A Denial of Service Attack to GSM Networks via Attach Procedure

Nicola Gobbo, Alessio Merlo, Mauro Migliardi

A Denial of Service attack to GSM networks via Attach Procedure

Nicola Gobbo[1], Alessio Merlo[2], and Mauro Migliardi[1]

[1] Università degli Studi di Padova
gobbonic@dei.unipd.it, mauro.migliardi@unipd.it
[2] Università degli Studi E-Campus
alessio.merlo@uniecampus.it

Abstract. Mobile Network Operators (MNOs) keep a strict control over users accessing the networks by means of the Subscribe Identity Module (SIM). This module grants the user to access the network, by performing the registration and authentication of the user's device. Without a valid SIM module and a successful authentication, mobile devices are not granted access and, hence, they are not allowed to inject any traffic in the mobile infrastructure. Nevertheless, in this paper we describe an attack to the security of a mobile network allowing an unauthenticated malicious mobile device to inject traffic in the mobile operator's infrastructure. We show that even with devices without any SIM module it is possible to inject high levels of signaling traffic in the mobile infrastructure, causing significant service degradation up to a full-fledged Denial of Service (DoS) attack.

Keywords: Mobile Security, GSM, cellular networks security, DoS attack

1 Introduction

Mobile phones are one of the most pervasively deployed technology in the world and cellular networks have reached worldwide coverage. On one hand, the evolution from early analog networks to recent 4G LTE solutions has allowed operators to offer new services to their customers. On the other hand, the same evolution has pushed new needs into the customers; such needs have evolved from simple phone calls and SMS to internet connections and high speed access to streaming data.

The availability of smartphones with wide touch-screen displays as well as the always-on, high bandwidth IP connectivity have generated a growing set of services and applications ranging from e-mail to remote banking, from e-shopping to music streaming, from video on demand to social geo-localized networks. In turn, the ease of use and the availability of a rich a set of functionalities have instilled into users a growing familiarity and a sense of dependency. This dependency does not exist only for leisurable activities, but has a definite onset also in business and critical tasks. In particular, the last year has seen a significant

penetration in govern agencies and public bodies. To this aim, we can cite the recent security certification of Android smartphones by the US Department of Defense [29] that allows the deployment of Dell hardware with Froyo (Android OS v2.2) in the Pentagon. A second example is the adoption of tablet PCs (Apple iPad) by the Chicago hospital and the Loyola University Medical Center in Maywood. Finally, several research projects are focusing on the deployment of health-care services onto the tablet PC platform with widely goals from simple access to medical records [14], to reminders for medication intake [30], to decision support systems [22], to automatic recognition of pathological states [25], to systems for memory support [23]. For these reasons, mobile networks security analysis should emphasize availability along with confidentiality and integrity.

However, the introduction of new technologies cannot be decoupled from the support to legacy ones, since i) a high number of older terminals are still active, and ii) some manufactures keep producing 2G-only phones to satisfy low-end market. For these reasons, each new radio access technology has to be deployed alongside existing ones, leading to hybrid architectures where some network components are shared among different technological infrastructures. This condition is driving operators toward single Radio Access Network solutions, causing a cellular site to broadcast signals related to up to 3 different technologies in 5 different frequency bands. Such a composite network architecture co-exists with a design traditionally focused on making mobile networks smarter and smarter, while keeping devices crowding their cells as "dumb" as possible [18,28]. Today's smartphones are far more intelligent and powerful than their predecessors. However, networks still don't profit from their enhanced processing power; on the contrary they assume the lowest possible capability in order to maintain compatibility with older devices. This assumption results in higher signaling traffic levels between network nodes[3] and more complex system management.

The complexity of the network structure may hide both unknown and known vulnerabilities. For an interesting survey on threats undermining the world of mobile telecommunication, the reader can refer to [10]. For the case of known vulnerabilities, the true impact on the mobile phone network may have not been sufficiently assessed in a way that is similar to what happens in mobile OSes [5]. To this aim, in this paper we extend the work by Khan et al. [21] focusing on the *attach phase* of GSM protocol and we show that it is possible to mount a complete attack even without hijacking or controlling a large number of user IDs recognized by the network. To achieve our goal, we study the amount of signalling traffic that a dedicated SIM-less device can inject into an operator's core network, by pushing air interface to its design limit. Such activity may obviously disable the signalling capabilities of the cells under attack, causing a local Denial of Service (DoS) similar to the one that can be achieved with a radio jammer; however, to reach a very critical level of disruption, the generated traffic may be targeted at the Home Location Register (HLR), i.e. the database containing information on mobile subscribers. Since this database is a critical

[3] `http://connectedplanetonline.com/mss/4g-world/the-lte-signaling-challenge-0919/` (accessed in May 2013).

component of the core network, an outage of its functionality may cause an interruption of other mobile services too, finally resulting in a mobile network DoS. In our study, we leverage the HLR performance measurement conducted by Traynor et al. [27], showing that it is possible to mount an attack without any SIM module.

The remainder of this paper is structured as follow: in Section 2 we provide a description of the architecture of GSM networks; in Section 3 we analyze the state of the art in the field and we discuss the results obtained in previous related works; in section 4 we describe how it is possible to launch a DoS attack with a number of SIMless devices; finally, in section 5, we provide some concluding remarks and we describe the future direction of our study.

