efficient semi-mobile Internet access is typically provided by the use of a WiFi connection according to IEEE 802.11 standard. Its typical range is some $100\,\mathrm{m}$ in the field and $20\,\mathrm{m}$ or $30\,\mathrm{m}$ in buildings depending upon the material of the walls. Several mobile devices can be associated to one and the same access point in parallel, and all of them can be freely moved. Hence, in order to guarantee a stable connection, there needs to be a key for identifying every device. The established key having been used for enabling a target-oriented delivery of packets is the *Media Access Control* or *MAC* address.

A serious problem with the technically well-motivated MAC address approach is the significant decrease of privacy for the end user carrying a mobile device. A static 1-to-1-link between a device and an identifier perfectly allows at least for tracking, and, by some reverse or social engineering finally to uncover the identity of the person who carries a particular device. Combining both mechanisms by data merging ultimately allows for tracking everyone all over the world, a scenario completely violating all privacy requirements. Note that such a kind of tracking is by far not only an academic issue, but can happen and happens on a grand scale [1].

A heavily promoted counter-action of mobile device sellers fighting this privacy issue was *MAC address randomization* as implemented by major companies, cf. [2], starting from 2014. Unfortunately, recent publications [3] [4] show clearly that attacking privacy has only become a little bit more difficult, but by far not impossible as initially claimed by mobile devices companies.

In this article, we will show by some experiments how such a tracking can be performed with a little bit of knowledge and some inexpensive equipment. Finally, we will present and discuss the results of our most advanced setup for performing a traffic jam analysis via a so-called *section control*[1].

Here, individual cars' average velocities are calculated via the measurement of their time passing a fixed-length (some kilometers) section of a road. Of course, exceeding the speed limit in terms of the average speed implies also an illegal speeding in terms of peak speed whatever the actual velocity profile looks like. On the other hand, on the majority part [5] of the German autobahn, there is no speed limit at all. While the first *section control* was installed in Austria more than 10 years ago, there are only plans to apply it in Germany as well. In 2011 in Poland, an experimental section control on a $16\,\mathrm{km}$ section close to the city of Gdańsk revealed 28 drivers driving at average velocities of more than $200\,\mathrm{km\,h^{-1}}$ while $140\,\mathrm{km\,h^{-1}}$ was the allowed top speed [6]. Conventional *section control* is based on automatic number plate recognition. We discuss here an alternative mobile-device-based approach.

The remainder of this paper is structured as follows: In section II we discuss projects and publications related to our findings. Next, we give a short overview of our experimental hard- and software in section III. In section IV we describe a series of experiments of increasing complexity we conducted. We discuss the setup and summarize the main results. The

[1]This is actually a pseudo-anglicism like *handy* for a mobile phone or *beamer* for a video/digital projector. The term used in UK is SPECS for *Speed Check Services*, see also http://www.jenoptik.co.uk/product/specs.

paper ends with a summary of our main findings and an outlook to open questions.

II. Related Work

Vehicular traffic monitoring is a very popular field of research [7][8]. Conventional sensor technologies use inductive loop, piezoelectric, magnetometer, pressure switch, video camera, microwave radar, ultrasonic, optical, and laser radar data [8]. None of them will be used in our experiments. Instead, our data will be MAC addresses extracted from probe requests.

The general topic of tracking mobile devices and finally end users via their MAC addresses passively via probe requests is a common topic in the literature.

Many authors are aware of the privacy issue of the approach. Demir [9] proposed a multiple hashing of MAC addresses. Fuxjäger *et al.* [10] show that brute-force attacks on just hashed MAC addresses are quite simple, and, thus suggest a truncated and hashed MAC address approach with a higher level of privacy. Finally, Martin *et al.* [4] recently showed that even the more advanced technique of MAC address randomization can be attacked with a $100\,\%$ success ratio.

Chilipirea *et al.* [3] performed experiments on WiFi tracking of pedestrians. They could improve the quality of the data sets by various data filters.

Fuxjäger *et al.* [10] report on traffic jam analysis experiments on Austrian roads, but they used a more expensive equipment with external antennae.

A comprehensive study of WiFi probe requests for tracking and monitoring was given by Freudiger [11]. He managed to recognize several phone and OS types via profiling. But—compared to us—he used as well a more expensive monitoring equipment.

