Magdeburger Journal zur Sicherheitsforschung

Gegründet 2011 | ISSN: 2192-4260
Herausgegeben von Stefan Schumacher und Jörg Sambleben
Erschienen im Magdeburger Institut für Sicherheitsforschung

This article appears in the special edition „In Depth Security – Proceedings of the DeepSec Conferences".
Edited by Stefan Schumacher and René Pfeiffer

How bluetooth may jeopardize your privacy.

An analysis of people behavioral patterns in the street.

Verónica Valeros and Sebastián García

Cell phones have become so personal that detecting them on the street means to detect the owners. By using the information of the phone along with its GPS position it is possible to record and analyze the behavioral patterns of the people in the street. Bluetooth devices are ubiquitous, but until recently, there were no tools to perform bluetooth wardriving with GPS position and behavioral analysis. A new tool called Bluedriving is presented for doing this type of bluetooth wardriving. Also, most people is not aware that their bluetooth device allows to abuse their privacy. The bludriving tool can visualize the devices on a map and set different alerts to follow people in the street. The tool is presented along with a large capture dataset and a deep privacy analysis. We conclude that it is possible to follow people in the street by detecting their bluetooth device.

Citation: Valeros, V. and García, S. (2013). How bluetooth may jeopardize your privacy. An analysis of people behavioral patterns in the street. *Magdeburger Journal zur Sicherheitsforschung*, 2, 394–405. Retrieved December 26, 2013, from http://www.sicherheitsforschung-magdeburg.de/publikationen.html
Version 2013/12/24 12:18

1 Introduction

Bluetooth devices has been incorporated to a myriad of different products. However the privacy issues of such a technology has been highlighted few times. We usually do not think about the possibility of a privacy issue because we consider that the technology is only used in short distances, but it has been demonstrated that with the proper antenna a bluetooth device can be accessed from more than 1km.

The short range misconception may be one of the root causes that allows the privacy abuse of this technology. Another cause may be that most of the devices belong to a unique individual and therefore they can be used to track him/her. If a cell phone is found on the street, it is most probable that the owner is carrying it.

In this paper we present some conclusions about several privacy concerns using this technology. Can we wardrive the bluetooth devices and correlate them with GPS information? Can we extract behavioral patterns from the data? Is it possible to track people using his/her bluetooth device? How many people has bluetooth activated and discoverable by default? We answered these questions by developing a new tool called *bluedriving*. The ultimate goal of this tool is to raise awareness about how this devices exposes information about our everyday movements, abusing our privacy.

Actually, the cell phone providers already have and use the information about the position of the cell phones. Also, some companies like Google and Apple has access to this information. However, this project makes this information *available to anyone*.

This tool creates the following new possibilities in the bluetooth analysis landscape:

It is possible to capture this information anonymously. Unlike the cell phone providers and companies, no one knows that you are capturing the bluetooth data. So your own privacy is guaranteed.

It is possible to *extract the behavioral patterns* of people.

It is possible to *follow people* (or cars) in the street. Opening the possibility of targeted attacks.

The bluedriving tool can be downloaded from following website[1] and the repository[2]

2 Previous tools

Some previous tools have been developed to capture information about bluetooth. Btscanner is a tool developed with ncurses and the BlueZ libraries. The main drawback of this tool is that it does not uses GPS information, making it useless for bluetooth wardriving.

Bluesniff is a tool that was presented on defcon 11[3]. It has interesting features like the possibility to make a brute force scan of bluetooth devices and it is able to show the signal strength of a device among other information. This tool also doesn't include GPS information.

Wigle.net has an android app for bluetooth wardriving called wigle bluetooth but, as the previous tools, it only shows the bluetooth devices information and does not include the GPS information.

3 Bluedriving tool set

The bluedriving tool consists in a console program, a web server along with its web page, a database analysis program and a sqlite database. The console is responsible for getting the bluetooth data and to show it on the console. The web server is the backend of a nice interface designed to give more flexibility to the behavioral analysis. The sqlite database act as a communication point between the console and the web server. The console and the web server were made with python. The web page uses jquery.

3.1 Console program

The `bluedriving.py` python program executes the console. An example of the console's output can be seen in Fig. 1 It has the following features:

- Uses threads to speed up the discovering process
- Searches for new bluetooth devices continuously.
- Gets the GPS information from the gpsd daemon in the system.
- Shows the approximate address of the GPS coordinates.
- Gets the basic bluetooth information from each device.
- It is prepared to describe its inner state with sounds, so it can be used while walking in the street.
- If a device matches a sound alarm, it pays a sound (useful while in the street).
- If a device matches a mail alarm, it sends an email using gmail
- It is possible to toggle options on/off on the run.

The GPS support has two interesting features. First, it can get the real address from the GPS coordinates using an Internet connection. This option is useful to debug the GPS system and to really know where you are. Second, the tool can read a pair of GPS coordinates from command line and it will consider that those are the real GPS coordinates. This trick, or 'Poor's man GPS' mode, is useful to use the tool without having a GPS dongle of cell phone. This mode is also useful when you are not moving, for example at your home, to be sure that you are not going to lose your GPS signal or run out of battery.

1 http://mateslab.weebly.com/bluedriving.html

2 https://github.com/verovaleros/bluedriving

3 http://bluesniff.shmoo.com



Figure 1: Bluedriving console output without the detailed device information

One of the major features of the console tool is that it is designed to be used in the street without looking at the display. Usually, during wardriving sessions, you can not look at the display of your notebook because you are walking, or perhaps you don't want to be seen looking at a suspicious display. The console will use different sounds for each of the following states:

- No device detected, and there is *no* GPS signal.
- No device detected, and there is GPS signal.
- Device detected. It is the *first time* that this device is detected.
- Device detected. We have seen this device before.
- GPS signal was successfully retrieved

With these sounds it is easy to know if the system is working, if we lose the GPS signal (maybe you want to stop walking), if we get a GPS signal again, if we found a device for the first time and if we found a previously seen device. This last option is useful for following people.

