Transparent Botnet Control for Smartphones over SMS

Georgia Weidman

Why Smartphone Botnets?

Nearly 62 million smartphones sold in Q2 2010

Development is similar to standard platforms

- Android = Linux
- iPhone = OSX
- Windows Mobile = Windows

Technical specs not as good as top of the line desktops. They are as good as the desktops you might have at work.

Why SMS C&C?

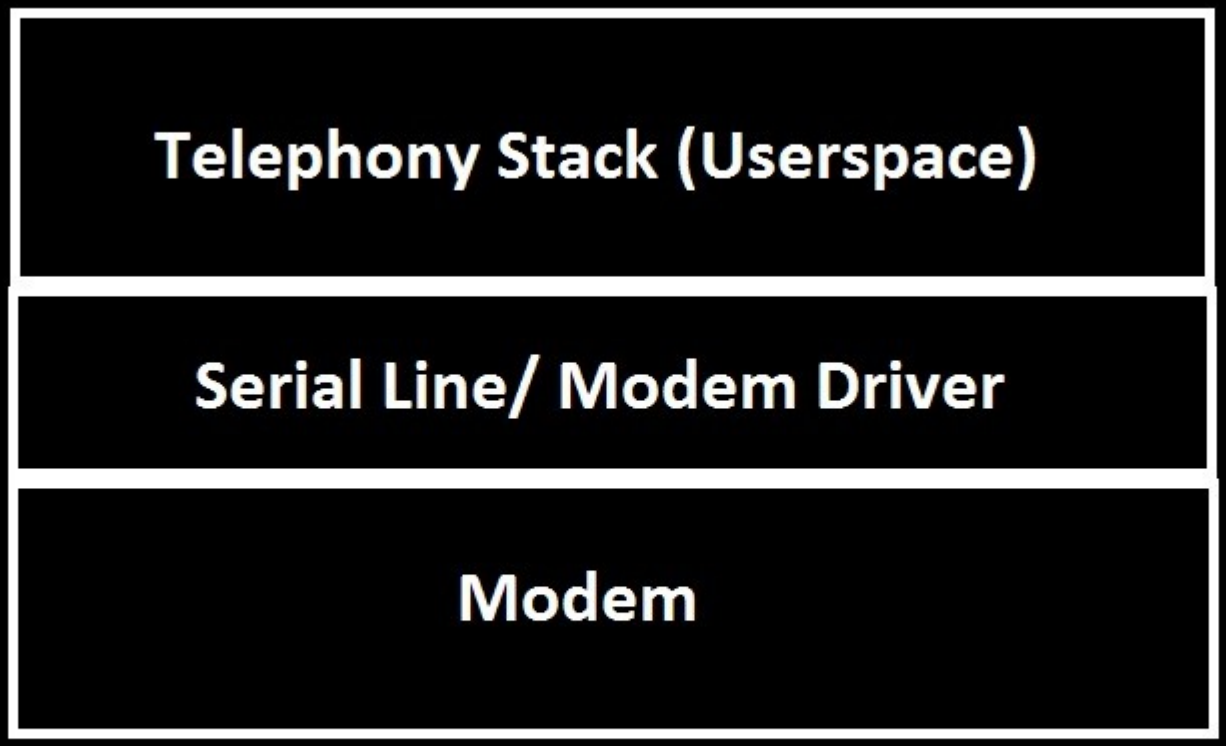
Battery Management

Fault Tolerant

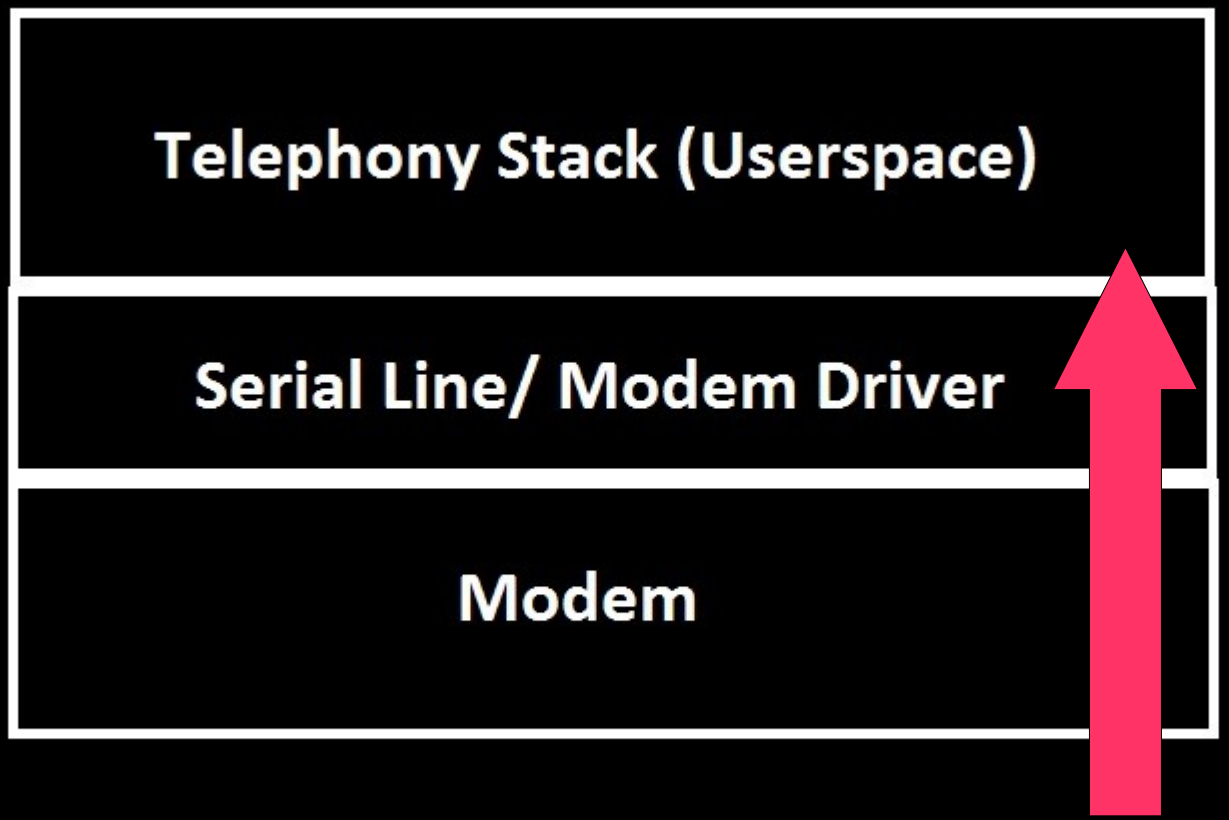
Always On

Difficult for security researchers to monitor

