Geolocalization Techniques using Log Files in Android

T. Ramakrishnaiah
Assistant Professor (Sr. Grade), ECE
Vardhaman college of engineering, Hyderabad, INDIA

A. Sandeep
M.Tech student, ECE
Vardhaman college of engineering, Hyderabad, INDIA

ABSTRACT

The location-sensing technology is core technology for location based services and applications. In these days Localization for mobile handset devices are gaining huge importance in various fields, they are globally useful, such as navigation, advertisements, safety security and also observation of tracking. This paper presents various techniques for location-sensing method, called a cell Id based location positioning method and its positioning accuracy estimation. In addition this paper proposed Geolocation technique which is used to track a specific user without permissions. This includes tracking, reading call history, past location history, reading SMS. To evaluate proposed system this paper implemented location-sensing technique on Android-based phones. There are different ways how information about the location of a mobile device is generated. This paper explains the state of art technology of WiFi and Cell Id based positioning and describes some of its applications. By knowing how location data is collected and processed the reader understands how location-based solutions work. Some examples are given at the end of the paper.

General Terms

Geolocalization, log files, android, GSM, cell location, radio logs, tcpdump, attack vectors.

Keywords

Location-sensing, location determination, cell ID, log files.

1. INTRODUCTION

The idea behind WiFi and Cell-ID positioning is to exploit the knowledge of GSM-Cell's or Access Points' (APs) geographic position and to calculate the position of a mobile device out of this data. It is an advantage that the infrastructure of needed for Cell-ID and WiFi based positioning already is installed almost everywhere in the world. Often location-based services are useful in urban areas. There the density of WiFi Access Points and GSM-Cells is very high. Therefore the positioning is accurate enough to meet today's standards.

To Geolocate, it has several techniques, they are GPS [2], WiFi and GSM Cell-ID.

1.1 GPS

GPS [2] is considered one of the most well-known localization techniques, it is not available in many cell phones, requires direct line of sight to the satellites, and consumes a lot of energy, and it doesn't work inside the buildings, subways. We can't locate our self in the subways and in the buildings. Therefore, research for other techniques for obtaining cell phones location has gained momentum fueled by both the users' need for location-aware applications and government requirements, e.g. FCC [3]. City-wide WiFi-based localization for cellular phones has been investigated in [3], [4] and commercial products are currently available [6].

1.2 WiFi

However, WiFi chips, similar to GPS [2] , are not available in many cell phones and not all cities in the world contain sufficient WiFi coverage to obtain ubiquitous localization, and far less accuracy compared to GPS. Similarly, using augmented sensors in the cell phones. e.g. accelerometers and compasses, for localization have been proposed in. However, these sensors are still not widely used in many phones.

1.3 GSM-Cell ID

GSM-based localization [3],[5],[11],[12] by definition, is available on all GSM-based cell phones, which presents 80-85% of today's cell phones, works all over the world, and consumes minimal energy in addition to the standard cell phone operation. Many research works have addressed the problem of GSM localization, including time-based systems, angle-of-arrival based systems, and received signal strength indicator (RSSI) [8] based systems. Only recently, with the advances in cell phones, GSM-based localization systems have been implemented. These systems are mainly RSSI-based as RSSI information is easily available to the user's applications. Since RSSI is a complex function of distance, due to the noisy wireless channel, RSSI-based systems usually require building an RF fingerprint of the area of interest. A fingerprint [5] stores information about the RSSI received from different base stations at different locations in the area of interest. This is usually constructed once in an offline phase. During the tracking phase, the received RSSI at an unknown location is compared to the RSSI signatures in the fingerprint and the closest location in the fingerprint is returned as the estimated location. Constructing the fingerprint is a time consuming process. However, this is typically done in a process called war driving, where cars scan the streets of a city to map it. Current commercial systems, such as Skyhook, Google's MyLocation and StreeView services already perform scanning for other purposes. Therefore, constructing the fingerprint for GSM localization can be piggybacked on these systems without extra overhead.

1.4 Cell Location Resolution

A GSM Cell ID (CID) [3] is a generally unique number used to identify each (BTS) or sector of a BTS within a Location area code (LAC) if not within a GSM network.

In some cases the last digit of CID represents cells' Sector ID.

