Mobile self-defense

Karsten Nohl <nohl@srlabs.de>
Agenda

SS7 attacks

- 3G security
- Self-defense options
SS7 network enables exchange of SMS and cryptographic keys

SS7 is used between operators... and network-internally
Tracking over SS7 has become commonplace.

Phone number ➔ Subscriber location (Cell ID)

AnytimeInterrogation

For sale: Systems that can secretly track where cellphone users go around the globe
Tracking can happen using many more signaling messages

Phone number

- AnytimeInterrogation
- AnytimeModification

Subscriber location (Cell ID)

- SRI/-SM/-LCS
- AnytimeSubscription-Interrogation

SendIMSI

Brute-force all MSCs

Impersonate HLR towards MSC:
- PSI
- PSL

IMSI & MSC

IMSI

MSC
SS7 enables mobile abuse on five frontiers

**Attacker objective**

A Tracking
- Find subscriber’s whereabouts

B Intercept
- Listen to calls, read short messages, intercept Internet traffic

C DoS
- Interfere with user connectivity or network availability

D Fraud
- Make illegitimate calls/send SMS; disable usage limits

E Spam
- Send unsolicited messages
2G + 3G transactions can be decrypted with help of SS7
SS7 enables 3G IMSI Catcher

Here is my identity (IMSI), now prove that you are the real network

3G Fake Base Station (“IMSI catcher”)

I. Prove your authenticity
II. Request key
III. Sends auth. proof

Global SS7

Mobile operator
Rerouting attacks over SS7 allow for remote intercept

<table>
<thead>
<tr>
<th>SS7 Man-in-the-middle attacks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Capture incoming calls</strong></td>
</tr>
<tr>
<td>Demo</td>
</tr>
<tr>
<td>Attacker activates call forwar-ding over SS7 for target number</td>
</tr>
<tr>
<td>- When a call is received, the attacker forwards it back to the original number</td>
</tr>
<tr>
<td><strong>Capture outgoing calls</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>- Attacker adds a number rewriting rule for dialed numbers</td>
</tr>
<tr>
<td>- Called numbers are rewritten to reach attacker and are then forwarded to intended recipient</td>
</tr>
</tbody>
</table>
Not all SS7 attacks can simply be blocked

<table>
<thead>
<tr>
<th>Abuse scenario</th>
<th>Offending SS7 message</th>
<th>Mitigation effort</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Local passive intercept</td>
<td>- SendIdentification</td>
<td>Easy – Block message at network boundary</td>
</tr>
<tr>
<td>2 IMSI Catcher</td>
<td>- SendAuthenticationInfo</td>
<td>More complex – Messages are required for operations, need to be plausibility-checked</td>
</tr>
<tr>
<td>3 Rerouting attacks</td>
<td>- SS_activate/register, UpdateLocation, Camel messages, (Probably others)</td>
<td></td>
</tr>
<tr>
<td>No.</td>
<td>Time</td>
<td>Source</td>
</tr>
<tr>
<td>-----</td>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>4</td>
<td>20:42:09</td>
<td>127.0.0.1</td>
</tr>
<tr>
<td>5</td>
<td>20:42:09</td>
<td>127.0.0.1</td>
</tr>
<tr>
<td>6</td>
<td>20:42:46</td>
<td>127.0.0.1</td>
</tr>
<tr>
<td>7</td>
<td>20:42:46</td>
<td>127.0.0.1</td>
</tr>
<tr>
<td>8</td>
<td>20:42:48</td>
<td>127.0.0.1</td>
</tr>
<tr>
<td>9</td>
<td>20:42:48</td>
<td>127.0.0.1</td>
</tr>
</tbody>
</table>

- **TP-Service-Centre-Time-Stamp**
  - Year: 14
  - Month: 12
  - Day: 18
  - Hour: 17
  - Minutes: 50
  - Seconds: 38
  - Timezone: GMT + 1 hours 0 minutes

- **TP-User-Data-Length**: (26) depends on Data-Coding-Scheme
  - **TP-User-Data**

**SMS text**: This is your new PIN: 31C3

0000 09 01 37 01 07 91 94 71 06 00 40 34 00 2b 24 .7......q..@4.+s
0010 0d 91 94 71 56 73 20 40 f8 00 00 41 21 81 71 05 ...qVs@...A!.q.
0020 83 40 1a 54 74 7a 0e 4a bb 41 f9 77 5d 0e 72 97 @.Itz.J .A.w].r.
0030 ef 20 68 d2 a9 03 cd 62 c3 19 .h....b..
Agenda

- SS7 attacks
- **3G security**
- Self-defense options
Remember? Intercepting GSM A5/1 calls and SMS is cheap

- A reprogrammed EUR 20 phone captures 2G calls and SMS
- Multiple such phones could be clustered for wide-scale intercept

Intercept:
GSM call

Crack:
A5/1 key

Standard server cracks key in seconds
Intercepting 3G is also surprisingly cheap, thanks to SS7

- Software-defined radio captures 3G transactions
- We use: BladeRF – USD 420
- Development took 3 months

SS7 query SendIdentification provides decryption key
- Also works for GSM A5/3
Some networks are so poorly configured that SS7 is not even needed to intercept their 3G transactions