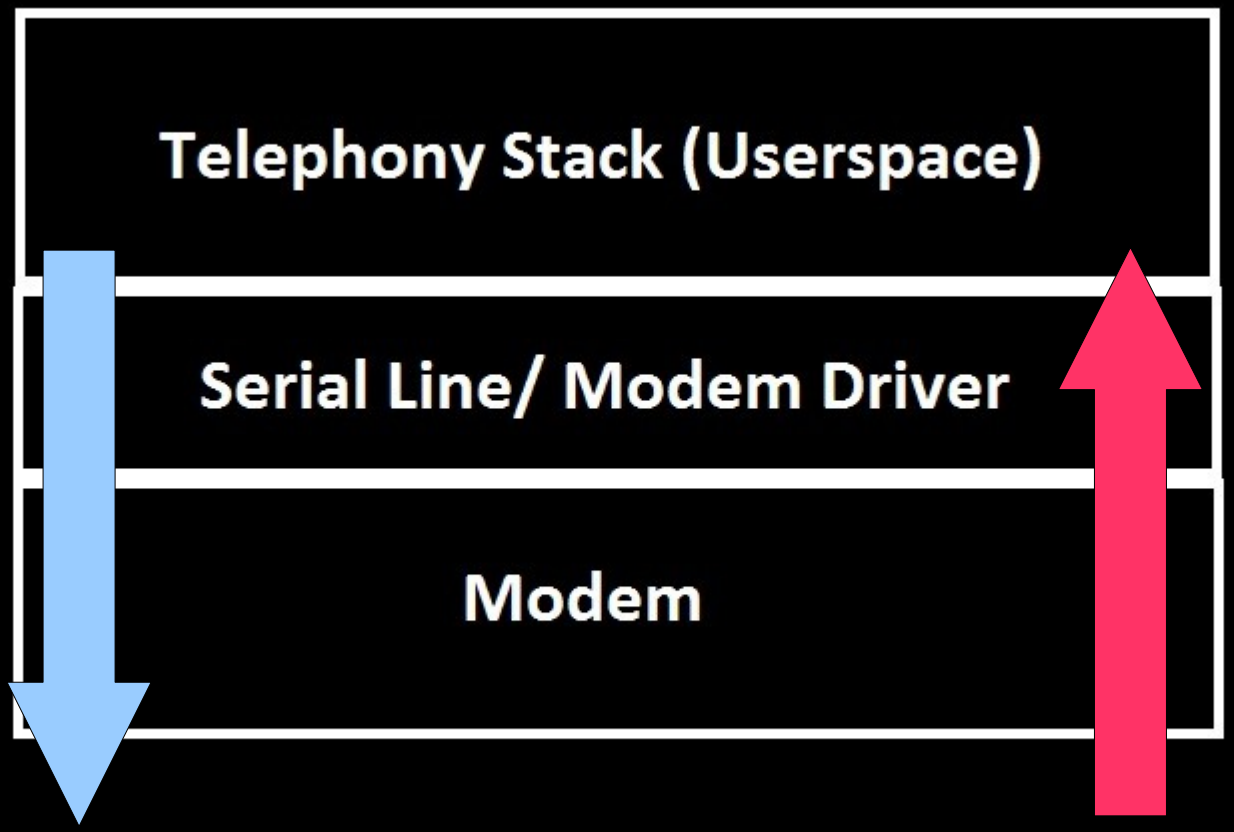
How an SMS is sent and received



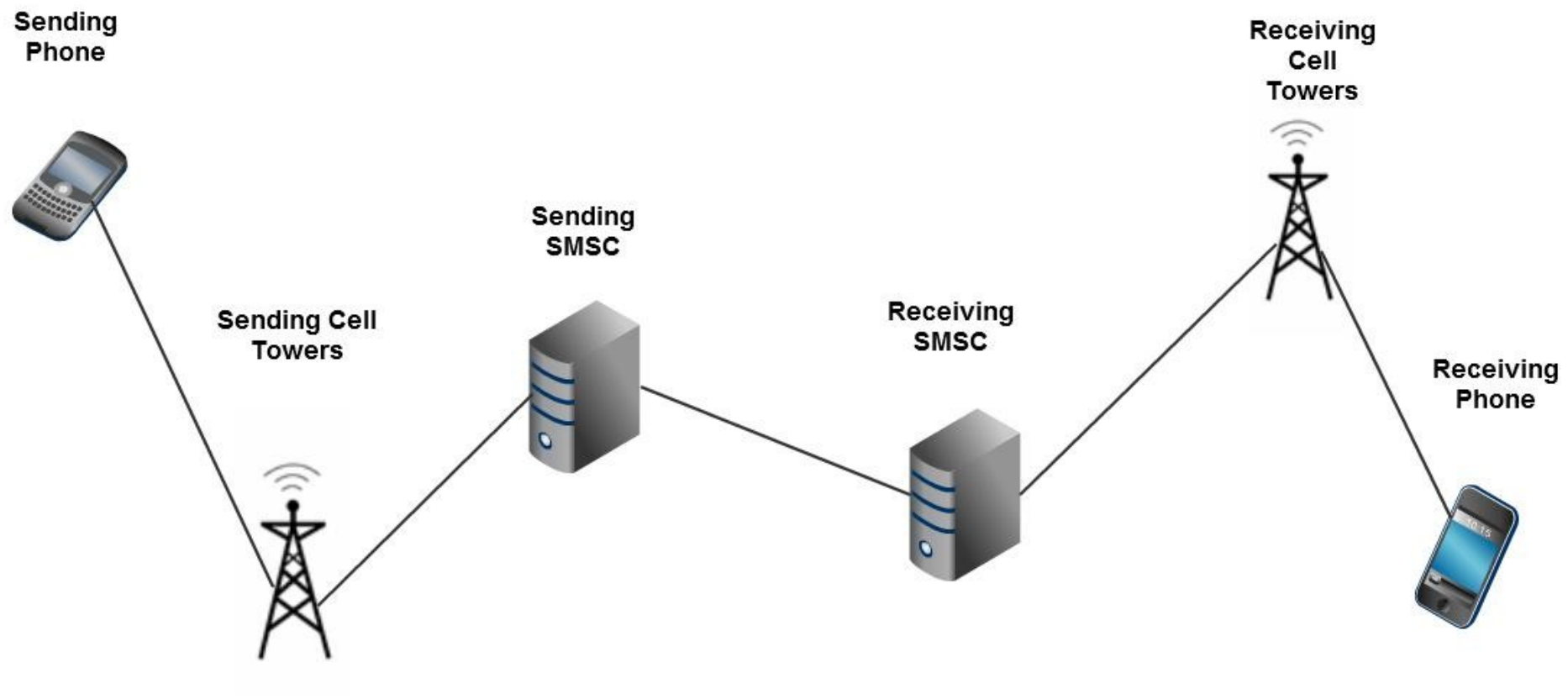
How an SMS is sent and received



How an SMS is sent and received

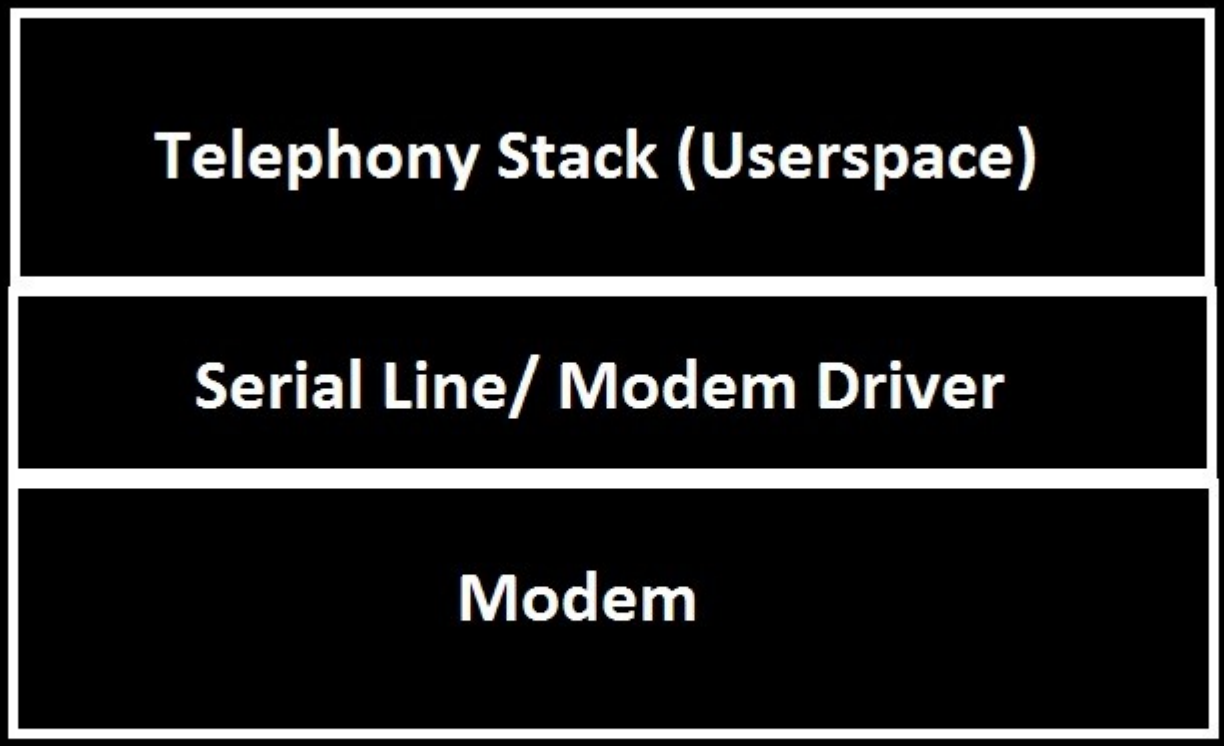


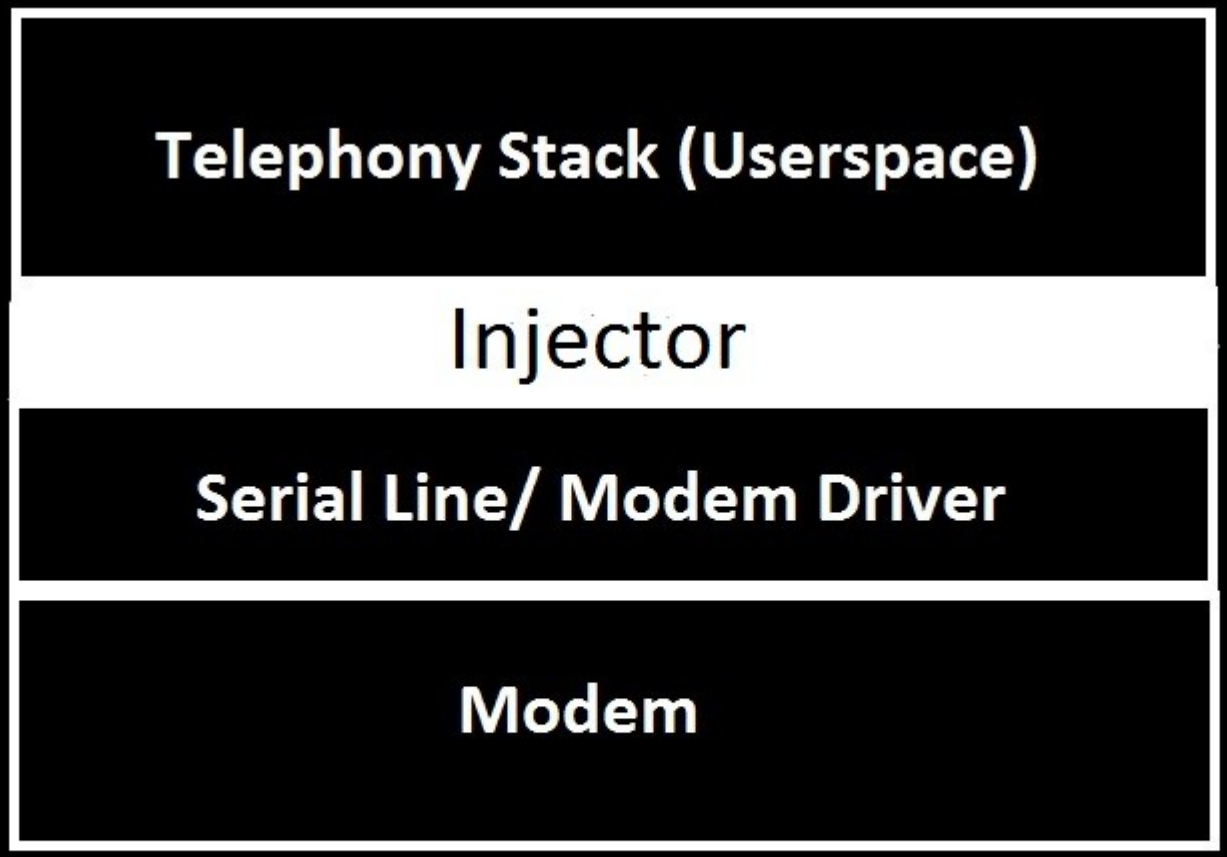
How an SMS is sent and received

