GMR Introduction

GMR-1 Speech codec

GMR-1 Cipher

Final words 0000

# GMR-1 Speech codec

・ロト・日本・日本・日本・日本・今日・

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher 0000000000	Final words 0000
GMR-1 Speech The problem	i codec			

- AMBE: Advanced Multi-Band Excitation
- Not documented in the standard
  - Barely a high level description
  - No reference code
- Proprietary codec by DVSI Inc.
  - Not supported by their "cheap" hardware USB decoder
  - Cheapest hardware is the NET-2000 appliance (2kEUR)
- But :
  - mbelib: Code for other documented IMBE/AMBE variants (P25)
  - Implemented in SO2510 phone DSP (TI C55x)

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher 0000000000	Final words 0000
AMBE Codec Description				

- Highly specialized for voice (vocoder)
- Divides speech in small segments
  - For GMR-1: 20 ms frames subdivided into co-quantized 10 ms sub-frames
- Represent each speech (sub)frame as a set of parameters
  - f<sub>0</sub> : Fundamental frequency (pitch)
  - G : Gain (volume)
  - Voiced / Unvoiced decision (per band)
  - Spectral Magnitudes
- Decoding can be summarized as 3 steps:
  - **Unpacking**: Unpack the raw frame bits into quantized parameters
  - De-Quantization: From quantized parameters to actual values
  - Synthesis: From the parameters set to actual audio

∃ ► < ∃ ►</p>

Introduction 00	GMR Introduction	GMR-1 Speech codec ○○●○○○○	GMR-1 Cipher	Final words
AMBE Codec				
Synthesis				



イロト イ団ト イヨト イヨト

Introduction 00	GMR Introduction	GMR-1 Speech codec ○○○●○○○	GMR-1 Cipher 0000000000	Final words
AMBE Reversi	ng			

- Target: SO-2510 phone
- Codec has to be in the DSP, nowhere else it could be !
- DSP firmware extracted from firmware update package
  - Supported by IDA
- But where ?

DSP Code analysis

- 250k binary blob
- No strings
- Obscure TI C55x assembly
- Dieter Spaar to the rescue !
  - Identified entry points for encode/decode functions
  - Look for Audio DMA / Interrupts
  - Search for constants
  - Stack Switching



Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher 0000000000	Final words
AMBE Reversi	ng			

- TI Code Composer Studio Simulator
  - Accurately simulates supported DSP
  - Arbitrary memory layout
  - fread()/fwrite() from host
  - Tracing of all memory access
  - Windows only :(
- Use the original firmware to decompress audio for us
  - DSP dump converted to a valid COFF .OBJ file for linking
  - Custom linker script
  - Simple main() that fread() frames and fwrite() audio
- Success !

Simulator

- It took quite a few tries, lots of traps
- But it works and we get audio out
- Slow (not real-time) and not practical though



n GMR li ooooc

GMR Introduction

GMR-1 Speech codec

GMR-1 Cipher

Final words 0000

# AMBE Reversing

- Real HW would be faster and more convenient. But :
  - Code has to run at the physical address it has been linked for
  - OMAP has a DSP MMU, but standalone DSP don't
  - Need a cheap board with a compatible memory map
- Dieter found one with SDRAM where needed and Ethernet
  - Success ! About 16x faster than real-time
  - SDRAM is not fast, relocate some data tables to SRAM
- I indented to buy the same board
  - But in my haste ... I ordered the wrong one ... \*facepalm\*
  - No SDRAM, more SRAM, but at the wrong physical address
  - Easy, just relocate the code ! Can't be that hard, right ?
  - Use IDAPython + simulator trace mode
  - Success !





ヘロト く得ト くほト くほ

Introduction 00	GMR Introduction	GMR-1 Speech codec ○○○○○○●	GMR-1 Cipher 0000000000	Final words
AMBE Reversi	ng			