Every GSM cell (BTS) is identified by 4 numbers

- MCC: Mobile Country Code
- MNC: Mobile Network Code
- LAC: Location Area Code

- CID: Cell Id

There have been several attempts to build database of GSM cells

2. EXISTING SYSTEM

In previous GSM location based technique [1],they proposed a probabilistic fingerprinting based technique for GSM localization. The CellSense probabilistic technique provides more accurate localization. However, constructing a probabilistic fingerprint is challenging, we need to stand at each fingerprint location for a certain amount of time to construct the signal strength histogram. This adds significantly to the overhead of the fingerprint construction process.

3. PROPOSED SYSTEM

In this project, proposed the GSM phone localization by sniffing the Cell Ids with more accuracy. In addition also implementing to reading past location history, call history, SMS.

To evaluate proposed system also implemented location-sensing technique on Android-based phones. There are different ways how information about the location of a mobile device is generated. This paper explains the state of art technology of WiFi and Cell Id based positioning and describes some of its applications.

By knowing how location data is collected and processed the reader understands how location-based solutions work. Some examples are given at the end of the paper

4. THE GEO LOCALIZATION SYSTEM

This paper investigates the Location based techniques in various methods, it uses Google indexing power, it works using google cars, these are the cars that build google street views and takes pictures of street views. The google cars not only taking pictures but also index all WiFi access points and all GSM BTS. They have complete data base of GSM [2] cells and all android users also. All android users are updating to this very large google date base. Our proposed system divided into small modules, they are given as bellow.

- Cell location resolution
- Attack vectors
- Attack basics
- Physical attacks
- Remote attacks
- Demo

4.1 Cell location resolution

There is a Google API which is quite confidential. Try to run google map in android mobile with GPS and WiFi disabled. Now android mobile will only be able to geo locate using GSM. Google API is used to geolocate.

4.1.1.1 Reverse Engineering

Android google map internels: Compile tcpdump for android mobile. This will need two cross compile tcpdump for ARM architecture. There are exchanges in binary called proprietary binary protocol, let it use HTTP and HTTP POSTed to http://www.google.com/glm/mmap.The four main codes MCC, MNC, LAC & CID are should be pack in one proper binary format and send them to google server .It will give answer also in binary, It is quite buggy. When we basically do twice the query it doesn't give the same answer.

Source: "Poor Man's GPS" by Dhaval Motghare http://www.orangeapple.org/?p=82.

4.1.1.2 Sniffing GSM network

By mapping the entire GSM network in given area using sniffing, every single BTS broadcast its cell id and locationary code, by sniffing all LAC and CID'S from given network can be collected. This can be done following,

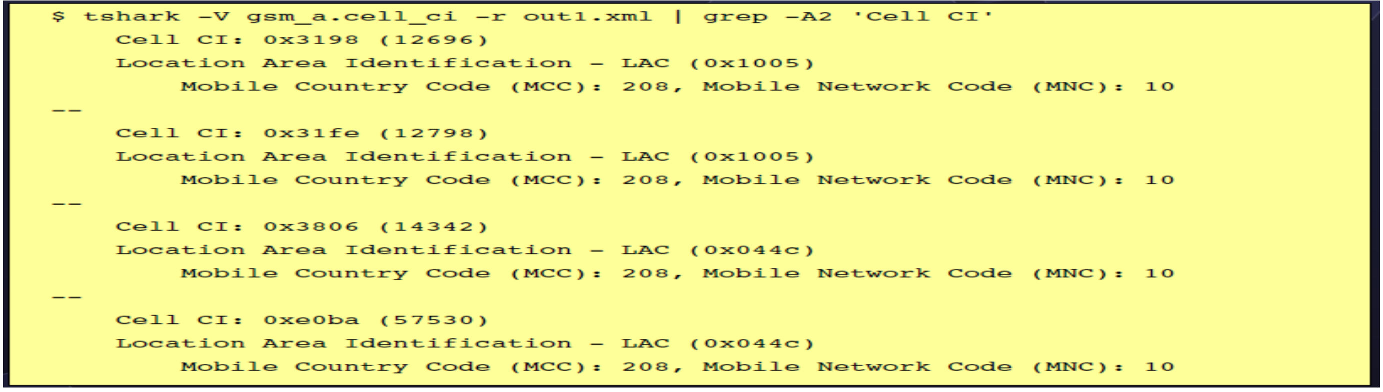
To sniff a given GSM Band, steps are explained below,

- SDR (Software Defined Radio).
- USRP 1 from ETTUS Research LLC.