III. Experimental Platform

As cheap and ubiquitous hardware platform we used the Raspberry Pi Version 3 which offers an integrated WiFi chipset (Broadcom bcm43438). As mass storage medium we utilized cheap microSD cards of 32 GiB size. To ensure a maximum of autonomous operability, the systems were powered by external power-banks with a capacity of 20.000 mAh, which appeared to be somewhat over-sized. Hardware cost for one system amount to 50$. We utilized off-the-shelf Raspbian[2] Linux Version 8 as operating system base which provides a tailored Linux kernel version 4.4.50-v7+. Both systems were configured and used in headless mode.

The Raspbian standard firmware for the WiFi chip is not able to switch to monitor mode, therefore we installed the alternative firmware `nexmon`[3], version 7_45_41_26. The received data frames were captured using `dumpcap`, version 1.12.1, which is part of the well-known `wireshark` tool suite. By means of a capture filter, only probe requests were logged to persistent memory.

[2]http://www.raspbian.org

[3]https://github.com/seemoo-lab/nexmon

The resulting dumps were transferred to an external computer and converted to text records using `tcpdump`. Afterwards, we eliminated all irrelevant information except sender MAC addresses and accompanying timestamps within the measurement interval with the help of standard UNIX tools.

IV. Experiments

A. Receiving Probe Requests while Driving on the Autobahn

As a first attempt, we wanted to find out whether the Raspberry Pi is able to capture probe requests when moving fast. We placed the board under the windshield just like a dashcam and captured while driving.

On 04/11/2017, we entered the German autobahn A17 at access no. 3 "Dresden-Südvorstadt" at 15:45, headed for Dresden, changed to the A4 heading to Erfurt and left it at 16:33 at A4 exit no. 66 "Wüstenbrand". The distance was $83\,\mathrm{km}$ the average speed amounted to $104\,\mathrm{km\,h^{-1}}$.

During these 48 minutes, we captured 3379 MAC addresses, 609 of them were unique. It seemed that we were able to receive probe requests not only from cars driving in the same but also in the opposite direction, especially when both were using the leftmost lane.

This and the result of the next experiment were encouraging and proved that the board is very well capable of capturing a large number of probe requests while moving.

B. Receiving Probe Requests on a Train

In this experiment, we took the regional train *RE 26984* departing from *Dresden Hbf* to *Plauen(Vogtl) ob Bf* on 3rd May 2017. Only the section *Dresden Hbf* to *Chemnitz Hbf* corresponding to a scheduled travel from 15:52 to 16:54 was part of this experiment. Due to the recording of approximately one hour, we hoped for many probe requests with a lot of various MAC addresses.

We recorded as many as 6752 probe requests, i.e., on the average almost 2 per second. Among them, there could be 219 different sender MAC addresses of broadcast probe requests extracted. This number gives us a raw estimation of the order of magnitude of the number of travelers in this part of the train.

C. Receiving Probe Requests at the Road

Description: To receive probe requests from passing vehicles on a multi-lane highway, two principal positions could be used: a) on a bridge above the middle lane of one travel direction or b) by the right side of the road. Position a) seems favorable due to its elevation (and probably better receiving conditions) but requires constructions which cross the highway such as bridges. Position b) seems better suited in terms of cost and convenience (the system could easily be attached to some post or crash barriers).

In contrast to the scenario described in section IV-A we statically positioned the receiver a) on a bridge three meters above the middle lane (it is the same as measurement point B in section IV-D) and b) ten meters to the right of the rightmost

