Exploiting Open Functionality in SMS-Capable Cellular Networks

Enck, W., Traynor, P., McDaniel, P., & La Porta, T.
CCS '05

Mobile phones in 2005

Mobile phones in 2005

Not just for games!

Cellular voice calling

Downloadable ringtones

Mobile phones in 2005

Not just for games!

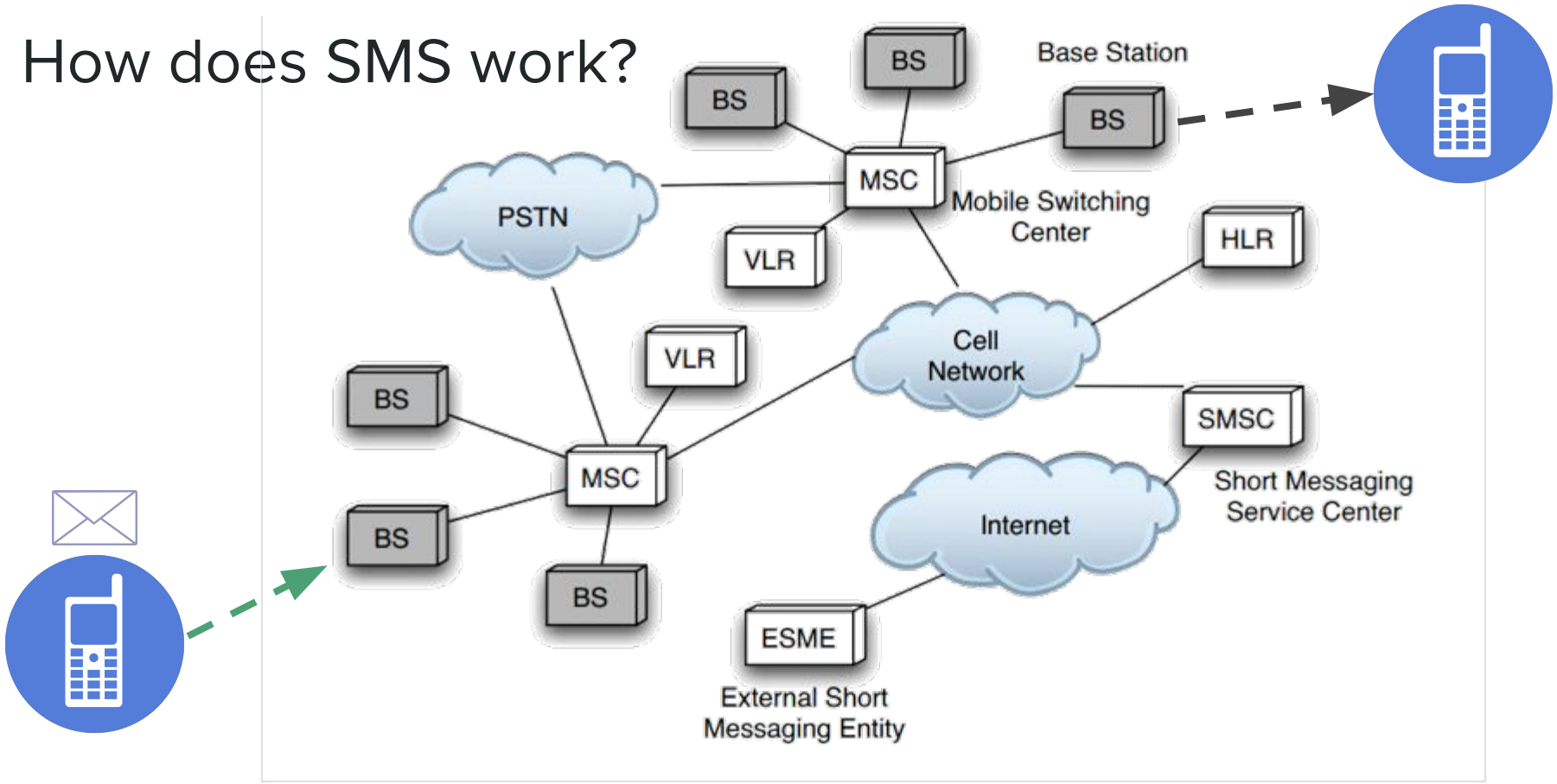
Cellular voice calling