2 GSM network description

Global System for Mobile Communications (GSM) standard (2G) was initially designed to carry efficiently circuit switched voice communications in full duplex, with a main advantage over previous analog generation: all the processing happens in the digital domain. The standard protocol set expanded over time with addictions that, from Mobile Network Operators (MNOs) point of view, require just a software upgrade on already deployed hardware; consumers, instead, need modern and more powerful devices to experiment newly offered services. The first addiction to GSM has been General Packet Radio Service (GPRS) that introduced data delivery alongside of voice communications, in both circuit switched and —the more efficient— packet switched mode. Apart from calls GPRS permits data connection throughputs roughly ranging in the 9–170$k$bps interval; augmenting this modest numbers has been the main target of the second GSM enhancement: Enhanced Data Rates for GSM Evolution (EDGE). EDGE is a backward-compatible extension to GSM/GPRS network that introduce new coding and transmission techniques thus allowing for data rates up to 470$k$bps.

A typical GSM Public Land Mobile Network (PLMN) consists at least of the infrastructures depicted in figure 1. It is mainly split up in three different portions: i) the Mobile Station (MS) or User Equipment (UE), ii) the GSM/EDGE Radio Access Network (GERAN), iii) the Core Network (CN) or Network Switching Subsystem (NSS) with fully separated packet and circuit switched domains.

The MS may be a mobile phone or a mobile broadband modem with appropriate protocol stack and capabilities as defined by specifications. Nonetheless whichever device is used to connect to the network, there will be a Subscribe Identity Module (SIM) in it. SIMs are smart cards usually referred to as the furthest extension of mobile operator's network; it securely stores user identity, represented by the International Mobile Subscriber Identity (IMSI), and its related secret key, as long as the algorithms needed during the Authentication and Key Agreement (AKA) phase.

MSs communicate over air interface with the Base Transceiver Station (BTS). This is the first element composing the Radio Access Network (RAN), in GSM

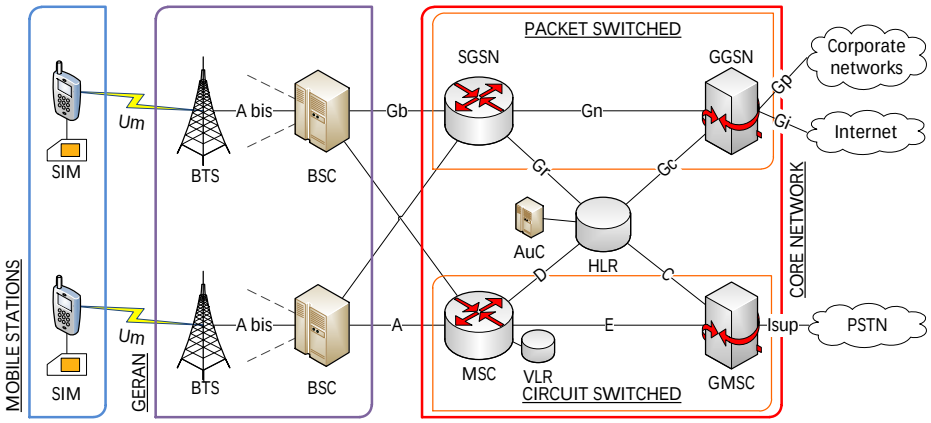

Fig. 1: Representation of the main components of a standard GPRS network.

it has minimum functionality in the sense that it just consists of a transceiver that controls the physical layer transmission. A typical GSM BTS serves three 120˚ sectors —also called cells— by means of one or more antennas per sector; antennas are powered by amplifiers that gets their pilot signals from one or more baseband modules which are finally connected to the transceiver. BTS are grouped together in tens or hundreds and connected with Base Station Controllers (BSCs), which are the devices accounting for radio resource management, MSs mobility management functions and encryption of user data prior to transmission over the air interface.

Each BSC has a couple of connections toward the core network: one linking the Serving GPRS Support Node (SGSN) carrying packet switched data, the other linking the Mobile Switching Center (MSC) and transporting circuit switched informations. This division is due to the fact that the data delivery capability of the GPRS has been a posthumous addendum to the NSS, so it has been designed for deployment in environments where GSM core networks have been already running. Both SGSN and MSC act as switching and end point for end-to-end connections in their own domains; they manage hand-overs between different BSCs as well as authentication checking and charging functions. The most valuable operation of these equipments, however, is the mobility management: they keep track of MS movements inside their service area and locate it whenever required. To carry out this operation an auxiliary database called Visitor Location Register (VLR) is used: it contains the user identity at the BSC-level along with an indication of its current location and a pointer to the main user record which is contained in another database called Home Location Register (HLR). The HLR maintains a record for each mobile phone subscriber with details like the telephone number, IMSI and a secret key (i.e. the same contained in the SIM), call blocking and forwarding and a pointer to the most updated VLR the user is known to be roaming on. HLR is a core component for the networks because it has to be queried for phone call and SMS delivery,

billing procedures and authentication: this latter function is supported by the Authentication Center (AuC) which calculates challenges and responses that are sent to the MSC/SGSN for actual user validation.

