Journal of Computer Security 16 (2008) 713–742
DOI 10.3233/JCS-2007-0308
IOS Press

Exploiting open functionality in SMS-capable cellular networks

Patrick Traynor *, William Enck, Patrick McDaniel and Thomas La Porta
Systems and Internet Infrastructure Security Laboratory, Department of Computer Science and Engineering, The Pennsylvania State University, University Park, PA 16802, USA
E-mail: {traynor, enck, mcdaniel, tlp}@cse.psu.edu

Cellular networks are a critical component of the economic and social infrastructures in which we live. In addition to voice services, these networks deliver alphanumeric text messages to the vast majority of wireless subscribers. To encourage the expansion of this new service, telecommunications companies offer connections between their networks and the Internet. The ramifications of such connections, however, have not been fully recognized. In this paper, we evaluate the security impact of the SMS interface on the availability of the cellular phone network. Specifically, we describe the ability to deny voice service to cities the size of Washington DC and Manhattan with little more than a cable modem. Moreover, attacks targeting the entire United States are feasible with resources available to medium-sized zombie networks. This analysis begins with an exploration of the structure of cellular networks. We then characterize network behavior and explore a number of reconnaissance techniques aimed at effectively targeting attacks on these systems. We conclude by discussing countermeasures that mitigate or eliminate the threats introduced by these attacks.

Keywords: Telecommunications, SMS, denial-of-service, open-functionality

1. Introduction

The majority of mobile phone subscribers are able to receive both voice and alphanumeric text via *Short Messaging Service* (SMS) transmissions. Text messaging allows users to interact with each other in situations where voice calls are not appropriate or possible. With countries such as the US experiencing volumes of five billion messages per month [32], this service is rapidly becoming as ingrained into modern culture as its voice counterpart [9,11].

Text messaging services are extremely popular with the telecommunications industry. Whereas voice traffic typically yields a fixed amount of revenue per user, service providers earn up to US$ 0.10 per text message sent or received by a mobile device [15,40,56]. Seeing this tremendous potential for revenue, cellular providers have opened their networks to a number of additional services designed to increase

*Corresponding author: P. Traynor, 344 IST/CSE Building, University Park, PA 16802, USA. Tel.: 814 865 6245; Fax: 814 865 3176; E-mail: traynor@cse.psu.edu.

SMS messaging volume. Through service provider website interfaces, email, and a wide variety of applications including instant messaging, users across the Internet can contact mobile subscribers without the use of a cell phone. Such *open functionality*, however, has serious negative consequences for these networks.

This paper evaluates the security impact of Internet-originated text messages on cellular voice and SMS services. The connections between the Internet and phone networks introduce open functionality that detrimentally affects the fidelity of a cellular provider's service. Through the generation and use of large, highly accurate phone hit-lists, we describe the ability to *deny voice service* to cities the size of Washington DC and Manhattan with little more than a cable modem. Moreover, attacks targeting the entire United States are feasible with resources available to medium-sized zombie networks. Even with small hit-lists, we show that these cyberwarfare attacks are sustainable for tens of minutes. These attacks are especially threatening when compared to traditional signal jamming in that they can be invoked from anywhere in the world, often without physical involvement of the adversary.

There are many dangers of connecting digital and physical domains. For example, a wide array of systems with varying degrees of connectivity to the Internet were indirectly affected by the Slammer worm. The traffic generated by this worm was enough to render systems including Bank of America's ATMs and emergency 911 services in Bellevue, Washington unresponsive [35].

There is nothing fundamentally different about the ways in which these victimized systems and cellular networks are connected to the Internet; all of the above systems were at one time both logically and physically isolated from external networks, but have now attached themselves to the largest open system on the planet. Accordingly, we show that mobile phone networks are equally as vulnerable to the influence of the Internet.

In evaluating Internet-originated SMS attacks on cellular networks, we make the following contributions:

- **System characterization:** Through analysis of publicly available cellular standards and gray-box testing, we characterize the resilience of cellular networks to elevated messaging loads.
- **Refining target search space:** We discuss a variety of techniques that, when used in combination, result in an accurate database of targets ("hit-lists") for directed attacks on cellular networks. These lists are absolutely essential to mounting effective attacks against these networks.
- **SMS/cellular network vulnerability analysis:** We illuminate the fragility of cellular phone networks in the presence of even low-bandwidth attacks. We describe and quantify the ability to incapacitate voice and SMS service to neighborhoods, major metropolitan areas and entire continents.
- **Examination of technical progress:** After more than a year since the initial publication of the vulnerability, we examine both short and long term responses.

The remainder of this paper is organized as follows: Section 2 gives a high-level overview of GSM network architecture and describes text message delivery; Section 3 investigates cellular networks from an attacker's perspective and identifies the mechanisms necessary to launch *Denial of Service* (DoS) attacks; Section 4 models and quantifies DoS attacks in multiple environments; Section 5 proposes various solutions to help alleviate these problems; Section 6 examines public response to the original paper and provides a glimpse into the current state of affairs; Section 7 discusses important related works; Section 8 presents concluding remarks.

2. SMS/cellular network overview

This section offers a simplified view of an SMS message traversing a GSM-based system from submission to delivery. These procedures are similar in other cellular networks including CDMA.

2.1. Submitting a message

There are two methods of sending a text message to a mobile device – via another mobile device or through a variety of *External Short Messaging Entities* (ESMEs). ESMEs include a large number of diverse devices and interfaces ranging from email and web-based messaging portals at service provider websites to voice mail services, paging systems and software applications. Whether these systems connect to the mobile phone network via the Internet or specific dedicated channels, messages are first delivered to a server that handles SMS traffic known as the *Short Messaging Service Center* (SMSC). A service provider supporting text messaging must have at least one SMSC in their network. Due to the rising popularity of this service, however, it is becoming increasingly common for service providers to support multiple SMSCs in order to increase capacity.

Upon receiving a message, the contents of incoming packets are examined and, if necessary, converted and copied into SMS message format. At this point in the system, messages from the Internet become indistinguishable from those that originated from mobile phones. Messages are then placed into an SMSC queue for forwarding.

2.2. Routing a message

The SMSC needs to determine how to route messages to their targeted mobile devices. The SMSC queries a *Home Location Register* (HLR) database, which serves as the permanent repository of user data and includes subscriber information (e.g. call waiting and text messaging), billing data, availability of the targeted user and their current location. Through interaction with other network elements, the HLR determines the routing information for the destination device. If the SMSC receives a reply stating that the current user is unavailable, it stores the text message for later

delivery. Otherwise, the response will contain the address of the *Mobile Switching Center* (MSC) currently providing service. In addition to call routing, MSCs are responsible for facilitating mobile device authentication, location management for attached *base stations* (BS), performing handoffs and acting as gateways to the *Public Switched Telephone Network* (PSTN).

When a text message arrives from the SMSC, the MSC fetches information specific to the target device. The MSC queries a database known as the *Visitor Location Register*, which returns a local copy of the targeted device's information when it is away from its HLR. The MSC then forwards the text message on to the appropriate base station for transmission over the air interface. A diagram of a mobile phone network is depicted in Fig. 1(a), followed by a simplified SMS message flow in Fig. 1(b).

2.3. Wireless delivery

The air interface is divided into two parts – the *Control Channels* (CCH) and *Traffic Channels* (TCH). The CCH is further divided into two types of channels – the Common CCH and Dedicated CCHs. The Common CCH, which consists of logical channels including the *Paging Channel* (PCH) and *Random Access Channel* (RACH), is the mechanism used by the base station to initiate the delivery of voice and SMS data. Accordingly, all connected mobile devices are constantly listening to the Common CCH for voice and SMS signaling.

The base station sends a message on the PCH containing the *Temporary Mobile Subscriber ID* (TMSI) associated with the end destination. The network uses the TMSI instead of the targeted device's phone number in order to thwart eavesdroppers attempting to determine the identity of the receiving phone. When a device hears its TMSI, it attempts to contact the base station over the RACH and alerts the network of its availability to receive incoming call or text data.[1] When the response arrives, the base station instructs the targeted device to listen to a specific *Standalone Dedicated Control Channel* (SDCCH). Using the SDCCH, the base station is able to facilitate authentication of the destination device (via the subscriber information at the MSC), enable encryption, deliver a fresh TMSI and then deliver the SMS message itself. In order to reduce overhead, if multiple SMS messages exist on the SMSC, more than one message may be transmitted over an SDCCH session [5]. If a voice call had been waiting at the base station instead of a text message, all of the above channels would have been used in the same manner to establish a connection on a traffic channel.

An illustration of this final stage of delivery over the air interface is shown in Fig. 2.

[1] A high number of call initiations at a given base station slows this response as the RACH is a shared access channel running the Slotted Aloha protocol.