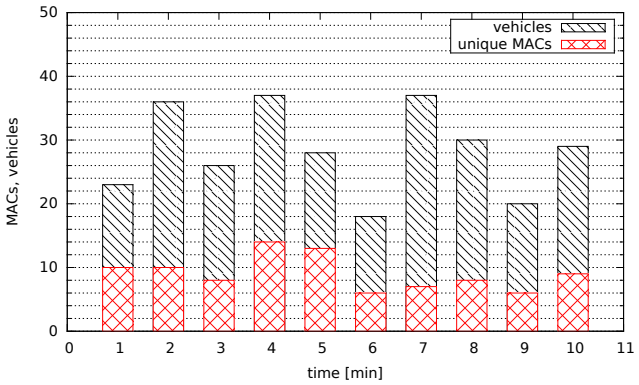

Fig. 1. Capturing on a bridge

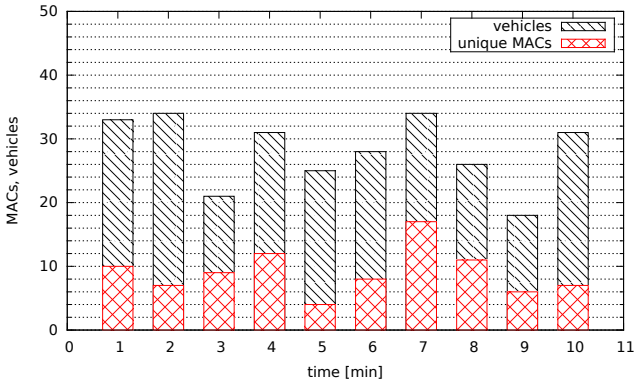

Fig. 2. Capturing at the side of the road

lane of the German autobahn A4[4] at a height of $1.50\,\mathrm{m}$ above ground (the GPS position was N 50.821650°, E 012.761600°). Due to restriction fences we were not able to get any closer to the lane.

Results: During the 10 minute capture interval, a total of 284 vehicles passed the position on the bridge and 281 vehicles were observed at the side of the road. During that interval, we were able to capture 91 unique MAC addresses resulting from probe requests at both positions which is almost a third. Figures 1 and 2 compare the number of cars and received unique MAC addresses on a minute-per-minute basis for both positions.

The ratio between passing cars and unique MAC addresses varies from $16\,\%$ to $50\,\%$ (both extrema were observed at the side of the road) with an average of $32\,\%$. Variance seems also a bit higher when capturing for the position at the side of the road but the short measurement interval prohibits deeper analysis.

Of course, there is no 1:1 relation between vehicles and MAC addresses for several reasons. First, some drivers may have switched off the WiFi functionality or could even have no smartphone at all. Second, other vehicles could carry more than one smart device, especially all kind of buses. Receiving more than one MAC from the same vehicle is redundant when trying to estimate vehicle speed (cf. section IV-D), but increases the chance of receiving two probe requests at different locations. Third, received MAC addresses from outside the context (passing bicyclists, vehicles from the opposite lane) could deteriorate our perceived numbers.

Nevertheless we can conclude, that a reasonable fraction of the passing vehicles sends probe requests such that our receiver hardware is able to capture them. Further, both logging positions seemed equally suitable.

D. Estimating Vehicle Speed

Description: In the final experiment, we tried to measure (or at least estimate) the average velocity of vehicles cruising in one direction for a certain section of the German autobahn. To this aim, we positioned two Raspberry Pis at a height of 3 to 4 meters above the middle lane of the A4 in direction of traffic Erfurt on two crossing bridges (The GPS coordinates are N 50.833591°, E 012.792370° for Point A, and N 50.819305°, E 012.745936° for Point B, respectively). Between A and B, the track runs almost straight. Figure 3 depicts the relevant topographical aspects. The distance between both points amounts to $5.03\,\mathrm{km}$ according to *openrouteservice.org*.

One motorway access is located between A and B therefore the numbers of passing vehicles may not be identical for both positions. During the time of our experiments, no explicit speed limit was mandated, visibility was very good.

Beforehand, the system clocks were synchronized manually with a Δ of ca. one second. Both systems logged all received probe requests for a fixed time interval of 15 minutes starting at 17:12 on 09/05/2017, a normal workday. Additionally, we manually recorded the number of passing vehicles per minute.

Because most MAC addresses were broadcast in short bursts we eliminated all but the first occurrence of a new unique MAC address. Then we searched for MAC addresses occurring in both log files (at different times) representing one and the same vehicle passing sequentially both measurement positions. We then determined the temporal difference t of the respective time stamps rounded to full seconds. Using the equation $v = s/t$ and the driving distance $s = 5.03\,\mathrm{km}$ between both points A and B, we finally computed the average speed of the vehicles.

Results: During the measurement interval of 15 minutes, a total of 453 vehicles passed point A. During that time, we observed a total of 115 unique MAC addresses. That 25 percent fraction seems to be somewhat optimistic, because a certain number of probe requests might also result from the opposite driving direction (see below). Figure 4 illustrates the number of passing cars and the number of received unique MAC addresses on a minute-per-minute basis. Nevertheless, we consider the number of unique MAC addresses surprisingly high given the cheap hardware platform and the high velocity of the passing vehicles which results in a visibility interval of a few seconds at the most.

