A Silent SMS Denial of Service (DoS) Attack

N.J Croft and M.S Olivier
Information and Computer Security Architectures (ICSA) Research Group
Department of Computer Science
University of Pretoria
Pretoria
South Africa
Email: ringtingting@gmail.com

***Abstract*— Global System for Mobile communications (GSM) is a popular mobile communications network. Short Message Service (SMS) is an easily adopted person-to-person communications technology for mobile devices. The GSM architecture allows for the insertion of mass application-generated SMS messages directly into the network infrastructure. This is achieved through a SMS Mobile Switching Centre (SMSC) using a variety of request-response protocols, for example Short Message Peer-To-Peer Protocol (SMPP).**

Through protocol manipulation, an application may generate an SMS which neither displays on the mobile handset nor provides an acoustic signal. Known as a "Silent" SMS, this occurs where the mobile handset must acknowledge receipt of the short message but may discard its contents. A "Silent" SMS may help police services detect the existence of a mobile handset without the intended party knowing about the request. In contrast, a mass continuous send of "Silent" SMS messages will constitute an invisible Denial of Service (DoS) attack on a mobile handset. Such a mobile handset DoS attack may be conducted for economic advantage to elude another party from communicating.

This paper describes, from a technical perspective, how a silent application-generated denial of service (DoS) SMS attack is conducted. We then investigate possible ways of thwarting such an attack at a GSM network level. Furthermore we explore related SMS attacks on the GSM network.

I. Introduction

The Global System for Mobile communications (GSM) is a popular digital circuit switched network [1]. GSM is a common telecommunications standard originally issued by the European Telecommunications Standards Institute (ETSI) [2]. GSM is an open standard which is currently developed by the 3rd Generation Partnership Project (3GPP) [3]. GSM provides recommendations, not requirements. The GSM specifications define the functions and interface requirements in detail but do not address the hardware. The reason for this is to limit the designers as little as possible but still to make it possible for the operators to buy equipment from different suppliers.

The name GSM first comes from a group called Group Special Mobile (GSM), which was formed in 1982 by the European Conference of Post and Telecommunications Administrations (CEPT) to develop a pan-European cellular system that would replace the many existing incompatible cellular systems already in place in Europe. However, when GSM service started in 1991, the abbreviation "GSM" was renamed to Global System for Mobile Communications from Group Special Mobile.

GSM has become an international cooperation and collaboration between people, companies and governments, creating a truly global wireless communication network. At the time of writing, GSM service had surpassed the 2 billion people mark and is currently available across more than 214 countries and territories worldwide [4].

The initial Short Message Service (SMS) standard was first discussed in the early 1980s but the world's first commercial SMS service was not introduced until 1992. SMS was created as part of Phase I of the GSM standard. SMS is widely adopted with approximately 1 billion SMS messages sent every day worldwide [5].

The SMS message, as specified by the ETSI organization in documents GSM 03.40 [6] and GSM 03.38 [7], can be up to 160 characters long, where each character is 7 bits according to the 7-bit default alphabet. Eight-bit messages (max 140 characters) are usually not viewable by the phones as text messages; instead they are used for data in e.g. smart messaging (images and ringing tones) and Over The Air (OTA) provisioning of Wireless Application Protocol (WAP) settings (to be discussed in more detail later). 16-bit messages (max 70 characters) are used for Unicode (UCS2) text messages, viewable by most phones. A 16-bit text message will on some phones appear as a Flash SMS (aka blinking SMS or alert SMS).

The Short Message Peer-To-Peer Protocol (SMPP) [8], [9] is a telecommunications industry protocol for exchanging SMS messages between SMS peer entities such as short message service centres (SMSCs). It is often used to allow third parties to submit messages at an application level, often in bulk.

Silent messages, often referred to as "Silent SMS" or "Stealth SMS" is indicated neither on the display nor by an acoustic signal. GSM 03.40 [6] describes a short message of type 0 which indicates that the ME must acknowledge receipt of the short message but may discard its contents. Such an SMS is useful, in particular, for the police services to detect the existence of a mobile handset without the intended party knowing about the request. However, a "Silent" SMS may be used for more sinister reasons.

