Security in DECT

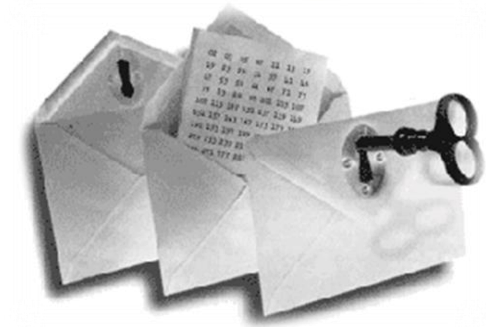
Marc Seeger
Computer Science and Media
HdM Stuttgart

Digital
Enhanced
Cordless
Telecommunications

You‘re going to hear about…

- the DECT standard

- security in DECT



- deDECTed

Usage

Usage	My personal security concerns
Babyphones	¯\(º_o)/¯
Wireless ISDN	**O_o**
Telephones	Ò_ó
Emergency Call Systems	:-/
Door opening systems	:-O
Wireless EC-Cardreaders	X-/
Traffic control systems	X-O

DECT: the numbers

- Before (analog): CT1(+), CT2
- ETSI Standard: 1992
- Audio codec: G.726
- Net bit rate: 32 kbit/s
- GFSK
- Frequency:
  - 1880 MHz–1900 MHz in Europe
  - 1900 MHz–1920 MHz in China
  - 1910 MHz–1930 MHz in Latin America
  - 1920 MHz–1930 MHz in the US
- Average transmission power:
  - 10 mW (250 mW peak) in Europe
  - 4 mW (100 mW peak) in the US

3 main parts

PP (portable part)

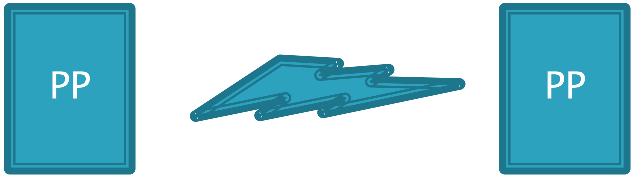
FP (fixed part)

RFP (radio fixed part)

A DECT system:

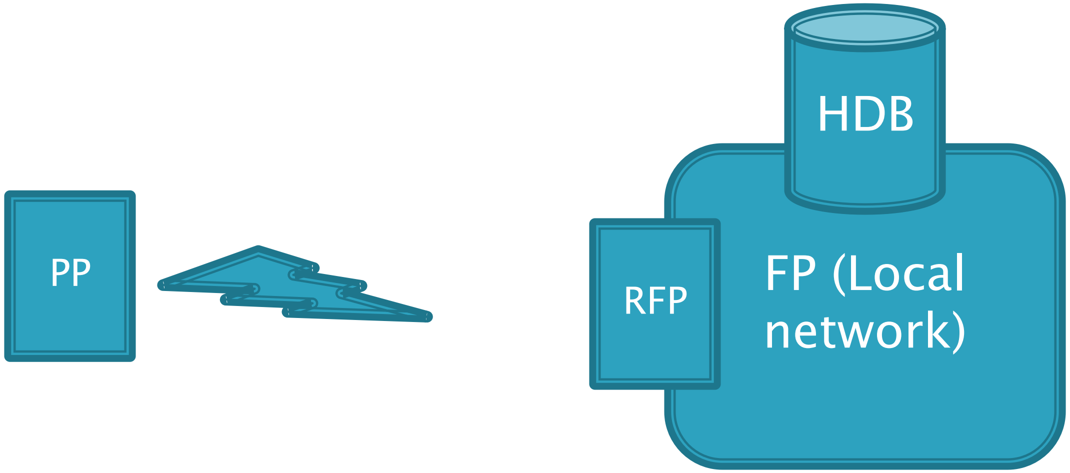
- 1 DECT Fixed Part (FP)
- 1+ radio fixed part (RFPs)
- 1+ DECT Portable Parts (PPs)