- Hardware USB decoder is nice, but not enough
- Decompression process:
  - Unpacking
    - Early, simple bit manipulation, easy to follow
  - Dequantization
    - Easily 95% of the work
    - Hard to follow fixed point math in DSP assembly
  - Synthesis
    - Started by just re-using mbelib code
    - Then rewrote using P25 specs and some guessing
- Resulting PoC/reference code in GIT
  - Not same audio quality as the original, but perfectly intelligible

amar	*+ARO(#857h)
call	sub_103540
mov	*AR6(#100h), AR1
btst	@0, AR1, TC1
mov	#0, <mark>T2</mark>
xccpart	ITC1
bcc	loc_10B148, !TC1
amar	*AR7, XAR0
amar	*AR5, XAR1
mov 1	2, TO
call	sub_109EC8
mov	T0, <mark>T2</mark>
amar	*AR5, XAR3
amar	*AR5, XAR4
amar	*+AR3(#684h)
amar	*AR3, XAR2
amar	*+AR4(#684h)
amar	*AR5, XAR3
amar	*+AR3(#584h)
rpt	#0FFh
mov	*AR4+, *AR3+
mov	*AR5(#785h), AR1
mov	AR1, *AR5(#784h)
amar	*SP(#var_0), XAR3
rpt #	¢0FFh
mov	*AR3+, *AR2+
add	#102h, mmap(@SP)
mov	T3, *AR5(#785h)
popboth	XAR6

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher ○○○○○●○○○○○	Final words 0000
A better attac Overview	k			

- Based on the same A5/2 GSM attack
  - Don't do anything fancy, just tweak for A5-GMR-1
- Both known-plaintext and ciphertext-only variant
- Targets FACCH3 control frames instead of TCH3 voice frames
- FACCH3 advantages :
  - $\blacksquare$  Simpler modulation and better training sequence  $\rightarrow$  less bit-errors
  - Predictable plaintext  $\rightarrow$  known-plaintext attacks
  - Much more redudancy (more FEC)  $\rightarrow$  less bursts needed for ciphertext-only attacks
  - $\blacksquare$  Used to negotiate TCH6/TCH9 channels  $\rightarrow$  attack works for CSD/Fax

Introduction 00		GMR Intro	duction		GMR-1 Spe 0000000	ech codec	0	MR-1 Cipher	Final wor
A bett Known pla	er atta aintext	ck							
	<u>File E</u> dit <u>V</u> iew	/ <u>G</u> o <u>C</u> apture <u>/</u>	Analyze <u>S</u> tatistics	Felephony	<u>T</u> ools <u>I</u> nternals	Help	0, 17   24	🗹 🍋 💥 l 🛱	
	Filter: gsmtap.g	mr1 chan type =	= 18	- Exp	pression Clea	r Apply Save			
	No. Time	Source	Destination S	rc Port Dst	Port Protocol	Fn Info			<u> </u>
	178 0.114724 183 0.11967 185 0.12515 187 0.122066 188 0.124150 199 0.125702 191 0.125702 193 0.125702 193 0.126720 193 0.126720 193 0.126720 1 F Finne 183: 60 F Ethernet IT, ▷ Ethernet IT, ▷ Ethernet IT, ▷ GSM TAP Head CSM TA	127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 127.0.0.1 8 bytes on wire 5rc: 00:00:00:00 tocol Version 4, m Protocol, Srci 5rc: 00:00:00:00 tocol Version 4, m Protocol, Srci 00:00 Procedure, Satel eld: 0x80 eld: 5, func=RR, F Payload last n. d: 0	127.0.0.1 127.0.0.0.1 127.0.0.0.1 127.0.0.0.1 127.0.0.0.1 127.0.0.0.0.0.0.0.	60051 60054 60051, r7.0.0.1), r. 0.0.1), r. 0.0.1), r. 0.0.1), j. D.S. Port annel: FAC tt)	Mr29 LAPSat 4729 LAPSat 4729 LAPSat 4729 LAPSat 4729 LAPSat 4729 LAPSat 4729 LAPSat 4729 LAPSat 4729 LAPSat 544 bits) 054: 00:00:00 Dst: 02:00:00 Dst: 02:00:00	22996 I, N(R)=4, 23024 S, func=RF 23024 S, func=RF 23024 S, func=RF 23028 S, func=RF 23032 I, N(R)=7, 230340 I, N(R)=7, 23044 I, N(R)=7, 23044 I, N(R)=7, 23044 I, N(R)=7, 23046 I, N(R)=7, 20046	N(s)=5 (DTAP) ( , N(R)=5 , N(R)=5 , N(R)=6 (Fragmer N(s)=6 (Fragmer N(s)=6 (DTAP) ( N(s)=8 (DTAP) ( N(s)=9 (DTAP) ( N(s)=9 (DTAP) ( N(s)=9 (DTAP) ( N(s)=8 (DTAP) ( N	<pre>RR) Ciphering Mode Command t) M) MM Information M) Location Updating Accep RR) Channel Release</pre>	
	0000         00         00         00         00           0010         00         36         ea           0020         00         01         ec           0630         00         00         00         00           0040         00         00         00         00	00 00 00 00 00 c3 40 00 40 11 ee 12 79 00 22 00 59 ec 12 00 00	00 00 00 00 00 08 00 51 f1 7f 00 00 01 fe 35 02 04 0a 0d 00 00 80 28 10 00	45 00 7f 00 .6 00 00 00 00					
	🔘 💅 File: "/tmp/	/tnt-locupd-deciph	ere Packets: 261	· Displayed:	33 · Marked: 3 ·	Load time: 0:00	Profile: Default		