The console also has two types of alarms. Alarms are set using the web page, but are implemented on the console. Each time a device is found all the alarms are analyzed. If a sound alarm match is found, then the proper sound is played. If a mail alarm is found, then the proper information is send by email. The sound alarm is useful to follow people in the street and the email alarm is useful when your bluedriving tools is stationary and you are not looking at the display continually. The email is sent only using a gmail account. You should provide the username and password. They are used directly on the email libraries and they are not stored.

Every time that a new device is found, the console can search for all the services served by the device. Fig. 2 shows an example of this information. This is a useful option to know in which way the device can be attacked.

The most important parameters for the bluedriving console are:

-w, --webserver It runs the webserver to visualize and interact with the collected information. Defaults to port 8000.

-s, --not-sound Do not play the beautiful discovering sounds.

-i, --not-internet If you don't have internet use this option to save time to avoid getting the addresses from the web.

-l, --not-lookup-services Use this option to avoid the lookup of services on each device. It make the discovery faster.

-g, --not-gps Do not try to get the GPS information.

-f, --fake-gps Fake gps position. Useful when you don't have a gps but know your location from google maps. Example: -f '38.897388,-77.036543'

-m, --mail-user Gmail user to send mails from and to when a mail alarm is found. The password is entered later.

3.2 Web Server and web page

The web server is also implemented in python and is designed as a backend for configuring and displaying the information. It is automatically executed by the console if the -w parameter is used. The webserver can be used standalone without the main bluedriving.py program, making it a useful offline analysis tool. It is mainly divided in four sections: Results, Device info, Device Map and All devices Map.

The results section displays, in real time, all the devices found. As Fig. 3 shows, each device is presented with its GPS position, so there is a line for each pair of device-position. This information is useful while wardriving to see what is being detected and it is also useful as for an offline analysis. The information shown in the table is:

- Last date and time seen
- First date and time seen
- Mac address
- GPS coordinates
- Address corresponding to the GPS coordinates
- Name of the device

This information makes easy the identification of valuable information and interesting devices. Finally, in the results section, if a device-position pair is selected with the mouse, all the positions of the same device are highlighted, so it is easier to find the desired information.

The Device Info section is designed to show information about the device in the current position. You



Figure 2: Bluedriving console output with the detailed device information

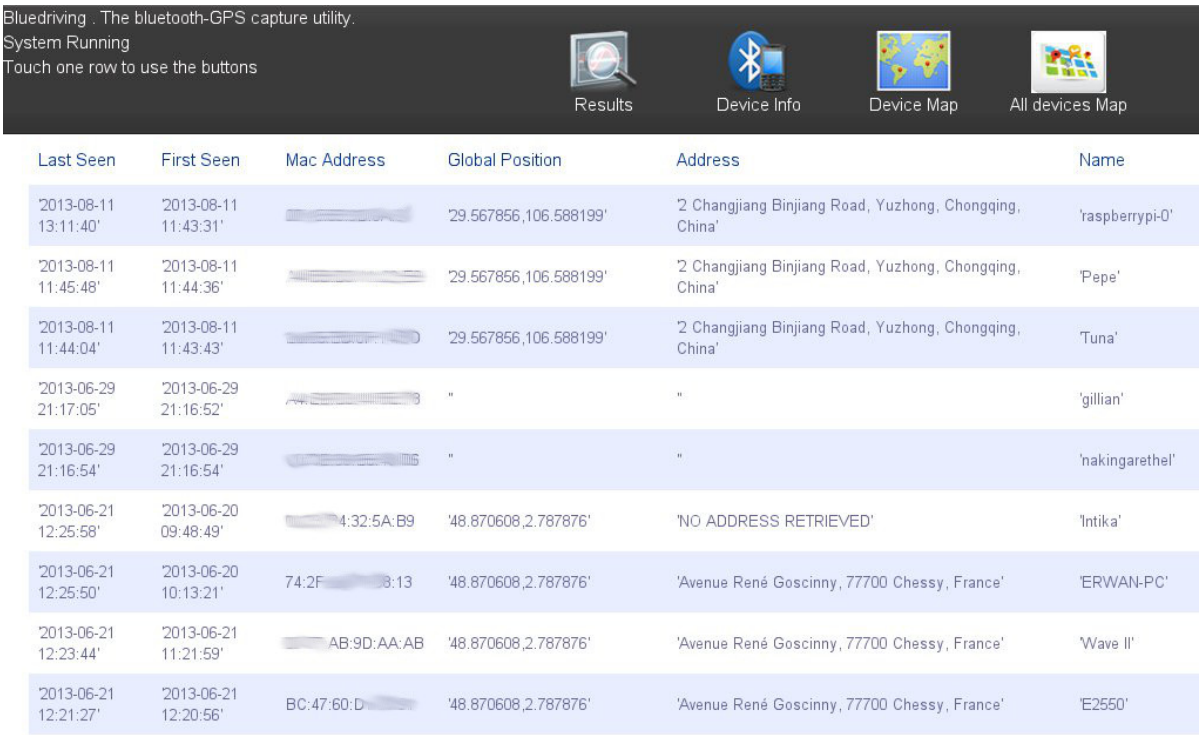


Figure 3: Web server Results section displaying the devices found.

should first select a line in the Results section and then click on the Device Info button. Fig. 4 shows how is the information organized. At the top of the page, the main information about the device is presented, including the device services available. The services are only stored if the program is run without the `-l` option.

This section allows the user to set notes on the devices. They are useful to store additional information about a device, such as the owner or the plate number of the car. Also, this section allows the user to set up two different alarms on the devices: sound alarms and mail alarms. Sound alarms will be played and mail alarms will be sent each time that the device is found in the future. These alarms are set using the web page but are active even if the web server is not being used. Finally, this section shows a map of the position selected.

