Improving 112 Location by Reducing Android ELS Error Rate

May 25, 2021

Email: android-emergency-location@google.com
Web site: crisisresponse.google/els
Agenda

- ELS errors and error rates
- Troubleshooting ELS errors
- One ELS partner’s perspective
- Case studies
ELS Errors and Error Rate: Cause & Effect
ELS errors/error rate: what it is, why it happens

Two reasons for ELS errors (failures):

1. No location computed in the message sent to the Endpoint
2. Location computed and transmitted from device but
   ○ ELS message not received by Endpoint
   or
   ○ No network acknowledgement within 30 secs (Data SMS)*

ELS ERROR RATE =

# of Failed ELS messages / Total # of ELS Messages

* HTTP timeout = 10 secs, considered an ELS error if we don't get a 200-299 response
Error rates can vary across and within countries

Country 1

Country 2

Country 1

Country 2
Common root causes behind errors

1. **Network/Hardware/Software Policies**
   - Some MNOs, SMS infra vendors or device OEMs have configured their systems/devices to hold all SMSes:
     - i. During an emergency call
     - ii. When device is emergency camped (roaming/in SOS mode)

2. **Circuit Switched Fallback (CSFB)**
   - When an LTE network is not available to make/receive a 112 call or SMS message, the device “falls back” to a more accessible 3G/2G network to finish the call or send the SMS message

3. **SMS-over-IP on VoLTE devices**
   - Some LTE phones don’t support SMSoIP > unable to transmit ELS message(s)

* Issue specific to 1 MNO, may be an edge case
Quick & happy fix example (your mileage may vary)
Troubleshooting ELS Errors
ELS data flow primer

Emergency call initiated by Android device

ELS data sent directly to endpoint as Data SMS AML or HTTPS message

Endpoint is set up and managed by ELS Partner, who is responsible for making ELS data available to Emergency Services (push or pull).

Google’s Responsibility

Partner’s Responsibility

AML - Advanced Mobile Location, open standard for sending emergency location (supported by Android ELS)

ELS Endpoint: a SMSc or HTTPS server maintained by partner that can receive ELS emergency location data

ELS Partner: carrier/MNO, government or public safety vendor that meets ELS partner requirements

PSAP/ECC (Public Safety Answering Point/Emergency Communications Center): call center & dispatch control for emergency services
ELS data flow value chain for Data SMS
No/Limited E2E visibility for all parties (esp Google!)

Mobile Network Operator

Google

ELS Partner

SMS Aggregation Service

Location Database

ELS Endpoint

SMS Center in each network

Voice call

ECC/PSAP
Troubleshooting requires collaboration among all Parties in the ELS data value Chain ...

... driven by the ELS Partner
Troubleshooting 101

**What:** Dedicated debug session with MNO’s network elements engineer(s), ELS Partner and PSAP call taker(s)

**How:** Make 50-100 emergency calls using the ELS Manager app and a SIM from the MNO

**Why:** Send & trace the progress of ELS test messages through the mobile network to see if the messages are delivered successfully to/from the Endpoint. If not, understand *why* and *where in the process/network issues occur* or the ELS message fails.
# Elements for an effective troubleshooting session

<table>
<thead>
<tr>
<th>Role</th>
<th>Tools</th>
<th>Key Tasks</th>
<th>Google</th>
<th>ELS Partner</th>
<th>MNO</th>
<th>PSAP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Review bug reports of failed messages</td>
<td>Manage project, coordinate across all parties</td>
<td>Track &amp; interpret incoming/outgoing ELS messages</td>
<td>Receive ELS test calls &amp; messages</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Google</td>
<td>ELS Partner</td>
<td>MNO</td>
<td>PSAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ELS Manager app + Android device in eng/userdebug mode</td>
<td>Live probes monitoring of Data SMS network element paths</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ELS Manager app + Android device in eng/userdebug mode</td>
<td>Live probes monitoring of Data SMS network element paths</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Monitor status of test calls (successful/failure reception)</td>
<td>Capture and share network protocol exchanges</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Retrieve logs from device using ADB commands</td>
<td>Network elements expertise to interpret suspicious messages</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Get PSAP approval for test calls</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Google**
  - Review bug reports of failed messages

- **ELS Partner**
  - Manage project, coordinate across all parties

- **MNO**
  - Track & interpret incoming/outgoing ELS messages

- **PSAP**
  - Receive ELS test calls & messages

**Key Tasks**

- **Google**
  - Technical consulting re: ELS service & features
  - Analyze bug reports

- **ELS Partner**
  - ELS Manager app + Android device in eng/userdebug mode
  - Monitor status of test calls (successful/failure reception)
  - Retrieve logs from device using ADB commands
  - Get PSAP approval for test calls

- **MNO**
  - Live probes monitoring of Data SMS network element paths
  - Capture and share network protocol exchanges
  - Network elements expertise to interpret suspicious messages

- **PSAP**
  - Awareness/ability to recognize a test call
What we’re looking for during troubleshooting

<table>
<thead>
<tr>
<th>Device / Google</th>
<th>Mobile Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>● Incomplete/error messages not inline with the feature design</td>
<td>● Identify &amp; explain generated network errors (if any)</td>
</tr>
<tr>
<td>● Incoming error messages from network side</td>
<td>● Which network element blocked the incoming/outgoing protocol message(s)</td>
</tr>
<tr>
<td>● Data SMS timings in device logs (to compare with those on the network/Endpoint side)</td>
<td>● Identify delta between <em>Fail</em> and <em>Pass</em> scenarios (in case issue is not 100% reproducible)</td>
</tr>
<tr>
<td>● Call not received by PSAP</td>
<td>● Confirm that complete protocol exchange went as expected per design (by logging and sharing logs with Google team when needed)</td>
</tr>
<tr>
<td></td>
<td>● Timing of data SMS received with respect to the timing the SMS was sent from the device (investigating a buffering issue)</td>
</tr>
<tr>
<td></td>
<td>● Network configuration related to SMS on emergency are well defined</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ELS Partner (Endpoint)</th>
</tr>
</thead>
<tbody>
<tr>
<td>● Number of Data SMS messages received by ELS Endpoint per call</td>
</tr>
<tr>
<td>● Timing of Data SMSes received with respect to the time the SMS was sent from the Device/Mobile Operator network</td>
</tr>
</tbody>
</table>
ELS Partner Perspective: Carbyne