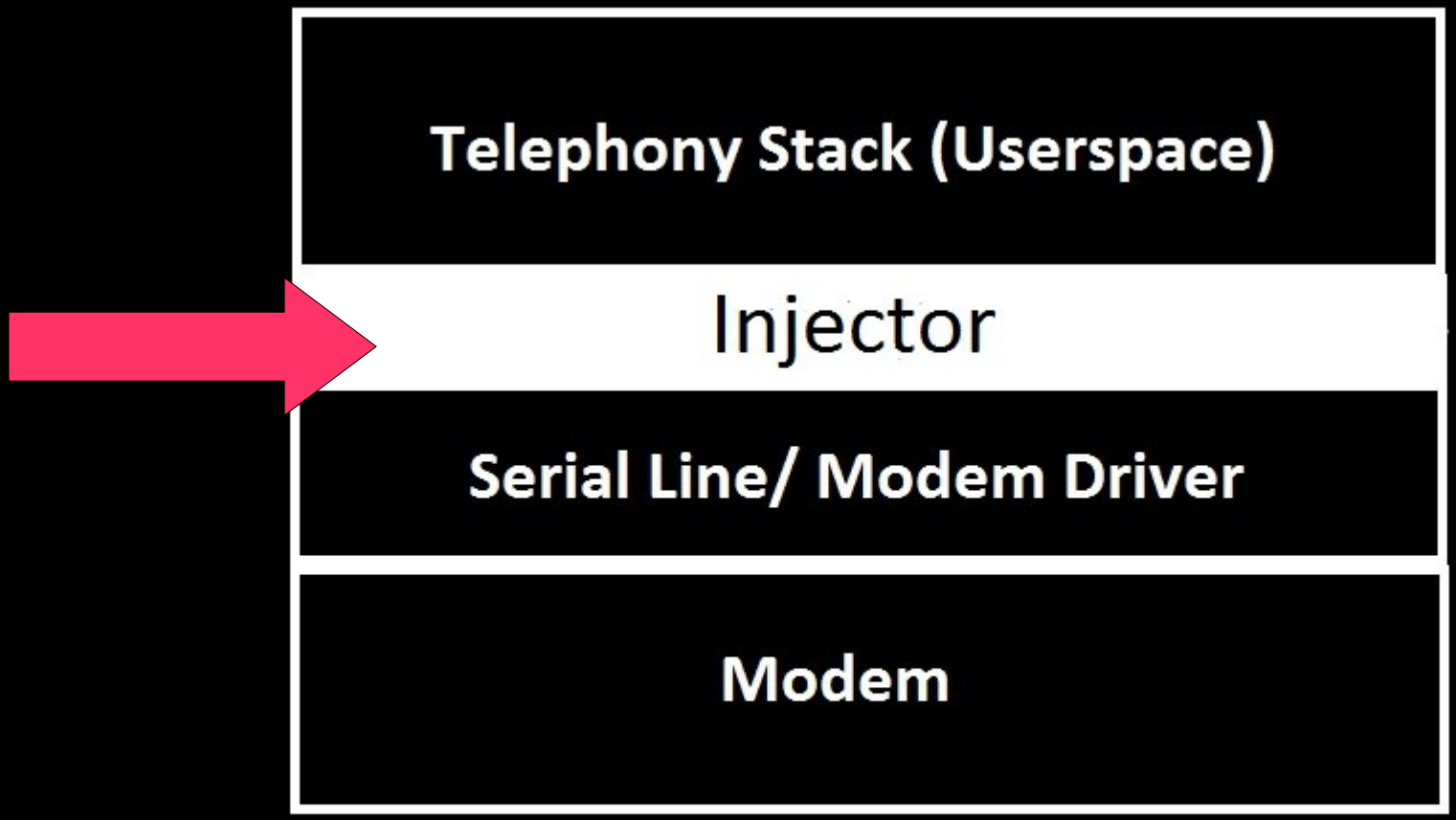


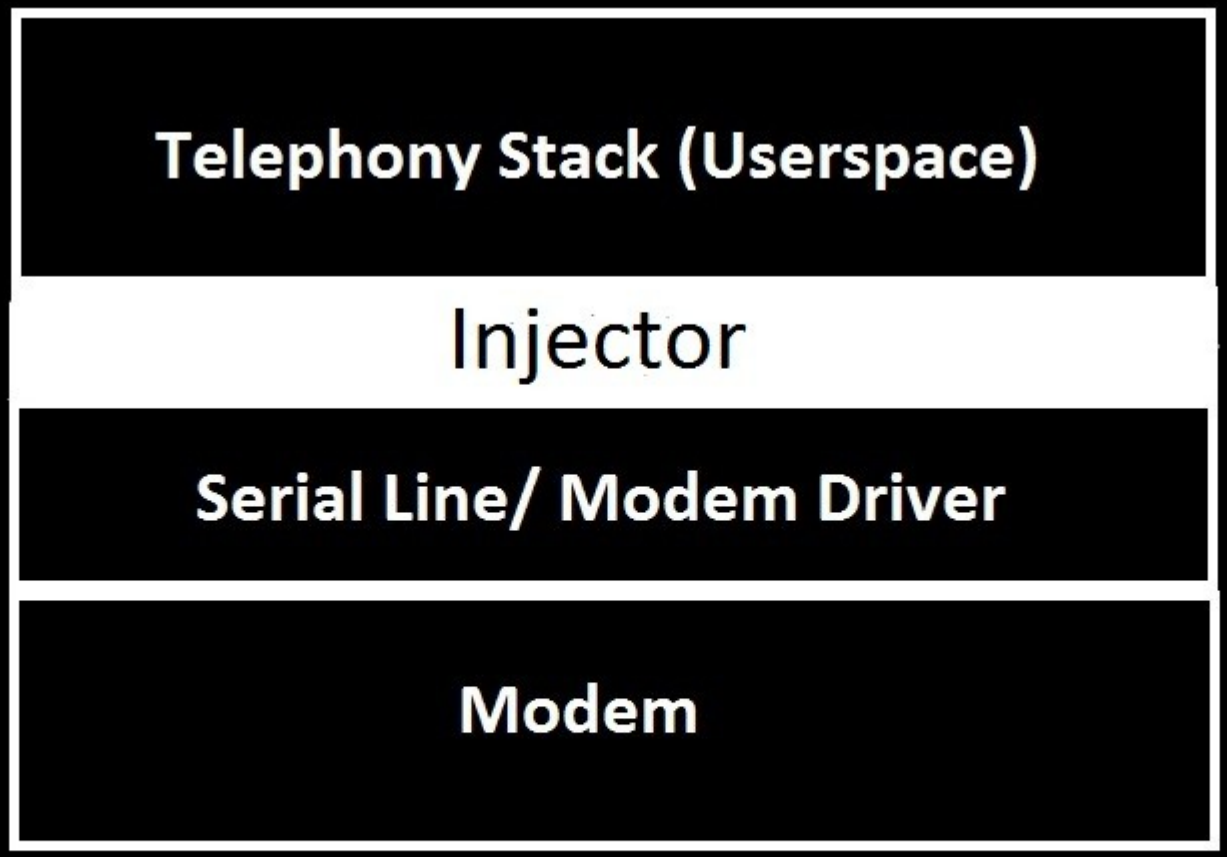
Previous Work: SMS Fuzzing

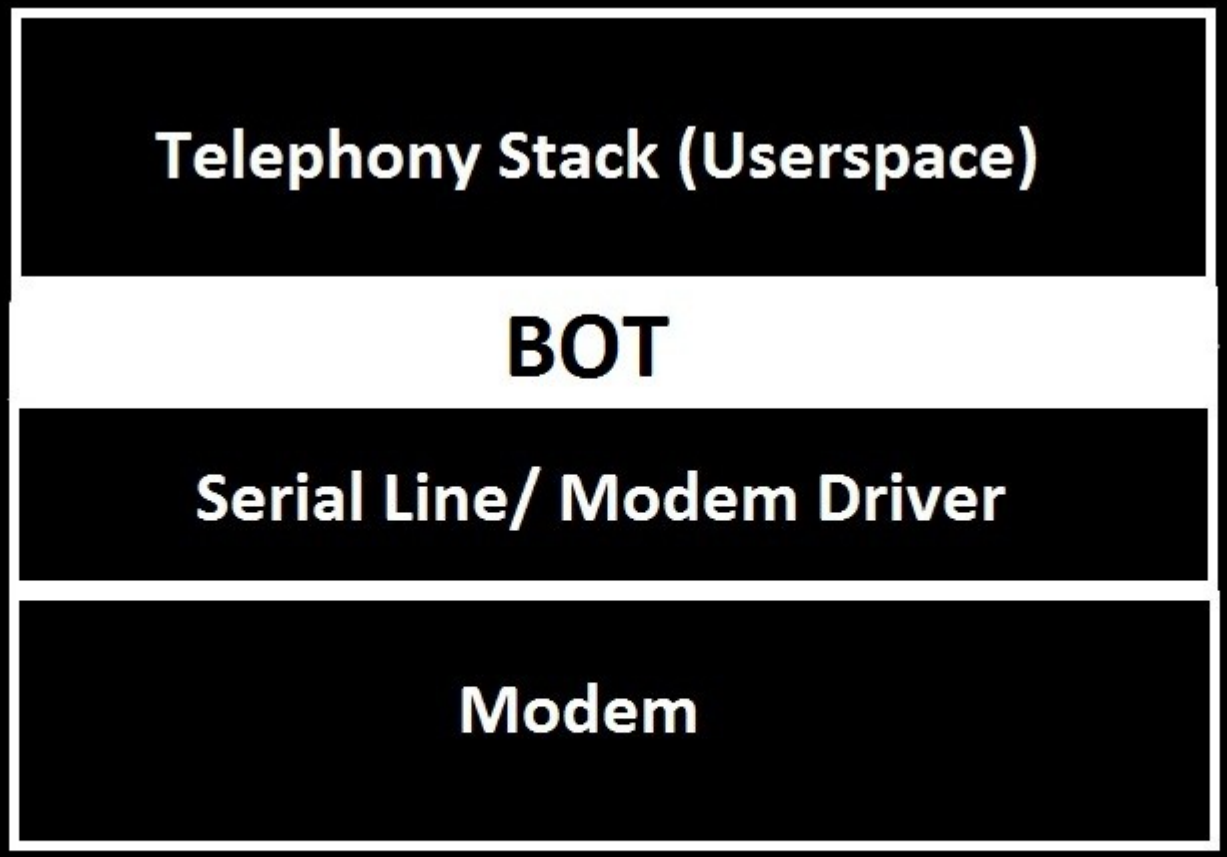
At Blackhat 2009, Charlie Miller & Collin Mulliner proxied the application layer and modem to crash smartphones with SMS.