3 Related works

Cellular networks seem unaffected by the same threats that, almost daily, came up in the newspapers regarding other types of widely spread systems like the Internet. Nonetheless, even if a large security outbreak has not already made its way through the news, mobile operators' network security has been studied in the literature for quite a long time. Initially, most of the attention of researchers was focused on confidentiality and integrity [8], [7], [12], [9] of data travelling over the wireless portion of the system; however, in more recent works, the problem of the actual availability of the services provided by the network, both in the wireless segment and in the core network segment, has gained popularity, becoming the focus of different studies.

The simplest way to prevent a mobile network from offering its services is using a radio jammer. Moving from physical towards upper layers increases both the complexity of the attack and the size of the involved network segment. In order to be able to prove higher layer attacks possible, however, researchers have had to wait for a device with extensible capabilities, a kind of device that made its first market appearance in 2000 but actually had a significant deployment only in 2007: the *smartphone*[4]. Until late 1990s mobile phones had only basic phone features so the user had complete control over what the terminals were doing. This fact, however, has been subverted by the first iPhone release in 2007 and, more specifically, by the introduction of Apple App Store. The iPhone, in fact, as all the smartphones marketed today, ran an operating system over which a series of applications are executed. The advent of this application-enabled phones and centralized software distribution systems attracted the attention both of attackers[5] and of security researchers. In particular, the research community has proved that the open feature set nature of the smartphone makes it the device capable of massive and distribute mobile network attacks [13].

Past Internet security studies prove that in order to mount a DoS attack a botnet is the tool that provides the most suitable characteristics; however, mobile networks have constraints and peculiarities that should be taken into consideration both during the infection phase [16] and in the setup of the command and control mechanism [24] [11]. An attacker capable of controlling a botnet can use infected devices for multiple purposes: spam delivery, sending calls or SMS toward premium price services, spoofing user identity and remote wiretapping become all straightforward for an attacker [18,15]. A malicious entity may also try to kick mobile network elements out of service. As an example, Guo et al. [18] predicted that a few dozens of subverted smartphones, served by the same

[4] `en.wikipedia.org/wiki/Smartphone` (accessed in May 2013).

[5] `http://arstechnica.com/security/2013/04/family-of-badnews-malware-in-google-play-downloaded-up-to-9-million-times/` (accessed on May 2013).

base station, can jeopardize its availability by making no-answer calls and thus saturating provisioned voice channels. If phones are not located in the same place, authors outlined that it is still possible to put call aggregation points to a halt by means of a *distributed* denial of service: the number of needed controlled devices is indeed higher than the one needed in the previous case, but, due to the fact that PSTN, cellular switches and call centers are designed for a limited Busy Hour Call Attempts, the attack is still feasible.

Later studies still focusing on DoS attacks show that it is possible to achieve the needed level of service degradation in a more efficient way: instead of consuming traffic (or user-plane) channels, an attacker may try to flood control channels which are usually separated from traffic ones and significantly more limited in terms of available bandwidth. One of the first work in this direction is from Traynor et al. [26]. In a strict sense, the attack described here doesn't use a botnet but, in a broader sense, every mobile phone is an accomplice because what it has to do is just receiving incoming requests. They show how the interconnection between the mobile network and the Internet via, for example, on-line SMS delivery capabilities, may be exploited by an attacker continuously sending text messages to an especially crafted hit-list of telephone numbers. Such a data flood, estimated in roughly 580kbps, is enough to keep a control channel shared by voice and SMS busy, thus unavailable to accept or delivery new voice calls. Another study from Traynor et al. [28] focuses on the GPRS network and characterizes two different types of DoS attacks targeting data connection setup and tear-down mechanisms. Tear-down mechanism affects only the data portion of the network trying to keep reserved all Temporary Flow Identifiers (TFIs) that distinguish different data flows. In the setup attack, instead, authors moves the focus from resource exhaustion to control channel depletion, analysing the Random Access Channel (RACH). They find out that, for the Manhattan borough, 3Mbps of malicious traffic cause a data and voice connection blocking probability of 65% and, along with that, they point out that, this time, attacking data realm affects voice realm too, because of the single shared control channel.

A significant advancement in the analysis of mobile network security has been achieved when researchers found a way to attack core network elements, proving that network-wide service deterioration possible. Khan et al. and Kambourakis et al. [21,20] examine UMTS security architecture finding some protocols flaws that can be used to delete, modify or replay some unauthenticated or not integrity protected messages. This flaws may permit revealing user identities (IMSI), launching DoS attacks against both user phones and network nodes or impersonating the network acting as a man-in-the-middle. These studies, however, do not detail the amount of resources needed to mount a successful DoS attack. An attempt to evaluate the amount of resources needed can be found in the work by Traynor et al. [27]. The first step is a performance characterization of different HLR devices in different network deployments. The authors identify the transaction most suitable to mount an HLR DoS attack, searching for a compromise between resource consumption and execution time. By means of a simulation of the network behaviour they find that about 11750 infected devices

submitting an "insert call forwarding" every 4.7 seconds are sufficient to reduce HLR throughput of legitimate traffic by more than 93%.