Traditionally a denial-of-service (DoS) attack is an attempt to make a computer resource unavailable to its intended users. One such method is to flood a network, thereby preventing legitimate network traffic. Typically the targets are high-profile web servers, and the attack attempts to make the hosted web

pages unavailable on the Internet. Such an attack is extendable to any mobile environment. A mobile device is rendered ineffective should a mobile device be flooded with this type of SMS messages. Furthermore, should a "Silent" SMS DoS attack takes place on the handset, the intended victim would be oblivious to the attack. The only visible symptom would be an abnormal decline in battery charge capacity and the inability to receive calls etc. This ineffectiveness of the handset is due to SMS messages making use of the signalling layer, also used in performing other network events.

Not only will a "Silent" SMS consume battery power but it will clog the signalling channel. This may be the reasoning behind the motivation in performing a "Silent" DoS attack. Primarily it may be done for economic advantage to elude another party to communicate, or may be used to ensure that a given party is not notified of some events. As another example, consider an Intrusion Detection System (IDS) that informs a network administrator via mobile phone if an attack occurs. By launching a DoS attack on the mobile phone, the network intrusion may occur for much longer without the knowledge of the network administrator.

In this paper we explore the technical detail in executing a mobile application-generated Denial of Service (Dos) SMS Attack. We investigate possible ways of thwarting such an attack at a network level. Furthermore we explore related attacks using SMPP and SMS on a GSM network.

This paper is structured as follows: Section II covers a brief overview of GSM. Section III covers in detail the composition of an SMS message. This includes both 7-bit and 8-bit messaging. Section IV investigates the various application generated SMS protocols with a specific focus on the industry leader, SMPP. Section V illustrates the technical aspects of sending a "Silent" 7-bit and 8-bit SMS using SMPP. Section VI explores the possibilities of thwarting "Silent" SMS DoS attacks. Section VII shows possible spin off attacks aimed in particular at the SMSC itself. Finally Section VIII concludes this paper.

Our decision to position our work in the GSM context is based on the popularity of GSM and its wide spread adoption in comparison to other mobile communication networks such as Universal Mobile Telecommunications System (UMTS). In addition, work reported on in this paper forms part of a larger privacy and security project [10]–[14] set in the GSM and next generation wireless communication context.

II. Background

The Global System for Mobile Communications (GSM) [15] [1] is a common standard issued by the European Telecommunications Standards Institute (ETSI). The most basic service supported by GSM is telephony; however GSM also allows data to be transported (both synchronous and asynchronous) as a bearer service [10]. The GSM standard is considered to be a "second generation" or 2G cellular system and was designed to be secure, have strong subscriber authentication and Over The Air (OTA) transmission encryption [10]. In order to understand the origin and SMS application in GSM, the respective underlying architecture needs to be understood.

A. GSM Architecture

The GSM system has two major components: the fixed installed infrastructure (network) and the Mobile Station (MS) [16]. Mobile users make use of the serving GSM network's services by communicating over a radio interface. Figure 1 illustrates the GSM architecture.

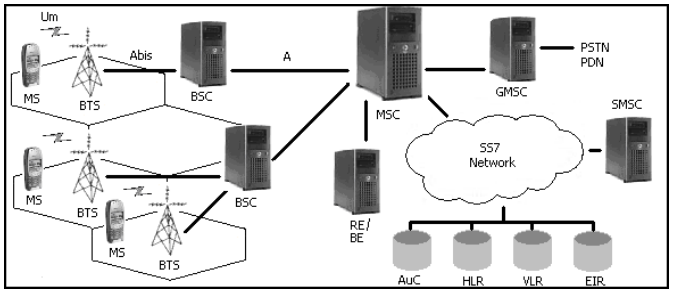

Fig. 1. GSM Architecture (adapted from [10])

The Mobile Station (MS) is the mobile phone or GSM compliant device. The MS provides access to the GSM network. The MS consists of Mobile Equipment (ME) and a Subscriber Identity Module (SIM) [17]. The Base Transceiver Station (BTS) is a radio tower or pico (single) cell with which the Mobile Station communicates. The Base Station Controller (BSC) acts as a common node between multiple BTSs and the network's backbone. The Mobile Switching Centre (MSC) performs the switching functions of the network. The MSC has an interface to one or more BSCs and to external networks. Signalling between functional entities in the network system uses the Signalling System Number 7 [18]. Several databases are available for control and network management. The following are usually considered to be part of the MSC: i) Home Location Register (HLR) - contains permanent (user's profile) and temporary (location information) data for all registered users with a network operator, ii) Visitor Location Register (VLR) - is responsible for a group of location areas and stores the data of those users who are currently in its area of responsibility, iii) Authentication Centre (AuC) - provides for authentication of an MS on the network and encryption of communication transmissions, iv) Equipment Identity Register (EIR) - registers equipment data.