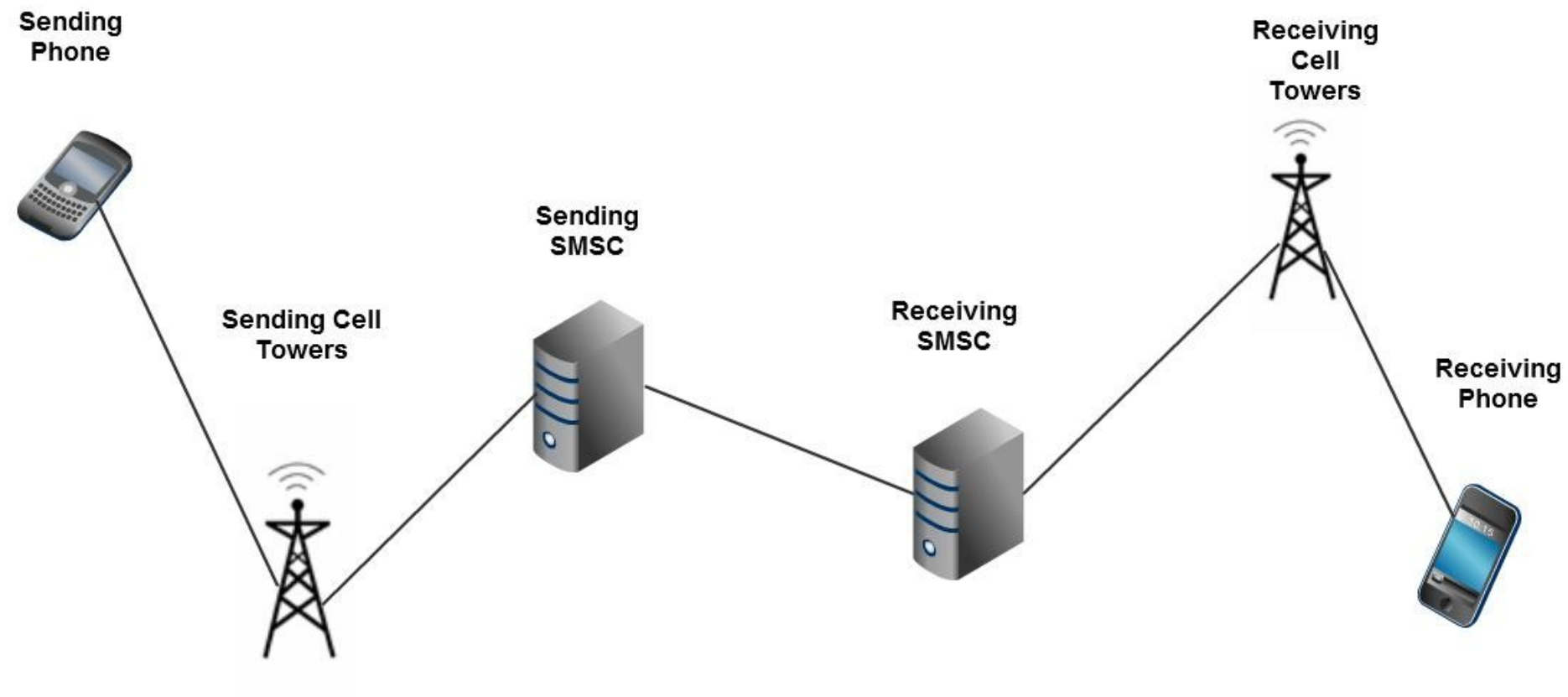

Telephony Stack (Userspace)
Serial Line/ Modem Driver
Modem

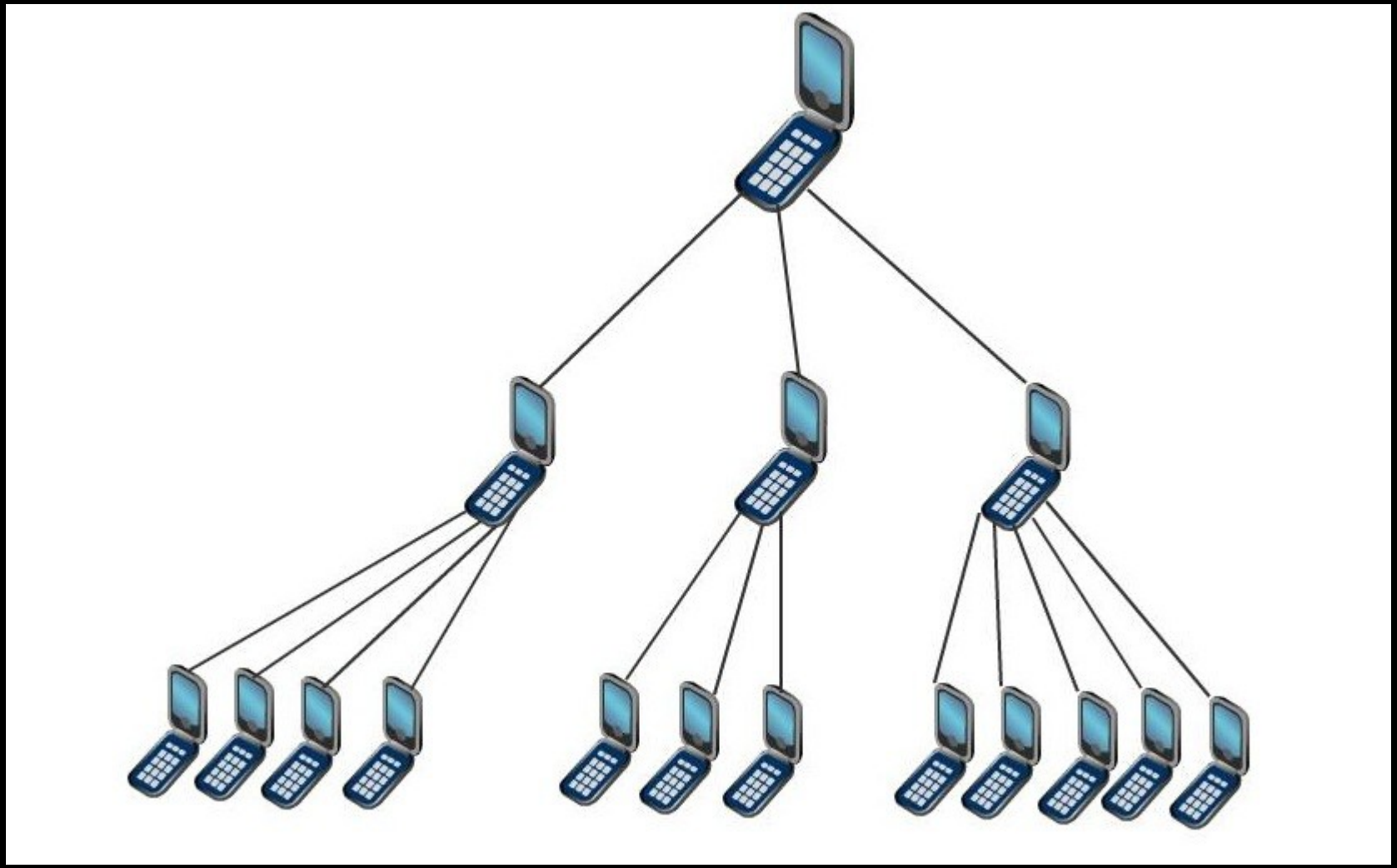

Telephony Stack (Userspace)
Injector
Serial Line/ Modem Driver
Modem


Telephony Stack (Userspace)
Injector
Serial Line/ Modem Driver
Modem