Concluding this summary of works related to DoS attacks in mobile cellular networks, it is interesting to notice the "big picture" that [18] and [28] try to draw. Currently studied mobile network DoS attacks roots their cause in the fact that this networks were designed to manage traffic with highly predictable properties but, once connected to the Internet, such constraints hold no more. The Internet was designed with architectural assumptions that are in complete opposition from the ones adopted for cellular networks; this creates a disparity in the effort spent to set up and tear down a connection, necessarily leading to a bottle neck. Moreover mobile terminals have been traditionally considered dumb because of their limited battery life and computational power: this second assumption, however, holds no more in the smartphone era and its underestimation both increases network design complexity and forces core elements to early commit far more resources than those needed by an unauthenticated device. In the following sections we show how it is possible to leverage these facts to greatly reduce the amount of resources needed to mount a successful DoS attack against cellular networks.

4 Squeezing radio access protocols

When a mobile phone is switched on, GSM and UMTS protocols define what operations should be performed in order to *attach* to the network. Despite differences between the two technologies that derive from the fact that they use different radio interfaces —GSM uses TDMA while UMTS uses WCDMA— a high level description of these procedures can be described as follows: i) cell discovery, ii) best server synchronization, iii) attachment request, iv) authentication and key agreement (AKA) and v) temporary identity creation. The peculiarity of this procedure is that it cannot leverage previously accrued knowledge as it must accommodate for new devices of which there is no previous information. Moreover the design described in the introduction, i.e. the model of a smart-network and of dumb terminals, requires the whole procedure to be computationally light for the terminals and to delegate to the network most of the operations and resources. Thus, the terminals do not have to commit significant resources but the network does. These two facts are the basis of the vulnerability to DoS that is present in the attach procedure; in fact, during the AKA step, an unauthenticated device may force the core network to carry on computations that are more resource consuming than the request itself. As described by Khan et al. work [21], the way an attack could be mounted is straightforward: in a preliminary phase an attacker builds a database of valid IMSIs, then, he floods the network with *attach requests* each one carrying a different IMSI chosen from said database. The cellular network forwards the requests to HLR/AuC where each IMSI is validated and, being authentic, triggers the calculation of authentication information that are sent to SGSN that, in turn, must submit the challenge back to the mobile station and verify the reply correctness. As the attacker is not in

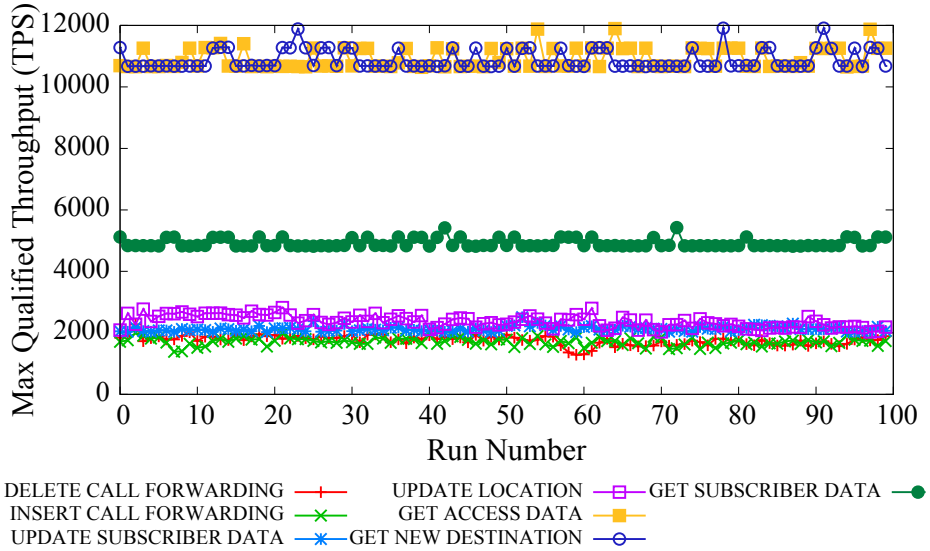

Fig. 2: HLR throughput for each transaction type with 500k subscribers [27].

possess of the SIM corresponding to the IMSI used, he doesn't know the correct answer; however, has not need for it, in fact his goal is to exhaust HLR/AuC computing resources thus he is already hitting the target with all the valid *attach requests* he is injecting. Although authors describe this attack with UMTS architecture in mind, it is important to notice that it can be performed, with minimal changes, both to old GSM [19] and new LTE [1] networks.

Khan work, however, does not provide a value for the HLR/AuC performance, thus it does not provide the number of terminals needed by an attacker in order to considerably degrade HLR services, using the attack described above. A partial analysis of this problem comes from Traynor et al. article [27]. In this work they outline an attack targeting HLR, but they adopt a different approach that leverages a botnet of authenticated devices, repeatedly injecting resource-demanding transactions available only to already attached terminals. In order to find the transaction that best suits their needs, the authors measure the average throughput —in transactions per second (TPS)— of an HLR setup, with respect to different transaction types. Their results are presented in figure 2. They choose the *insert call forwarding* procedure as the attack vector because it offers the best trade-off between computational load and execution speed. As the next step, authors simulate the effect of injecting attack traffic on an HLR already serving a typical mix of transactions: doing so they found that injecting 2500TPS the HLR capability to handle legitimate requests —under low-traffic assumptions— is reduced by 93%.