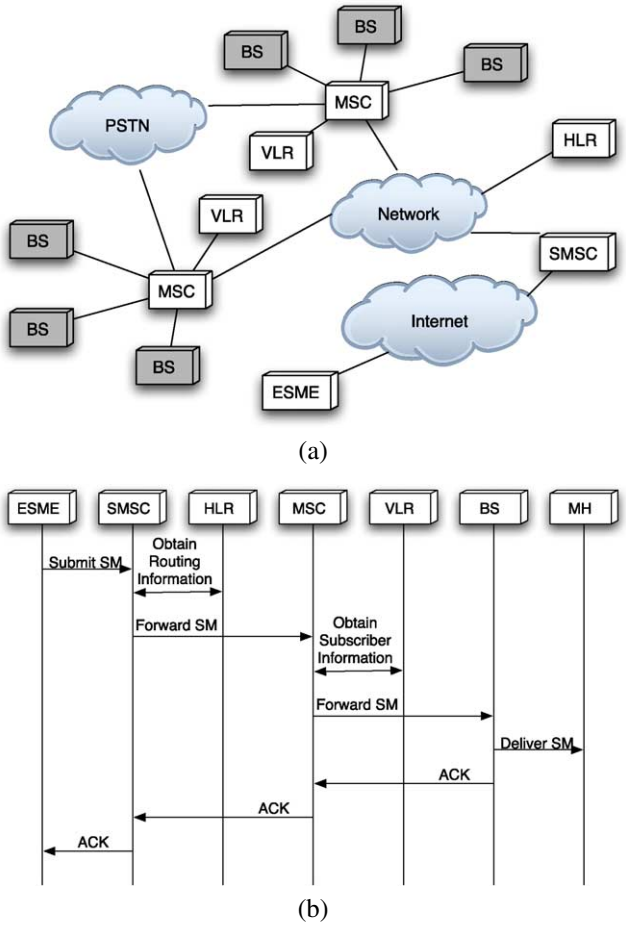

Fig. 1. Simplified examples of an SMS Network and message flow: (a) SMS Network; (b) SMS Flow.

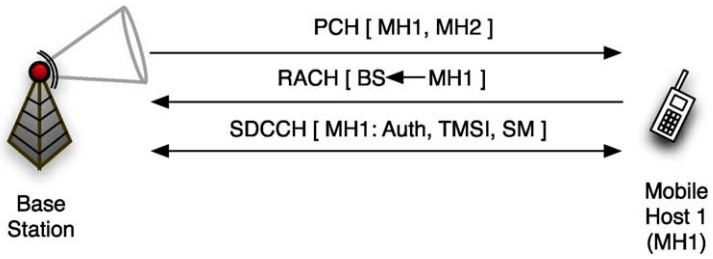

Fig. 2. A simplified SMS air interface communication. The base station notifies two mobile hosts (MH1 and MH2) of new messages. MH1 hears its identifier and responds. After authenticating and establishing an encrypted channel, the text message is delivered over a dedicated control channel.

3. SMS/cellular network vulnerability analysis

The majority of legitimate uses for SMS can often be characterized as nonessential, ranging from social interactions to low priority business-related exchanges. The salient feature of these communications is that they can typically be accomplished through a number of other, albeit potentially less convenient channels. During the terrorist attacks of September 11, 2001, however, the nature of text messaging proved to be far more utilitarian.

With millions of people attempting to contact friends and family, telecommunications companies witnessed tremendous spikes in cellular voice service usage. Verizon Wireless, for example, reported voice traffic rate increases of up to 100% above typical levels; Cingular Wireless recorded an increase of up to 1000% on calls destined for the Washington DC area [39]. While these networks are engineered to handle elevated amounts of traffic, the sheer number of calls was far greater than capacity for voice communications in the affected areas. However, with voice-based phone services being almost entirely unavailable due to TCH saturation, SMS messages were still successfully received in even the most congested regions because the control channels responsible for their delivery remained available.

Text messaging allowed the lines of communication to remain open for many individuals in need in spite of their inability to complete voice calls. Accordingly, SMS messaging is now viewed by many as a reliable method of communication when all other means appear unavailable.

Due to this proliferation of text messaging, we analyze Internet-originated, SMS attacks and their effects on voice and other services in cellular networks. We first characterize these systems through an extensive study of the available standards documentation and gray-box testing. From this data, we discuss a number of attacks and the susceptibility of mobile phone networks to each. Lastly, from gray-box testing, we assess the resilience of these networks to these attacks.

Before discussing the specifics of any attack on cellular networks, it is necessary to examine these systems from an adversary's perspective. In this section, we present simple methods of discovering the most fragile portions of these networks by determining system bottlenecks. We then investigate the creation of effective targeting systems designed to exploit these choke points.

3.1. Determining bottlenecks in cellular networks

There is an inherent cost imbalance between injecting SMS messages into the phone network and delivering messages to a mobile user. Such imbalances are the root of DoS attacks.

Recognizing these bottlenecks requires a thorough understanding of the system. The cellular network standards documentation provides the framework from which the system is built, but it lacks implementation specific details. In an effort to bridge this gap, we performed gray-box testing [6,12].

We characterize these systems by delivery disciplines, delivery rates, and interfaces. All tests were performed using our own phones. At no time did we inject a damaging volume of packets into the system or violate any service agreement.

3.1.1. Delivery discipline

The delivery discipline of a network dictates the way messages move through the system. By studying this flow, we determine system response to an influx of text messages. The overall system response is a composite of multiple queuing points. The standards documentation indicates two points of interest – the SMSC and the target device.

SMSCs are the locus of SMS message flow; all messages pass through them. Due to practical limitations, each SMSC only queues a finite number of messages per user. As SMSCs route messages according to a store and forward mechanism, each message is held until either the target device successfully receives it or it is dropped due to age. The buffer capacity and eviction policy therefore determine which messages reach the recipient.

The SMSC buffer and eviction policy were evaluated by slowly injecting messages while the target device was powered off. Three of the most prominent service providers were evaluated: AT&T (now part of Cingular), Verizon, and Sprint. For each provider, 400 messages were serially injected at a rate of approximately one per 60 seconds. When the device was reconnected to the network, the range of the attached sequence numbers indicated both buffer size and queue eviction policy.

We found that AT&T's SMSC buffered the entire 400 messages. While seemingly large, 400 160-byte messages is only 62.5 kB. Tests of Verizon's SMSC yielded different results. When the device was turned on, the first message downloaded was not sequence number one; instead the first 300 messages were missing. This demonstrates that Verizon's SMSC has a buffer capacity of 100 messages and a FIFO eviction policy. Sprint's SMSC proved different than both AT&T and Verizon. Upon reconnecting the device to the network, we found only 30 messages starting with message number one. Therefore, Sprint's SMSC has a message capacity of 30 messages and a LIFO eviction policy.

Messages also remain in the SMSC buffer when the target device's message buffer is full. This occurs, as noted in the GSM standards [5], when the mobile phone returns a *Mobile-Station-Memory-Capacity-Exceeded-Flag* to the HLR. Because it is impossible to determine the inbox capacity of every phone, we chose to test three representative devices of varying age and expense: the Nokia 3560 (AT&T), the slightly newer LG 4400 (Verizon), and the recently released high-end Treo 650 (Sprint) containing a 1 GB removable memory stick. Mobile device capacity was observed by slowly sending messages to the target phone until a warning indicating a full inbox was displayed. The resulting device buffer capacities varied as shown in Table 1.

The delivery discipline experimentation results indicate how the SMS system will react to an influx of text messages. We confirmed that finite buffer capacities exist in most SMSCs and mobile devices. In the event of a DoS attack, messages exceeding these saturation levels will be lost. Therefore, a successful DoS attack must be distributed over a number of subscribers.

Table 1
Mobile device SMS capacity

Device	Capacity (number of messages)
Nokia 3560	30
LG 4400	50
Treo 650	500*

*500 messages depleted a full battery.

3.1.2. Delivery rate

The speed at which a collection of nodes can process and forward a message is the delivery rate. In particular, bottlenecks are discovered by comparing injection rates with delivery rates. Additionally, due to variations in injection size for different interfaces, the injection size per message is estimated.

Determining the maximum injection rate for a cellular network is an extremely difficult task. The exact number of SMSCs in a network is not publicly known or discoverable. Given the sheer number of entrances into these networks, including but not limited to website interfaces, email, instant messaging, and dedicated connections running the *Short Messaging Peer Protocol* (SMPP), we conservatively estimate that it is currently possible to submit between several hundred and several thousand messages per second into a network from the Internet using simple interfaces.

A brief sampling of available interfaces is provided in Table 2. These interfaces can be grouped into three main categories: instant messaging, information services, and bulk SMS. Instant messaging provides the same functionality as text messaging, but connects new networks of users to cellular networks. With 24 hour news, customers are frequently flooded with "on the go" updates of headlines, sports, and stocks from information service providers such as CNN and MSNBC. Lastly, through bulk SMS providers, companies can provide employees with updates ranging from server status to general office notifications.

While injection rates for instant messaging and the information services are unknown, the bulk SMS providers offer plans with rates as high as 30–35 messages per second, per SMPP connection. Furthermore, by using multiple SMPP connections, START Corp. (`www.startcorp.com`) offers rates "an order of magnitude" greater. Combining all of these conduits provides an adversary with the ability to inject an immense number of messages.