In order to achieve use the "AIR Probe project" which is a project for GSM sniffing. Scan the frequency to find out all the different BTS with GNU RADIO. Few simple steps are explained below

1. Scan with GnuRadio: First scan the frequency to find all the different BTS
2. Demodulate with Airprobe: If few good frequencies identified, demodulate sine wave with Airprobe.
3. Decode with Wireshark: Decode the Demodulated sinewave using Wireshark.

Here is the result from the demodulated capture

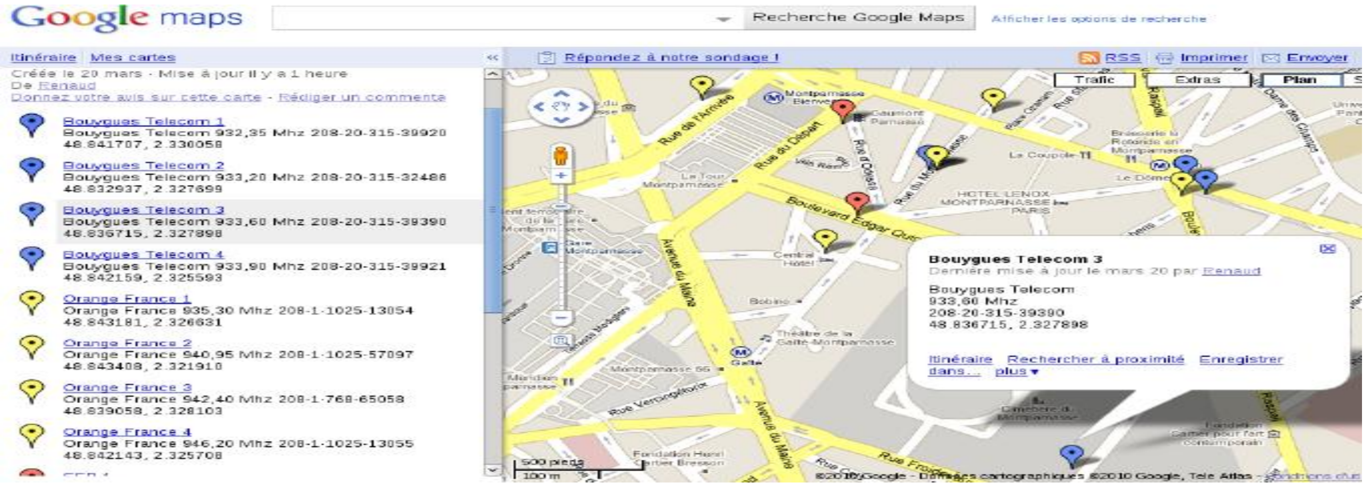


```
$ tshark -V gsm_a.cell_ci -r out1.xml | grep -A2 'Cell CI'
    Cell CI: 0x3198 (12696)
    Location Area Identification - LAC (0x1005)
        Mobile Country Code (MCC): 208, Mobile Network Code (MNC): 10
--
    Cell CI: 0x31fe (12798)
    Location Area Identification - LAC (0x1005)
        Mobile Country Code (MCC): 208, Mobile Network Code (MNC): 10
--
    Cell CI: 0x3806 (14342)
    Location Area Identification - LAC (0x044c)
        Mobile Country Code (MCC): 208, Mobile Network Code (MNC): 10
--
    Cell CI: 0xe0ba (57530)
    Location Area Identification - LAC (0x044c)
        Mobile Country Code (MCC): 208, Mobile Network Code (MNC): 10
```

Fig 1: Demodulated Capture

The above result was taken as an XML file and applied a filter in order to only to keep the packets with CELL IDS. Here in the capture four different BTS broadcasts with MCC MNV LAC and CID.

From the demodulated results, mapping according to the above result.



