Enabling the Ciphering Indicator on Android

Felician ALECU, Paul POCATILU
Department of Economic Informatics and Cybernetics
The Bucharest University of Economic Studies, Romania
ROMANIA
felician.alecu@ie.ase.ro, ppaul@ase.ro

Abstract: This paper exemplifies the use of AT commands to retrieve (and eventually override) the ciphering indicator status on Android by directly communicating with the phone modem. Curiously, his indicator is disabled by default both at operating system level and SIM card settings. By turning it on, the mobile handset will inform the user each time the communication becomes unencrypted, so a proper decision could be made just in time.

Key-Words: GSM encryption, IMSI catcher, A5, Ciphering Indicator, SIM, AT Commands.

1. Introduction

One of the most globally used cellular networks are based on GSM standard. Generally speaking, the GSM communication is encrypted, but there are networks that do not support encryption and it will be relative easier for a third party to intercept the traffic. As GSM specification states that the users has to be aware of such kind of communication. Also, even the communication is encrypted, researchers and practitioners manage to break the ciphers.
This is why the security of mobile and wireless traffic is a very important issue that affects billions of users worldwide. This complex topic is approached in numerous papers such as [6], [7] and [8]. The researches focus on certain aspects of mobile and wireless communication security.
In this respect, the paper aims to present the most relevant researches related to this field.
The paper is structured as follows.
Section *GSM Service* presents a short description of GSM architectural components.
GMS Security section aims to presents the security issues related to GSM networks, including encryption algorithms for data and voice.
In *IMSI (International Mobile Subscriber Identity)* catcher is described the architecture and the components of fake service that intercepts all incoming and outgoing generated traffic.
The section *Spoofing a GSM Network* presents the mechanisms used to intercept the communication in a GSM network.
The section, *Turning off the GSM Encryption*, focuses on current implementation of ciphering indicators on mobile devices.
The last section, *Obtaining the SIM Ciphering Indicator status on Android*, presents a practical method to access the SIM cards parameters using AT commands on Android phones, connected as modems.
The paper ends with conclusions and future work.

2. GSM Service

GSM was launched as a service in 1991 but its development started in 1982 with the intention of replacing the incompatible cellular systems from Europe. Today it is spread all around the globe, being the most used standard for cellular devices worldwide. Compared with CDMA, GSM networks counts over 60% of the market worldwide, with significant differences in America and Europe.

ISSN 2067 – 4074

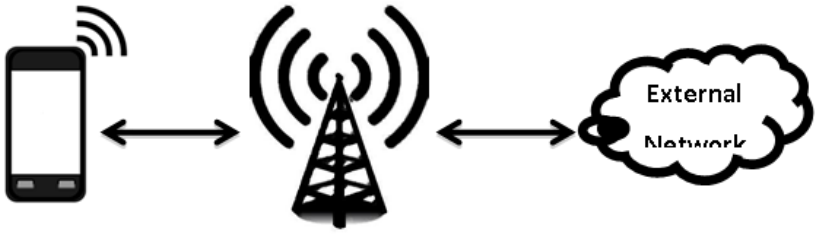

Figure 1. The Simplified GSM Architecture

The GSM architecture (Figure 1) consists of the following devices:

- Mobile phones – in order to function properly, each GSM device is using a Subscriber Identity Module (SIM) that is removable, so it can be used on several devices (but not in parallel, of course). Duplicate SIM cards are not allowed to operate into any GSM network. Also, a phone cannot be used for voice or data transfers without having a SIM card installed;
- Base stations – are usually connecting the users of mobile phones with the fixed networks by an air interface.

In [7] is presented a more detailed architecture of GSM networks.

The SIM card stores the user IMSI (International Mobile Subscriber Identity) that is sent to the base station when the phone wants to connect to the GSM network. Actually, this identifier can be captured by a fake base station that can intermediate the entire traffic between the mobile device and the real provider base stations. Fortunately, this traffic is encrypted by using specific algorithms (see below).

3. GSM Security

Due to the use of an air interface, the GSM communications are considered as being less secure than wired networks, so the traffic is encrypted by using dedicated algorithms. In order to be able to encrypt/decrypt data on the fly, any mobile phone has a built in A5 encryption algorithm implemented directly at hardware level.