When message delivery time exceeds that of message submission, a system is subject to DoS attacks. We therefore compare the time it takes for serially injected messages to be submitted and then delivered to the targeted mobile device. This was accomplished via a PERL script designed to serially inject messages approximately once per second into each provider's web interface. From this, we recorded an average send time of 0.71 seconds.

Measurement of incoming messages was more difficult due to a lack low-level access to the device operating system. Via informal observation, we recorded interarrival times of 7–8 seconds for both Verizon and AT&T. Interarrival times for

Table 2
A brief sampling of SMS access services

Service	URL
Instant Messaging	
AOL IM	mymobile.aol.com/portal/index.html
ICQ	www.icq.com/sms/
MSN Messenger	mobile.msn.com
Yahoo Messenger	messenger.yahoo.com/messenger/wireless/
Information Services	
CNN	www.cnn.com/togo/
Google	sms.google.com
MSNBC	net.msnbc.com/tools/alert/sub.aspx
Bulk SMS	
Clickatell	www.clickatell.com
SimpleWire	www.simplewire.com/services/smpp/
START Corp.	www.startcorp.com/StartcorpX/ Mobile_Developer.aspx

Sprint were undetermined due to sporadic message downloads occurring anywhere between a few seconds and few minutes apart. The experiments clearly demonstrate an imbalance between the time to submit and the time to receive.

While SMS messages have a maximum size of 160 bytes, each submission requires additional overhead. Using `tcpdump`, we observed both raw IP and user data traffic. Not considering TCP/IP data overhead, Sprint, AT&T, and Verizon all required under 700 bytes to send a 160 byte SMS message. This included the HTTP POST and browser headers.

Due to the ACKs required for downloading the web page (8.5 kB for Sprint, 13.6 kB for AT&T, 36.4 kB for Verizon), the actual data upload size was significantly higher. While the overhead is relative to retransmissions and window size, we recorded upload sizes of 1300 bytes (Sprint), 1100 bytes (AT&T), and 1600 bytes (Verizon). In an effort to reduce the overhead induced by TCP traffic, we observed the traffic resulting from email submission. Even with TCP/IP traffic overhead, less than 900 bytes was required to send a message. For the purposes of the following analysis, we conservatively estimate 1500 bytes (a standard MTU size) as the required data size to transmit an SMS message over the Internet.

3.1.3. Interfaces

Lost messages and negatively acknowledged submit attempts were observed. We expect this was due to web interface limitations imposed by the service providers. It is therefore important to determine both the mechanisms used to achieve rate limitation on these interfaces and the conditions necessary to activate them.

A group of 50 messages was submitted serially at a rate of approximately one per second. This was followed by a manual send via the web interface in order to check

for a negative acknowledgment. If an upper bound was not found, the number of sequential messages was increased, and the test was repeated.

During the injection experiments performed for rate analysis, we encountered interface limitations.[2] After 44 messages were sent in a serial fashion through Verizon's web interface, negative acknowledgments resulted. Further investigation revealed that blocking was subnet based.

Message blocking was also observed for the AT&T phone. Even though the web interface blindly acknowledges all submissions, we observed message loss after 50 messages were sent to a single phone. This time, further investigation revealed that even messages originating from a separate subnet were affected. Seeing an opportunity to evaluate policy at the SMSC, we sent a text message from the Verizon phone. The message was received, therefore, AT&T's SMSC must differentiate between its inputs.

While both Verizon and AT&T use IP based limitations, Sprint deployed an additional obstacle. In order to submit a message through the web interface, a session cookie[3] value was required. While circumventing this prevention scheme was accomplished through automated session ID retrieval, further analysis showed it had no effects on rate limitation.

Due to the above determined SMSC buffer capacity of 30 messages and the sporadic download times, approximately 30 messages can be injected before loss occurs.

In summary, through gray-box testing, we found SMSCs typically hold far more messages than the mobile devices. While high end multifunction platforms hold over 500 messages, common phones only hold 30 to 50 messages. When the target device cannot receive new messages, continued injection from the Internet results in queuing at the SMSC. Therefore, to launch a successful DoS attack that exploits the limitations of the cellular air interface (discussed in Section 4), an adversary must target multiple end devices. To accomplish this, effective reconnaissance must occur.

3.2. Hit-list creation

The ability to launch a successful assault on a mobile phone network requires the attacker to do more than simply attempt to send text messages to every possible phone number. Much like the creation of hit-lists for accelerated worm propagation across the Internet [48], it is possible to efficiently create a database of potential targets within a cellular phone network. The techniques below, listed from the most coarse to fine-grain methods, are only a subset of techniques for creating directed attacks; however, the combination of these methods can be used to create extremely accurate hit-lists.

The most obvious first step would be simply to attempt to capture phone numbers overheard on the air interface. Because of the use of TMSIs over the air interface, this approach is not possible. We therefore look to the web as our source of data.

[2] Presumably for mitigating cell phone spam.

[3] The session cookie is referred to as a "JSESSIONID" at this particular website.

3.2.1. NPA/NXX

The United States, Canada and 18 other nations throughout the Caribbean adhere to the *North American Numbering Plan* (NANP) for telephone number formatting. NANP phone numbers consist of ten digits, which are traditionally represented as "NPA-NXX-XXXX".[4] These digit groupings represent the area code or *Numbering Plan Area*, exchange code,[5] and terminal number, respectively. Traditionally, all of the terminal numbers for a given NPA/NXX prefix are administered by a single service provider.

A quick search of the Internet yields a number of websites with access to the NPA/NXX database. Responses to queries include the name of the service provider administering that NPA/NXX domain, the city where that domain is located and the subdivision of NPA/NXX domains among a number of providers. For example, in the greater State College, PA region, 814-876-XXXX is owned by AT&T Wireless; 814-404-XXXX is managed by Verizon Wireless; 814-769-XXXX is supervised by Sprint PCS.

This information is useful to an attacker as it reduces the size of the domain to strictly numbers administered by wireless providers within a given region; however, this data does not give specific information in regards to which of the terminals within the NPA/NXX have been activated. Furthermore, as of November 23, 2004, this method does not account for numbers within a specific NPA/NXX domain that have been transferred to another carrier under new number portability laws. Nonetheless, this approach is extremely powerful when used in conjunction with other methods, as it reduces the amount of address space needed to be probed.

3.2.2. Web scraping

As observed in the Internet [42], a large number of messages sent to so-called "dark address space" is a strong indicator that an attack is in progress. A more refined use of domain data, however, is readily available.

Web Scraping is a technique commonly used by spammers to collect information on potential targets. Through the use of search engines and scripting tools, these individuals are able to gather email addresses posted on web pages in an efficient, automated fashion. These same search tools can easily be harnessed to collect mobile phone numbers listed across the web. For example, the query `Cell 999-999-0000..9999` at Google (*www.google.com*) yields a large number of hits for the entire range of the NPA/NXX "999-999-XXXX". Through our own proof-of concept scripts, we were able to collect 865 unique numbers from the greater State College, PA region, 7,308 from New York City and 6,184 from Washington DC with minimal time and effort.

The difficulty with this method, much like the first, is that it does not give a definitive listing of numbers that are active and those that are not. As personal web pages

[4]Numbers in the last two subsets can take the form of N(2-9) or X(0-9).

[5]The "NXX" portion of a phone number is sometimes referred to as the "NPX" or *Numbering Plan Exchange*.

are frequently neglected, the available information is not necessarily up to date. Accordingly, some portion of these numbers could have long since been returned to the pool of dark addresses. Furthermore, due to number porting, there is no guarantee that these numbers are still assigned to the service provider originally administering that domain. Regardless, this approach significantly narrows down the search space of potential targets.

3.2.3. Web interface interaction

All of the major providers of wireless service in the United States offer a website interface through which anyone can, at no charge to the sender, submit SMS messages. If a message created through this interface is addressed to a subscriber of this particular provider, the message is sent to the targeted mobile device and a positive acknowledgment is delivered to the sender. A message is rejected from the system and the user, depending on the provider, is returned an error message if the targeted device is a subscriber of a different provider or is addressed to a user that has opted to turn off text messaging services. An example of the both the positive and negative acknowledgments is available in Fig. 3. Of the service providers tested (AT&T Wireless, Cingular, Nextel, Sprint PCS, T-Mobile and Verizon Wireless[6]), only AT&T did not respond with a positive or negative acknowledgment; however, it should be noted that subscribers of AT&T Wireless are slowly being transitioned over to Cingular due to its recent acquisition.

The positive and negative acknowledgments can be used to create an extremely accurate hit-list for a given NPA/NXX domain. Every positive response generated by the system identifies a potential future target. Negative responses can be interpreted in multiple ways. For example, if the number corresponding to a negative response was found through web scraping, it may instead be tried again at another provider's website. If further searching demonstrates a number as being unassigned, it can be removed from the list of potential future targets.

While an automated, high speed version of this method of hit-list creation may be noticed for repeated access to dark address space, an infrequent querying of these interfaces over a long period of time (i.e. a "low and slow" attack) would be virtually undetectable.

A parallel result could instead be accomplished by means of an automated dialing system; however, the simplicity of code writing and the ability to match a phone to a specific provider makes a web-interface the optimal candidate for building hit-lists in this fashion.