Créée le 20 mars · Mise à jour il y a 1 heure
De Renaud
Donnez votre avis sur cette carte - Rédiger un commente

Bouygues Telecom 1
Bouygues Telecom 932,35 Mhz 208-20-315-39920
48.841707, 2.330058

Bouygues Telecom 2
Bouygues Telecom 933,20 Mhz 208-20-315-32486
48.832937, 2.327699

Bouygues Telecom 3
Bouygues Telecom 933,60 Mhz 208-20-315-39390
48.836715, 2.327898

Bouygues Telecom 4
Bouygues Telecom 933,90 Mhz 208-20-315-39921
48.842159, 2.325593

Orange France 1
Orange France 935,30 Mhz 208-1-1025-13054
48.843181, 2.326631

Orange France 2
Orange France 940,95 Mhz 208-1-1025-57097
48.843408, 2.321910

Orange France 3
Orange France 942,40 Mhz 208-1-768-65058
48.839058, 2.328103

Orange France 4
Orange France 946,20 Mhz 208-1-1025-13055
48.842143, 2.325708

Bouygues Telecom 3
Dernière mise à jour le mars 20 par Renaud
Bouygues Telecom
933,60 Mhz
208-20-315-39390
48.836715, 2.327898

Itinéraire Rechercher à proximité Enregistrer dans... plus ▾

Fig 2: Mapping according to the demodulated result

4.2 Attack Vectors

Android uses a specific logging facility and it is quite unknown. Unfortunately enable by default. Android has 3 or 4 different logs depends on the phone model. These logs are not plain text files like in Unix or Linux. These logs are in fact circular memory buffers. It means they have very unique size. These logs are handles by Character device. Files like character device drivers in linux and these drivers are not easy to handle. There is a built in tool in every single android phone. Which will help us to manage or to read these logs.

Log is a logging class that you can utilize in your code to print out messages to the LogCat. Common logging methods include

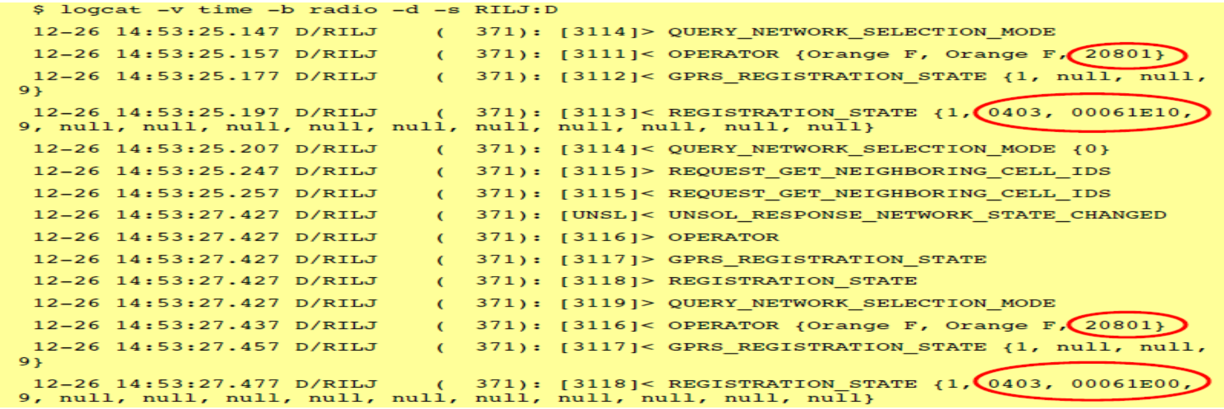
- v(String, String) (verbose)
- d(String, String) (debug)
- i(String, String) (information)
- w(String, String) (warning)
- e(String, String) (error)

These are character device files, the system log, the radio log, the main log and the events log. In logcat utility look at log size using logcat,

- **-b**: to select a given log
- **-g**: to query the character device for its size

By looking at the contents of the log file, we can notice the following, From the image file, the complete data is in binary format.