DECT Infrastructure: Direct mode



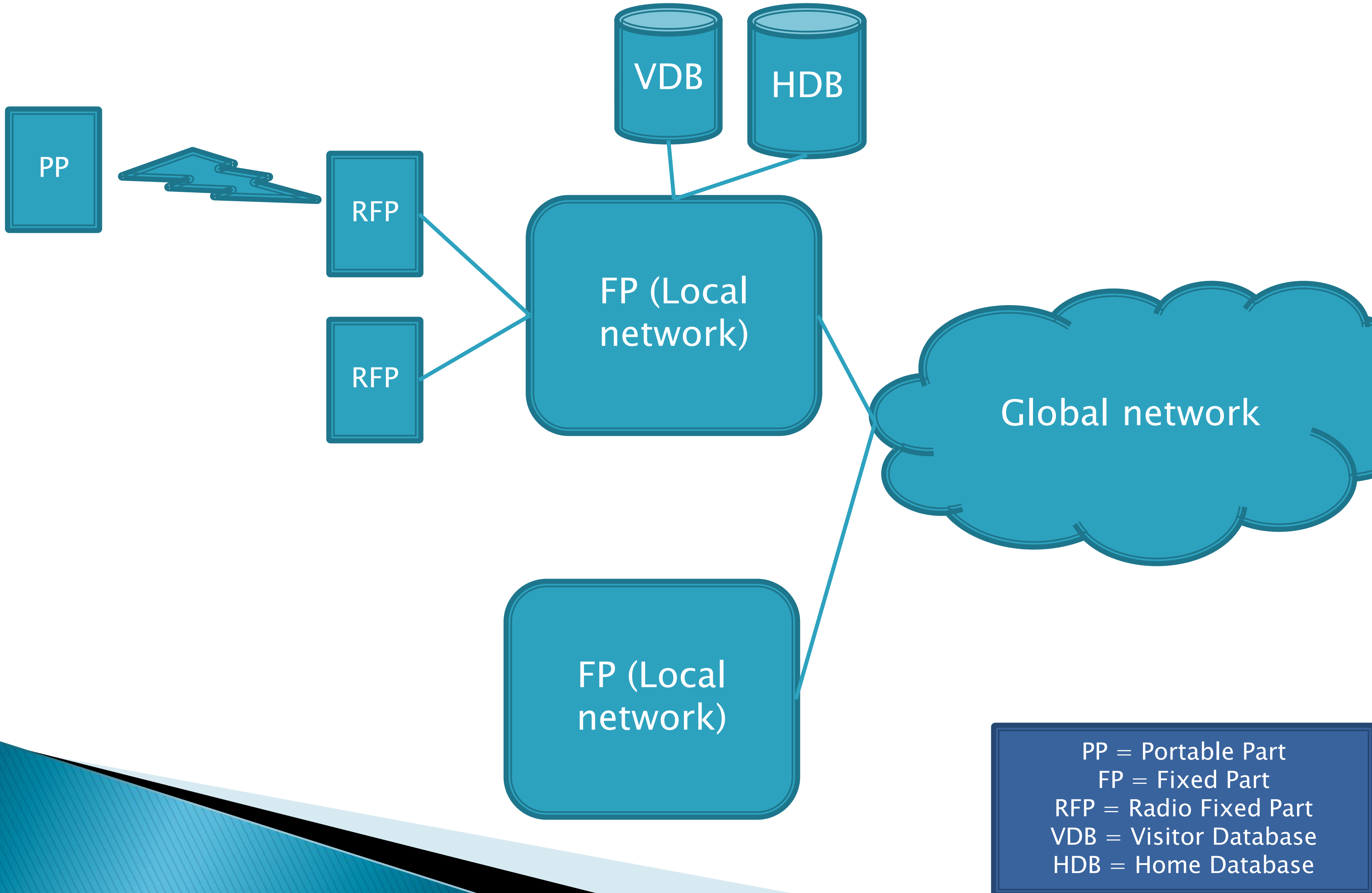
PP = Portable Part

DECT Infrastructure: single cell



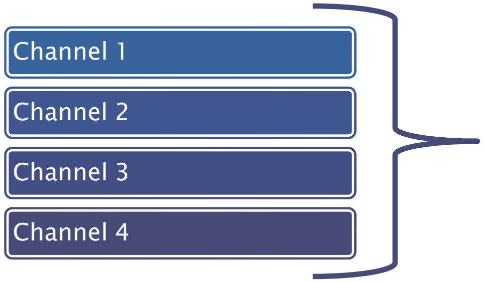
PP = Portable Part
FP = Fixed Part
RFP = Radio Fixed Part
HDB = Home Database

DECT Infrastructure: multi cell



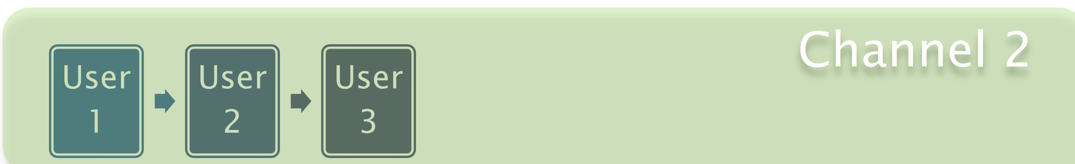
DECT: Seperation

- Frequency division multiple access (FDMA)