The section Device map is used to show a map of all the positions of the device selected in the results section. Fig. 4 shows this section. This is an interesting map to show where the device has been seen. It can be used to find common paths during several days or where the device is most commonly found in the city.

The section All Devices Map presents a map of all the positions of N last devices in the database. The number of devices should be selected in the web page. This section allows the user to see several different devices at the same time, showing a complete map of the behaviors and positions. Fig. 6 shows this complete map. The map also includes the date and time when each device was seen.

3.3 Database analysis program

The `manageDB.py` program executes the database analysis part of the bluedriving system. It is a simple tool to get information about the devices in the database. It's main parameters are:

-d, --database-name Name of the database to store the data.

-l, --limit Limits the number of results when querying the database

-e, --get-devices List all the MAC addresses of the devices stored in the DB

-n, --get-devices-with-names List all the MAC addresses and the names of the devices stored in the DB

-E, --device-exists <mac> Check if a MAC address is present on the database

-R, --remove-device <mac> Remove a device using a MAC address

-g, --grep-names <string> Look names matching the given string

-r, --rank-devices <limit> Shows a top 10 of the most seen devices on the database

-m, --merge-with <db> Merge the database (-d) with a given database

-L, --get-locations-with-date <mac> Prints a list of locations and dates in which the mac has been seen.

-q, --quiet-devices Print only the results of the requested option

-C, --count-devices Count the amount of devices on the database

-c, --create-db Create an empty database.Useful for merging.

One of the best features of this tool is the ability to merge different bluedriving databases. This way it is possible to set up several bluedriving nodes in different parts of the city and then mix them together to get quicker results. If we combine this with the fact that the bluedriving tool can run on notebooks and other devices like Raspberry Pi, the amount of nodes that a group of people can set up will raise considerable.

This tool is a vital part of the tool set because it allows to query the database for particular data, search devices, pool statistics of the information stored; it allows to give real value to the data captured.

4 Data captured

During several months we have captured a bunch of bluetooth devices in several places. We have been walking, driving cars and using public transportation services in Mar del Plata city in Argentina, Tandil city in Argentina, Buenos Aires city in Argentina, Prague city in Czech Republic and Paris city in France. We used several different notebooks computers and two Raspberry Pi devices executing our tool. The bluetooth devices used to gather the information ranged from the internal devices on notebooks to different bluetooth USB dongles.

So far, the approximate total amount of unique devices captured is *3.000* and it is growing fast. Considering that we have not travel more than a few kilometers on each city, it is an average of 600 bluetooth devices on each city. With a simple 10 block walk every day we can capture as much as 70 new devices per day.

A lot of different devices were captured during the bluetooth wardriving. Fig. 7 shows the distribution of the bluetooth devices manufacturers. This information is very useful from an attacker perspective. You can focus on attacks that will work on the most used devices.

The most interesting data is about the position and behavior of the devices during the experiments. The following paragraphs describe some of the most interesting findings.

We manage to capture some Samsung TVs in the street. This information is really private information about a very expensive home appliance. The information even describe how big they are. This can be a serious issue in cities with a high level of insecurity. For example a robber can easily pick the building where these TVs are located.