Downloadable ringtones

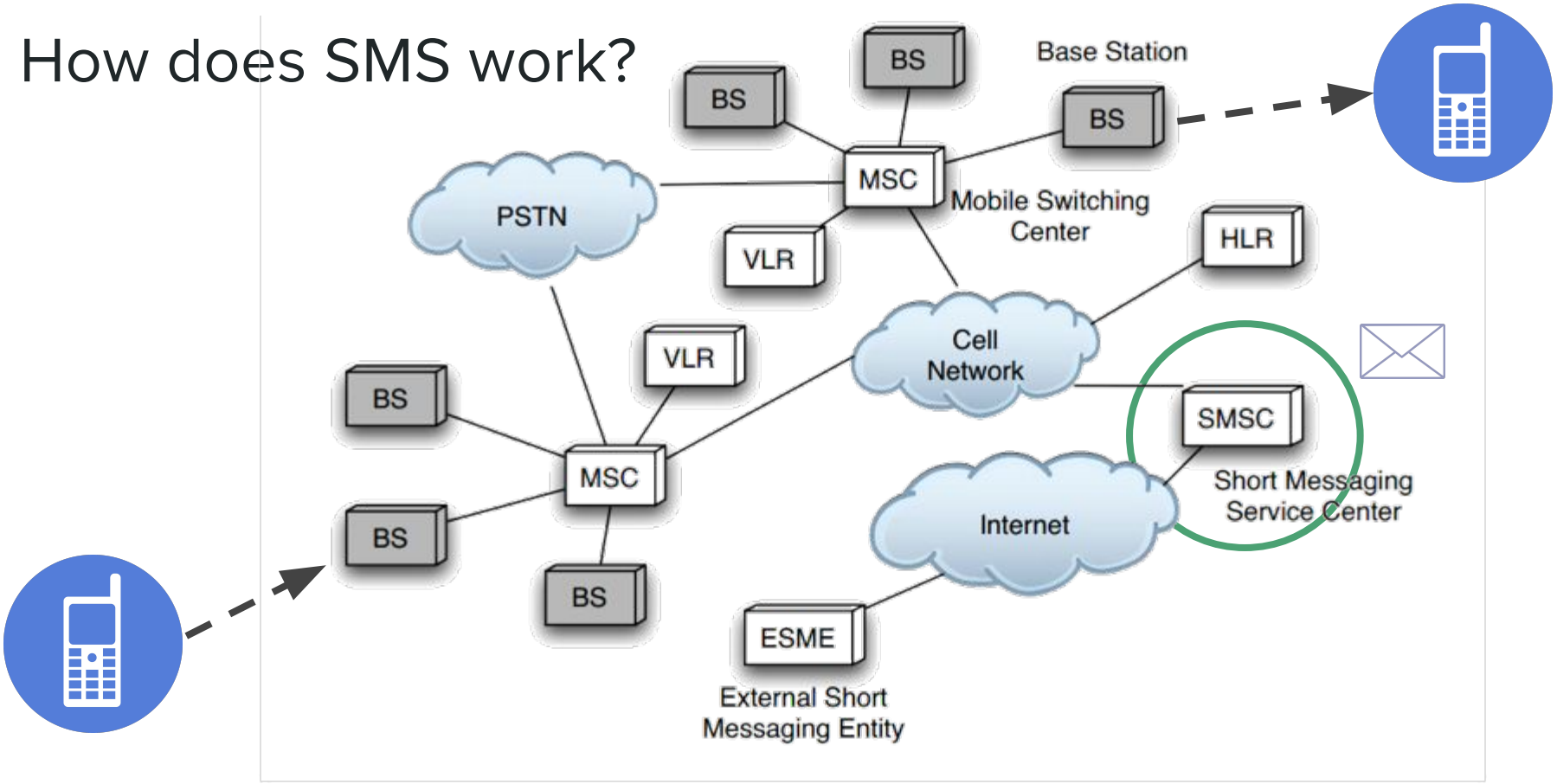
SMS - Short Messaging Service

69M msgs/day in UK

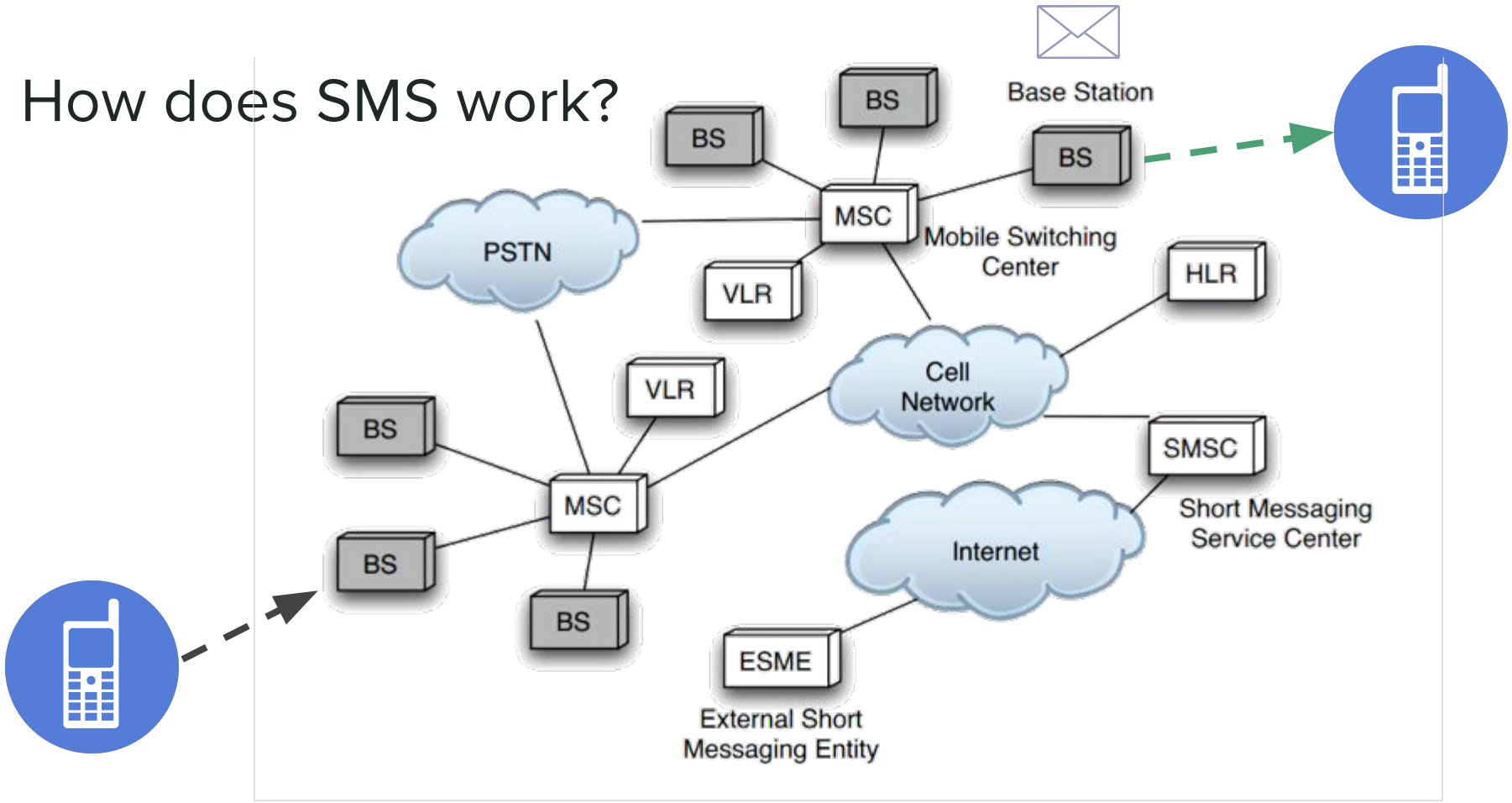
How does SMS work?