The Short Message Service (SMS) is a store and forward service, in other words, short messages are not sent directly from sender to recipient, but always via an SMS Centre (SMSC) instead. Each mobile telephone network that supports SMS has one or more messaging centers to handle and manage the SMS messages. The service center is responsible for the collection, storage, and delivery of short messages, and is outside the scope of GSM [1]. Thus the provider of the SMS service does not necessarily have to be the serving GSM operator. However, a default SMSC number is usually provided by the network operator and this number is stored at the Mobile Station (MS).

III. Short Message Service (SMS)

Short Message Service (SMS), is a universal text messaging system, allowing the transmission of messages up to 160

alphanumeric characters to be sent to or from a GSM Mobile Station (MS). SMS is characterized by an out-of-band packet delivery and low-bandwidth message transfer, which results in a highly efficient means for transmitting short bursts of data. Message delivery forms part of the GSM infrastructure where every SMS has to pass via a Short Message Service Centre (SMSC).

The benefit of an SMS to a user centers around convenience, flexibility and the seamless integration of a complete messaging solution. SMS works on a store-and-forward basis and when received, is usually stored on the SIM card or on the MS's internal store. An SMS is transferred in a connectionless packet mode over the signalling channel of the serving GSM network. Once a message is sent, it is received by a SMSC (refer to Figure 1), which must then get it to the appropriate recipient mobile device via the MSC.

An SMS comprises of the following elements, of which only the User Data (the message) and originating address (mobile number) is displayed on the recipient's mobile device: i) Header - identifies the type of message, ii) Service Center TimeStamp, iii) Originating Address - mobile number of the sender, iv) Protocol Identifier, v) Data Coding Scheme, vi) User Data Length - the length of the message, vii) User Data - the message (***140 bytes***: 160 7-bit characters, 140 8-bit characters or 70 16-bit characters).

SMS messages travel between several network nodes before being delivered. The sender of a Mobile Terminating (MT) message is charged for the sending of the SMS. Usually the charge for receiving an SMS is zero.

We now describe the process flow when an SMS message is sent from one sender MS (handset) to a recipient MS [14].

1) The SMS message is submitted from the sender MS to the SMSC
2) After the message is processed at the SMSC, it sends a request to the HLR and receives routing information for the recipient MS
3) The SMSC sends the SMS to the MSC
4) The MSC retrieves the recipient's information from the VLR. This may include an authentication operation between the MSC and VLR
5) The MSC forwards the message to the recipient MS
6) If delivered successfully, the SMS is stored on the recipient MS's SIM card under USER-DATA
7) The MSC returns to the SMSC the outcome of the SMS delivery status
8) If requested by the sending MS, the SMSC reports delivery status of the SMS back to the sender

As described above, messages are sent to a Short Message Service Centre (SMSC) which provides a store-and-forward mechanism. It is a "best effort" attempt on the networks side to send messages to the intended recipients. If a recipient is not reachable, the SMSC queues the message for later retry. The re-try process rules differ per SMSC. There may be a fixed re-try upper bound transmission count or time-elapsed constraint, after which the message is discarded. Some SMSCs merely provide a "forward and forget" option where transmission is tried only once. Both Mobile Terminated (MT), for messages sent to a mobile handset, and Mobile Originating (MO), for those that are sent from the mobile handset, operations are supported. As message delivery is best effort, there is no guarantee that a message will actually be delivered to its recipient and delay or complete loss of a message is not uncommon, particularly when sending between networks. Users may choose to request delivery reports, which can provide positive confirmation that the message has reached the intended recipient, but notifications for failed deliveries are unreliable at best.

