A Game Boy and his cellphone
What is the Mobile Adapter GB?

- Make and receive calls
  - Send/receive data over the phone line
- Call an ISP and connect to the internet
- Associated online game services: Mobile System GB
- Very early attempt at online connectivity by Nintendo
What is the Mobile Adapter GB?

- A few different versions
  - PHS never released
- Service sunset in December 2002
Games

Because a gameless peripheral is like a breadless sandwich
Games: Mobile Trainer

- Bundled, used to configure the adapter
- Usage manual
- Mail client
  - SMTP and POP
  - Contact with friends
- Web browser
  - HTTP, very basic HTML
  - Read news
Games: Pocket Monsters: Crystal Version

- Battling/Trading
- Battle Tower
- Trade Corner
- News Machine
- Events
Games: Net de Get

- Minigame collection
- Custom mapper
- Downloadable games
Games: Various

- Mobile Golf
  - Bundled with adapter later in its life
- Starcom: Star Communicator
- Game Boy Wars
- Mario Kart Advance
Storytime!

Anything's better with a good story
Storytime

- Thread on Glitch City Labs
- Python script
- Shonumi articles and Dan Docs
- Hardware support
• Group of enthusiasts, developers and preservationists
• Recreating the system faithfully
  - Emulators
  - Servers
  - Translations
The system

The system: High-level overview
The system: Emulated

USB setup

Game

P2P Relay
Emulation: Peripheral

• **libmobile**: callback-based library
  - BGB
  - mGBA
  - RP2040 (Raspberry Pi Pico)
  - ATmega (Arduino Uno)

• **GBE+**

• Complete documentation in Dan Docs
Emulation: Peripheral
Emulation: Peripheral
Emulation: Servers

- P2P: Relay server
- Mail server (SMTP, POP)
- Game servers
  - Pokémon Crystal
  - Mario Kart
  - Monopoly EX
Emulation: Servers

- **Game servers**
  - Net de Get @ 100 (GBE+)
  - Game Boy Wars 3 (GBE+)
  - All-Japan GT Championship (GBE+)
  - Hello Kitty’s Happy House (GBE+)
Translations

- Pokémon Crystal
  - All features restored
  - All 5 languages, all features
- Mobile Golf - @marc_max
- Mobile Trainer - @TrEpToR
Development tools

• **Adapter emulation: libmobile**
  - No dependencies, easy to reuse
  - https://github.com/REONTeam/libmobile

• **Server emulation: reon repository**
  - Infrastructure reused for most games
  - https://github.com/REONTeam/reon

• **Client library: MIA**
  - libma, decompilation of nintendo SDK
Conclusion
Links

https://github.com/REONTeam

https://discord.gg/ceaFvwbkSJ

https://shonumi.github.io/
Diving deeper

Exploring the depths unknown
Protocol: Basics

- Bidirectional serial data
- Request/response protocol

<table>
<thead>
<tr>
<th>0x99</th>
<th>0x66</th>
<th>CMD</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEN</td>
<td>DATA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATA</td>
<td>SUM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEV</td>
<td>ACK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Protocol: Command flow

10 Start session: NINTENDO
10 Start session: NINTENDO
19 Read EEPROM (offset: 00; size: 60)
19 Read EEPROM (offset: 00)
... data ...
19 Read EEPROM (offset: 60; size: 60)
19 Read EEPROM (offset: 60)
... data ...
12 Call (prot 0): #9677
12 Call
21 PPP connect (id: g111111111; dns1: 210.196.3.183; dns2: 210.141.112.163)
21 PPP connect (ip: 127.0.0.1; dns1: 0.0.0.0; dns2: 0.0.0.0)
22 PPP disconnect
22 PPP disconnect
13 Disconnect
13 Disconnect
11 End session
11 End session

Connect to ISP

HTTP request

28 DNS request: gameboy.datacenter.ne.jp
28 DNS request: 127.0.0.1
23 TCP connect: 127.0.0.1:80
23 TCP connect (conn 0)
15 Transfer data (conn 0)
GET /01/CGB-B9AJ/index.html HTTP/1.0
User-Agent: CGB-B9AJ-00
15 Transfer data (conn 0)
15 Transfer data (conn 0)
15 Transfer data (conn 0)
HTTP/1.0 200 OK
[... data ...]
15 Transfer data (conn 0)
15 Transfer data (conn 0)
1F Transfer data end (conn 0)

Cleanup
Protocol: HTTP endpoints

- /cgb/download?name=<path>
- /cgb/upload?name=<path>
- /cgb/utility?name=<path>
- /cgb/ranking?name=<path>
Protocol: HTTP auth

- crypt(md5(key + password) + userid)