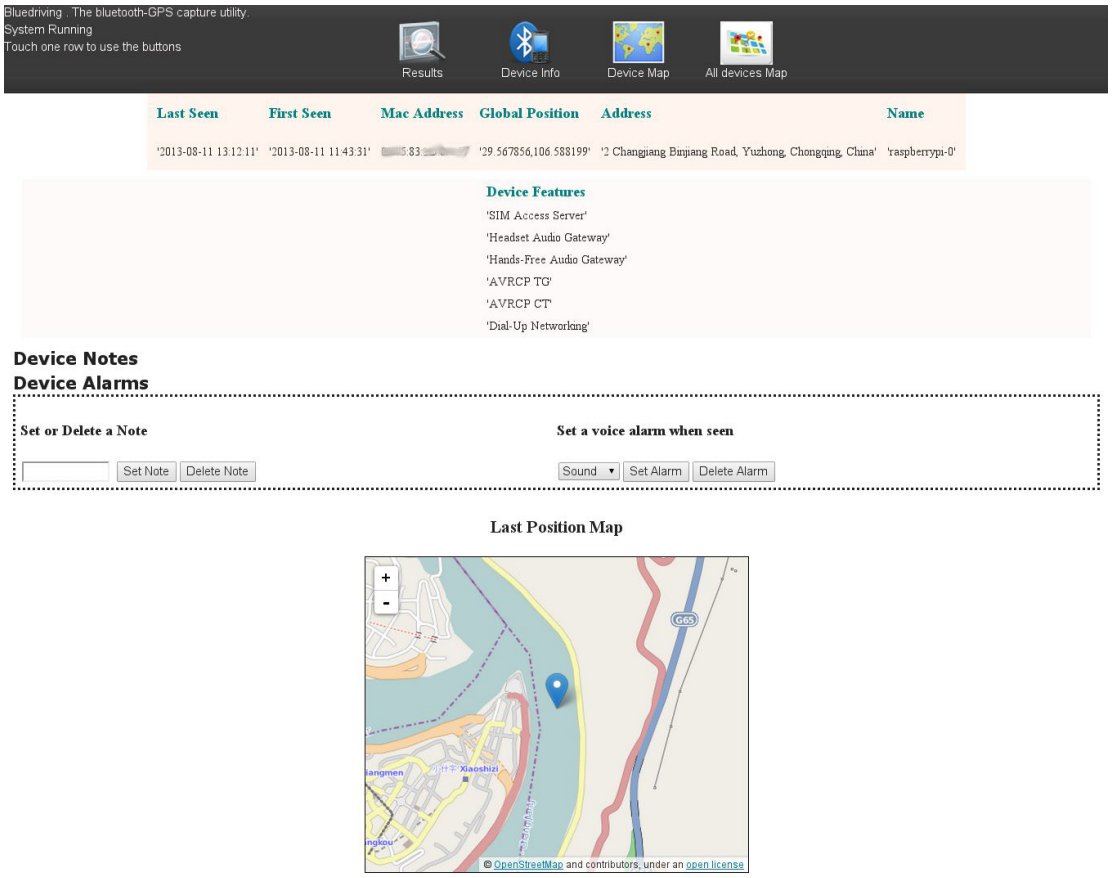


Figure 4: Web server Device Info section

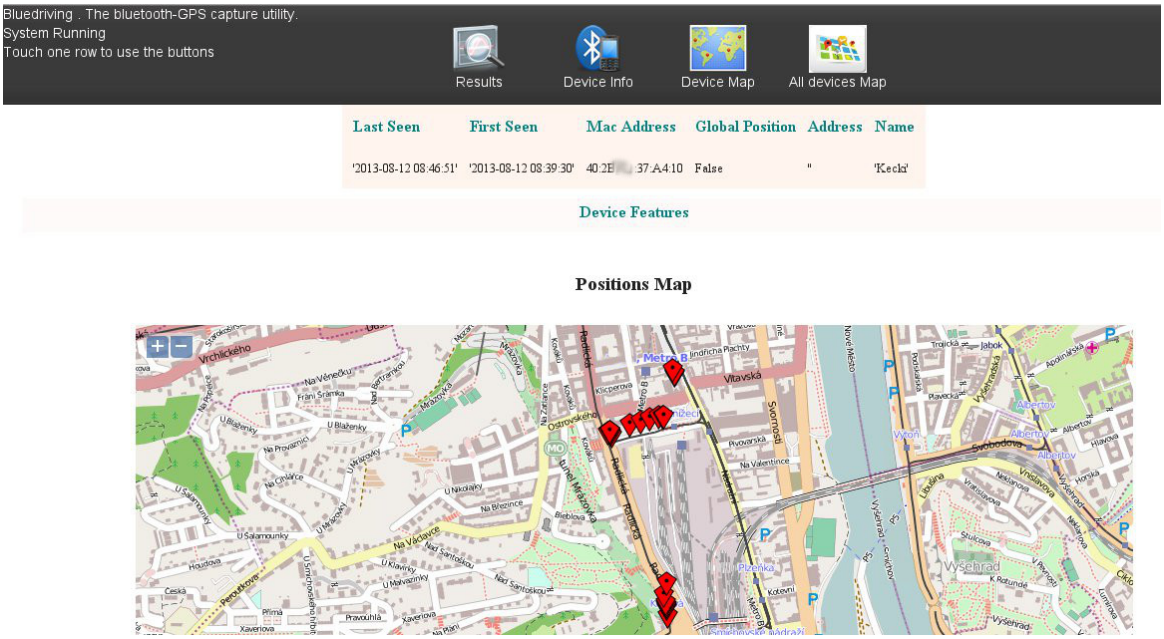


Figure 5: Web server Device Map section

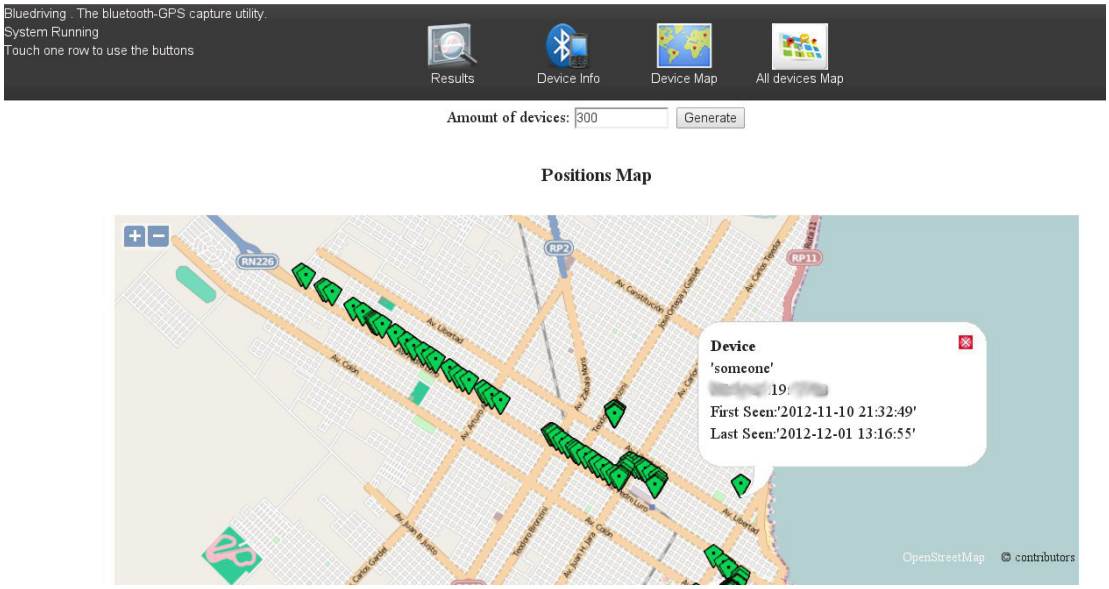


Figure 6: Web server All Devices Map section

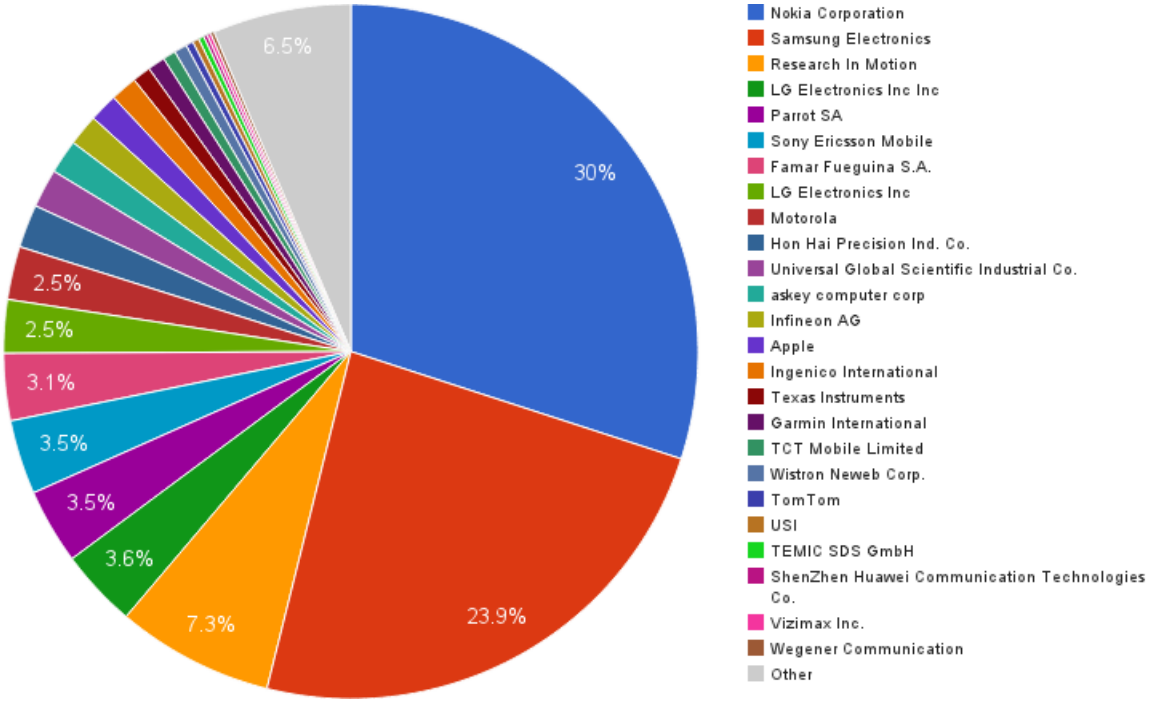


Figure 7: Distribution of Bluetooth devices manufacturers

[TV]Samsung LED*46*: '2013-08-12 07:30:17'-'2013-08-12 07:46:34'

[TV]Samsung PDP*51*: '2013-08-12 07:35:57'-'2013-08-12 07:46:32'

[TV]Samsung LED*46*: '2013-08-12 07:37:29'-'2013-08-12 07:46:37

[TV]Samsung LED*40*: '2013-08-12 07:36:03'-'2013-08-12 07:46:33'

[TV]Samsung LED*40*: '2013-08-12 07:36:04'-'2013-08-12 07:45:31'

[TV]Samsung LED*40*: '2013-08-12 07:33:08'-'2013-08-12 07:45:25'

[TV]Samsung PDP64: '2013-08-12 07:30:12'-'2013-08-12 07:46:33'

[TV]Samsung LED*75*: '2013-08-12 07:33:10'-'2013-08-12 07:45:26'

[TV]Samsung LED*46*: '2013-08-12 07:35:59'-'2013-08-12 07:45:26'

TVBluetooth: '2013-08-12 07:35:54'-'2013-08-12 07:44:45'

LGE DTV BCM20702A1: '2013-08-12 07:33:06'-'2013-08-12 07:45:23'

DTVBluetooth: '2013-08-11 20:22:59'-'2013-08-11 20:22:59', '50.070983333,14.404236667' ( )

DTVBluetooth: '2013-08-11 20:25:50'-'2013-08-11 20:25:50', '50.070881667,14.404413333' ( )