Transmission of the short messages between SMSC and phone can be done through different protocols such as SS7 within the standard GSM framework or TCP/IP within the same standard. Messages, whose payload length is limited by the constraints of the signalling protocol to precisely 140 bytes (140 bytes = 140 * 8 bits = 1120 bits). In practice, this translates to either 160 7-bit characters, 140 8-bit characters, or 70 16-bit characters. Characters in languages such as Arabic, Chinese, Korean, Japanese or Slavic languages (e.g. Russian) must be encoded using the 16-bit UCS-2 character encoding.

When a mobile terminated message is class 0 and the MS has the capability of displaying short messages, the MS shall display the message immediately and send an acknowledgement to the SMSC when the message has successfully reached the MS [7]. This effectively means that if the mobile is incapable of displaying a message it may simply be ignored and discarded by the handset. The SMSC will still, however, receive a successful delivery receipt acknowledgement. This forms the basis of a possible Denial of Service (DoS) attack whereby the mobile handset is flooded with messages it simply wont display and discard. In order to flood a MS (handset) with "Silent" SMS messages, a mechanism is required to auto-generate mass messages which can be sent to the Mobile Station continuously. We now investigate bulk SMS messaging protocols which allow for the mass sending of SMS messages providing the platform for a silent DoS mobile handset attack.

IV. Application-Generated SMS Protocols

There are numerous protocols for the generation of application originating SMS messages. Industry protocols include UCP/EMI, CIMD [19] and SMPP [8] amongst others. These protocols provide third parties the capability of submitting SMS messages, often in bulk and at reduced costs.

Computer Interface to Message Distribution (CIMD) is a proprietary SMSC protocol developed by Nokia for their Artus SMSC. The External Machine Interface (EMI), an extension to Universal Computer Protocol (UCP), was developed by LogicaCMG [20], the current SMSC market leader. The Short Message Peer-To-Peer protocol (SMPP) is the most common of industry protocols for exchanging SMS messages between SMS peer entities such as SMSC. SMPP was originally designed by Aldiscon, a small Irish company that was later bought by Logica, now LogicaCMG [20]. In 1999, SMPP was formally handed over to the SMPP Developers Forum, which was later renamed as The SMS Forum [21].

The most commonly used versions of SMPP are v3.3 and v3.4 [8] where the latter adds transceiver support functionality (single connections that can send and receive messages).

Data exchange may be synchronous, where each peer must wait for a response for each PDU (protocol data units, or packets) being sent, and asynchronous, where receiving and transmitting execute independently with the use of buffers and timers. The latest version of SMPP is v5.0 [9]. The protocol is based on pairs of request/response PDUs exchanged over OSI layer 4 (TCP session) connections. PDUs are binary encoded for efficiency.

Using the SMPP protocol, an SMS application system called the "External Short Message Entity" (ESME) may initiate an application layer connection with an SMSC over a TCP/IP or X.25 network connection and may then send short messages and receive short messages to and from the SMSC respectively [8]. Every SMPP operation must consist of a request PDU and associated response PDU. The receiving entity must return the associated SMPP response to an SMPP PDU request. As an example, a *Submit_Sm* PDU (refer to Table I) is used to send an SMS message which expects a *Submit_SM_Resp* from an SMSC or bulk messaging gateway.

Field Name	Size Octets	Type	Description
command_length	4	Integer	Set to overall length of PDU
command_id	4	Integer	submit_sm (0x00000004)
command_status	4	Integer	Not used
sequence_number	4	Integer	Unique sequence number
service_type	max 6	COctet String	NULL for default SMSC settings
source_addr_ton	1	Integer	Type of Number for source
source_addr_npi	1	Integer	Numbering Plan Indicator for source
source_addr	max 21	COctet String	Originating address
dest_addr_ton	1	Integer	Type of Number for destination
dest_addr_npi	1	Integer	Numbering Plan Indicator for destination
dest_addr	max 21	COctet String	Destination address
esm_class	1	Integer	Indicates Message Mode & Message Type
protocol_id	1	Integer	Protocol Identifier (refer to [6])
priority_flag	1	Integer	Priority level of the message
schedule_delivery_time	1 or 17	COctet String	NULL for immediate message delivery
validity_period	1 or 17	COctet String	NULL for SMSC default validity period
registered_delivery	1	Integer	SMSC delivery receipt
replace_if_present_flag	1	Integer	Replace existing message
registered_delivery	1	Integer	SMSC delivery receipt
data_coding	1	Integer	Encoding scheme used (refer to [7])
sm_default_msg_id	1	Integer	NULL for default SMSC msg_id
registered_delivery	1	Integer	SMSC delivery receipt
sm_length	1	Integer	Length in octets of *short_message*
short_message	Var. 0-254	Octet String	Up to 254 octets of short message user data