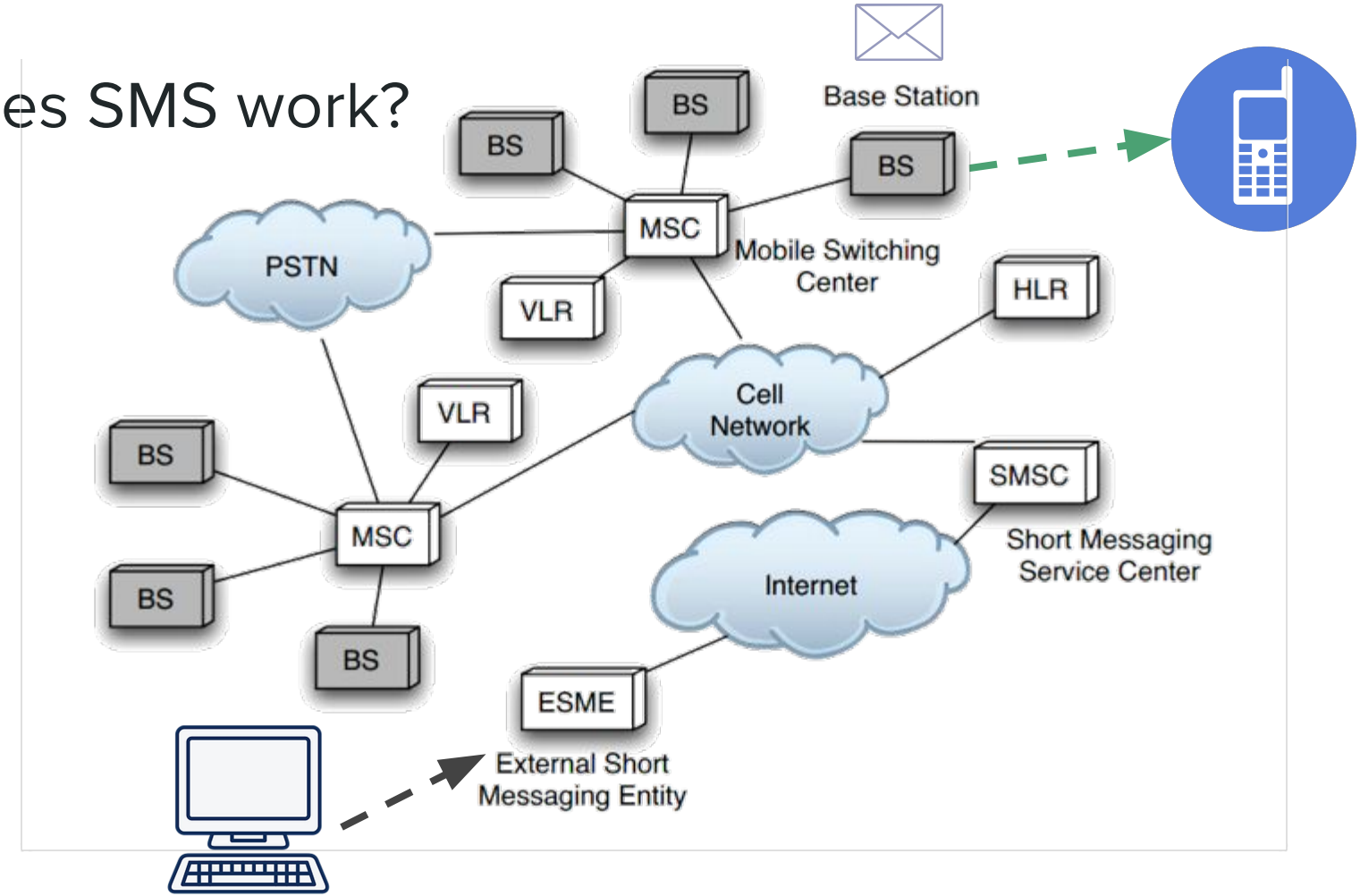
How does SMS work?