From figure 2 it is possible to determine that the *get access data* procedure is roughly 5 times faster than the *insert call forwarding* one, so, in order to achieve the same level of service degradation, we assume that the attack traffic

must be multiplied by 5. This puts our target to 12500TPS, however, for the attacker this is a worst case scenario: in fact Traynor's tests focus only on the HLR, disregarding the computations at the AuC that is needed to calculate authentication information.

4.1 Regular mobile phones are a limiting factor

To launch the attack Traynor needs a smartphone botnet for two reasons: first, clients must be authenticated before submitting an *insert call forwarding* request; second, this very kind of procedure is a standard one, so it is possible for an application to ask the underlying operating system to begin its execution. In our scenario, instead, regular phones are a limiting factor. First, from a smartphone's OS there's no way to distinguish among the steps of the GSM authentication procedure once it has been started: OSes control the modem component via a Radio Interface Layer[6] which converts high level actions such "call number" or "send SMS" into AT commands that the modem logic can understand [3]. Both high level actions and AT commands, however, are too abstract for our needs because the only way to force the attach procedure would be switching the radio off and on again. This operation is completely contained inside the GSM protocol stack and operatively hidden inside the baseband module itself, thus the module informs the OS only after the completion or failure of the entire procedure. More in details, in a mobile phone the access to the network can take only one of these three roads: 1) if the device has a valid SIM module, then the attach procedure completes unless there is a failure on the network side; 2) if the device has an invalid SIM module, then it initiate the attach procedure, but the network rejects it without needing a significant amount of resources; 3) if the device has no SIM module at all, then it does not even initiate the attach procedure. The only way to use a standard phone for performing multiple attach procedures is to equip it with a programmable SIM card and instruct the card to return a different IMSI as well as a random challenge response at each invocation. However, in this case too the solution is definitely sub-optimal because of the phone itself. Built-in mobile protocol stack is implemented strictly following 3GPP specifications which, in turn, are full of transmission wait times, exponential backoffs, maximum re-transmission trials and other artifices [2] designed with the precise purpose to induce a fair use of the network resources. As a proof of this fact Traynor highlights that, during his network behaviour measurements, he was forced to insert a 2s delay between each request: its removal, otherwise, caused extended execution times. The very goal of a DoS attack, on the contrary, is to unfairly squander the network resources in order to prevent legitimate devices to access the service; furthermore we want to reach the limits of the air interface in order to cut down the number of attacking point. For these reasons we claim that the tool best suited to an

[6] RIL specifications are available for Windows Mobile® `http://msdn.microsoft.com/en-us/library/aa920475.aspx` (accessed on May 2013) and Android `http://www.kandroid.org/online-pdk/guide/telephony.html` (accessed on May 2013).

attacker needs is a dedicated device capable of accessing the network without needing a valid SIM, and without the timing guards and the strict adherence to the protocol that are normally introduced in components aimed at the consumer market.

4.2 Analysing the Air Interface

We now analyze the peculiarities of GSM air interface protocol to evaluate its limits in terms of number of *attach requests* sent to the base station per second. In this process we suppose to be the only device communicating with the target cell; this hypothesis is unrealistic, but is a direct consequence of the unfairness of the attacking device: while legitimate mobile phones would backoff when facing a traffic problem, our device actively works toward the consumption of all the cell's resources. Thus, most of the time a mobile phone tries to get access, it won't be served because of the high number of requests injected by the attacking device, moreover, as soon as a legitimate request completes, the high number of requests injected by the attacking device generates a high probability that the just freed resources will be grabbed by the attacker and made unavailable to legitimate, well behaved devices.

GSM protocol. GSM attach procedure involves only three channels as depicted in figure 3: RACH, AGCH and SDCCH. Channels are logical entities used to carry specific traffic types; they are laid over GSM's frequency and time division multiple access (FDMA / TDMA) texture. For each carrier frequency, the fundamental building block is the TDMA frame that, in turn, is divided into 8 time slots, each during 577μs. Channel are broadcast over the air interface time-multiplexed into the multiframe structure: we focus on control-type multiframes which are dedicated to signalling and are made up of 51 frames, thus are repeating with periodicity $235.38ms$. The standard dictates the available configurations for control-type multiframes: they differ for the number of available SDCCHs and, even if combined, it is possible to have at most 12 SDCCHs. [19]

In order to evaluate the design limits of the GSM protocol, we need to analyse each channel and to find out which one introduces the maximum bottleneck. The RACH —the Random Access Channel— is the uplink channel used to carry mobile phone's access requests; in normal conditions, it is governed by the slotted ALOHA protocol, so, in order to maximize its performances, protocol developer designed RACH messages to fill just a single timeslot. We specified "normal conditions" because, in our scenario, we don't care about contention that may be caused by other devices, thus, differently from the normal scenario, we do not apply any backoff and we aim directly at the full channel consumption. In such a scenario, a 12 SDCCHs configuration provides 27 RACH access slots each multiframe and this means a capacity of:

$$\rho_{RACH} = \frac{27}{235.38ms} \approx 114.7 \text{ requests per second} \tag{1}$$