TABLE I
SMPP SUBMIT_SM PDU

The *command_length, command_id, command_status and sequence_number* all form part of the PDU header while the remaining field names constitutes the message body. The *data_coding* field indicates the data coding scheme and is used to usually indicate if the SMS message is 7-bit, 8-bit or 16-bit encoded. This field will play an important role during the attack to be described later. Further details are available in the relevant specification document [8].

An example of a 7-bit SMPP Submit_Sm PDU is provided for in Listing 1. SMS messages are binary encoded according to the 7-bit GSM Default Alphabet table, found in GSM 03.38 [7]. Listing 1 is GSM 03.38 encoded and represented in Hexadecimal (Hex) as it is easier to read hexadecimal numbers rather than a binary representation.

Listing 1. Example of SMPP Submit_Sm PDU Encoding

```
Encoding PDU Header..
'command_length', (71) ... 00 00 00 47
'command_id', (4) ... 00 00 00 04
'command_status', (0) ... 00 00 00 00
'sequence_number', (1) ... 00 00 00 01

Encoding PDU Body..
'service_type', (0) ... 30 00
'source_addr_ton', (1) ... 01 **
'source_addr_npi', (1) ... 01 **
'source_addr', (27829239812) ... 32 37 38 32 39 32 33 39 38 31 32 00
'dest_addr_ton', (1) ... 01 **
'dest_addr_npi', (1) ... 01 **
'dest_addr', (27829239812) ... 32 37 38 32 39 32 33 39 38 31 32 00
'esm_class', (0) ... 00
'protocol_id', (0) ... 00
'priority_flag', (0) ... 00
'schedule_delivery_time', (0) ... 30 00
'validity_period', (0) ... 30 00
'registered_delivery', (1) ... 01
'replace_if_present_flag', (0) ... 00
'data_coding', (0) ... 00
'sm_default_msg_id', (0) ... 00
'sm_length', (0) ... 00
'short_message', (satnac.org.za) ... 73 61 74 6E 61 63 2E 6F 72 67 2E 7A 61

Full PDU (71 octets++).. 00 00 00 47 00 00 00 04 00 00 00 00 00 00 00
00 01 30 00 01 01 32 37 38 32 39 32 33 39 38 31 32 00 01 01 32 37 38
32 39 32 33 39 38 31 32 00 00 00 00 30 00 30 00 01 00 00 00 00 73 61
74 6E 61 63 2E 6F 72 67 2E 7A 61

** (0) indicates local numeric numbering formatting
   (1) indicates international numeric number formatting

++ Octet is a group of 8 bits, often referred to as a byte
```

Short messages can also be used to send binary content such as ringtones, logos, or WAP Push messages as well as Over The Air (OTA) programming or configuration data. A WAP Push message provides a direct link to an Internet web reference (URL) via an SMS message. An OTA message is usually used to send handset specific settings via an SMS message. Such SMS messages are sometimes vendor-specific extensions of the GSM specification. A WAP push is a binary SMS message consisting of a header, a URL and a message. WAP push messages are 8-bit encoded messages and are therefore limited to 140 octets. It is important to note that not all phones on the world market support WAP pushes and may be discarded by the handset upon receipt.

Listing 2 illustrates a 8-bit WAP push template represented in Hexadecimal (Hex).