[4]It is part of the longest European route, E40 from France to Kazakhstan. The majority of the A4 in Germany is a 3-lane-per-direction highway, including the part considered here.

Fig. 3. Measurement points for the estimation of vehicle speed

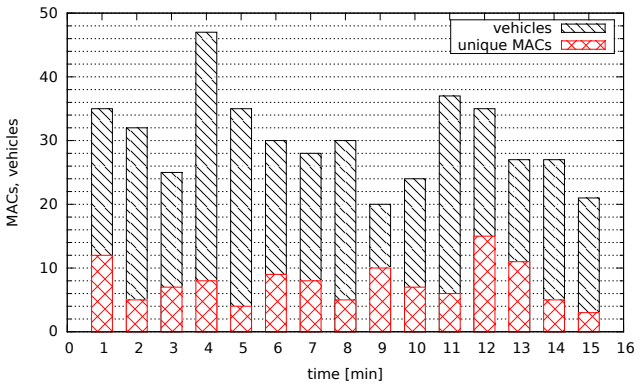

Fig. 4. Numbers of vehicles and unique MAC addresses per minute

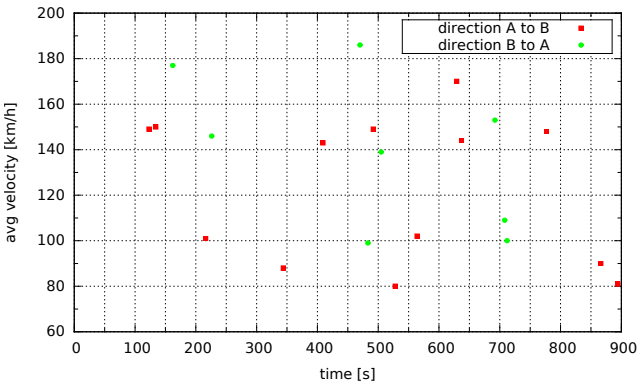

Fig. 5. Vehicle velocities measured over an interval of 15 minutes

Out of 115 unique MAC addresses recorded at point A and 128 unique MAC addresses recorded at B, we obtained 21 MAC addresses occurring at *both* positions. The slightly higher number of addresses at point B could result from a $1\,\mathrm{m}$ to $1.5\,\mathrm{m}$ lower position relative to the road surface. Figure 5 illustrates the individual velocities during our 15 minutes measurement interval.

Two of the address pairs had exactly the same time difference (121 seconds). Obviously, we monitored two different smart devices residing in the same car and sending their probe requests at the same instant. Further, it is interesting to note that we monitored 8 out of the 21 address pairs stemming from vehicles driving in the opposite direction. Third, a clear distinction between slow-driving trucks ($v < 120\,\mathrm{km\,h^{-1}}$) and faster passenger cars ($v > 140\,\mathrm{km\,h^{-1}}$) can be made. This reflects our empirical perception of the traffic situation and very good driving conditions. All obtained velocity values are plausible.

We can conclude that our setup allows to reliably estimate average velocity of vehicles for a given section on the autobahn and one driving direction. Further, it seems effortlessly possible to cover both driving directions by placing the Raspberries above the middle of the highway.

V. Conclusion and Outlook

Our considerations and experiments have shown that a tracking of mobile devices based on MAC addresses is feasible even with low-end equipment like a *Raspberry Pi Version 3* without any external antennae. Recently added features of modern smartphones like MAC address randomization render a tracking of such devices more difficult. But there is still a high market share of older mobile phones and such ones where the feature is not (yet?) implemented due to compatibility issues, cf. [4].

Use cases of such a tracking could be traffic jam analysis on a road. Here, we don't need to track many cars, some representative data is perfectly sufficient. Such real-time data can be used immediately for congestion alerts in navigation systems or in the good old FM radio.

Second, a preliminary assessment of road sections in terms of the percentage of speeding cases and their respective severity can be done. Our setting is very inexpensive and works anonymously. The recorded MAC addresses serve only as a matcher between two measurement points and can even be deleted directly after raw data processing. Our configuration is much cheaper than a conventional section control system and, thus, can be used more extensively for finding out where to place speed control points or sections.