<table>
<thead>
<tr>
<th>Network</th>
<th>Encrypts</th>
<th>Authenticates calls / SMS</th>
<th>Protects integrity</th>
</tr>
</thead>
<tbody>
<tr>
<td>🇰cesso</td>
<td>✗</td>
<td>✗</td>
<td>✔</td>
</tr>
<tr>
<td>🇮🇳India</td>
<td>✗</td>
<td>✗</td>
<td>✔</td>
</tr>
<tr>
<td>🇰🇷Korea</td>
<td>✗</td>
<td>✗</td>
<td>✔</td>
</tr>
<tr>
<td>🇸🇰Korea</td>
<td>✗</td>
<td>✗</td>
<td>✔</td>
</tr>
<tr>
<td>🇷🇸Slovenia</td>
<td>✗</td>
<td>✗</td>
<td>✔</td>
</tr>
</tbody>
</table>

**Risk** – Calls, SMS, and Internet traffic on these networks can be intercepted passively with a programmable radio (but without SS7)
Protection status of 3G networks is tracked in online tool

gsmmap.org network security comparison

German networks encrypt 3G, but do not all change TMSIs
Networks without USIMs are vulnerable to brute-force attacks

<table>
<thead>
<tr>
<th>Encryption</th>
<th>SIM</th>
<th>USIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSM</td>
<td>A5/3 64 bit 64 bit</td>
<td>NSA-vulnerable</td>
</tr>
<tr>
<td></td>
<td>A5/4 64 bit 128 bit</td>
<td>Not brute-forceable</td>
</tr>
<tr>
<td>UMTS</td>
<td>UEA/1 or 2 64 bit 128 bit</td>
<td>Not brute-forceable</td>
</tr>
</tbody>
</table>

NSA apparently broke 64-bit A5/3

Source – The intercept: wolframite-encryption-attack.pdf
Agenda

- SS7 attacks
- 3G security

Self-defense options
Many mobile network abuse scenarios can be detected

<table>
<thead>
<tr>
<th>Attack scenario</th>
<th>Detection heuristic</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SMS Attacks</strong></td>
<td>SIM OTA attacks</td>
</tr>
<tr>
<td></td>
<td>Silent SMS</td>
</tr>
<tr>
<td><strong>SS7 Attacks</strong></td>
<td>SS7 abuse: Tracking</td>
</tr>
<tr>
<td></td>
<td>Empty paging</td>
</tr>
<tr>
<td><strong>IMSI Catcher</strong></td>
<td>Tracking or Intercept</td>
</tr>
<tr>
<td></td>
<td>Unusual cell config</td>
</tr>
<tr>
<td></td>
<td>Unusual cell config</td>
</tr>
<tr>
<td><strong>Network Security</strong></td>
<td>Insufficient encryption</td>
</tr>
<tr>
<td></td>
<td>Encryption level</td>
</tr>
<tr>
<td></td>
<td>Encryption level</td>
</tr>
<tr>
<td></td>
<td>Lack of TMSI updates</td>
</tr>
<tr>
<td></td>
<td>TMSI update frequency</td>
</tr>
</tbody>
</table>
New tool detects common abuse scenarios

**SnoopSnitch**

*App ID: 25ea434b*

**Last analysis:** Dec 22, 2014 4:35:09 PM

**Tool name**

**SnoopSnitch**

**Purpose**

- Collect network traces on Android phone and analyze for abuse
- Optionally, upload to GSMmap for further analysis

**Requirements**

- Android 4.1 or newer
- Rooted, (but no CyanogenMod)
- Qualcomm chipset: Samsung S5/S4/S3 Neo, Sony Z1, LG G2, Moto E, and many more

**Source**

Google Play: Search for *SnoopSnitch*
IMSI catcher detection analyzes a cell’s configuration and behavior

SnoopSnitch combines a number of IMSI Catcher heuristics

Suspicious cell configuration
- Encryption downgrade / no encryption
- High cell reselect offset
- Large number of paging groups
- Low registration timer

Suspicious cell behavior
- Delayed *Cipher Mode Complete* acknowledgement
- *Cipher Mode Complete* message without IMEISV
- ID requests during location update
- Paging without transaction
- Orphaned traffic channel

A number of other rules could not be implemented based on data available from Qualcomm chipsets. (Future work?)
SnoopSnitch collects data in the background and on request.

Directed attacks are constantly analyzed in a background process.

Network tests are uploaded only on demand.

Alerts can be shared for further analysis.
It’s now on you to contribute data and progress the toolbox of self-defense apps

**Mobile self-defense strategy**

1. Check your network operator on [gsmmap.org](http://gsmmap.org) for vulnerabilities; possibly switch to a more secure operator

2. Install **SnoopSnitch** from Google Play (needs Android 4.1+, Qualcomm chipset, root, but no custom ROM)

3. Conduct a network test and **upload any attack alarms** (SMS, SS7, IMSI catcher) for further analysis

4. Contribute to the SnoopSnitch code or use the source to build your own application based on raw 2G/3G/4G data
Thank you!

Research supported by OPEN TECHNOLOGY FUND

Many thanks to Alex Senier, Luca Melette, Lukas Kuzmiak, Linus Neumann, Jakob Lell, and dexter for making this release possible!

Questions?

Karsten Nohl <nohl@srlabs.de>