There are several versions of the A5 algorithm, as follows [14]:

- A5/0 – no encryption at all, for example there are countries where encrypting the phone calls is illegal, so the A5/0 is a good option in these cases;
- A5/1 – original encryption algorithm used in Europe;
- A5/2 – a weaker version created for export (and used in USA);
- A5/3 – stronger version used by 3G networks.

A5/1 is the most used encryption algorithm today because it is preferred over the 2G networks while most of the phones are still containing 2G only options to be used to preserve the battery life while the mobile device is not performing any data traffic.

Even if the A5/1 algorithm specifications were initially secret, it was disclosed in 1999 by the use of reverse engineering. Successful attacks over the A5/1 were reported in the past years [13], so a good option for self-protection is to avoid using 2G networks at all since the 3G is providing a much better protection for the voice, text and data transfers [12]. But, setting such an option seems to be not an easy task. For Android, for example, the 2G connections only option is listed right under the settings while the feature of allowing only 3G or upper links is deeply hidden under a special set of options (called testing menu) that cannot be accessed very easy.

So, to disallow 2G connections on an Android phone, the user has to [4]:

- dial a special code, *#*#4636#*#* that is opening Android testing menu
- go to the phone information
- locate the set preferred network type option and set it to WCDMA only. Each time the phone is restarted, the option will go back to default. Also, in areas where no 3G is available, the phone will be simply out of connection, so the user has to access again the special menu to modify the settings.

This seems quite complicated while the opposite option, choosing a less secure

connection, is directly accessible from the settings menu and checking it is very tempting simply because it says it saves the battery life (Figure 2).

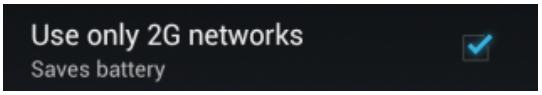

Figure 2. 2G/3G selection

But, as we will see next, such an option could be dictated (overwritten) by the base station connecting to. And, if this base station is a fake one, the entire traffic could be intercepted very easy.

4. IMSI (International Mobile Subscriber Identity) catcher

Since a mobile device should authenticate to the network while the network should not do the same, it seems pretty clear any device can pretend it represents a mobile tower acting as a base station intercepting the mobile devices in a totally unnoticeable way, so the user of a mobile phone connected to the fake mobile tower has no chance to detect such a situation.

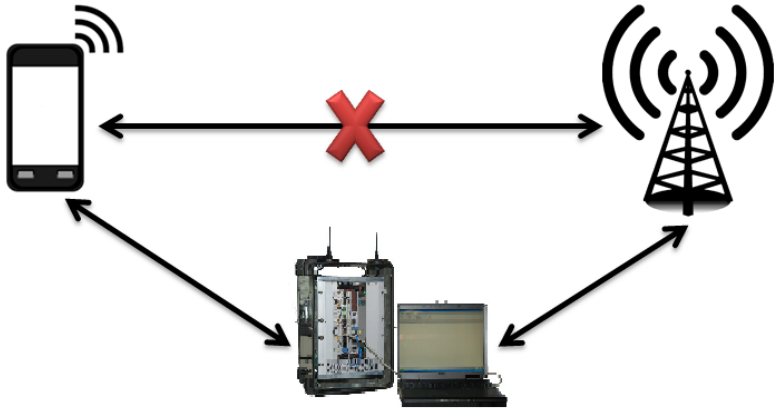

Figure 3. IMSI Catcher – a false base station considered as being trusted

The fake tower is placing itself between the mobile device and the real base stations, so the entire incoming and outgoing traffic will flow through it (Figure 3).
In Romania, the IMSI catchers are used by the police from 2013 by accessing an OLAF Hercules2 European funded project [11] and intelligence agencies to track and intercept mobile communications.
IMSI catchers were first introduced by Rohde & Schwarz, an electronics company having the headquarters in Munich, Germany. The patent was issued in 1993 and invalidated in 2012 because it was considered as being evident.

5. Spoofing a GSM Network

Usually the mobile catcher is placed nearby the location of the phone to be intercepted waiting for the phone to authenticate itself. The only issue for such an approach is to simulate the original network, actually the user will presume something is wrong if the GSM network reported on the device screen is suddenly changing. So the false tower should identify itself as being a real service provider tower which is not very complicated since the GSM mobile networks are identified by the MCC (Mobile Country Code)/MNC (Mobile Network Code) tuples freely available on the Internet [5]. Table 1 presents the MCC/MNC tuples currently available in Romania. Since the mobile phones are trying to connect to the tower providing the best signal, the IMSI catcher will always be preferred and the device will wrongly presume the network trying to connect to, is a trusted one.