One of the most interesting types of devices captured are cars, or to be more precise car's audio systems that include bluetooth. In some cars this function can not be even deactivated. If you believe that following a car is not possible, just think about the usual path that you do with your own car. Do you always follow the same path to go to work or back home? Well, using this tool people can know when and where is your car. Some examples are:

Audi UHV 5719: '2013-06-30 14:27:08'-'2013-06-30 14:27:08'

Audi UHV 5347: '2013-06-30 17:25:14'-'2013-06-30 17:25:14', '50.077508333,14.41412'

Ford Audio: '2013-08-11 13:34:08'-'2013-08-11 13:34:08', '50.08627,14.404335'

Skoda BT: '2013-08-12 08:46:07'-'2013-08-12 08:46:07'

The Parrot system is a small device to connect your cell phone to your cars audio. This car was seen two different days:

Parrot CK3100: '*2013-08-11* 15:45:10'-'2013-08-11 15:45:10', '50.072173333,14.414185' (Rašínovo nábřeží 60, 128 00 Prague 2, Czech Republic)

Parrot CK3100: '*2013-08-11* 15:45:22'-'2013-08-11 15:45:22', '50.072316667,14.41419' (Rašínovo nábřeží 60, 128 00 Prague 2, Czech Republic)

Parrot CK3100: '*2013-08-12* 08:37:11'-'2013-08-12 08:37:11', False

Another type of device found are printers. The printers may come with bluetooth or they can use a bluetooth adapter to allow users send jobs directly from their phone. Here is an example:

Canon MP800R-1: '2013-06-30 22:18:26'-'2013-06-30 22:19:44', False

Canon MP800R-1: '2013-06-30 22:21:01'-'2013-06-30 22:21:01', '50.076788333,14.418331667' (Charles Square 14, 120 00 Hl.m. Praha-Praha 2, Czech Republic)