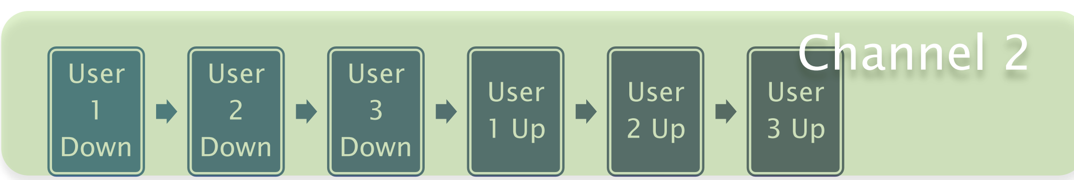


Frequency Range: 10 (1,728 kHz spacing) in Europe; 5 (1,728 kHz spacing) in the US

- Time division multiple access (TDMA)



- Time division duplex (TDD)



Time slots: 2 x 12 (up and down stream)

Profiles

- Generic Access (GAP)
  - mandatory minimum requirement for all DECT voice telephony equipment as from October 1997
- Radio in the Local Loop applications (RAP)
  - the "last mile"
- ISDN and GSM interworking (GIP).
- …

Signaling

FP (station)

- Broadcasting network informations (RFPI,...)
- Scanning for PP activity

Signaling

PP (phone)

- Radio: Passive in idle mode
- Scanning for pages
- Scanning and making a list of channels avg. RSSI < every 30 seconds
- Synchronizing with base station
- Selecting best carrier/slot-combination for communication and opening a conne
- Initiating encryption

Signaling II

- When authenticating with an FP, the PP receives a unique 20 Bit identifier called TPUI (Temporary User Identity).
- This TPUI is used when the FP uses paging because of incoming calls

Security: Executive summary

General problems:

- digital radio access technology
  - Eavesdropping

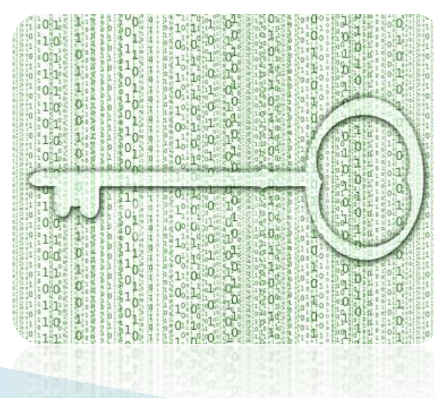
  - Third party accesses equipment

  - Man-in-the middle attack

2 main topics

- Authentication

- Encryption

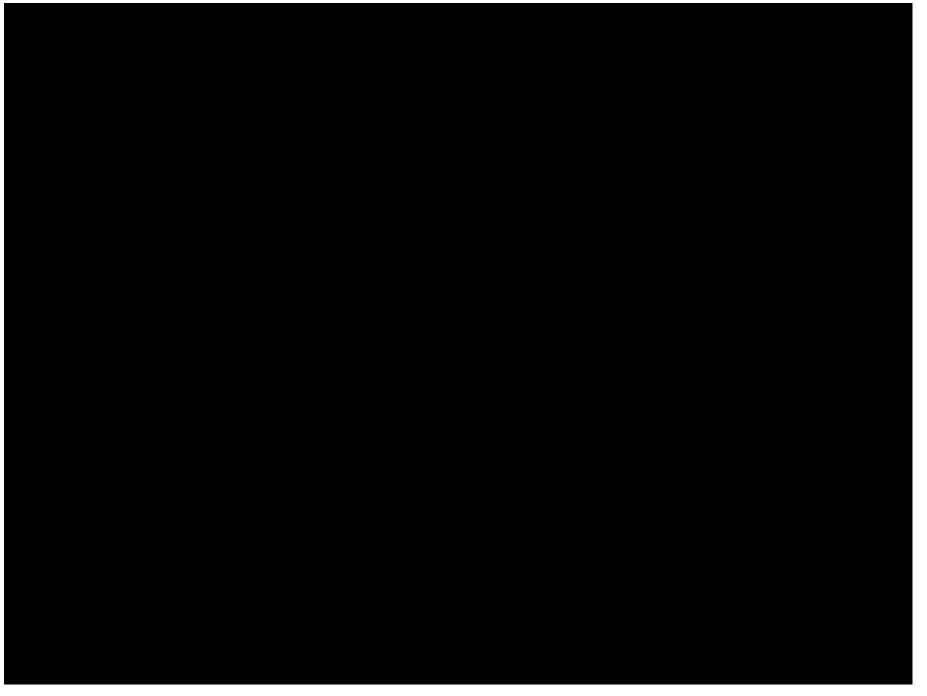


Authentication summary

- „DSAA“ = DECT Standard Authentication Algorithm
- Subscriber and base station share an authentication key after first „pairing“

→ challenge + response

...as explained by Monty Python:



Encryption summary

- DSC = DECT Standard Cipher
- During authentication, both sides also calculate a cipher key.
- This key is used to de/encrypt data sent over the air.
- The ciphering process is part of the DECT standard (but not mandatory).