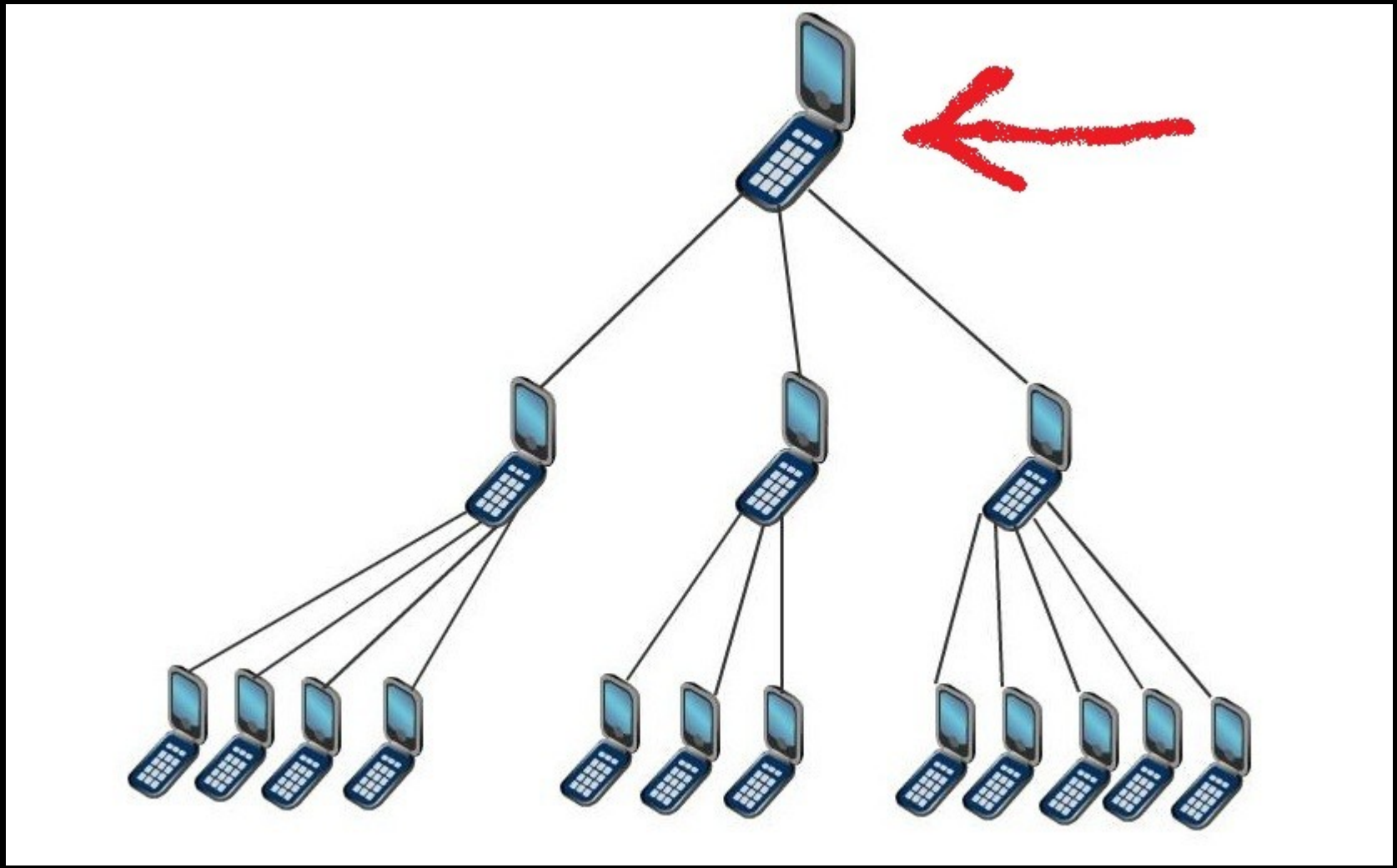

Telephony Stack (Userspace)
Injector
Serial Line/ Modem Driver
Modem


Telephony Stack (Userspace)
BOT
Serial Line/ Modem Driver
Modem


Sending Phone
Sending Cell Towers
Sending SMSC
Receiving SMSC
Receiving Cell Towers
Receiving Phone