Canon MP800R-1: '2013-08-11 15:22:14'-'2013-08-11 15:22:14', '50.07684,14.41832' (Charles Square 14, 120 00 Hl.m. Praha-Praha 2, Czech Republic)

Canon MP800R-1: '2013-08-11 15:22:18'-'2013-08-11 15:22:18', '50.07682,14.418283333' (Charles Square 14, 120 00 Hl.m. Praha-Praha 2, Czech Republic)

Canon MP800R-1: '2013-08-11 15:22:41'-'2013-08-11 15:22:41', '50.076786667,14.41805' (Na zbořenci 271/2, 120 00 Prague 2-New Town, Czech Republic)

Canon MP800R-1: '2013-08-11 15:22:45'-'2013-08-11 15:22:45', '50.076778333,14.418056667' (Na zbořenci 271/2, 120 00 Prague 2-New Town, Czech Republic)

Canon MP800R-1: '2013-08-11 15:22:49'-'2013-08-11 15:22:49', '50.076793333,14.418063333' (Na zbořenci 271/2, 120 00 Prague 2-New Town, Czech Republic)

Canon MP800R-1: '2013-08-11 15:22:53'-'2013-08-11 15:22:53', '50.076805,14.418061667' (Na zbořenci 271/2, 120 00 Prague 2-New Town, Czech Republic)

The most important information that can be extracted from this database are the behavioral patterns. An example of how the behavior of the same device can be found is the following information:

8752: '2013-08-11 14:18:15'-'2013-08-11 14:18:20', '50.072453333,14.407486667' (Lidická 796/20, 150 00 Prague 5, Czech Republic)

8752: '2013-08-12 08:30:31'-'2013-08-12 08:30:31', '50.072958333,14.412783333' (Palackého most 22, 120 00 Prague 2, Czech Republic)

We can see how the same device was found two different days in very near locations. This may mean that this individual may usually take this path. Looking at the hour we may assume that on 12th August he/she was going to work.

Also the behavioral pattern can be distributed along several days, like in the following example:

STIG: '2013-06-29 23:43:53'-'2013-06-29 23:43:53', '50.072786667,14.414255' (Rašínovo nábřeží 1571/62, 120 00 Prague-Prague 2, Czech Republic)

STIG: '2013-06-29 23:44:10'-'2013-06-29 23:44:10', '50.073,14.414405' (František Palacký, Palackého náměstí, 128 00 Prague-Prague 2, Czech Republic)

STIG: '2013-06-29 23:44:18'-'2013-06-29 23:44:18', '50.073096667,14.41447' (Palackého náměstí 1571/1, 128 00 Prague-Prague 2, Czech Republic)

STIG: '2013-06-29 23:44:33'-'2013-06-29 23:44:33', '50.073171667,14.41477' (Palackého náměstí 357/3, 128 00 Prague-Prague 2, Czech Republic)

STIG: '2013-06-30 13:40:07'-'2013-06-30 13:40:07', '50.072795,14.414271667' (Rašínovo nábřeží 1571/62, 120 00 Prague-Prague 2, Czech Republic)

STIG: '2013-06-30 13:40:43'-'2013-06-30 13:40:43', '50.07325,14.414546667' (Palackého náměstí 357/3, 128 00 Prague-Prague 2, Czech Republic)

The following device was seen twice within a two months difference in two far away places. It is a rare capture. This is a clear example on how this tool allows to start finding patterns on people's behaviour that were not visible before.

Anna: '2013-06-30 17:17:44'-'2013-06-30 17:17:44', '50.083688333,14.423221667' (*Jungmannovo náměstí 770/8*,

110 00 Prague-Prague 1, Czech Republic)

Anna: '2013-08-11 15:39:03'-'2013-08-11 15:39:03', '50.081388333,14.41959' (*Spálená 30*, 110 00 Prague 1, Czech Republic)

The impact of the bluetooth technology on privacy may be better appreciated when some medical devices are found on the street. This not only means that this individual may be followed but also that she/he has a medical condition. Fig. 8 show this type of device. In this case we found a Spirometer, a device to measure the volume of air inspired and expired by the lungs. This kind of devices are carried usually by people that has asthma. This is a lot of private information we can obtain by only knowing a MAC address.

Spirobank G-USB- SN806181: '2013-06-30 02:13:03'-'2013-06-30 02:18:48', False

Spirobank G-USB- SN806181: '2013-06-30 17:07:33'-'2013-06-30 17:07:33', '50.084975,14.421146667' (Havelská 504/17, 110 00 Prague-Prague 1, Czech Republic)

A curious case was when we detected two Wiimotes control devices from the Nintendo Wii game console. We can see that one of the controllers has Wiiplus (with -TR) while the other does not.

Nintendo RVL-CNT-01-TR: '2013-08-14 15:56:15'-'2013-08-14 15:56:31'

Nintendo RVL-CNT-01: '2013-08-14 15:40:51'-'2013-08-14 15:40:51'

We manage to also capture some more specialized bluetooth devices, such as some China-based Universal Global Scientific Industrial Co. (E0:2A:82:78:B7:B4) devices that can be embedded in other computers.

E0:2A:82:78:B7:B4 WIN-E66LMJPU5C9: '2013-06-29 23:20:26'-'2013-06-29 23:56:50'

Another type of unusual device found were external speakers:

SRS-BTX300: '2013-06-30 22:53:52'-'2013-06-30 22:27:17'

SRS-BTX300: '2013-08-12 06:28:32'-'2013-08-12 06:28:32',

'50.077778333,14.419408333' ( )

Bluetooth-enabled devices are used in a lot of different purposes, and sometimes people is not aware that a computer can be contacted using this protocol, for example by using an old iMac as Cash Desk:

Cash Desk I's iMac (2): '2013-06-30 17:18:54'-'2013-06-30 17:18:54', '50.08332,14.422098333' ( )

Cash Desk I's iMac (2): '2013-06-30 17:19:16'-'2013-06-30 17:19:16', '50.083216667,14.421863333' ( )

4.1 Following people

Following people in the street using this tools is not an easy task. We can start from two different situations: or you do have the MAC address of your target, or you do not. In case that you already have the MAC address of your target, it is easier to follow it by putting a sound alarm in the bluedriving tool. In this way, each time that the device is found, a sound will be played and you can walk in the street without nothing more that a headphone.