Now have a look at the radio log in the below image, it showing a logcat with the radio log in Verbose modem to show the time -v , and –s is to set a filter on the log. In order to have only the log files from the RIL (Radio Interface Layer) Daemon which is Daemon that handles the GSM chip on android phone. The MCC and the MNC of the operator can be find at the end of the line, and in other lines with the keyword registration state, it has LAC and the CID. It has two different CID's and the corresponding time in the logs that means it has complete history during the day.



```
 $ logcat -v time -b radio -d -s RILJ:D
  12-26 14:53:25.147 D/RILJ     (  371): [3114]> QUERY_NETWORK_SELECTION_MODE
  12-26 14:53:25.157 D/RILJ     (  371): [3111]< OPERATOR {Orange F, Orange F, 20801}
  12-26 14:53:25.177 D/RILJ     (  371): [3112]< GPRS_REGISTRATION_STATE {1, null, null,
9}
  12-26 14:53:25.197 D/RILJ     (  371): [3113]< REGISTRATION_STATE {1, 0403, 00061E10,
9, null, null, null, null, null, null, null, null, null, null}
  12-26 14:53:25.207 D/RILJ     (  371): [3114]< QUERY_NETWORK_SELECTION_MODE {0}
  12-26 14:53:25.247 D/RILJ     (  371): [3115]> REQUEST_GET_NEIGHBORING_CELL_IDS
  12-26 14:53:25.257 D/RILJ     (  371): [3115]< REQUEST_GET_NEIGHBORING_CELL_IDS
  12-26 14:53:27.427 D/RILJ     (  371): [UNSL]< UNSOL_RESPONSE_NETWORK_STATE_CHANGED
  12-26 14:53:27.427 D/RILJ     (  371): [3116]> OPERATOR
  12-26 14:53:27.427 D/RILJ     (  371): [3117]> GPRS_REGISTRATION_STATE
  12-26 14:53:27.427 D/RILJ     (  371): [3118]> REGISTRATION_STATE
  12-26 14:53:27.427 D/RILJ     (  371): [3119]> QUERY_NETWORK_SELECTION_MODE
  12-26 14:53:27.437 D/RILJ     (  371): [3116]< OPERATOR {Orange F, Orange F, 20801}
  12-26 14:53:27.457 D/RILJ     (  371): [3117]< GPRS_REGISTRATION_STATE {1, null, null,
9}
  12-26 14:53:27.477 D/RILJ     (  371): [3118]< REGISTRATION_STATE {1, 0403, 00061E00,
9, null, null, null, null, null, null, null, null, null, null}
```

Fig 3: History of user's visited MCC's+MNC's, Cell ID's, LAC's in radio logs

4.3 Attack Basics

- Collect history of visited GSM cells on the victim's side (no prior access needed)
- Send them to the attacker
- Resolve them into latitude and longitude

4.3.1.1 Attack Range

Two ways to attack on a target mobile,

- Local (i.e. physical attack)
- This can be your friend phone or kid phone, if physical access is available, this can be done.
- Remote (here remote means remote vulnerability). Remotely using local vulnerability and a bit social engineering.

4.4 Physical Attack

- Connect the victims phone to the attack computer via USB
- Requires:
- -Physical access to the victim's phone for a few seconds

Works even if the victims phone is locked (Using USB debugging function).

4.5 Remote Attack

- Remotely spy the victim
- Malware application who abuse either

-User trust
-Android security model

Requires a bit of social engineering (or not)This will allow you to spy the victim using malware ,the best way to do this is to install malware in phone .Which will abuse either trust of the user or the android security model. This will require a bit social engineering. How does security works with android ? When a user download an application from the android market, it will ask you for the require permissions. An application can't use permission which is not declared in manifest file. This is fine grade permission model which is far more secure than the iphone application. For this Java codes are helpful.

- Android permissions model:

-Dalvik(java) sandbox

- Permission: android.permission.*
- What can a user fear?

-Dangerous combination of 2 permissions:
ACCESS_COARSE_LOCATION
or ACCESS_FINE_LOCATION + INTERNET

In the above permissions, both uses GSM and WiFi networks. With internet access the application will be able to report users location.

4.5.1 1st Attack

- Use both permissions:

-Internet permission is needed for free ad sponsored application.
-Official Geolocation permission is needed for location aware applications

In 1st attack, use both permissions, most users don't care permissions and will install a game which uses the location permission and the internet permission even that game isn't related to those permissions. Because Internet permission is needed for free ad sponser application. Geolocation permission is needed for location aware application.

4.5.2 2nd Attack

Use the radio logs:
-Instead of using Geolocation API, use radio logs of a user to collect the Cell Ids
-Write Result in to the system log(no permission needed)
-Voluntarily crash the application when needed (no permission needed!)
-If the user reports the crash, System log is sent to the developer using integrated google feedback client.

2nd attack is based on radio logs, instead of using android location API only read radio logs, it requires only read logs permissions to collect different LAC and Cell IDs. Write those results in the system logs and permissions not required to write in to system log because this is use for debugging. Voluntarily crash the application when needed, If the user reports the crash, System log is sent to the developer using integrated google feedback client. It doesn't require any internet access to remotely collect the CIDs, It use the Integrated feedback client.