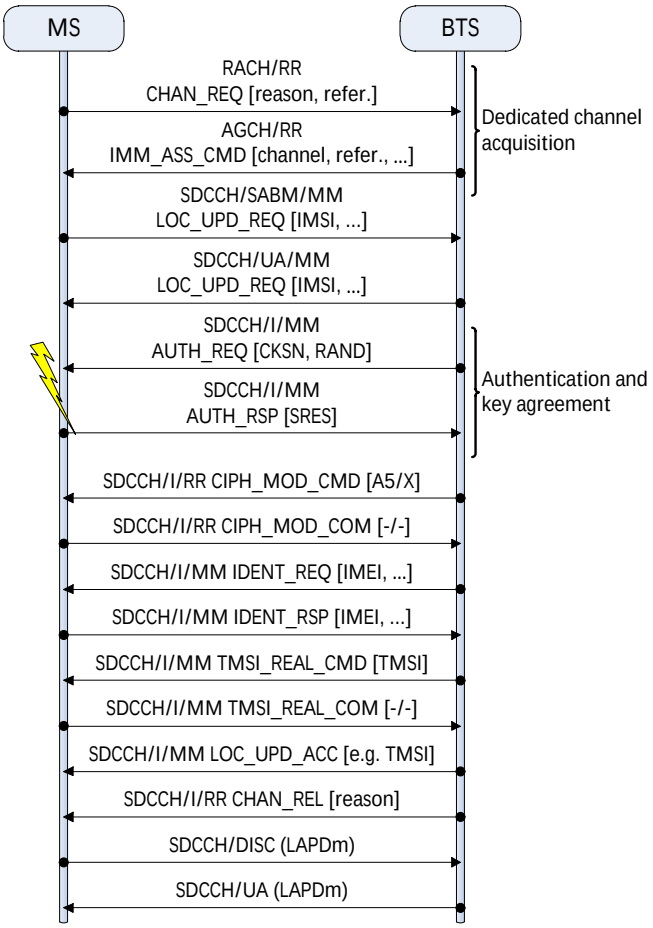

Fig. 3: Messages exchanged between MS and BTS during the GSM attach procedure [19]. The lighting on the left mark the message replaced during the attack.

This result is not fully consistent with the 80TPS calculated by [27] for the slotted ALOHA instance: authors assume a multiframe entirely dedicated to RACH slots, but this is not the case when 12 SDCCHs are deployed [19, page 99].

The Access Grant (AGCH) downlink channel is used to answer incoming random access request; it carries the information needed by the mobile phone to access the dedicated channel used for further communications. Messages over AGCH fills 4 consecutive time slots due to channel coding and interleaving; this

scheme allows the BSS to answer up to 3 RACH requests every multiframe[7]:

$$\rho_{AGCH} = \frac{3}{235.38ms} \approx 12.7 \text{ requests per second} \tag{2}$$

which represent a tighter limit than RACH.

The main part of the attach procedure is delivered via Standalone Dedicated Control Channel (SDCCH) that is an bidirectional channel assigned to a mobile terminal and is reserved to it until a special *channel release* message is issued by the BSC. As we stated above, in our scenario we assume the presence of 12 SDCCHs; determining their occupation time, however, is quite tricky. Traynor et al. [27] measured an average time of 3s to perform a complete attach where 0.5s are needed by the core network to contact HLR/AuC, calculate the authentication information and receive data back. We prove that the remaining 2.5s are spent to send messages back and forth between the mobile phone and the BTS. A multiframe can carry just one message for each SDCCH in each direction, but, when the BTS requires information to the mobile phone, the latter one can answer in the same multiframe: in fact the GSM protocol states a displacement between downlink and uplink multiframes that allows the MS to compute its reply. Given these two rules and assuming two multiframes needed for the RACH-AGCH exchange, we may conclude that completing the attach procedure requires 11 multiframes, that is $11 \times 235.38ms = 2.6$s that is almost exactly the time obtained in Traynor's measurements. Thus we say that, during message exchange between the MS and the BTS, the only wait time is related to the HLR/AuC interrogation; this, in turn, allows us to estimate SDCCH utilization time during our attack. Message exchange will be modified just from *authentication response* message on, in the way depicted in figure 4. After receiving the *authentication request* the device answers back with a LAPD_m DISC message that request BTS to terminate the multiple frame operation, releasing its Layer 2 connection [4]. We use this procedure instead of replying with a wrong `SRES` for two reasons: first, it speeds up the SDCCH release cutting the number of needed messages from 10 to 7; second, the *authentication request* message, containing the challenge, already carries the proof that the HLR/AuC has been consulted. Using the same rule, we now require 6 multiframes, 4 of which are carried over SDCCH, leading to a channel holding time of $4 \times 235.38ms + 0.5\text{s} = 1.44$s, thus a 12 SDCCHs capacity of:

$$\rho_{SDCCHs} = \frac{12}{1.44\text{s}} \approx 8.3 \text{ requests per second} \tag{3}$$

Comparing each channel capacity and choosing the lower one, we argue that GSM attacking capabilities are limited by the SDCCH channel at a rate of 8TPS. This result tell us that a GSM-only attack can be mounted with 1563 SIMless

[7] The BSS may use the extended version of the *immediate assignment* command that allow channel assignment to two mobile phones simultaneously, thus doubling AGCH capacity: we will see, however, that also in the more stringent case the AGCH is not the attack bottleneck.