Listing 2. WAP Push template

```
HEADER
<STRING>
<SI>
<INDICATION>
<PROTOCOL INDICATOR/>
<STRING>
URL
</STRING>
</INDICATION>
<STRING>
MESSAGE
</STRING>
</INDICATION>
</SI>

Now replace the WAP Push Tags as follows:
HEADER ... 06 05 04 0B 84 23 F0 DC 06 01 AE 02 05 6A 00
<STRING> ... 03
</STRING> ... 00
<SI> ... 45
<INDICATION> ... C6
<PROTOCOL INDICATOR> ... 0C for http://, 0D for http://www.
</INDICATION> ... 01
</SI> ... 01
URL ... 73 61 74 6E 61 63 2E 6F 72 67 2E 7A 61 (satnac.org.za)

Full PDU (40 octets++).. **
0605040B8423F0DC0601AE02056A0045C60C037361746E61632E6F72672E7A610001036869000101

** place FULL PDU in short_message field of Submit_Sm PDU
++ Octet is a group of 8 bits, often referred to as a byte
```

If an application encodes GSM User Data Header Information (UDHI) in the *short_message* user data, it must set the UDHI flag in the *esm_class* field [8]. In other words, we must indicate that this message is a 8-bit binary message (WAP Push SMS). This is achieved in SMPP by setting the *esm_class* field to 64 (0x40). Likewise, the encoding of the message must be set to represent 8-bit binary encoding. This is achieved by setting the *data_coding* field to 4 (0x04) (refer to [7], [8]).

V. Sending a "Silent" SMS

This section considers strategies to launch a silent SMS attack. A successful attack strategy will be one that i) sends an SMS to an MS without displaying the SMS on the MS and ii) is useable on as many SMPP gateways as possible.

In order to find possible attack strategies the SMS specifications were scrutinized and bulk SMS providers questioned regarding the sending of a "Silent" SMS. Once possible strategies have been found, it would be necessary to test a "Silent" SMS in principle. Furthermore, to identify the number of bulk SMS providers providing "Silent" SMS capabilities.

We have currently found two known ways to send a "Silent" SMS. There are however, countless ways to malform an SMS PDU which may cause the handset to malfunction or SMSC to crash.

A. Manipulating the Data Coding Scheme

Using GSM 03.38 [7], we set the *data_coding* to 192 (0xC0) (11000000). This sets the Message Waiting Indication Group identifier, which translates to "Discard Message". With bits 7..4 set to 1100, the mobile may discard the contents of the message [7]. Figure 2 shows a "Silent" SMS example by calling an exposed web service method at an SMPP supported SMS gateway.

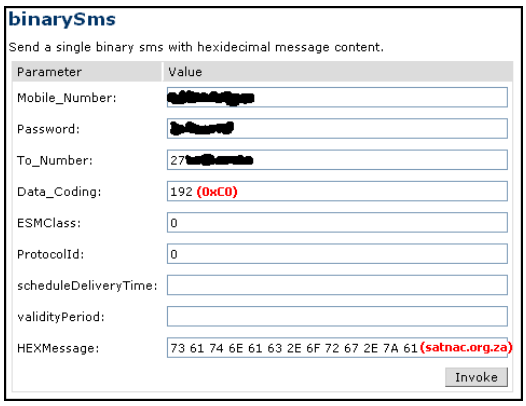

Fig. 2. Web Service "Silent" SMS Call using [22]

Figure 3 shows the PDU dump of a "Silent" SMS and the delivery receipt notification received. The SMS message status is ***DELIVRD***, however the message never displays on the mobile handset.

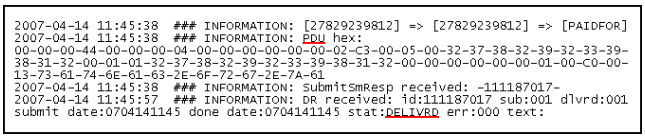

Fig. 3. PDU dump of "Silent" SMS from [22]

The "Silent"SMS defined by manipulating the data coding scheme worked on **every** SMPP enabled gateway we tested (six different SMPP gateways in total). SMPP gateways are stringently built according GSM standards [8]. This effectively guarantees the sending of a "Silent" SMS via manipulating the Data Coding Scheme through any SMPP gateway.

B. Manipulating Timing in a WAP Push Message

Another example of a "Silent" SMS is to manipulate the *scheduled_delivery_time* or *validity_period* when sending a WAP push SMS. However, this approach was not transparent through all gateways we tested and warrants investigation.

By setting the *scheduled_delivery_time* or *validity_period* before today's date, we were able to achieve similar results. The format of the *scheduled_delivery_time* and *validity_period* is in the form (YYMMDDhhmmsstnn). Figure 4 illustrates a WAP Push "Silent" SMS example by calling an exposed web service method again at an SMPP supported SMS gateway.