3.2.4. Additional collection methods

A number of specific techniques can also be applied to hit-list development. For example, a worm could be designed to collect stored phone numbers from victim

[6]Since the original publication of this work, the above providers have changed. AT&T Wireless was absorbed by Cingular, which recently changed its name back to AT&T. Sprint and Nextel have also merged into a single entity.

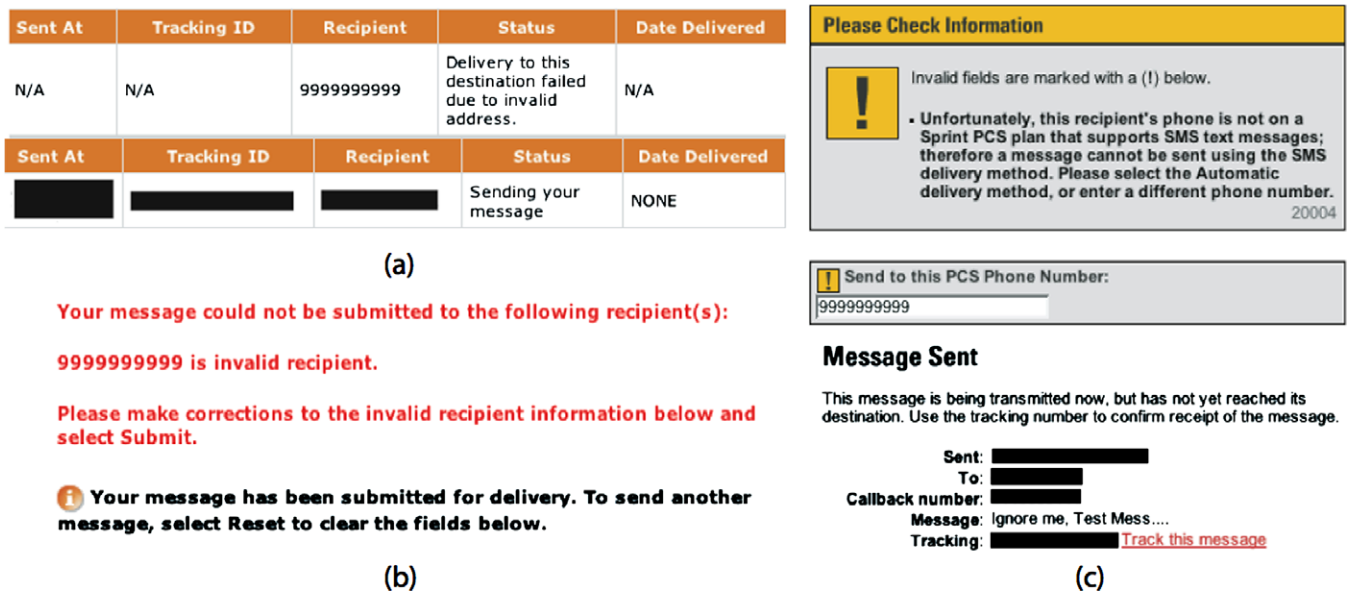

Fig. 3. The negative (top) and positive (bottom) response messages created by message submission to (a) Verizon, (b) Cingular and (c) Sprint PCS. Black rectangles have been added to preserve sensitive data.

devices by address book scraping. In order to increase the likelihood that a list contained only valid numbers, the worm could instead be programmed to take only the numbers from the "Recently Called" list. The effectiveness of his method would be limited to mobile devices running specific operating systems. The interaction between many mobile devices and desktop computers could also be exploited. An Internet worm designed to scrape the contents of a synchronized address book and then post that data to a public location such as a chat room would yield similar data. Lastly, Bluetooth enabled devices have become notorious for leaking information. Hidden in a busy area such as a bus, subway or train terminal, a device designed to collect this sort of information [50] through continuous polling of Bluetooth-enabled mobile phones in the vicinity would quickly be able to create a large hit-list. If this system was left to run for a number of days, a correlation could be drawn between a phone number and a location given a time and day of the week.

4. Modeling DoS attacks

Given the existing bottlenecks and the ability to create hit-lists, we now discuss attacks against cellular networks. An adversary can mount an attack by simultaneously sending messages through the numerous available portals into the SMS network. The resulting aggregate load saturates the control channels thereby blocking legitimate voice and SMS communication. Depending on the size of the attack, the use of these services can be denied for targets ranging in size from major metropolitan areas to entire continents.

4.1. Metropolitan area service

As discussed in Section 2, the wireless portion of SMS delivery begins when the targeted device hears its *Temporary Mobile Subscriber ID* (TMSI) over the *Paging Channel* (PCH). The phone acknowledges the request via the *Random Access Channel* (RACH) and then proceeds with authentication and content delivery over a *Standalone Dedicated Control Channel* (SDCCH).

Voice call establishment is very similar to SMS delivery, except a *Traffic Channel* (TCH) is allocated for voice traffic at the completion of control signaling. The advantage of this approach is that SMS and voice traffic do not compete for TCHs, which are held for significantly longer periods of time. Therefore, TCH use can be optimized such that the maximum number of concurrent calls is provided. Because both voice and SMS traffic use the same channels for session establishment, contention for these limited resources still occurs. Given enough SMS messages, the channels needed for session establishment will become saturated, thereby preventing voice traffic to a given area. Such a scenario is not merely theoretical; instances of this contention have been well documented [2,3,14,24,33,41].

In order to determine the required number of messages to induce saturation, the details of the air interface must be examined. While the following analysis of this vulnerability focuses on GSM networks, other systems (e.g. CDMA [49]) are equally vulnerable to attacks.

The GSM air interface is a timesharing system. This technique is commonly employed in a variety of systems to provide an equal distribution of resources between multiple parties. Each channel is divided into eight timeslots and, when viewed as a whole, form a frame. During a given timeslot, the assigned user receives full control of the channel. From the telephony perspective, a user assigned to a given TCH is able to transmit voice data once per frame. In order to provide the illusion of continuous voice sampling, the frame length is limited to 4.615 ms. An illustration of this system is shown in Fig. 4.

Because the bandwidth within a given frame is limited, data (especially relating to the CCH) must often span a number of frames, as depicted in Fig. 5. This aggregation is known as a multiframe and is typically comprised of 51 frames.[7] For example, over

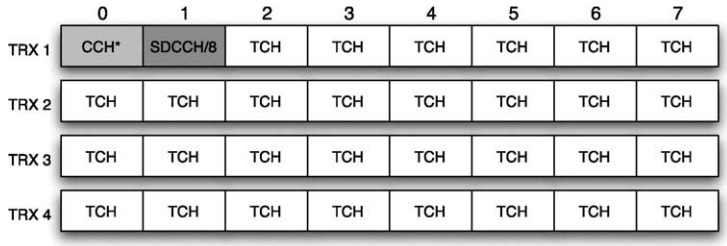

Fig. 4. An example air interface with four carriers (each showing a single frame). The first time slot of the first carrier is the Common CCH. The second time slot of the first channel is reserved for SDCCH connections. Over the course of a multiframe, capacity for eight users is allotted. The remaining time slots across all carriers are designated for voice data. This setup is common in many urban areas.

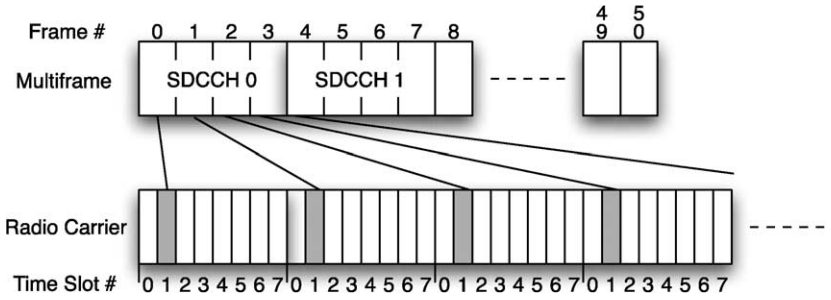

Fig. 5. Timeslot 1 from each frame in a multiframe creates the logical SDCCH channel. In a single multiframe, up to eight users can receive SDCCH access.

[7] Multiframes can actually contain 26, 51 or 52 frames. A justification for each case is available in the standards [4].

the course of a single multiframe, the base station is able to dedicate up to 34 of the 51 Common CCH slots to paging operations.

Each channel has distinct characteristics. While the PCH is used to signal each incoming call and text message, its commitment to each session is limited to the transmission of a TMSI. TCHs, on the other hand, remain occupied for the duration of a call, which on average is a number of minutes [39]. The SDDCH, which has approximately the same bandwidth as the PCH across a multiframe, is occupied for a number of seconds per session establishment. Accordingly, in many scenarios, this channel can become a bottleneck.

In order to determine the characteristics of the wireless bottleneck, it is necessary to understand the available bandwidth. As shown in Fig. 5, each SDCCH spans four logically consecutive timeslots in a multiframe. With 184 bits per control channel unit and a multiframe cycle time of 235.36 ms, the effective bandwidth is 782 bps [4]. Given that authentication, TMSI renewal, the enabling of encryption, and the 160 byte text message must be transferred, a single SDCCH is commonly held by an individual session for between four and five seconds [39]. The gray-box testing in Section 3.1 reinforces the plausibility of this value by observing no messages delivered in under six seconds.