Table 1. Romanian MCC/MNC tuples

ISSN 2067 – 4074

MCC	MNC	Brand	Operator	Status
226	01	Vodafone	Vodafone Romania	Operational
226	02	Romtelecom	Romtelecom	Operational
226	03	Cosmote	Cosmote Romania	Operational
226	04	Cosmote/Zapp	Cosmote Romania	Not operational
226	05	Digi.Mobil	RCS&RDS	Operational
226	06	Cosmote/Zapp	Cosmote Romania	Operational
226	10	Orange	Orange Romania	Operational

Now all the traffic (voice, messages, data, etc.) is flowing through the false tower but in an encrypted way, so there is not possible to determine the real meaning of the bits without spending some supplementary work.

6. Turning off the GSM Encryption

After a successfully connection to a GSM tower, since the phone always assumes the network is trusted, the base station is actually dictating the settings to the mobile device, thus the encryption could simply be turned off by telling the phone to disable the GSM encryption. Basically, the false base station can instruct the mobile device to use A5/0 as encryption algorithm meaning no encryption at all will take place during the voice and data transfer between the mobile handset and the base station (real or fake).

In such cases, according to the GSM specifications, the phone normally should warn about using an unencrypted connection but the GSM providers consider such a warning as being confusing for the users, so the ciphering indication is usually disabled directly from the SIM card settings.

So it turns easier to force the phone to use an unencrypted connection rather than spending some processing capabilities to decrypt the A5 algorithm. Finally it seems there is no need to crack the code since the encryption can be just turned off without the user being noticed about.

According to the GSM standard ([2]), "[…] *whenever a connection is in place, which is, or becomes unenciphered, an indication shall be given to the user. Ciphering itself is unaffected by this feature, and the user can choose how to proceed*."

Mobile phones are supposed to include a ciphering indicator, as exemplified in [9]. Unfortunately, only a few mobile phones are giving such indications to the users. For Android, such an issue is opened from 2009 [1] and it is still unresolved, being labeled as an enhancement. Also, for Windows Phone devices this is a feature request as seen in [10].

According to the GSM11.11 (Specifications of the SIM-ME Interface) [18], the SIM content is stored in a special file hierarchy, as presented in Figure 4.

Each file (called EF – Elementary File) is identified by a number. For the one we are interested in (AD), the number is 0x6FAD. Currently, the EF_{AD} file has only 3 bytes defined, as the following [18]:

- 1st – operation mode, like normal, specific activities, maintenance (offline), etc.
  - 00 – normal operation;
  - 80 – type approval operations;
  - 01 – normal operation + specific facilities;
  - 81 – type approval operations + specific facilities;
  - 02 – maintenance (off line);
  - 04 – cell test operation;
- 2nd – additional information, like manufacturer specific data;
- 3rd – additional information, including the Ciphering Indicator on the bit number one (the right side one).

The Ciphering Indicator feature is enabled when bit1 of byte1 is set to 1 (meaning special facilities are on) and the bit1 of byte3 is also on (the ciphering indicator is on).

