GPRS Intercept: Wardriving your country

Karsten Nohl, nohl@srlabs.de
Luca Melette, luca@srlabs.de
Executive summary – Do not send sensitive data over GPRS

- GPRS/EDGE networks provide the data backbone of smart phones and industry automation systems
- The cryptographic protection of GPRS/EDGE is out-dated and vulnerable to several attacks
  - Lack of mutual authentication allows for ‘fake base stations’ to harvest data
  - Lack of encryption (some countries) allows for passive intercept with EUR10 phone and software released during this talk
  - Weak encryption (remaining countries) enables cryptanalysis,
- Ever more applications are building up on mobile data networks, thereby amplifying the exposed risks instead of mitigating them
Agenda

- GPRS basics
  - Practical GPRS attacks
  - Mitigation measures
GPRS provides the communication backbone for mobile societies

- **Industry automation**
- **Mobility management**
- **Mobile phones, Pads, PCs**
- **Smart grid**
GPRS can encrypt data packets

Layer 3 – Data packets of typically 1,520 bytes are exchanged with backend. Encryption should prevent intercept over-the-air and on transport links.

Layer 1/2 – GPRS/EDGE share channels with GSM and only differ in the modulation and multiplexing.
GPRS support different encryption levels, but predominantly the weak ones are used

<table>
<thead>
<tr>
<th>Protection function</th>
<th>Encryption</th>
<th>Key length</th>
<th>Used by</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEA/0</td>
<td>No encryption</td>
<td>N/A</td>
<td>Anybody?</td>
</tr>
<tr>
<td>GEA/1</td>
<td>Proprietary stream cipher (96 bit state)</td>
<td>64 bit</td>
<td>Most operators use both GEA/1 and GEA/2</td>
</tr>
<tr>
<td>GEA/2</td>
<td>Proprietary stream cipher (125 bit state)</td>
<td>64 bit</td>
<td>Some, mostly newer networks</td>
</tr>
<tr>
<td>GEA/3</td>
<td>Standard block cipher (128 bit state)</td>
<td>64 bit</td>
<td>Nobody</td>
</tr>
<tr>
<td>GEA/4</td>
<td></td>
<td>128 bit</td>
<td></td>
</tr>
</tbody>
</table>
Agenda

- GPRS basics
- Practical GPRS attacks
- Mitigation measures
GPRS networks are valuable to multiple attacks

<table>
<thead>
<tr>
<th>Attacks</th>
<th>Passive</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEA/0</td>
<td>Simple intercept</td>
<td>Fake base station</td>
</tr>
<tr>
<td>GEA/1</td>
<td>Crypt-analysis</td>
<td>Rainbow tables (where IV is constant)</td>
</tr>
<tr>
<td>GEA/2</td>
<td></td>
<td>Phones downgrade to GEA/0 when asked by (fake) base station</td>
</tr>
<tr>
<td>GEA/3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Covered in this talks

Covered in 27C3, 28C3 talks
GPRS interception only requires open source tools

Function

Capture bursts

Layer 2 parsing

Layer 3 parsing

Implemented Adaptations

1. Start with Sylvain’s burst_ind branch
2. Pimp the USB cable
3. Add multi-time-slot support
4. Multiplex data from multiple phones
5. Channel decoding
6. RLC parsing (block defrag)
7. LLC parsing (more block defrag)
8. Optional – Native RLC / LLC decoder

Osmocom BB → Raw data → GPRS decode → gsmtap → Wireshark
GPRS decode consists of 16 decoding chains

GPRS decode starts with raw data obtained from Osmocom BB. The data is then processed through a series of stages:

1. **Frame FIFO** for TS0 and 2x (UL, DL) TS1...TS7.
2. **Channel decoding** with 4 modes.
3. **RLC parsing**.
4. **TBF queues + block defrag**.
5. **LLC message**.

The processed data is then fed into Wireshark via gsmtap.
GPRS “overcapsulates”

- Frame FIFO
- Channel decoding (4 modes)
- RLC parsing
- TBF queues + block defrag
- LLC message

Example

Demux of 32 identifiers in 2 directions

BSN
0 1 2 3 4 5 ...

BSN 3  BSN 4  BSN 5
LLC 0  LLC 1  LLC 2  LLC 3  LLC 4  Padding

[63 x the same decode chain]
Some GPRS networks do not use any encryption

Supposedly encryption hinders in-line data monitoring. Hence some commercial networks use GEA/0—no encryption!

Now off to some actual cryptanalysis on all the other networks …
GEA/1 mostly mitigates A5/1’s rainbow table attacks but opens new crypto holes

<table>
<thead>
<tr>
<th></th>
<th>A5/1</th>
<th>GEA/1</th>
<th>Relevant for</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key size</td>
<td>64bit</td>
<td>64bit</td>
<td>Brute force/(TMTO)</td>
</tr>
<tr>
<td>Internal state</td>
<td>64bit</td>
<td>96bit</td>
<td>TMTO</td>
</tr>
<tr>
<td>LFRSs</td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Output nonlinearity</td>
<td>degree 1</td>
<td>degree 4</td>
<td>Algebraic attacks</td>
</tr>
<tr>
<td>Non-linear update</td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Output</td>
<td>114bit</td>
<td>up to 1500 bytes</td>
<td></td>
</tr>
</tbody>
</table>
GPRS lacks good non-linearity
Agenda

- GPRS basics
- Practical GPRS attacks
- Mitigation measures
Not securing mobile data would be negligent

GPRS is here to stay

Securing GPRS requires actions from networks and application authors

Mobile data traffic* [TB/month]

- **Short term mitigation:** Application must protect themselves
- **Mid/long term need:** Networks must upgrade encryption

*Source: Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update (February 1, 2011)
Mobile applications should start using internet-grade encryption

Example – iPhone applications:

- Mail
- Most critical applications
- 20% of tested messaging apps
- Many mobile web sites

GPRS / Internet

- Some mobile application and most mobile web sites send data unencrypted over GPRS
- SSL, proudly used on the internet since 1994, could easily protect all this data

*Some iOS versions use vulnerable SSL implementations that can be abused in a fake base station attack
GPRS network wish list – Continuous improvements

Immediately – Switch on encryption

Mid term – Add mutual authentication

Long term – Upgrade to USIM + 128bit GEA/4

Mobile data finally secure against today's threats

1. Deploy Java applet to SIM card
2. Execute mutual authentication from Java Applet before generating GPRS key
3. Use GEA/3 to secure connection

Network operator
GPRS currently is a risk to mobile societies

| Risk: The level of protection widely differs among networks but is typically outdated. |
| Mitigation: Protect applications through SSL and start demanding better protection from your operator |

**Osmocom GPRS sniffing tutorial:** srlabs.de/gprs

**Questions?**
Karsten Nohl, nohl@srlabs.de
Luca Melette, luca@srlabs.de

Lots of thanks to Mate Soos, Dieter Spaar, Harald Welte, Sylvain Munaut and Dexter