This service time translates into the ability to handle up to 900 SMS sessions per hour on each SDCCH. In real systems, the total number of SDCCHs available in a sector is typically equal to twice the number of carriers,[8] or one per three to four voice channels. For example, in an urban location such as the one demonstrated in Fig. 4 where a total of four carriers are used, a total of eight SDCCHs are allocated. A less populated suburban or rural sector may only have two carriers per area and therefore have four allocated SDCCHs. Densely populated metropolitan sectors may have as many as six carriers and therefore support up to 12 SDCCHs per area.

We now calculate the maximum capacity of the system for an area. As indicated in a study conducted by the National Communications System (NCS) [39], the city of Washington DC has 40 cellular towers and a total of 120 sectors. This number reflects sectors of approximately 0.5–0.75 mi^2 through the 68.2 mi^2 city. Assuming that each of the sectors has eight SDCCHs, the total number of messages per second needed to saturate the SDCCH capacity C is:

$$\begin{aligned} C &\approx (120 \text{ sectors}) \left(\frac{8 \text{ SDCCH}}{1 \text{ sector}} \right) \left(\frac{900 \text{ msgs/h}}{1 \text{ SDCCH}} \right) \\ &\approx 864{,}000 \text{ msgs/h} \\ &\approx 240 \text{ msgs/s}. \end{aligned}$$

[8] Actual allocation of SDCCH channels may vary across implementations; however, these are the generally accepted values throughout the community.

Manhattan is smaller in area at 31.1 mi^2. Assuming the same sector distribution as Washington DC, there are 55 sectors. Due to the greater population density, we assume 12 SDCCHs are used per sector.

$$\begin{aligned} C &\approx (55 \text{ sectors})\left(\frac{12 \text{ SDCCH}}{1 \text{ sector}}\right)\left(\frac{900 \text{ msg/h}}{1 \text{ SDCCH}}\right) \\ &\approx 594{,}000 \text{ msg/h} \\ &\approx 165 \text{ msg/s}. \end{aligned}$$

Given that SMSCs in use by service providers in 2000 were capable of processing 2500 msgs/s [55], such volumes are achievable even in the *hypothetical* case of a sector having twice this number of SDCCHs.

Using a source transmission size of 1500 bytes as described in Section 3.1 to submit an SMS from the Internet, Table 3 shows the bandwidth required at the source to saturate the control channels, thereby incapacitating legitimate voice and text messaging services for Washington DC and Manhattan. The adversary's bandwidth requirements can be reduced by an order of magnitude when attacking providers including Verizon and Cingular Wireless due to the ability to have a single message repeated to up to ten recipients.

Due to the data gathered in Section 3.1, sending this magnitude of messages to a small number of recipients would degrade the effectiveness of such an attack. As shown in the previous section, targeted phones would quickly see their buffers reach capacity. Undeliverable messages would then be buffered in the network until the space allotted per user was also exhausted. These accounts would likely be flagged and potentially temporarily shut down for receiving a high number of messages in a short period of time, thereby fully extinguishing the attack. Clever usage of well constructed hit-lists keeps the number of messages seen by individual phones far below realistic thresholds for rate limitation on individual targets.

Table 3
Required upload bandwidth to saturate an empty network

Area	# Sectors	# SDCCHs/sector	SMS capacity (msgs/s)	Upload bandwidth* (kbps)	Multi-recipient bandwidth* (kbps)
Washington DC	120	8	240	2812.5	281.25
(68.2 mi^2)		12	360	4218.8	421.88
		24	720	8437.5	843.75
Manhattan	55	8	110	1289.1	128.91
(31.1 mi^2)		12	165	1933.6	193.66
		24	330	3867.2	386.72

*Assuming 1500 bytes per message.

Using the conservative population and demographic numbers cited from the NCS technical bulletin [39][9] and assuming 50% of the wireless subscribers in Washington are serviced by the same network, an even distribution of messages would require the delivery of approximately 5.04 messages to each phone per hour (1 message every 11.92 minutes) to saturate Washington DC If the percentage of subscribers receiving service from a provider is closer to 25%, the number is only 10.07 messages per hour (1 message every 5.96 minutes). In a more densely populated city such as Manhattan, with a population estimated at 1,318,000 with 60% wireless penetration and 12 SDCCHs, only 1.502 messages would have to be received per user per hour if half of the wireless clientele use the same provider. That number increases slightly to 3.01 if the number is closer to 25%.

Depending on the intended duration of an attack, the creation of very large hit-lists may not be necessary. An adversary may only require a five minute service outage to accomplish their mission. Assuming that the attacker created a hit-list with only 2500 phone numbers, with each target having a buffer of 50 messages and launched their attack in a city with 8 SDCCHs (e.g. Washington DC), uniform random use of the hit-list would deliver a single message to each phone every 10.4 seconds, allowing the attack to last 8.68 minutes before buffer exhaustion. Similar to the most dangerous worms in the Internet, this attack could be completed before anyone capable of thwarting it could respond.

When compared to the requisite bandwidth to launch these attacks listed in Table 3, many of these scenarios can be executed from a single high-end cable modem. A more distributed, less bandwidth intense attack might instead be launched from a *small* zombie network. Figure 6 summarizes the impact of such an attack.

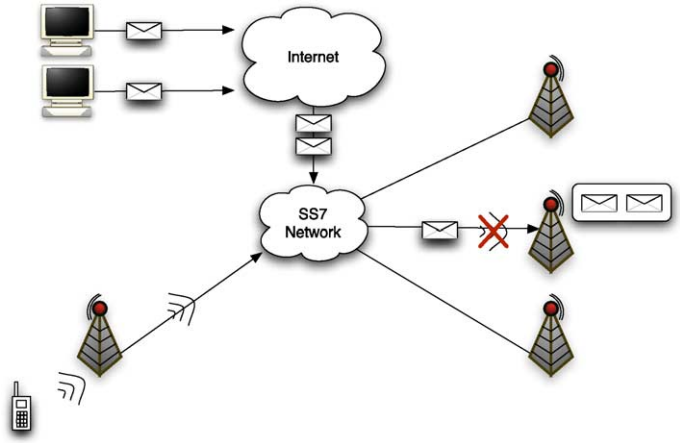

Fig. 6. A high level overview of the effects resulting from a targeted text messaging attack on a metropolitan area.

[9] 572,059 people with 60% wireless penetration and 8 SDCCHs (and that devices are powered on).

4.2. Regional service

Both popularity and the potential for high revenue have forced service providers to investigate methods of increasing SMS capacity in their networks. Already, a number of major industrial players [17,26] offer solutions designed to offload SMS traffic from the traditional SS7 phone system onto less expensive, higher bandwidth IP-based networks. New SMSCs, each capable of processing some 20,000 SMS messages per second, would help to quickly disseminate the constantly increasing demand.

Advanced services including *General Packet Radio Service* (GPRS) and *Enhanced Data rates for GSM Evolution* (EDGE) promise high speed data connections to the Internet for mobile devices. While offering to alleviate multimedia traffic at the SMSC and potentially send some SMS messages, these data services are widely viewed as complimentary to SMS and will thus not replace SMS's functionality in the foreseeable future [10].[10] In terms of SMS delivery, all aspects of the network are increasing available bandwidth except the SDCCH bottleneck.

We examine a conservative attack on the cellular infrastructure in the United States. From the United States Census in 2000, approximately 92,505 mi^2 [54] are considered urban. This 2.62% of the land is home to approximately 80% of the nation's population. We first model the attack by assuming that all urban areas in the country have high-capacity sectors (8 SDCCHs per sector). This assumption leads to the results shown below:

$$\begin{aligned} C &\approx \left(\frac{8 \text{ SDCCH}}{1 \text{ sector}}\right)\left(\frac{900 \text{ msg/h}}{1 \text{ SDCCH}}\right)\left(\frac{1.7595 \text{ sectors}}{1 \text{ mi}^2}\right)(92{,}505 \text{ mi}^2) \\ &\approx 1{,}171{,}890{,}342 \text{ msg/h} \\ &\approx 325{,}525 \text{ msg/s}. \end{aligned}$$

This attack would require approximately 3.8 Gbps and a nation-wide hit-list to be successful. If the adversary is able to submit a single message to up to ten different recipients, the requisite bandwidth for the attacker drops to approximately 370 Mbps. Considering that previous distributed DoS (DDoS) attacks have crippled websites such as Yahoo! (*www.yahoo.com*) with gigabit per second bandwidth, this attack on the entire cellular infrastructure is wholly realizable through a relatively small zombie network.

4.3. Targeted attacks

While total network degradation attacks can occur, Internet attacks can be targeted. Internet driven attacks directed at specific targets in the physical domain are not new.

[10] SMS over GPRS is already in service; however, it is not the default method of SMS delivery on GPRS-capable phones and must be activated by the user. Furthermore, SMS over GPRS still defaults to the standard SMS delivery mechanism when GPRS is unavailable.