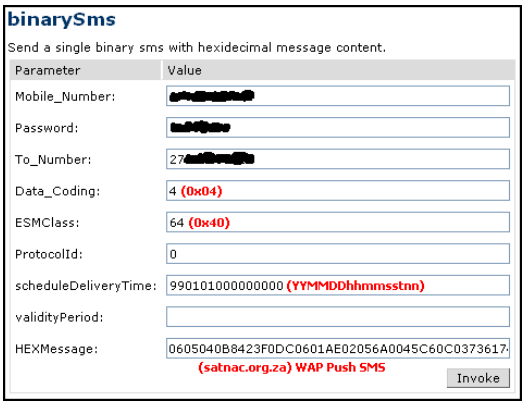

Fig. 4. Web Service "Silent" SMS Call (WAP Push) using [22]

Figure 5 shows the PDU dump of a WAP Push "Silent" SMS and the delivery receipt notification received. The SMS message status is ***DELIVRD***, however the message never displays on the mobile handset.

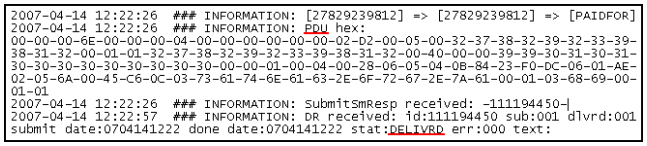

Fig. 5. PDU dump of WAP Push "Silent" SMS from [22]

C. Cost of the attack

Now that we have established how to send both 7-bit and 8-bit messages using SMPP, how much would it cost to send a message? Purchasing SMS messages in large prepaid volume bundles, from messaging providers or network operators, results in substantial cost reductions. A few years ago, some operators were not charging for the sending of bulk SMS messages. Although rare, some network operators still provide for this free messaging service. Message costs may further be reduced if the application is assigned a fixed-number sender identifier. This effectively means that a message will always arrive on a handset from the application with a predetermined number. A base cost of 0.01 Euro cents is not unobtainable per message. Then for example the cost associated in sending one "Silent" SMS every second for one hour will amount to Euro 36 (1 x 60 x 60 x 0.01 = Euro 36). This shows that a "Silent" SMS DoS attack is economically feasible.

VI. Thwarting "Silent" DoS SMS Attacks

We have proved a "Silent" SMS attack is indeed a reality and cheap enough to the detriment of all mobile GSM subscribers. The most obvious solution to this form of attack, or any bulk SMS attack, would simply be to check each message and discard it based on a predetermined set of criteria. One such criteria is to simply discard any identical messages (same content and recipient). With the current volumes [5] of SMS messages sent worldwide, a real-time check on each message is impracticable and will no doubt have an adverse effect on network performance.

The market leader in terms of SMS Fraud Management Systems (FMS) is WhiteCell [23]. Such a FMS counters threats by providing the mobile network operator with an additional security layer on top of its existing SMS infrastructure. This security program identifies potential threats, and prevents unwanted traffic from passing through the network. Although such technologies do exist, implementing this across GSM worldwide is a near impossible task given the global complexity of the GSM infrastructure. Added to this, there is the network profit motive. This insight to network operators thinking can be summed up in the following comment: "as long as messages are being sent, someone is being charged for it!".

Thwarting a bulk SMS attack may prove difficult, identifying and preventing a "Silent" SMS attack seems even less likely.

VII. SMS Related Attacks

A. SMSC Attacks

The first attacks is aimed specifically at the SMSC. The SMSC is flooded with malformed SMS PDUs or SMS PDUs are manipulated to never exit the SMSC re-try queue. This may effect the stability of the SMSC, overloading it and eventually causing it to crash.

B. Charged for Receiving SPAM

In the United States the GSM billing model includes billing the subscriber who ***receives*** an SMS message. Through the sending of mass SMS messages, the subscriber is charged for unsolicited messages (SPAM). This effectively means that subscribers are legitimately charged for receiving SPAM. In the case of a "Silent" SMS attack, the end user will be charged for messages that were never even displayed on the handset.