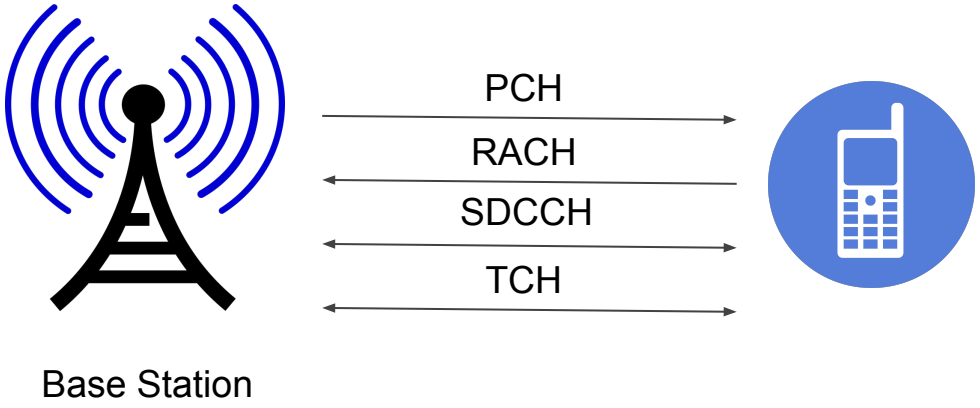
How does SMS work?



How does SMS work?



Wireless Last Hop



Paging Channel (PCH)

Random Access Channel (RACH)

Standalone Dedicated Control Channel (SDCCH) - SMS tacked on

Traffic Channel (TCH)

Finding Bottlenecks

Threat model: attacker has access to the external interfaces of the system

Storage bottlenecks

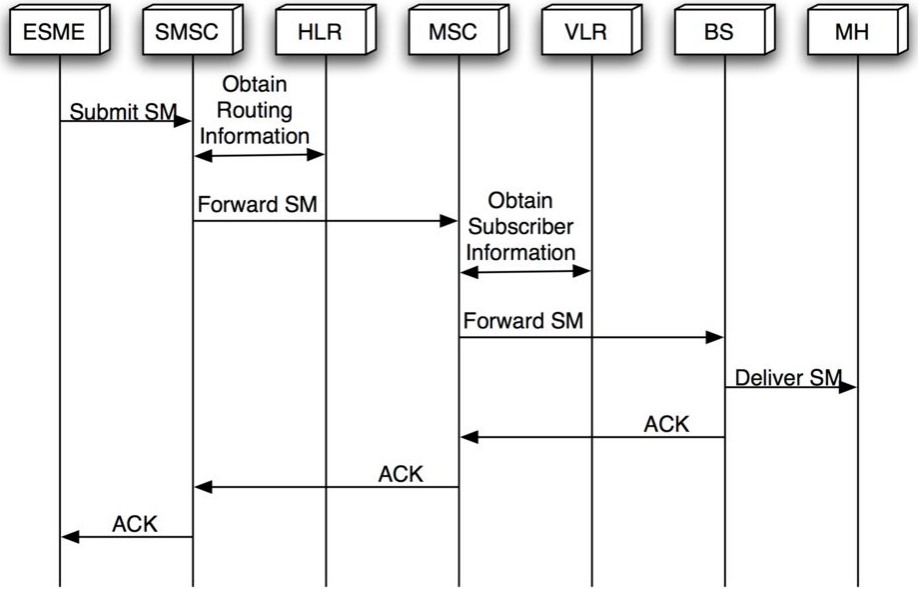
SMSC: 30 to >400 messages

Mobile Host: 30 to 500 messages

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.