4.5.3 Spying Users

- Much more interesting information in different logs:

-Phone calls (number & duration)
-SMS (PDU format)

- Combination of information:

-Where did phone calls take place?
-Where were SMS sent/received?
-Recovery of deleted SMS and calls

Proposed system can do more than the geolocate a user, there are much more interesting information in radio logs and other logs. All call history data and SMS data are encoded in PDU format. By decoding the information we can read them. Even able to recover deleted SMS and delete call history. Length of the history depend on log filling, if the user has moved quickly for the last hours or as sent several text messages and only few phone calls, the log will fill very quickly. If the user hasn't travelled a lot, hasn't phoned a lot it will take whole day to recovery the day. Log size can be changed using logcat tool. Now it has complete geolocation, calls and SMS tracking nearly no permission needed.

5. HOW TO ROTECT YOURSELF

- Carefully look at the applications using NDK (apk archives embedding .so files)
- Don't install any applications requiring READ_LOGS permissions.
- Don't submit bug reports (or at least choose not to include system logs with submission)
- Reduce logcat buffer size (seems tricky : logcat –r / logcat –n)
- Often clear your logcat (logcat –b logcat –c)
- Disable radio logs (seems tricky too !)

6. EXPERIMENT RESULTS

Collect the different cell Ids, and resolve the cell ids in latitude and longitude and build a google map with all the different positions in the day. They are associated with date and time. This was in last august A User have leaved the walk at 20/6 and arrived home at nearly 14 minutes after. But a user stayed at home up to 11 PM, a user went back to home after walk, a complete 5 hours of history shown in the below figure.

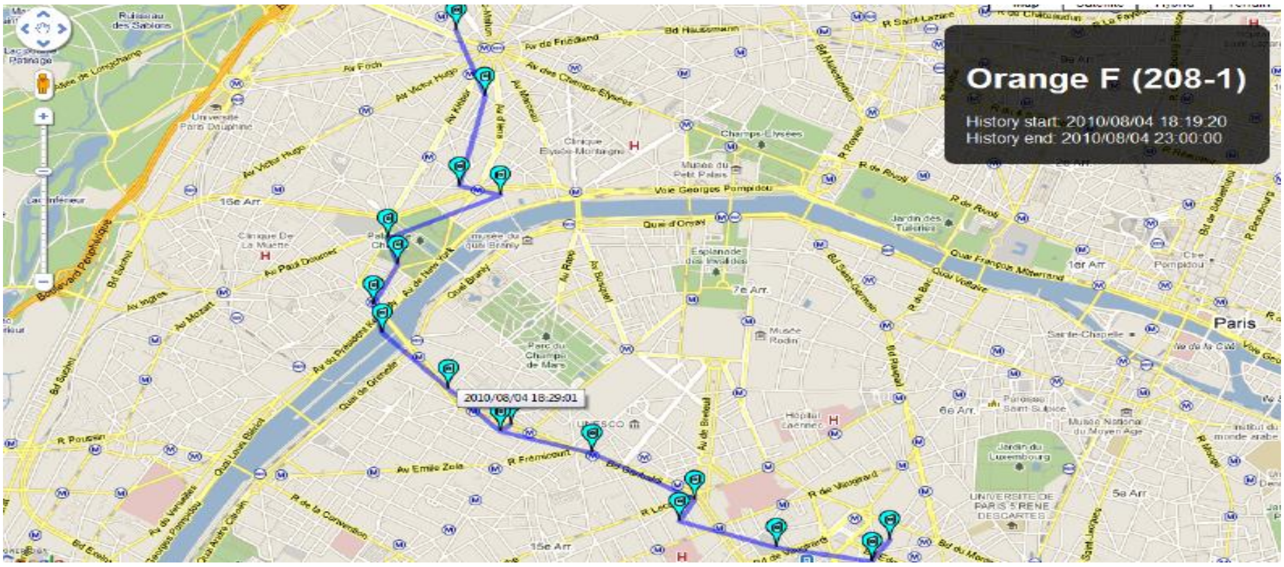

Orange F (208-1)
History start: 2010/08/04 18:19:20
History end: 2010/08/04 23:00:00
2010/08/04 18:29:01
Paris

Fig 4: Dumping and viewing a user's past location history

7. CONCLUSION

In this paper, several important issues explained for the GSM cell phones localization and proposed new methods to geolocate, and presented the details of the system and how it constructs the geolocation without incurring any additional overhead.