VIII. Conclusion

In this paper we described, from a technical perspective, how a silent application-generated denial of service (DoS) SMS attack is conducted. We illustrated through real-world examples how such an attack is conducted. We began by providing a detailed description of SMS messages, how they are composed and sent. We made use of the SMPP protocol for the sending of "Silent" SMS messages which may be used in performing a Denial of Service (DoS) attack on a mobile handset. We then investigated possible ways of thwarting such an attack at a GSM network level. Furthermore we explored related SMS attacks on an SMSC and billing of a subscriber within the GSM network.

References

[1] M. Rahnema, "Overview of the gsm system and protocol architecture," *IEEE Communications Magazine*, vol. 31, no. 4, pp. 92–100, April 1993.

[2] *Recommendation GSM 02.09; Security related network functions*, European telecommunications Standard Institute, ETSI, June 1993, tech. Rep.

[3] 3rd Generation Partnership Project, "3gpp," Web Reference: http://www.3gpp.org, accessed October 2006.

[4] GSM Association, "homepage," Web Reference: http://www.gsmworld.com/index.shtml, accessed April 2006.

[5] ——, "SMS (Short Message Service)," Web reference, http://www.gsmworld.com/yechnology/sms. Accessed May 2005.

[6] *Digital cellular telecommunications system (Phase 2+); Technical realization of the Short Message Service (SMS); Point to Point (PP)(GSM 03.40 version 6.0.0)*, European telecommunications Standard Institute, ETSI, March 1998.

[7] *Digital cellular telecommunications system (Phase 2+); Alphabets and language-specific information (GSM 03.38 version 7.0.0 Release 1998)*, European telecommunications Standard Institute, ETSI, July 1998.

[8] *Short Message Peer to Peer Protocol Specification v3.4*, The SMS Forum, October 1999.

[9] *Short Message Peer to Peer Protocol Specification v5.0*, The SMS Forum, February 2003.

[10] N. Croft, "Secure Interoperations of Wireless Technologies," Masters Dissertation, University of Pretoria, School of Computer Science, October 2003.

[11] N. Croft and M. Olivier, "Using compatible keys in achieving subscriber privacy channelling for billing in GSM Networks," in *Proceedings of the Fifth International Network Conference*, S. Furnell, P. Dowland, and G. Kormentzas, Eds., 2005, pp. 245–252.

[12] ——, "Using a Trusted Third Party Proxy in achieving GSM Anonymity," in *South African Telecommunication Network and Applications Conference*. SATNAC, September 2004.

[13] ——, "Codec-Hopping: Secure and Private Voice Communication in Bandwidth Constrained Networks," in *SecPerU, Workshop on Security an Privacy in Pervasive Ubiquitous Computing*, Santorini, Greece, April 2005.

[14] ——, "Using an approximated one-time pad for securing Short Message Service (SMS)," in *South African Telecommunication Network and Applications Conference*. SATNAC, September 2005.

[15] M. Mouly and M. Pautet, *The GSM System for Mobile Communications*. Telecom Publishing, 1992, foreword By-Thomas Haug.

[16] E. T. . 929, "Digital cellular telecommunications system (Phase 2); Security related network functions," European Telecommunications Standards Institute, November 1999.

[17] *European digital cellular telecommunications system (Phase 2); Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface (GSM 11.11)*, European Telecommunications Standards Institute, Sophia Antipolis, France, 1998.

[18] Y. B. Lin, "Signaling System Number 7," *IEEE Potentials*, pp. p. 5–8, August 1996.

[19] *CIMD Interface Specification, Nokia SMS Center 7.0*, Nokia, December 2004.

[20] LogicaCMG, Web reference, http://www.logicacmg.com. Accessed March 2007.

[21] The SMS Forum, Web Reference: http://www.smsforum.net, accessed April 2007.

[22] SMS BUG COMMUNICATIONS, Web reference, https://www.smsbug.com/api/webservice.asmx. Accessed April 2007.

[23] WhiteCell, Web reference,http://www.white-cell.com. Accessed April 2007.

Neil Croft Neil Croft is a final year PhD Computer Science student at the University of Pretoria. His research interests include security and privacy in current and next generation wireless communication networks. He completed his Masters degree at the University of Pretoria in October 2003 and undergraduate studies at the Rand Afrikaans University in 2001. He currently operates an SMS company called SMS BUG COMMUNICATIONS (www.smsbug.com).