Finding Bottlenecks

Threat model: attacker has access to the external interfaces of the system

Storage bottlenecks

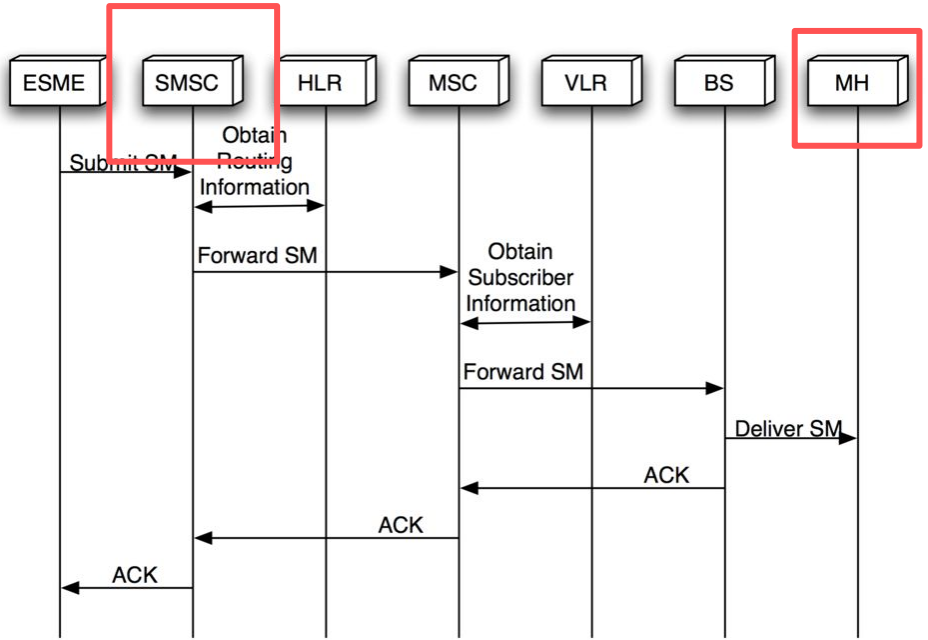
SMSC: 30 to >400 messages

Mobile Host: 30 to 500 messages

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.