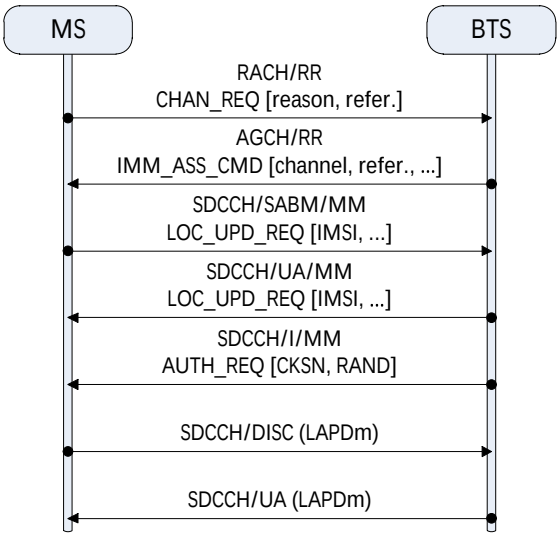

Fig. 4: Messages exchanged between MS and BTS during the attack: our device solicits an early disconnection right after receiving the `AUTH_REQ` from the network.

devices spread over the same number of cells. Furthermore, we have proved that using SIMless devices is not only possible but, compared with the number of devices required for a botnet based attack, allows reducing the amount of resources of an order of magnitude. Finally, it is important to notice that the devices enrolled in a botnet are still positioned by their rightful owners, independently from the attacker will. Thus, it is possible that an unusual clustering of users (e.g. an event in a theatre or a concert) could produce a concentration of devices that saturates the cell signalling bandwidth and prevents some of the botnets node to fulfil their full attacking potential. On the contrary, the device we envision is not owned by an unknowing user, it can be precisely placed by the attacker and even remotely triggered to start the attack. All of these factors represents a significant increase in the dangerousness of the described attack when compared with the ones described in previous works.

5 Conclusions and future works

Cellular networks are one of the infrastructure designated as critical both in the American and the European vision of the homeland security. This has lead to a large number of studies that have analysed the architecture of the networks to identify and possibly mend vulnerabilities that could be exploited to mount attacks.

Each infrastructure has been deeply analysed and many possible sweet spots for an attack have been neutralized; however, two new factors aggravate the com-

plexity in the infrastructure defence. The first of these factors is the appearance of programmable mobile phones; the second aggravating factor is, as it has been already pinpointed in previous works [5] [6] the interplay between different well known components: in this case coexisting different generations of networks. In past works several ways to mount DoS attack leveraging the programmability of modern smartphones have been described, however, these works described methodologies that needed hijacking more than 10.000 smartphones with valid SIM modules in order to mount a successful attack.

In this paper we have described a different approach, we have evaluated the possibility to bypass the strict timings enforced by the cellular network protocols by means of a dedicated radio device. This allowed us to prove that it is possible to inject into the cellular networks signalling traffic without having the control of valid SIM modules. The amount of resources that we can force the infrastructure to squander through the network of a single generation (e.g. 2G, the GSM network) is sufficient to produce a significant degradation of the service although the number of needed devices is still very high. Nonetheless, this result is very significant: first, the usage of a SIMless device allows gathering the resources needed to mount the attack without interfering with users and running the risk of being discovered; second, the usage of devices that are not in possession of unknowing users allows optimal distribution of attacking devices and removes the risk that the attack fails because of an incorrect placement of the botnet nodes. The possibility to hit a single infrastructure component through different generations of network, thus leveraging the interplay between network generations in the cellular infrastructure, would allow reducing the number of attacking devices need. In fact, combining the signalling bandwidth of GSM with the one made available by the 3G (UMTS) let each attacking device to inject more traffic into the network. The combination of this with what we presented in this paper, though, is part of a broader study [17] which has been submitted to the Journal of Ambient Intelligence and Humanized Computing.