900

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher ○○○○○○○●○○○	Final words 0000
A better attack	k			

- Goal is to describe cipher as a linear operation:  $A \cdot x = b$ 
  - A = matrix describing cipher, x = internal state and b = cipher stream
  - Each row of A and b is a bit of the output
- Internal cipher state dependency on FN and Kc is linear
  - Possible to combine equations from different bursts at different FN
  - Can recover Kc from the state
- Non-linear elements:
  - Majority function:  $\mathcal{M}(a, b, c) = a + b.c$ 
    - Introduces quadratic terms
    - Linearize by adding one new unknown for every possible quadratic term
    - 594 new unknowns
  - Irregular clocking depending on R4 value
    - R4 is 17 bits but one is forced to '1' at init. Small enough for brute force !
    - Assume a given value for R4
    - Repeat 65536 times

> < 国 > < 国 >

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher ००००००००●००	Final words 0000
A better attac R4 quick scan	k			

- In  $A_n \cdot x = b$ , some equations are redundant
- We can get a parity-check matrix  $H_n$  such that  $H_n \cdot b = \mathbf{0}$
- Those 65536  $H_n$  matrices can be precomputed offline
- With a single matrix-multiply we can check if a given R4 value is even a possibility
  - If result is non-zero, we can skip that R4 value
  - If result is zero, then we try to solve the system
  - In practice, only a few R4 value ever matches

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher	Final words
A better at	tack			

- Channel coding operation:  $m = d \cdot G + g$
- Let *H* be the parity-check matrix so that  $H \cdot (m + g) = \mathbf{0}$
- Encryption operation: y = m + b
- *H* can be used to derive equations from the ciphertext *y*:

$$H \cdot (y + g) = H \cdot (m + b + g)$$
  
=  $H \cdot b + \underbrace{H \cdot (m + g)}_{0}$   
=  $H \cdot A \cdot x$ 

- The same R4 quick-scan technique can also be used here
- To get enough equations for a unique solutions, multiple frames are needed

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher ○○○○○○○○○●	Final words 0000
A better attac Results	k			

- Known-plaintext variant
  - Requires between 4 and 8 bursts depending on alignement
  - Space: 50 Mb
  - Time: 500 ms
- Ciphertext-only variant
  - Requires 8 consecutive bursts belonging to 2 FACCH3 L2 frames
  - Space: 5 Gb
  - Time: 1 s

글 🕨 🖌 글

troduction	G

MR Introduction

GMR-1 Speech codec

GMR-1 Cipher

Final words

# Final words

<ロト < @ ト < 差 > < 差 > うへの

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher 0000000000	Final words ●000
Future				

- C-band
- Packet Data (GmPRS)
- Upper layers implementation
- CSN.1 and 04.008 code generators
- TX side

Help welcome :)

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher	Final words ○●○○
0.1				

## Other satellite phone systems

- We choose Thuraya because :
  - Visible from Europe
  - Cheapest sat phone on ebay
  - Specifications mostly available
- Don't think other are better without proof
  - Availability of commerical intercepts tend to say otherwise

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher 0000000000	Final words 00●0
Thanks				

Thanks to anyone who contributed to this projects and related ones. Most notably:

- Dimitri "horizon" Stolnikov
- Dieter Spaar
- RUB team

注▶ ★ 王

Introduction 00	GMR Introduction	GMR-1 Speech codec	GMR-1 Cipher 0000000000	Final words 000●
Further readin	g			

#### GMR-1 in general

OsmocomGMR http://gmr.osmocom.org/ 28C3 talk http://gmr.osmocom.org/trac/blog/28c3-recording GMR1 Specs http://www.etsi.org/standards-search GSM Specs http://webapp.etsi.org/key/queryform.asp

## AMBE Codec

DVSI Inc. http://www.dvsinc.com/

## GMR-1 Cipher

RUB GMR page http://gmr.crypto.rub.de/ Paper http://cryptome.org/gsm-crack-bbk.pdf