Master Bot



Master Bot

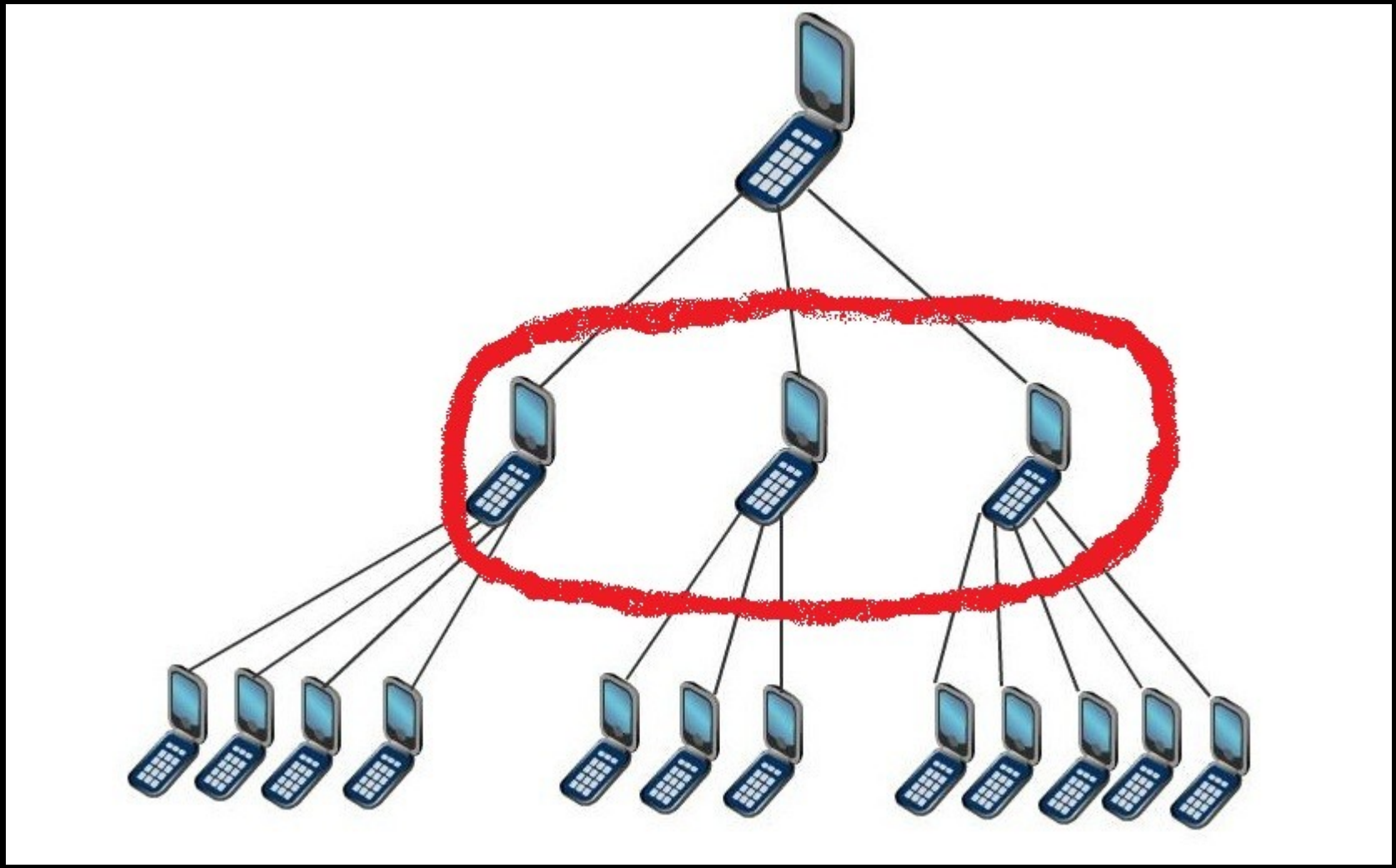
Handled by botherders

Switched out regularly to avoid detection
Prepay SIM Cards + Kleptomania

Sends instructions to Sentinel Bots

In charge of bot structure

Sentinel Bots



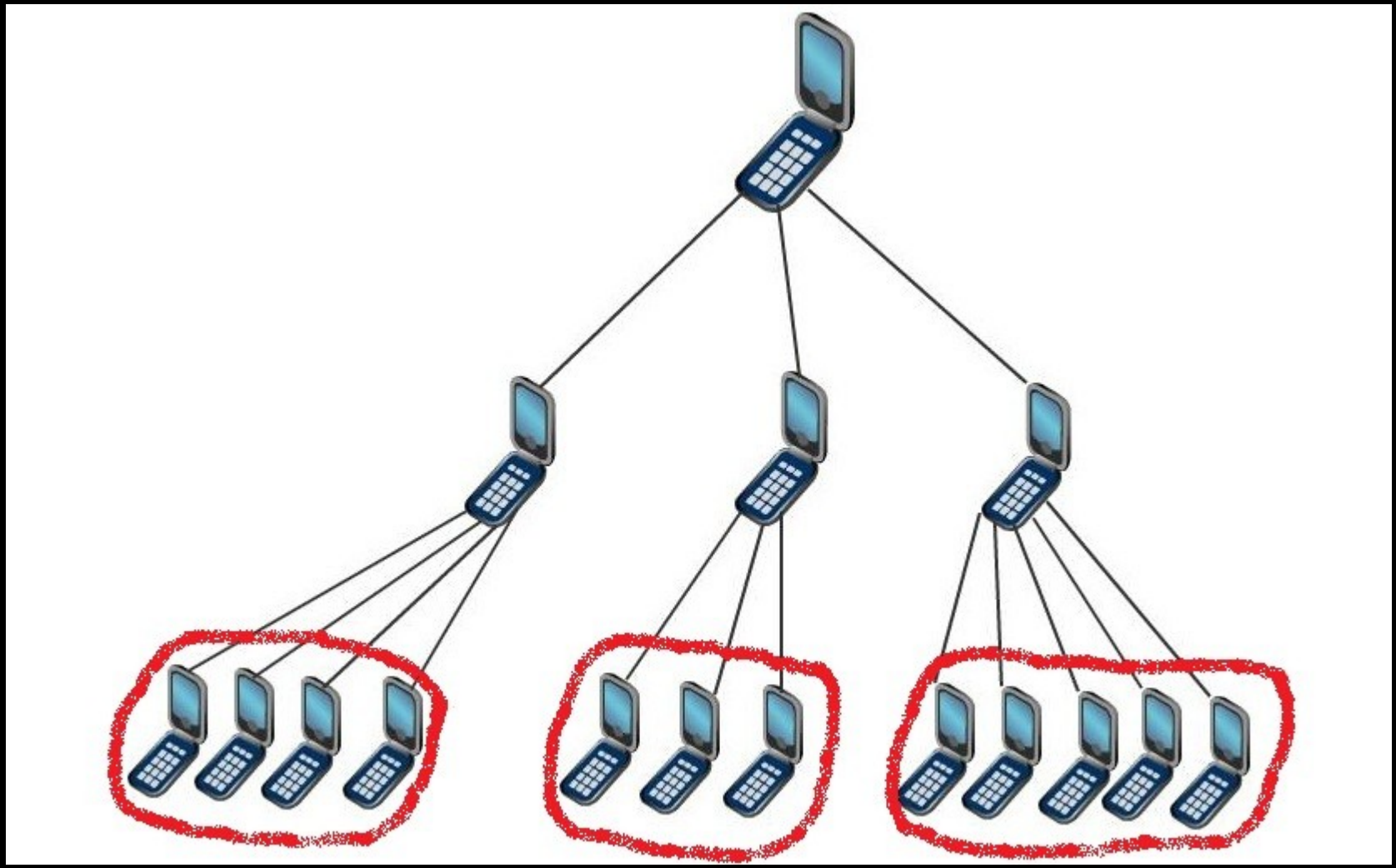
Sentinel Bots

Several “trustworthy” long infected bots

Receive instructions from master bot

Pass on instructions to a set of slave bots

Slave Bots



Slave Bots

Receive instructions from sentinel bots

Carry out botnet payload functionality (DDOS, SPAM, etc.)

Robustness

Master Bot:

- May change device, platform, SIM at will
- Prepaid phones are difficult to track
- Has knowledge of all active bots

Sentinel Bots:

- Reserved for long time bots
- The only bots that interact directly with the master
- Master may promote any slave when needed

Slave Bots:

- A compromise results in at most finding the identity of a single sentinel

Security Concerns

Impersonation:

- Use cryptographic keys to authenticate master bot and sentinel bots

Replay:

- SMS timestamps
- Sequence number
- One time keys

SMS-Deliver PDU

07914140540510F1040B915117344588F1000
0012103714004 4A0AE8329BFD4697D9EC37

Field	Value
Length of SMSC	07
Type of Address (SMSC)	91
Service Center Address (SMSC)	41 40 54 05 10 F1
SMS Deliver Info	04
Length of Sender Number	0B
Type of Sender Number	91
Sender Number	51 17 34 45 88 F1
Protocol Identifier	00
Data Coding Scheme	00
Time Stamp	01 21 03 71 40 04 4A
User Data Length	0A
User Data	E8 32 9B FD 46 97 D9 EC 37

SMS-Deliver PDU

07914140540510F1040B915117344588F1000
0012103714004 4A0A**E8329BFD4697D9EC37**

Field	Value
Length of SMSC	07
Type of Address (SMSC)	91
Service Center Address (SMSC)	41 40 54 05 10 F1
SMS Deliver Info	04
Length of Sender Number	0B
Type of Sender Number	91
Sender Number	51 17 34 45 88 F1
Protocol Identifier	00
Data Coding Scheme	00
Time Stamp	01 21 03 71 40 04 4A
User Data Length	0A
User Data	**E8 32 9B FD 46 97 D9 EC 37**

How It Works

1. Bot Receives Message