Our results raise some interesting questions. First of all, we want to provide bounds for the accuracy of our velocity values. To that aim, two influence factors must be studied: one has to know the receiving range of the built-in antenna of the Raspberry Pi which depends on the position of the system itself and the period and sending pattern of probe requests has to be known which depends, among other, from the particular device type, its operating system version and operating mode.

We think that our approach is robust against heterogeneous and lane-less vehicular traffic being typical for countries like India due to lacking lane discipline or lacking lane infrastructure [8]. But this hypothesis needs to be verified by some additional experiments.

Further, because for our use case the Raspberry Pi has quite some unnecessary peripheral components, it seems promising to try even cheaper platforms, such as the Raspberry Pi Zero or the Espressif ESP8266 and ESP32. To drastically increase the duration of our measurements or even let the platform work autonomously, some means for solar powering the Pi should be investigated.

Finally, cost of an individual system approaching five dollars or even less would permit to equip a longer section of the highway with systems who are able to connect to each other via WiFi and propagate traffic data accordingly. Such a *smart highway* will probably lead to new interesting use cases.

References

[1] S. Khandelwal, "Spying agencies tracking your location by capturing mac address of your devices," *The Hacker News - Security in a serious way*, 2014, http://thehackernews.com/2014/01/spying-agencies-tracking-your-location_31.html.

[2] A. Mamiit, "Apple implements random mac address on ios 8. goodbye, marketers," *Tech Times*, 2014, http://www.techtimes.com/articles/8233/20140612/apple-implements-random-mac-address-on-ios-8-goodbye-marketers.htm.

[3] C. Chilipirea, A. C. Petre, C. Dobre, and M. van Steen, "Presumably simple: Monitoring crowds using wifi," in *2016 17th IEEE International Conference on Mobile Data Management (MDM)*, vol. 1, June 2016, pp. 220–225.

[4] J. Martin, T. Mayberry, C. Donahue, L. Foppe, L. Brown, C. Riggins, E. C. Rye, and D. Brown, "A study of MAC address randomization in mobile devices and when it fails," *CoRR*, vol. abs/1703.02874, 2017. [Online]. Available: http://arxiv.org/abs/1703.02874

[5] S. Anker, "Freie Fahrt – Wo geht das noch in Deutschland?" *PS-Welt*, 2013, https://www.welt.de/motor/article121455433/Freie-Fahrt-Wo-geht-das-noch-in-Deutschland.html.

[6] WP moto, "Odcinkowy pomiar prędkości na a1 działał tydzień," *WP moto*, 2011, http://moto.wp.pl/odcinkowy-pomiar-predkosci-na-a1-dzialal-tydzien-6068745844532353a.

[7] P. Bellavista, F. Caselli, A. Corradi, and L. Foschini, "Cooperative Vehicular Traffic Monitoring in Realistic Low Penetration Scenarios: The COLOMBO Experience," *Sensors (Basel, Switzerland)*, vol. 18, no. 3, March 2018. [Online]. Available: http://europepmc.org/articles/PMC5876597

[8] N. K. Singh, L. Vanajakashi, and A. K. Tangirala, "Segmentation of vehicle signatures from inductive loop detector (ILD) data for real-time traffic monitoring," in *2018 10th International Conference on Communication Systems Networks (COMSNETS)*, Jan 2018, pp. 601–606.

[9] L. Demir, "Wi-Fi tracking : what about privacy," Master's thesis, M2 SCCI Security, Cryptologyand Coding of Information - UFR IMAG, Sep. 2013. [Online]. Available: https://hal.inria.fr/hal-00859013

[10] P. Fuxjaeger, S. Ruehrup, T. Paulin, and B. Rainer, "Towards privacy-preserving wi-fi monitoring for road traffic analysis," *IEEE Intelligent Transportation Systems Magazine*, vol. 8, no. 3, pp. 63–74, 2016.

[11] J. Freudiger, "How talkative is your mobile device?: An experimental study of wi-fi probe requests," in *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, ser. WiSec '15. New York, NY, USA: ACM, 2015, pp. 8:1–8:6. [Online]. Available: http://doi.acm.org/10.1145/2766498.2766517