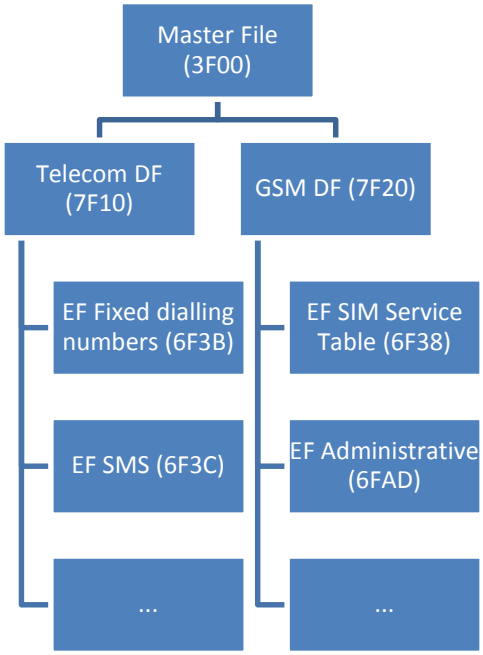

Figure 4. Excerpt of the SIM file structure

7. Obtaining the SIM Ciphering Indicator status on Android

Speaking about the Ciphering Indicator that must be revealed on the phone's display each time the connection is not encrypted, we can easily notice this feature was not implemented at all for most of the mobile oriented operating systems like Android or Windows Phone.
But, even if such a feature may exists, it can be bypassed by the SIM card settings because the service provider may decide to disable the warning by setting the corresponding SIM card bits to OFF. By default, the vast majority of the service providers are issuing SIM cards having this feature already turned off, so it doesn't matter if the operating system is able to show the warning because the ciphering indicator will simply not appear due to the fact it is disabled at the SIM card level.
So, to be able to display the ciphering warning, a phone must meet simultaneously the following two conditions:

- the feature must be supported by the phone's operating system;
- the indicator must not be disabled by the network operator via the SIM card settings.

Unfortunately, these conditions are not met by the vast majority of mobile phones of our days because the most popular operating systems are not implementing the GSM requirement related to the ciphering indicator and the network operators all over the world are turning off the feature by the SIM card default settings.
This is why we can easily assume most of the unencrypted GSM calls and data transfers are taking place without informing the phone user about the total lack of encryption. Related to the vulnerable context of GSM phone/data transfer by wireless communication, the encryption is crucial, so the user must be really informed as soon as the data encryption is missing for any reason (technical or legal limitations).
For a few phones in the world (depending on the handset and/or the mobile provider), the ciphering indicator status (enabled or disabled) can be found out by dialing a dedicated USSD code – *#32489#.
If the above USSD code is not working, the only chance to get the ciphering indicator status is to query the SIM card content. Even if it sounds really simple, due to the fact the ciphering indicator is located in a SIM area with restricted access (EFAD – Elementary File

ISSN 2067 – 4074

Administrative Data), the operating system will simply deny any requests coming from applications trying to access the Administrative Data section. For example, Android clearly states that "*Low level access to the SIM card is not available to third-party apps. The OS handles all communications with the SIM card including access to personal information (contacts) on the SIM card memory*". [15].

Unfortunately, there is no API to be used to access the Administrative Data restricted SIM card area.

Following, we can imagine we can write a dedicated application able to send AT commands [16] to the phone modem that has unlimited access to the SIM card content.

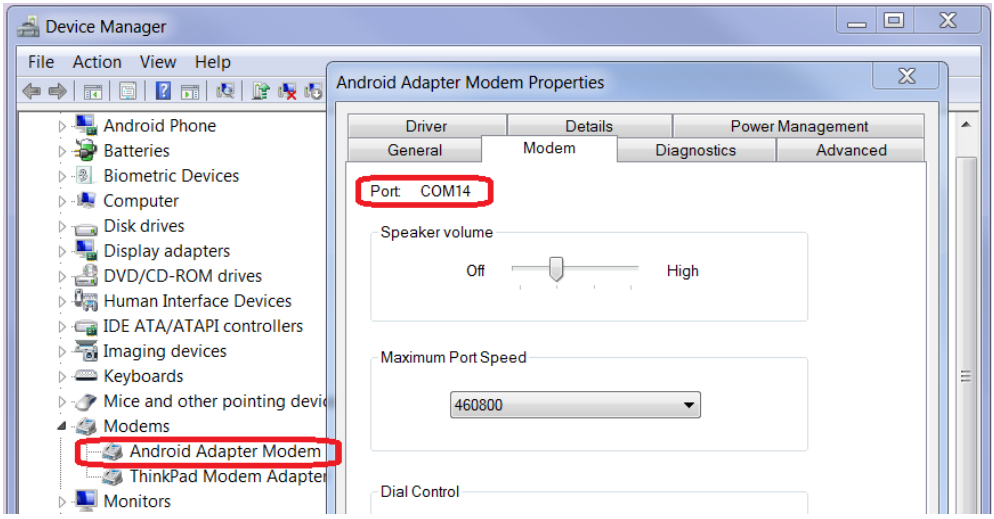

Figure 5. Device Manager Window with Android Adapter Modem installed

Android restricts the possibility of any application to directly discuss via AT commands with the phone modem: "Applications also cannot access AT commands, as these are managed exclusively by the Radio Interface Layer (RIL). The RIL provides no high level APIs for these commands." [15].

Without the existence of any API for SIM card data access (directly or via AT commands), checking and eventually enabling the ciphering indicator seems to be not a very easy task that can only be achieved by using a direct connection to the phone modem that is able to access any SIM card data.