In 2002, anonymous individuals inundated spammer Alan Ralsky with thousands of mail-order catalogs on a daily basis. Through the use of simple scripting tools and a lack of mechanisms to prevent automation [13], these individuals subscribed their target to postal mailing lists at a much faster rate than he could possibly be removed. In so doing, Mr. Ralsky's ability to receive normal mail at his primary residence was all but destroyed.

This same attack can be applied to SMS service. While the complete disruption of a user's SMS service is dangerous, a more interesting attack occurs when the adversary wishes to stop a victim from receiving useful messages. For example, a jealous ex-lover may wish to keep a message from being delivered; a stock trader may want to delay updates received by competitors; an attacker may want to keep a systems administrator from receiving a notification.

This attack is accomplished by flooding the user with a superfluous number of messages. This results in one of three outcomes: a buffer somewhere overflows and the message is lost, the message is delayed longer than its shelf-life,[11] or the user does not notice the message due to the deluge of meaningless messages.

In many cases, an attack allowing intentional message loss is ideal for the adversary. Mobile phones, like other embedded devices, have significant memory constraints, thereby limiting the number of messages a phone can hold. For all but the highest-end phones (see Section 3.1), this typically ranges from 30 to 50 messages. Once the phone can no longer receive messages, the service provider's network begins to buffers all subsequent messages. For reasons of practicality, providers impose limitations on the number of messages the network can store per user. Thus, if the adversary can exceed this value, messages become lost.

The SMSC is not the only source for message loss. As observed with the Nokia 3560, when the buffer became full, any message with content assumed to be known (any outbox message and read messages in the inbox) were automatically deleted. While this occurrence was isolated to the firmware of a specific phone, the potential to remotely maliciously destroy a user's data exists.

The onslaught of large numbers of packets helps accomplish the remaining two attack outcomes. During the testing in Section 3.1, where 400 messages were injected to determine the size of the SMSC buffers, the delivery of all packets took almost 90 minutes even with the constant monitoring and clearing of phone buffers. Temporally critical messages were potentially delayed beyond their period of usefulness. Additionally, the use of the "Clear Inbox" function significantly increases the possibility of a user accidentally deleting a legitimate text message that arrived among the attack messages.

While deleting an immense number of text messages is taxing on the user, as described in Section 3.1, the receipt of large amounts of data consumes significant battery power. This leads to yet another targeted DoS attack, a battery depletion attack.

[11] An SMS weather notification is useless if you are already stuck in the rain.

5. Solutions

Many of the mechanisms currently in place are not adequate to protect these networks. The proven practicality of address spoofing or distributed attacks via zombie networks makes the use of authentication based upon source IP addresses an ineffective solution [8]. As demonstrated in Section 4, limiting the maximum number of message received by an individual over a time period is also ineffective. Due to the tremendous earnings potential associated with open functionality, it is also difficult to encourage service providers to restrict access to SMS messaging. Solutions must therefore take all of these matters into consideration. The mechanisms below offer both long term and temporary options for securing cellular networks.

5.1. Eliminating Internet-originated text messaging

The most extreme approach to addressing the vulnerabilities discussed in this paper is by closing all text messaging gateways between cellular networks and the Internet. Unfortunately, such a solution is neither practical nor complete. In the case of the former, the closure of such interfaces would correspond to a significant loss of revenue. The ability to infect mobile phones with an increasing pool of malcode [22, 23] also reduces the feasibility of such measures. Whether through cellular data or Bluetooth connections, the ability to compromise and remotely control such devices allows attacks to be launched from within the network itself [53]. Given the size of these networks and the number of connected external entities, implementing this option may actually be impossible. Accordingly, we immediately discard this suggestion as unrealistic.

5.2. Separation of voice and data

It is highly unlikely that the numerous connections between the Internet and cellular networks will or can be closed by service providers. In light of this, the most effective means of eliminating the above attacks is by separating all voice and data communications. In so doing, the insertion of data into cellular networks will no longer degrade the fidelity of voice services.

This separation should occur in both the wired network and at the air interface. Dedicating a carrier on the air interface for data signaling and delivery eliminates an attacker's ability to take down voice communications. Dedicated data channels, however, are an inefficient use of spectrum and are therefore unattractive. Even if this solution is implemented, the bottleneck may be pushed into the SS7 network. More importantly, separating text messaging traffic onto IP or dedicated SS7 links does not prevent an attack from overloading the air interface. Until offloading schemes [17,26] are fully implemented in these networks, overload controls [29] based upon origin priority should be implemented to help shape traffic. As mentioned in Section 4.2, a partial separation has already begun with the introduction of data services including

GRPS and EDGE; however, these networks will remain vulnerable to attack as long as interconnections between voice and data flows exist.

The separation of voice and data is not enough to completely ensure unaffected wireless communications. In situations similar to September 11th where voice capacity is saturated, Internet-originated SMS messages can still be used to fill data channels such that legitimate text messaging is still impossible. SMS traffic should therefore be subject to origin classification. Text messages originating outside of the network should be assigned low priority on data channels. Messages originating within the phone network should receive high priority. This solution assumes that the SMSC is sufficiently protected from physical compromise by an attacker. If this expectation does not hold, more sophisticated, distributed mechanisms will have to be employed throughout the SS7 network.

5.3. Resource provisioning

Many service providers have experience dealing with temporary elevations in network traffic such as flash crowds. COSMOTE, the Greek telecommunications company responsible for providing service to the 2004 Olympic games, deployed additional base stations and an extra MSC in the area surrounding the Olympic Complex [19]. This extra equipment allowed this system to successfully deliver over 100 million text messages during the 17 day duration of the games [31]. Similarly, sporting events and large public gatherings in the United States regularly take advantage of so-called *Cellular-on-Wheels* (COW) services in order to account for location-dependent traffic spikes.

The effects of Internet-originated SMS attacks could be reduced by increasing capacity to critical areas in a similar fashion. Unfortunately, the cost of additional equipment makes this solution too expensive. Even if a provider rationalized the expense, the elevated provisioning merely makes DoS attacks more difficult but not impossible. Additionally, the increased number of handoffs resulting from reduced sector size would induce significant strain on the network core.

5.4. Rate limitation

Due to the time and money required to realize either of the above solutions, it is necessary to provide short term means of securing cellular networks. Many of these techniques harness well-known rate limitation mechanisms (e.g., fair queuing, etc.) and are explored in greater detail in our follow-on work [51].

On the air interface, the number of SDCCH channels allowed to deliver text messages could be restricted. Given the addition of normal traffic filling control channels, this attack would still be effective in denying service to all but a few individuals. Additionally, this approach slows the rate that legitimate text messages can be delivered, potentially elevating congestion in the core of the phone network. This approach is therefore not an adequate solution on its own.

Because many of these attacks are heavily reliant upon accurately constructed hit-lists, impeding their creation should be of the highest priority. Specifically, all of the web interfaces should cease returning both positive and negative acknowledgments for submitted SMS messages. Instead, a message indicating only that the submission was being processed should be returned so as to not permit an attacker from accurately mapping an NPA/NXX domain. This is currently the behavior seen when a mobile-to-mobile message is sent. Unfortunately, because legitimate users are unable to determine whether or not their message has been accepted by the system, the tradeoff for implementing this policy is a reduction in the reliability of Internet-originated text messages.

Furthermore, all web interfaces should limit the number of recipients to which a single SMS submission is sent. The ability to send ten messages per submission at both the Verizon and Cingular Wireless websites is particularly dangerous as flooding the system requires one-tenth of the messages and bandwidth necessary to interfere with other networks.

Reducing the ability to automate submissions is another approach that should be considered as a temporary solution for these interfaces. Having the sender's computer calculate tractable but difficult puzzles [7,28] before a submission is completed limits the frequency with which any machine can inject messages into a system. The use of CAPTCHAs [38,57], or images containing embedded text that is difficult for computers to parse, is also plausible. Because CAPTCHAs are not unbreakable [37] and puzzles only impede the submission speed for individuals, both of these countermeasures can be circumvented if an attacker employs a large enough zombie network.

A final option to keeping open web interfaces is to require a sender of a message through this mechanism to pay for the transaction. Accounts could be set up with a number of credits that could be increased at the cost of the user. This approach may be particularly attractive to service providers as it would provide an additional source of income; however, it may still be possible to purchase the requisite number of messages need to launch an attack through this interface, so careful planning would be necessary before any attempts at implementation were made. Additionally, it may be difficult to convince customers to pay for something they have been receiving for free for many years.

5.5. Education

While the above mechanisms are appropriate for the prevention of DoS attacks, they have limited success preventing other attacks. Phishers will still be able to send messages to individuals through the web interface with anonymity; however, their ability to blanket large prefixes in a short period of time is greatly reduced. Unfortunately, it may only require a single message for an attacker to get the sensitive information they seek. Malcode, whether downloaded or self-propagating, will also pose a similar problem. As mobile phones rapidly evolve from limited embedded

devices to general purpose computing platforms, they will become increasingly attractive targets [53]. While a number of software suites designed to protect mobile phones from such threats are available [21], these devices generally lack many of the most basic operating system protection mechansisms.