This paper implemented a system on Android-based phones and compared it to other GSM-localization systems.

In addition this paper proposed Geolocation technique which is used to track a specific user without permissions. This includes tracking, reading call history, past location history, reading SMS.

The Future scope for this paper can be possible by extending this system in different directions including using parametric distributions, clustering of fingerprint locations, experimenting with larger datasets, comparison with other city-wide commercial systems, targetting low-end phones [19], among others.

8. REFERENCES

[1] M. Ibrahim and M. Youssef, "CellSense: A probabilistic RSSI based GSM positioning system," in GLOBECOM, 2010.

[2] P. Enge and P. Misra, "Special issue on GPS: The Global Posiioning System," Proceedings of the IEEE, pp. 3172, January 1999.

[3] S. Tekinay, "Special issue on Wireless Geolocation Systems and Services," IEEE Communications Magazine, April 1998.

[4] Y.-C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm, "Accuracy characterization for metropolitan-scale wi-fi localization," in MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services. New York, NY, USA: ACM, 2005, pp. 233–245.

[5] I. Smith, J. Tabert, A. Lamarca, Y. Chawathe, S. Consolvo, J. Hightower, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, P. Powledge, G. Borriello, and B. Schilit, "Place lab: Device positioning using radio beacons in the wild," in Proceedings of the Third International Conference on Pervasive Computing Springer, 2005, pp. 116–133.

[6] Skyhook wireless, http://www.skyhookwireless.com.

[7] R. R. C. Ionut Constandache and I. Rhee, "Towards mobile phone localization without war-driving," in IEEE Infocom, 2010.

[8] R. S. Andrew Offstad, Emmett Nicholas and R. R. Choudhury, "Aampl: Accelerometer augmented mobile phone localization," in ACM MELT Workshop (with Mobicom 2008), 2008.

[9] I.C.Martin Azizyan and R. R. Choudhury, "Surroundsense: Mobile phone localization via ambience fingerprinting," in ACM MobiCom, 2009.

[10] Wikipedia, "Comparison of mobile phone standards — Wikipedia, the free encyclopedia," 2010, [Online; accessed 25-March-2010]. [Online]. Available:

\url{http://en.wikipedia.org/wiki/Comparison of mobile phone standards}.

[11] M. Y. Chen, T. Sohn, D. Chmelev, D. Haehnel, J. Hightower, J. Hughes, A. Lamarca, F. Potter, I. Smith, and A. Varshavsky, “Practical metropolitan-scale positioning for GSM phones,” in Proceedings of the Eighth International Conference on Ubiqui- tous Computing (UbiComp. Springer, 2006, pp. 225–242.

[12] A. Varshavsky, M. Y. Chen, E. de Lara, J. Froehlich, D. Haehnel, J. Hightower, A. LaMarca, F. Potter, T. Sohn, K. Tang, and I. Smith, “Are GSM phones THE solution for localization?” in WMCSA ’06: Proceedings of the Seventh IEEE Workshop on Mobile Computing Systems & Applications. Washington, DC, USA: IEEE Computer Society, 2006, pp. 2028.

[13] E. Elnahrawy, J. Austen-francisco, and R. P. Martin, “Adding angle of arrival modality to basic RSS location management techniques,” in In Proceedings of IEEE International Symposium on Wireless Pervasive Computing (ISWPCŠ07), 2007.

[14] E. Elnahrawy, J. austen Francisco, and R. P. Martin, “Poster abstract: Bayesian localization in wireless networks using angle of arrival,” in Proceedings of the Third ACM Conference on Embedded Networked Sensor Systems (SenSys’05), 2005.

[15] P. Biswas, H. Aghajan, and Y. Ye, “Integration of angle of arrival information for multimodal sensor network localization using semidefinite programming,” in In Proceedings of 39th Asilomar Conference on Signals, Systems and Computers, 2005.

[16] M. Li and Y. Lu, “Angle-of-arrival estimation for localization and communication in wireless networks,” 2008.

[17] Google maps, http://www.google.com/mobile/ maps/.

[18] M. Youssef, M. A. Yosef, and M. N. El-Derini, “GAC: Energy efficient hybrid gps-accelerometer-compass gsm localization,” in GLOBECOM, 2010.

[19] M. I brahim and M. Youssef, “A hidden markov model for localization using low-end GSM cell phones,” in ICC, 2011.