When a phone is connected to a Windows PC, the phone's modem appears under Device Manager, as pictured in Figure 5.

To check the modem is working properly, the Query Modem button of the Properties window can be used, as illustrated in Figure 6.

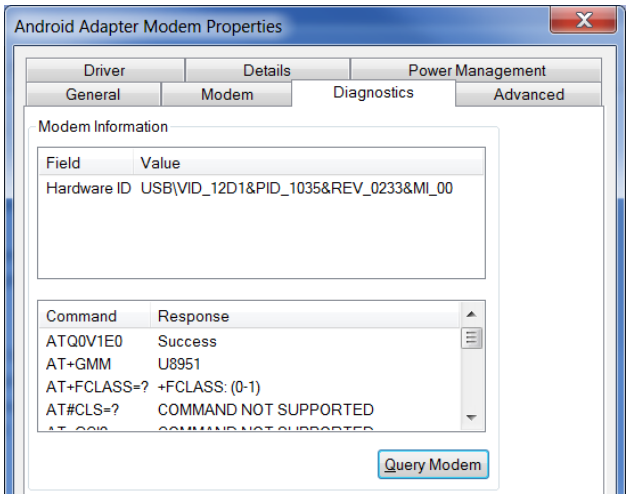

Figure 6. Android Adapter Modem Properties window

Since the Android Adapter Modem is connected on COM14, we can use a terminal application (like the standard HyperTerminal) to directly send AT commands to the phone's modem using that port as seen in Figure 7.

Figure 7. Connection to modem

The corresponding AT command used to get the Administrative Data SIM content is AT+CRSM (restricted SIM access) with the following parameters: operation type (176 means binary read), file to be accessed (0x6FAD = 28589), bytes to be read (3) [19].
The result is 00FFFF, as can be seen in Figure 8, which means:

- byte1 – 0 0 0 0 0 0 0 0
- byte2 – 1 1 1 1 1 1 1 1
- byte3 – 1 1 1 1 1 1 1 1

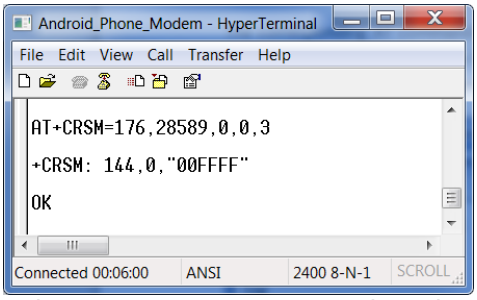

Figure 8. AT CRSM command results

Since the bit1 of 3rd byte is 1, we may presume the ciphering indicator is on, but soon after we notice the special facilities bit is off too (bit1 of 1st byte) so, in conclusion, for the VODAFONE RO SIM card we used, the ciphering indicator is disabled by the network operator default settings.
Fortunately this is not a problem because we can use again the AT commands to override the default settings. We would like to enable the bit1 of byte1 (special facilities), like the following:

- byte1 – 0 0 0 0 0 0 0 1
- byte2 – 1 1 1 1 1 1 1 1
- byte3 – 1 1 1 1 1 1 1 1

The corresponding AT command is illustrated in Figure 9. Please notice the user must have administrative rights to be allowed to alter the SIM content from a restricted area.

ISSN 2067 – 4074

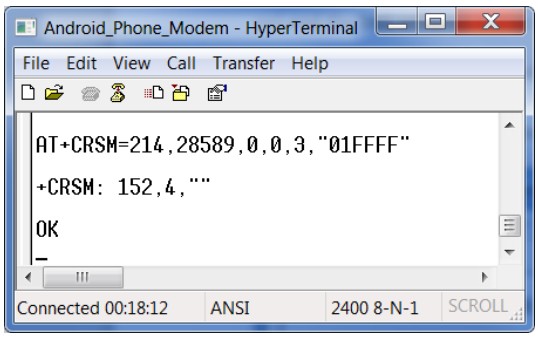

Figure 9. AT+CRSM command used to override settings

Now, since we know how to enable the ciphering indicator at the SIM card level, the next step could be to push Google (and Microsoft, etc.) to fix the issue in the next operating system versions.

8. Conclusions and future work

Turning off the GSM encryption is very possible today. In the most cases, such a situation appears without the user being informed about, simply because such a message is considered as being too confusing.