The most practical solution for this family of exploits is therefore education. Cellular service providers must launch an aggressive campaign to reach all of their clients to tell them that no such request for information will ever come via SMS text. To this date, we are unaware of any such effort.

6. Realization of technical solutions

The publication of vulnerabilities discovered during the course of research is hotly debated. Disclosure of such problems, as is frequently argued, ultimately leads to their exploitation. In some cases, such as those in which absolutely no short-term mitigation techniques are available or human life is imminently threatened, immediate public disclosure may in fact be irresponsible. Equally as dangerous, however, is the failure to discuss such problems in the larger community. As the recent increase in zero-day exploits makes painfully obvious, the public disclosure of a vulnerability does not necessarily indicate its independent discovery. The researchers and engineers responsible for designing such systems must be aware of previous weaknesses not only to correct problems, but also to avoid recreating them in the future. In order to balance these concerns, we engaged in significant discussions with federal and state officials prior to the release of the original paper [20]. Through these conversations, we were able to communicate with representatives within the telecommunications industry and eventually the research community. These discussions resulted in both long and short term changes. In order to more fully understand the current landscape, we look back a year later at the technical reactions to our disclosure.

One of the greatest misconceptions of this attack centered around SMS spam. Our initial work in fact briefly examined the presence of unsolicited messages made possible by connections between the Internet and cellular networks. Volumes of such spam have in fact increased since the initial publication of this work [43]; however, this is a tangential issue to the problem presented in this paper. To illustrate this point, we return to the results of the gray-box testing discussed in Section 3.1. To avoid losing messages to spam filtering, we used sets of text designed to pass as regular communications. Specifically, we took 160 byte sequences from the terms of service agreements available on service provider websites. At no time were any of our messages unexpectedly dropped. Less obvious text could be increasingly effective. For example, a message containing phrases such as “Stuck in the office – will be late for dinner.” would be virtually impossible to filter out. While spam filtering has certainly advanced significantly since its inception, it is not sufficiently developed such that it is able to determine sender intent. Accordingly, filters are a necessary defense, but ultimately fail to prevent such attacks from occurring.

Fig. 7. Approximately one year after the publication of the original paper, Cingular Wireless placed restrictions on its web interface. Only subscribers can now submit messages directly through their portal; however, email-submitted messages still remain open to all.

Because of the closed nature of cellular networks, observing the progress of measures addressing this problem has been difficult. However, there have been a number of encouraging signs. Nearly a year after the initial public disclosure of the vulnerability, Cingular Wireless (once again AT&T) closed their website's public SMS interface. Figure 7 shows that, due to the potential for misuse, this avenue for submitting text messages has been removed [16]. Instead, subscribers can log into their accounts to use this service. This technique, however, does not completely solve the problem. It is still possible to send unsolicited messages to end hosts via email, instant messaging programs and bulk providers. As mentioned in Section 5, mobile hosts compromised by malware also remain a threat. While this move is a significant step towards addressing the problem, it is by no means a silver-bullet. The steps taken by other providers are less obvious. At the time of this writing, the website text messaging interface for Verizon Wireless, Sprint and T-Mobile remain unchanged. Through additional gray-box testing, we were still able to extract valuable information from these websites to assist in the creation of hit-lists. This does not necessarily mean that changes have not been made internally – the presence of additional protection mechanisms, such as those discussed in our follow-on work [51], is unknown. Our ability to determine the existence of such mechanism through further gray-box is likely beyond our legal capacity.

7. Related work

Phone networks are among the oldest digital systems in the world. In spite of their distributed nature, these networks have traditionally enjoyed a relatively high level of security due to a logical and physical separation from external systems. As phone networks become increasingly interconnected with networks such as the Internet, previous security assumptions no longer hold. Since the initial convergence of these networks, a number of vulnerabilities have been discovered. Before 2002 messages between SS7 network nodes were transmitted in plaintext without authentication [47]. Additionally, the parsers for call routing information, which use the

ASN.1 language, were demonstrated to be vulnerable to buffer overflow attacks. Despite current efforts of securing mechanisms critical to network operation [30,36], little attention has been paid to directly securing end users against the consequences of connecting phone networks to the Internet.

Attaching systems to the Internet has been problematic in other contexts as well. By leveraging the combination of automation and anonymity in the digital domain, an adversary can negatively affect systems in the physical world. Byers et al. [13] demonstrated the ability to use simple automated scripting tools to register an individual for large volumes of postal junk mail. The speed of this attack far outpaces the ability of the targeted individual to remove him or herself from the mailing lists, thereby destroying all practical usability of one's physical mailbox.

A large number of websites have fallen victim to DoS attacks [1]. Access to Yahoo!, Amazon and eBay were all temporarily restricted when their servers were flooded with over a gigabit per second of traffic in 2002 [18]. Significant research has been dedicated to exploring and defending against these attacks on the Internet [25, 28,34,46]. The inability to differentiate the origin of SMS messages after arrival at end devices makes techniques used to trace and mitigate [27,45] these attacks ineffective. While attacks have been mounted against specific phones [44], the feasibility of a widespread a DoS and the effectiveness of traditional DoS countermeasures on a phone network have not been explored.

In an attempt to understand the parameters leading to non-malicious, congestion-based DoS scenarios in a wireless environment, the National Communications System published a study examining the effects of SMS messages [39]. This study primarily focused upon problems caused by mobile to mobile communications and the lack of privacy users relying on email for SMS delivery should expect. While the lack of capacity available in critical scenarios was well highlighted, little focus was given to the impact of an intentionally malicious intruder, especially one originating in the Internet.

8. Conclusion

Cellular networks are a critical part of the economic and social infrastructures in which we live. These systems have traditionally experienced below 300 seconds of communication outages per year (i.e., "five nines" availability). However, the proliferation of external services on these networks introduces significant potential for misuse. We have shown that an adversary injecting text messages from the Internet can cause almost twice the yearly expected network down-time in a metropolitan area using hit-lists containing as few as 2500 targets. With additional resources, cyberwarfare attacks capable of denying voice and SMS service to an entire continent are also feasible. By attacking the less protected edge components of the network, we elicit the same effects as would be seen from a successful assault on the well protected network core.

More critically, this vulnerability hints at a larger architectural conflict between cellular and traditional data networks. As our later work shows [52], it is not simply low bandwidth channels that endanger cellular networks. Rather, it is because the network expends significant effort in finding and establishing connections with mobile devices. Adding bandwidth to the SDCCHs simply pushes the vulnerability to other constrained portions of the connection establishment process. Because the network and not higher level protocols is responsible for connection establishment, these systems inherently create exploitable amplification points.

Mobile voice and text messaging have become indispensable tools in the lives of billions of people across the globe. The problems presented in this paper must therefore be addressed in order to preserve the usability of these critical services.

9. Acknowledgements

This work was supported in part by the National Science Foundation (CNS-0721579). Any opinions, findings and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation. We would also like to thank Matt Blaze, Somesh Jha, Gary McGraw, Fabian Monrose, Avi Rubin, the members of the SIIS Lab and the anonymous readers and reviewers for providing many insightful comments on this paper.

Appendix

- **BS:** Base Station
- **CCH:** Control Channel
- **CDMA:** Code Division Multiple Access
- **COW:** Cellular-on-Wheels
- **DoS:** Denial of Service
- **EDGE:** Enhanced Data rates for GSM Evolution
- **ESME:** External Short Messaging Service
- **GPRS:** General Packet Radio Service
- **GSM:** Global System for Mobile Communication
- **HLR:** Home Location Register
- **MSC:** Mobile Switching Center
- **NCS:** National Communications System
- **NANP:** North American Numbering Plan
- **PCH:** Paging Channel
- **PSTN:** Public Switched Telephone Network
- **RACH:** Random Access Channel
- **SDCCH:** Standalone Dedicated Control Channel

- **SMPP:** Short Messaging Peer Protocol
- **SMS:** Short Messaging Service
- **SMSC:** Short Messaging Service Center
- **TCH:** Traffic Channel
- **TMSI:** Temporary Mobile Subscriber Identity

References

[1] Denial of service attacks, Technical report, CERT Coordination Center, October 1997, http://www.cert.org/tech_tips/denial_of_service.html.

[2] Mobile networks facing overload, December 31, 2003, http://www.gateway2russia.com/st/art_187902.php.

[3] Record calls, text again expected for nye, December 31, 2004, http://www.itnews.com.au/newsstory.aspx?CIaNID=17434.

[4] 3rd Generation Partnership Project, Physical layer on the radio path; general description, Technical report 3GPP TS 05.01 v8.9.0.

[5] 3rd Generation Partnership Project, Technical realization of the short message service (sms), Technical report 3GPP TS 03.40 v7.5.0.

[6] A. Arpaci-Dusseau and R. Arpaci-Dusseau, Information and control in gray-box systems, in: *Proceedings of Symposium on Operating Systems Principles (SOSP)*, 2001, pp. 43–56.

[7] T. Aura, P. Nikander and J. Leiwo, Dos-resistant authentication with client puzzles, in: *Proceedings of Cambridge Security Protocols Workshop*, 2000.