If you do not have the MAC address, then you can try to find if her/his phone is discoverable by doing some street tricks. For example, you can try to follow your target and capture the devices around you. Most probably you are going to see several devices. Then you move and follow your target a little bit more, trying to find a new set of devices. If there is a device that was seen both times, then it is a probable candidate to be the targets cell phone. You can repeat this operation until you find the correct device or you find out that the target does not have a discoverable bluetooth device. The following paragraphs show some real examples.

A bluetooth device called *STIG* (Fig. 9) was found on almost the same spot two different days on two different hours. If we combine this information with the fact that public transportation is really good in this particular city, the citizens are more likely to follow the same path every day. This make the task of retrieving information of the devices more easy, because after some days of capturing data, you can safely presume where you will find a particular device and at which hour.

Another example of following a device using public transportation can be seen in the device called 'Mama' in Fig. 4.1. The device was seen in two different days and different times. The fact that a MAC address identifies a unique device makes the tracking task more easy as you don't need to completely rely on poor sight identification of individuals.

Another example of a device coming back to the same place two different days is the *1692* device:

1692: '2013-08-11 15:37:14'-'2013-08-11 15:37:14', '50.08029,14.41984' (Spálená 25-27, 110 00 Prague 1, Czech Republic)

1692: '2013-08-11 15:37:18'-'2013-08-11 15:37:18', '50.080325,14.419828333' (Spálená 25-27, 110 00 Prague 1, Czech Republic)

1692: '2013-08-12 06:33:37'-'2013-08-12 06:33:37', '50.080215,14.41986' (Spálená 78/12, 110 00 Praha-New Town, Czech Republic)

A good example of a large capture in the street was the *Guigui* device shown in Figure 10. The device was seen for almost 2 km.

4.2 Strange behavior of bluetooth devices

Analyzing the dataset we found several strange conditions or errors on some bluetooth devices. For example, the following device reported two different names in a short period of time:

05324723383: '2013-06-30 00:20:09'-'2013-06-30 00:20:29', '50.087581667,14.420585' (Old Town Square 934/5, 110 00 Prague-Prague 1, Czech Republic)

SGH-D900i antonio: '2013-06-30 00:20:43'-'2013-06-30 00:20:43', '50.087591667,14.420628333' (Old Town Square 934/5, 110 00 Prague-Prague 1, Czech Republic)

05324723383: '2013-06-30 00:20:59'-'2013-06-30 00:20:59', '50.087601667,14.42065' (Old Town Square 934/5, 110 00 Prague-Prague 1, Czech Republic)

We are not sure why this device behave like that. Maybe it was booting up or changing from a default configuration state to the user-defined name.

Another example of strange behaviors are the generic cell phones usually bought on online stores. These phones had an unregistered MAC address. For example the MAC address of the following phone, 29:94:44:C2:66:22, could not be identified, however the device name corresponds to this type of generic phone.

MTKBTDEVICE: '2013-08-12 06:27:08'-'2013-08-12 06:27:08',



Figure 8: Medical device found on the street with bluetooth

'2013-06-30 13:40:43' '2013-06-30 13:40:43' 00:16:DB:A2:9D:CC '50.07325,14.414546667' ' ' STIG

Device Features

Positions Map



Figure 9: The same device comes back at the same spot the other day

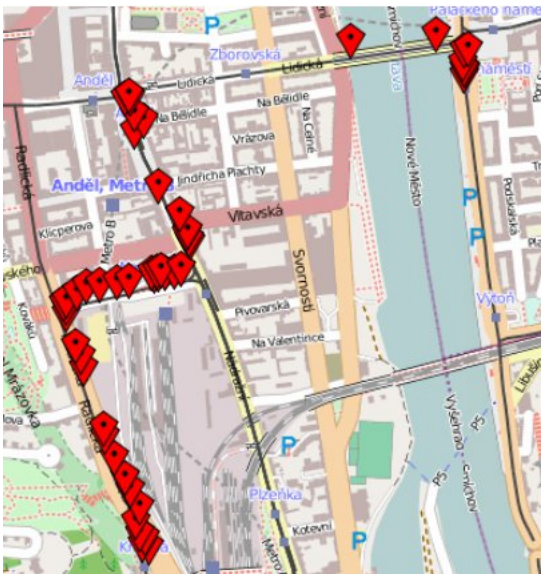


Figure 10: Following a device using public transportation

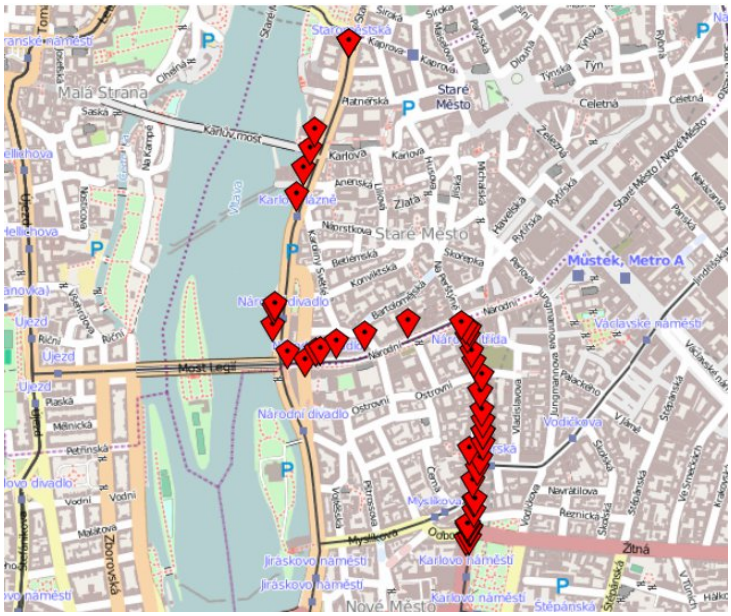


Figure 11: Following a device for a long time

'50.061963333,14.402515' (Radlická 35, 150 00 Prague 5, Czech Republic)