Even if the risks of directly using the AT commands are quite high because wrong inputs may wipe or brick the phone or the SIM card, enabling the ciphering indicator has the remarkable advantage of informing the user each time the communication becomes unencrypted, so he can take the proper decision about

It is not necessary the phone to be rooted to be able to use the AT commands to directly communicate with the phone modem in order to enable the ciphering indicator. Of course a rooted handset allows the direct execution of the AT commands from a dedicated application running on the phone, so there is no need for the PC connection and terminal application to discuss with the modem.

Future work includes deeper researches and the use and development of dedicated tools in order to validate the results.

Acknowledgment

Parts of this research have been published in the Proceedings of the 7th International Conference on Security for Information Technology and Communications, SECITC 2014 [17].

References

[1] Android Ciphering Issue, available at: https://code.google.com/p/android/issues/detail?id=5353

[2] The GSM Standard, available at: http://www.sans.org/reading-room/whitepapers/telephone/gsm-standard-an-overview-security-317

[3] K. Paget, Practical Cellphone Spying, available at: http://www.tombom.co.uk/blog/?p=262

[4] Forcing 3G only on Android, available at: http://siliconstation.com/how-to-force-android-only-3g/

[5] Mobile Country Code, available at: http://en.wikipedia.org/wiki/Mobile_country_code).

[6] U. Meyer, S. Wetze, On the Impact of GSM Encryption and Man-In-The-Middle Attacks on the Security Of Interoperating GSM/UMTS Networks, *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004*. (Volume: 4), pp. 2876 – 2883

[7] C. Toma, Future Developments in Non-Repudiation in GSM WAP Applications, *Journal of Mobile, Embedded and Distributed Systems*, vol. 1, no. 1, pp. 20-31, jun. 2009, available at: http://www.jmeds.eu/index.php/jmeds/article/view/Future-Developments-in-Non-Repudiation-in-GSM-WAP-Applications

[8] I. Bosoanca, A. Vargatu, An Overview of Vertical Handoff Decision Algorithms in NGWNs and a new Scheme for Providing Optimized Performance in Heterogeneous Wireless Networks, *Informatica Economică*, vol. 15, no. 1/2011, pp 5-21

[9] I. Androulidakis, D. Pylarinos, G. Kandus, Ciphering Indicator approaches and user awareness, *Maejo International Journal of Science and Technology*, 2012, 6(03), pp. 514-527

[10] Ciphering Indicator – Feature Suggestions for Windows Phone, available at: http://windowsphone.uservoice.com/forums/101801-feature-suggestions/suggestions/5825108-ciphering-indicator

[11] Centrul de presă Politia Romana – Comunicat, available at: http://www.politiaromana.ro/relatii_publice/detalii.aspx?id=16556

[12] Security consequences following the GSM encryption algorithm crack - What is the real-world risk from the cracking of the GSM encryption algorithm?, available at:

http://searchsecurity.techtarget.com/answer/Security-consequences-following-the-GSM-encryption-algorithm-crack
[13] GSM encryption code 'cracked', available at: http://www.zdnet.com/gsm-encryption-code-cracked-2062060205/
[14] Questions about the Interception of GSM Calls, available at: http://www.cryptophone.de/en/support/faq/questions-about-the-interception-of-gsm-calls/
[15] Android Security Overview - SIM Card Access, available at: http://source.android.com/devices/tech/security/#sim-card-access
[16] Hayes Command Set – GSM, available at: http://en.wikipedia.org/wiki/Hayes_command_set
[17] F. Alecu, P.Pocatilu, S. Capisizu, Interception of GSM Calls by Turning off the GSM Encryption, *Proceedings of the 7th International Conference on Security for Information Technology and Communications (SECITC'14)*, Bucharest, Romania, June 12-13, 2014, Bucharest University of Economic Studies Press, 2014, pp. 157-163
[18] Digital cellular telecommunications system (Phase 2+); Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) Interface, (3GPP TS 11.11 version 8.14.0 Release 1999, ETSI TS 100 977 V8.14.0 (2007-06), available at: http://www.etsi.org/deliver/etsi_ts/100900_100999/100977/08.14.00_60/ts_100977v081400p.pdf
[19] How to talk to the Modem with AT commands, available at http://forum.xda-developers.com/galaxy-s2/help/how-to-talk-to-modem-commands-t1471241