Finding Bottlenecks

Transport bottlenecks from ESME

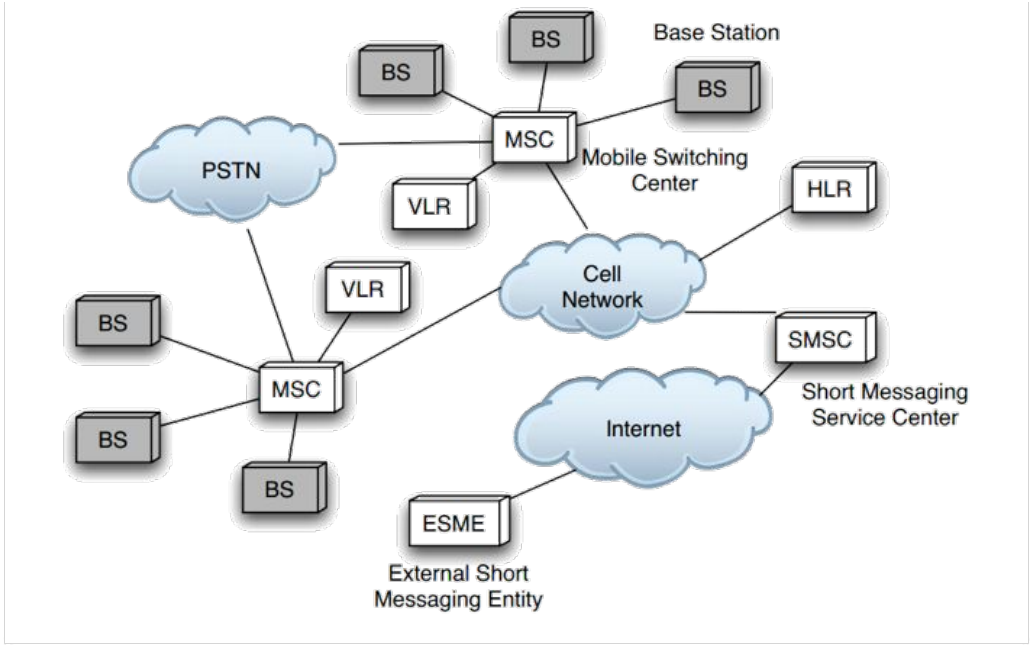
Input rate: 0.71 sec / msg

Approx. 1500 bytes / msg

Output rate: 7-8 sec / msg, sporadic

Approx. 160 bytes / msg

Input rate > output rate => DoS opportunity



Carrier DoS Defenses!

Rate-limiting (sent to a single phone)

Verizon - subnet based blocking

AT&T - input type based blocking

Sprint - Additional session cookie

Approx. 30 messages can be injected before loss

Carrier DoS Defenses!

Rate-limiting (sent to a single phone)

Verizon - subnet based blocking

AT&T - input type based blocking

Sprint - additional session cookie

Approx. 30 messages can be injected before loss

Weakness: distributed inputs (web services) and/or distributed outputs (hit lists)

Targeted Attack

Single out a specific set of phone numbers

Overwhelm SMSC

- dropped messages - Verizon FIFO, Sprint LIFO

Overwhelm Mobile Host

- Some firmware will delete messages

Constructing Hit-lists

Challenge: Telephone number address space is sparsely populated

Brute-force approach is costly and can trigger warning flags

1) NPA/NXX - *regional* registry of telephone # “subnets”
2) Web scraping via search engine - specific addresses
3) Web interface oracle - slow and low, hard-to-trace
4) Other - mobile worms, leaky Bluetooth, etc.

Regional Attack

Known vulnerability - SMS to block voice service

SDCCH - Encryption initiation, Subscriber authentication, Call set-up and **SMS**

Attack - overwhelm SDCCH with SMS, prevent other functionality

Modeling

Area	# Sectors	# SDCCHs/sector	SMS Capacity	Upload Bandwidth*	Multi-Recipient Bandwidth*
Washington D.C. ($68.2\ mi^2$)	120	8	240 msgs/sec	2812.5 kbps	281.25 kbps
		12	360 msgs/sec	4218.8 kbps	421.88 kbps
		24	720 msgs/sec	8437.5 kbps	843.75 kbps
Manhattan ($31.1\ mi^2$)	55	8	110 msgs/sec	1289.1 kbps	128.91 kbps
		12	165 msgs/sec	1933.6 kbps	193.66 kbps
		24	330 msgs/sec	3867.2 kbps	386.72 kbps

* assuming 1500 bytes per message

Email of Tomorrow

A.k.a. Everything old is new again!

SMS is more personal, trusted

Spam - can overwhelm limited user systems

Phishing - spoofing is extremely easy

Viruses

Mitigations

1) Separation of Voice and Data
    a) Completely separate channels (EDGE) or voice prioritization
2) Resource provisioning
    a) Beef up resources where/when they are needed (COWs)
3) Rate Limitation
    a) Make hit lists harder to construct - reduce legitimate usage rates
4) Education
    a) Always a good thing - can never have enough

Summary

SMS is a complex system with many different bottlenecks

Connecting SMS to the Internet has allowed scalable outsider access

Volume-based attacks

- Known SDCCH vulnerabilities
- Other DoS
- Email security issues

Discussion

Given the advantage of retrospection, what mitigations did / didn’t work? Why?

Do any of these systemic vulnerabilities persist? How can we fix them?

Contributions of the paper?

Things that could have been better?