[8] S. Bellovin, Security problems in the TCP/IP protocol suite, *Computer Communications Review* **19**(2) (1989), 32–48.

[9] S. Berg, A. Taylor and R. Harper, Mobile phones for the next generation: Device designs for teenagers, in: *Proceedings ACM SIGCHI Conference on Human Factors in Computing Systems*, 2003, pp. 433–440.

[10] S. Buckingham, What is GPRS?, 2000, http://www.gsmworld.com/technology/gprs/intro.shtml#5.

[11] J.V.D. Bulck, Text messaging as a cause of sleep interruption in adolescents, evidence from a cross-sectional study, *Journal of Sleep Research* **12**(3) (2003), 263.

[12] N. Burnett, J. Bent, A. Arpaci-Dusseau and R. Arpaci-Dusseau, Exploiting gray-box knowledge of buffer-cache management, in: *Proceedings of USENIX Annual Technical Conference*, 2002, pp. 29–44.

[13] S. Byers, A. Rubin and D. Kormann, Defending against an internet-based attack on the physical world, *ACM Transactions on Internet Technology (TOIT)* **4**(3) (2004), 239–254.

[14] A. Choong, Wireless watch: Jammed, September 7, 2004, http://asia.cnet.com/reviews/handphones/wirelesswatch/0,39020107,39186280,00.htm.

[15] Cingular Wireless, Text messaging, https://www.cingular.com/media/text_messaging_purchase.

[16] Cingular Wireless, Cingular Customer Forums: Re: send a text message from the Web, 2006, http://forums.cingular.com/cng/board/message?board.id=messaging&message.id=8997#M8997.

[17] Cisco Systems Whitepaper, A study in mobile messaging: The evolution of messaging in mobile networks, and how to efficiently and effectively manage the growing messaging traffic, Technical report, 2004, http://www.cisco.com/warp/public/cc/so/neso/mbwlso/mbmsg_wp.pdf.

[18] Computer Associates, Carko, http://www3.ca.com/securityadvisor/pest/pest.aspx?id=453075555.

[19] COSMOTE Whitepaper, COSMOTE and the 'Athens 2004' olympic sponsorship, Technical report, 2003, http://www.cosmote.gr/content/en/attached_files/investorrelations/COSMOTE_Annual_Report_2003_77-84.pdf.

[20] W. Enck, P. Traynor, P. McDaniel and T.F. La Porta, Exploiting open functionality in SMS-capable cellular networks, in: *Proceedings of the ACM Conference on Computer and Communication Security (CCS)*, November 2005.

[21] J. Evers, Is your cell phone due for an antivirus shot?, 2006, http://news.com.com/Is+your+cell+phone+due+for+an+antivirus+shot/2100-7349_3-6042745.html.

[22] F-Secure Corporation, F-Secure virus descriptions: Cabir.h, December 2004, http://www.f-secure.com/v-descs/cabir_h.shtml.

[23] F-Secure Corporation, F-Secure virus descriptions: Skulls.a, January 2005, http://www.f-secure.com/v-descs/skulls.shtml.

[24] M. Grenville, Operators: Celebration messages overload sms network, November 2003, http://www.160characters.org/news.php?action=view&nid=819.

[25] K. Houle and G. Weaver, Trends in denial of service attack technology, Technical report, CERT Coordination Center, October 2001, http://www.cert.org/archive/pdf/DoS_trends.pdf.

[26] Intel Whitepaper, SMS messaging in SS7 networks: Optimizing revenue with modular components, Technical report, 2003, http://www.intel.com/network/csp/pdf/8706wp.pdf.

[27] J. Ioannidis and S. Bellovin, Implementing pushback: Router-based defense against DDoS attacks, in: *Proceedings of Network and Distributed System Security Symposium*, February 2002.

[28] A. Juels and J.G. Brainard, Client Puzzles: A cryptographic countermeasure against connection depletion attacks, in: *Proceedings of Network and Distributed System Security Symposium (NDSS)*, 1999.

[29] S. Kasera, J. Pinheiro, C.L.M. Karaul, A. Hari and T.L. Porta, Fast and robust signaling overload control, in: *Proceedings IEEE Conference on Network Protocols (ICNP)*, November 2001, pp. 323–331.

[30] G. Lorenz, T. Moore, G. Manes, J. Hale and S. Shenoi, Securing ss7 telecommunications networks, in: *Proceedings of the IEEE Workshop on Information Assurance and Security*, 2001.

[31] S. Makris, Athens 2004 games: The "extreme makeover" olympics!, April 2005, Slides presented at *CQR 2005 Workshop*, St. Petersburg Beach, FL, USA.

[32] K. Maney, Surge in text messaging makes cell operators :-), July 27, 2005.

[33] S. Marwaha, Will success spoil sms?, March 15, 2001, http://wirelessreview.com/mag/wireless_success_spoil_sms/.

[34] J. Mirkovic and P. Reiher, A taxonomy of DDoS attacks and DDoS defense mechanisms, *ACM SIGCOMM Computer Communication Review* **34**(2) (2004), 39–53.

[35] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford and N. Weaver, Inside the slammer worm, *IEEE Security and Privacy* **1**(4) (2003), 33–39.

[36] T. Moore, T. Kosloff, J. Keller, G. Manes and S. Shenoi, Signalling system 7 network security, in: *Proceedings of the IEEE 45th Midwest Symposium on Circuits and Systems*, August 4–7, 2002.

[37] G. Mori and J. Malik, Recognizing objects in adversarial clutter: Breaking a visual captcha, in: *Proc. of Computer Vision and Pattern Recognition*, 2003.

[38] M. Naor, Verification of human in the loop or identification via the turing test, 1996, http://www.wisdom.weizmann.ac.il/~naor/PAPERS/human.ps.

[39] National Communications System, SMS over SS7, Technical Report Technical Information Bulletin 03-2 (NCS TIB 03-2), December 2003, http://www.ncs.gov/library/tech_bulletins/2003/tib_03-2.pdf.

[40] Nextel, Text messaging, http://www.nextel.com/en/services/messaging/text_messaging.shtml.

[41] J. Pearce, Mobile firms gear up for new years text-fest, December 30, 2003, http://news.zdnet.co.uk/communications/networks/0,39020345,39118812,00.htm.

[42] H. Project, The honeynet project, 2005, http://project.honeynet.org.

[43] M. Reardon, Text message spam could spell trouble for text-based ads, 2006, http://news.com.com/Text+message+spam+could+spell+trouble+for+text-based+ads/2100-1039-6135609.html?part=dht&tag=nl.e703.

[44] P. Roberts, Nokia phones vulnerable to dos attack, February 26, 2003, http://www.infoworld.com/article/03/02/26/HNnokiados_1.html.

[45] S. Savage, D. Wetherall, A. Karlin and T. Anderson, Practical network support for IP traceback, in: *Proceedings of ACM SIGCOMM*, October 2000, pp. 295–306.

[46] C. Schuba, I. Krsul, M. Kuhn, E. Spafford, A. Sundaram and D. Zamboni, Analysis of a denial of service attack on TCP, in: *Proceedings of the 1997 IEEE Symposium on Security and Privacy*, IEEE Computer Society, May 1997, pp. 208–223.

[47] G. Shannon, Security vulnerabilities in protocols, in: *Proceedings of ITU-T Workshop on Security*, May 13–14, 2002.

[48] S. Staniford, V. Paxson and N. Weaver, How to 0wn the internet in your spare time, in: *Usenix Security Symposium*, 2002, pp. 149–167.

[49] Telecommunication Industry Association/Electronic Industries Association (TIA/EIA) Standard, Short messaging service for spread spectrum systems, Technical report ANSI/TIA/EIA-637-A-1999.

[50] Tom's Hardware, How to: Building a bluesniper rifle, March 2005, http://www.tomsnetworking.com/Sections-article106.php.

[51] P. Traynor, W. Enck, P. McDaniel and T. La Porta, Mitigating attacks on open functionality in SMS-capable cellular networks, in: *Proceedings of the Twelfth Annual ACM International Conference on Mobile Computing and Networking (MobiCom)*, September 2006.

[52] P. Traynor, P. McDaniel and T. La Porta, On attack causality in internet-connected cellular networks, in: *Proceedings of the USENIX Security Symposium*, 2007.

[53] P. Traynor, V. Rao, T. Jaeger, P. McDaniel and T. La Porta, From mobile phones to responsible devices, Technical report NAS-TR-0059-2006, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, 2007.

[54] United States Census Bureau, United States Census 2000, 2000, http://www.census.gov/main/www/cen2000.html.

[55] S. van Zanen, Sms: Can networks handle the explosive growth?, 2000, http://www.wirelessdevnet.com/channels/sms/features/smsnetworks.html.

[56] Verizon Wireless, About the service, http://www.vtext.com/customer_site/jsp/aboutservice.jsp.

[57] L. von Ahn, M. Blum, N. Hopper and J. Langford, CAPTCHA: Using hard AI problems for security, in: *Proceedings of Eurocrypt*, 2003, pp. 294–311.