References

1. 3GPP: TS 23.401 — General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access, http://www.3gpp.org/ftp/Specs/html-info/23401.htm
2. 3GPP: TS 25.214 — Physical layer procedures (FDD), http://www.3gpp.org/ftp/Specs/html-info/25214.htm
3. 3GPP: TS 27.007 — AT command set for User Equipment (UE), http://www.3gpp.org/ftp/Specs/html-info/27007.htm
4. 3GPP: TS 44.006 — Mobile Station - Base Stations System (MS - BSS) interface Data Link (DL) layer specification, http://www.3gpp.org/ftp/Specs/html-info/44006.htm
5. Armando, A., Merlo, A., Migliardi, M., Verderame, L.: Would you mind forking this process? A Denial of Service attack on Android (and some countermeasures). In: Information Security and Privacy Research, pp. 13–24. Springer (2012)
6. Armando, A., Merlo, A., Migliardi, M., Verderame, L.: Breaking and fixing the Android Launching Flow . Computers & Security (0), – (2013), http://www.sciencedirect.com/science/article/pii/S0167404813000540
7. Castiglione, A., Cattaneo, G., Cembalo, M., De Santis, A., Faruolo, P., Petagna, F., Ferraro Petrillo, U.: Engineering a secure mobile messaging framework. Computers & Security 31(6), 771–781 (2012)
8. Castiglione, A., Cattaneo, G., De Maio, G., Petagna, F.: SECR3T: Secure End-to-End Communication over 3G Telecommunication Networks. In: Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on. pp. 520–526 (2011)
9. Castiglione, A., Cattaneo, G., De Santis, A., Petagna, F., Ferraro Petrillo, U.: SPEECH: Secure Personal End-to-End Communication with Handheld. In: ISSE 2006, Securing Electronic Busines Processes, pp. 287–297. Vieweg (2006), http://dx.doi.org/10.1007/978-3-8348-9195-2_31
10. Castiglione, A., De Prisco, R., De Santis, A.: Do You Trust Your Phone? In: Noia, T., Buccafurri, F. (eds.) E-Commerce and Web Technologies, Lecture Notes in Computer Science, vol. 5692, pp. 50–61. Springer Berlin Heidelberg (2009), http://dx.doi.org/10.1007/978-3-642-03964-5_6
11. Castiglione, A., De Prisco, R., De Santis, A., Fiore, U., Palmieri, F.: A botnet-based command and control approach relying on swarm intelligence. Journal of Network and Computer Applications (0), – (2013), http://dx.doi.org/10.1016/j.jnca.2013.05.002
12. De Santis, A., Castiglione, A., Cattaneo, G., Cembalo, M., Petagna, F., Ferraro Petrillo, U.: An Extensible Framework for Efficient Secure SMS. 2010 International Conference on Complex, Intelligent and Software Intensive Systems 0, 843–850 (2010)
13. Derr, K.: Nightmares with mobile devices are just around the corner! In: Portable Information Devices, 2007. PORTABLE07. IEEE International Conference on. pp. 1–5 (2007)
14. Doukas, C., Pliakas, T., Maglogiannis, I.: Mobile healthcare information management utilizing cloud computing and android os. In: Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE. pp. 1037–1040. IEEE (2010)
15. Felt, A.P., Finifter, M., Chin, E., Hanna, S., Wagner, D.: A survey of mobile malware in the wild. In: Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices. pp. 3–14. ACM (2011)

16. Fleizach, C., Liljenstam, M., Johansson, P., Voelker, G.M., Mehes, A.: Can you infect me now?: malware propagation in mobile phone networks. In: Proceedings of the 2007 ACM workshop on Recurring malcode. pp. 61–68. ACM (2007)
17. Gobbo, N., Merlo, A., Migliardi, M.: Attacking the attach procedure in cellular networks. . Journal of Ambient Intelligence and Humanized Computing (0), – (2014)
18. Guo, C., Wang, H.J., Zhu, W.: Smart-phone attacks and defenses. In: HotNets III (2004)
19. Heine, G., Horrer, M.: GSM networks: protocols, terminology, and implementation. Artech House, Inc. (1999)
20. Kambourakis, G., Kolias, C., Gritzalis, S., Hyuk-Park, J.: Signaling-oriented dos attacks in umts networks. In: Advances in Information Security and Assurance, pp. 280–289. Springer (2009)
21. Khan, M., Ahmed, A., Cheema, A.R.: Vulnerabilities of umts access domain security architecture. In: Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008. SNPD'08. Ninth ACIS International Conference on. pp. 350–355. IEEE (2008)
22. Kuntagod, N., Mukherjee, C.: Mobile decision support system for outreach health worker. In: e-Health Networking Applications and Services (Healthcom), 2011 13th IEEE International Conference on. pp. 56–59. IEEE (2011)
23. Migliardi, M., Gaudina, M.: Memory Support through Pervasive and Mobile Systems, in Inter-Cooperative Collective Intelligence: Techniques and Applications. In: Studies in Computational Intelligence. Springer (2013)
24. Mulliner, C., Seifert, J.P.: Rise of the iBots: Owning a telco network. In: Malicious and Unwanted Software (MALWARE), 2010 5th International Conference on. pp. 71–80. IEEE (2010)
25. Tacconi, C., Mellone, S., Chiari, L.: Smartphone-based applications for investigating falls and mobility. In: Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on. pp. 258–261. IEEE (2011)
26. Traynor, P., Enck, W., McDaniel, P., La Porta, T.: Mitigating attacks on open functionality in sms-capable cellular networks. In: Proceedings of the 12th annual international conference on Mobile computing and networking. pp. 182–193. ACM (2006)
27. Traynor, P., Lin, M., Ongtang, M., Rao, V., Jaeger, T., McDaniel, P., La Porta, T.: On cellular botnets: measuring the impact of malicious devices on a cellular network core. In: Proceedings of the 16th ACM conference on Computer and communications security. pp. 223–234. ACM (2009)
28. Traynor, P., McDaniel, P., La Porta, T., et al.: On attack causality in internet-connected cellular networks. In: Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium. pp. 1–16. USENIX Association (2007)
29. U.S. Department of Defense: Security Technical implementation Guide, `http://iase.disa.mil/stigs/net_perimeter/wireless/smartphone.html`
30. Wang, M.Y., Zao, J.K., Tsai, P., Liu, J.: Wedjat: a mobile phone based medicine intake reminder and monitor. In: Bioinformatics and BioEngineering, 2009. BIBE'09. Ninth IEEE International Conference on. pp. 423–430. IEEE (2009)