2. Bot Decodes User Data
3. Bot Checks for Bot Key
4. Bot Performs Payload Functionality

How It Works

1. Bot Receives Message
   - Bot receives all communication from modem
   - If SMS (code CMT) continue analysis
   - If not SMS pass up to user space
2. Bot Decodes User Data
3. Bot Checks for Bot Key
4. Bot Performs Payload Functionality

How It Works

1. Bot Receives Message
2. Bot Decodes User Data
    - Moves through PDU to User Data
    - Decode 7 bit GSM to plaintext
3. Bot Checks for Bot Key
4. Bot Performs Payload Functionality

How It Works

1. Bot Receives Message

2. Bot Decodes User Data

3. Bot Checks for Bot Key
   - Bot checks for secret key in message
   - If bot message continue analysis
   - If not bot message pass to user space
4. Bot Performs Payload Functionality

How It Works

1. Bot Receives Message
2. Bot Decodes User Data
3. Bot Checks for Bot Key
4. Bot Performs Payload Functionality
    - Bot reads functionality request in message
    - If found perform functionality
    - If not found fail silently

Getting The Bot Installed

Regular Users:

App + Local Root Exploit (Sendpage etc.)
Example: John Oberheide's Twilight
Android Botnet Defcon Skytalks 2010

Root-level/Jailbroken Users:

Root level app using proxy function for AWESOME + Bot

Remote:

Remote root exploit (rooted and nonrooted)
Example: iKee-B “Duh” Worm for iPhone

Example Payloads

- Spam
  - Creating SMS-Send PDUs and passing them to the modem
  - Example: SMS ads
- DDOS
  - Millions of smartphones vs. a server
- Loading New Functionality
  - Send URL in payload
  - Download the module into known payloads

Parallel Research: iPhone Base Code

Rise of the iBots: Owning a Telco Network
Collin Mulliner and Jean-Pierre Seifert

SMS and P2P smartphone botnets

DEMO :)

Android Bot with SMS Spam Payload

Released code has the bot without payloads (have fun)

Thanks

To Mom for helping me master character arrays in c.

Contact

Georgia Weidman
Email: Georgia@grmn00bs.com
Website: http://www.grmn00bs.com