MTKBTDEVICE: '2013-08-12 06:30:04'-'2013-08-12 06:30:04', '50.068155,14.402918333' (Za Ženskými domovy 4, 150 00 Prague 5, Czech Republic)

MTKBTDEVICE: '2013-08-12 06:32:24'-'2013-08-12 06:32:24', '50.07092,14.404438333' (Nádražní 222/23, 150 00 Prague 5-Smíchov, Czech Republic)

The most important problem that we found regarding privacy issues is related to *DELL* notebooks. Usually we think that the bluetooth device in our notebook is turned on only by the operating system and that we can control when it is working or not by switching a hardware or software button. This is the most common way to protect our computers from accidentally using a bluetooth communication. However we found that during the booting process of these DELL notebooks there is a short period of time where the BIOS activates the bluetooth device with a default name and makes it discoverable. This device is turned off later and the control is then passed to the operating system. This means that *even if the user turned off the bluetooth device in the operating system we are still able to capture its information* and found out its MAC address. The following are examples of this behavior from two different DELL computers:

Dell Wireless 365 Bluetooth Module: '2013-08-14 15:25:29'-'2013-08-14 15:29:03'

Dell Wireless 365 Bluetooth Module: '2013-06-30 23:27:17'-'2013-06-30 23:27:17'

Finally, the dataset is also useful also to evaluate the precision of your GPS device. Fig.4.2 shows how the GPS signal varies by leaving the capture device on the same spot during all night. This type of analysis helps us to figure it out the probable position error on the rest of the captures. At best the precision is about several meters, but it can be as bad as 150m.

5 Conclusion

In this work we have shown how critical the results are when the information of bluetooth devices is combined with GPS data. This information allows us to see the behavioral patterns of people and also to track down those individuals without them even noticing. But the problem is even deeper when we can deduce from this captured data the kind of disease of a person, the technological furnitures inside a home or the social status of the owner.

We conclude that:

Most people are not aware of the amount of information being leaked by their own bluetooth devices.

By mixing GPS data with bluetooth information our privacy can be easily abused.

The user must have the right to choose whether to share this information or not. In the case of car audio's bluetooth devices and DELL notebooks, the user doesn't have this choice.

It is clear that we need to raise people's awareness about this matter and why it is a privacy issue.

Is it worth the use of bluetooth technology in comparison with the amount of information disclosed?

We should have the right to power down these devices if we want to. But we demonstrated that in some cases, the user cannot turn out these devices and therefore he/she cannot prevent the leaking of the information.

Future work may include the use of bluetooth attacking tools to take controls of the devices, leveraging the security issue to a more serious stage.

6 About the Authors

Verónica Valeros is one of the founders of the MatesLab Hackerspace, the first hackerspace in Mar del Plata, Argentina. She is actually based in Czech Republic. Her passion lies on information security and privacy, python programming, networking analysis, lockpicking and traveling. Her

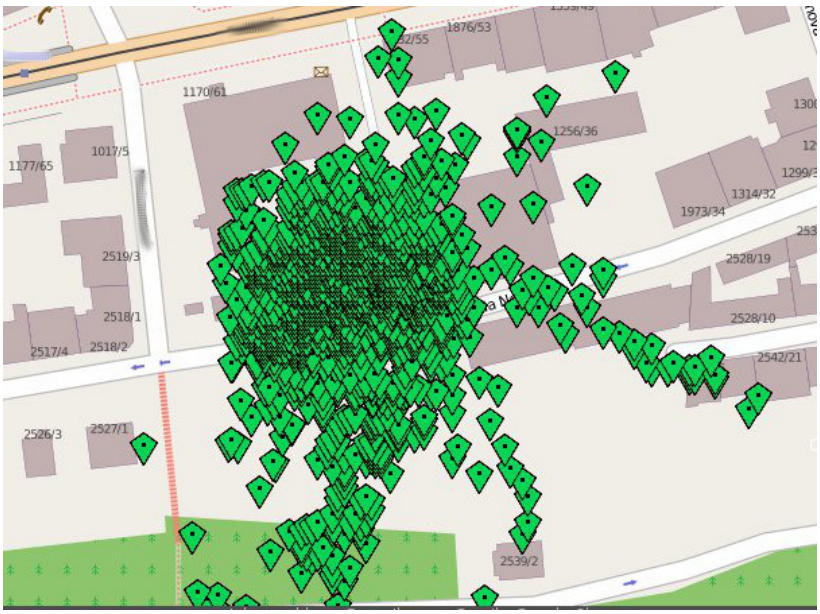


Figure 12: GPS signal error during all night in the same place.

work is focused now on malware research and anomaly detection.

Sebastián García is co-founder of the Mateslab HackSpace in Argentina and a PhD student in the UNICEN University in Argentina and the ATG of CVUT University in Czech Republic. His research interests include network-based botnet behavior detection, bluetooth analysis, anomaly detection, penetration testing, honeypots, malware detection and keystroke dynamics. His recent projects focus on using unsupervised and semi-supervised machine learning techniques to detect botnets on large networks based on their behavioral models.

They can be reached at vero.valeros@gmail.com and eldraco@gmail.com, respectively.