Nitty gritty details

Authentication

- First: Key allocation

  („pairing“)

- After that: Challenge Response

Key allocation

- Initial pairing of the FP with the PP
- Special „pairing mode“
- User has to enter PIN on FP and PP
  => shared secret for DSAA
- Key allocation results in a 128 bit secret key
  „UAK“ = User Authentication Key

DSAA DECT standard authentication algorithm

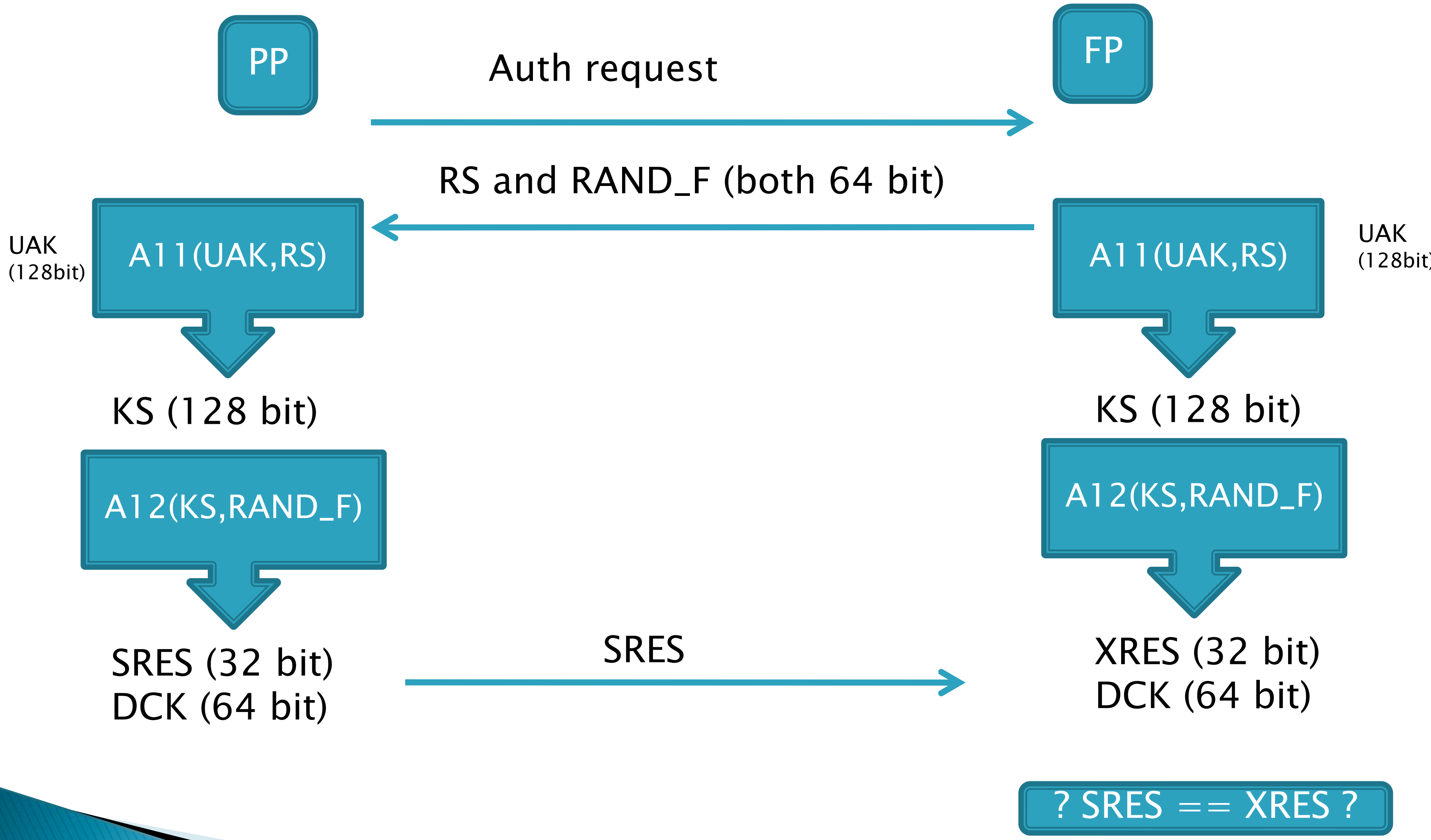
A11, A12, A21, A22

- A11 + A12
  - Authentication of PP
  - Generation of UAK: User Authentication Key (GAP)
  - Key generation for DSC
- A21 + A22
  - Authentication of FP

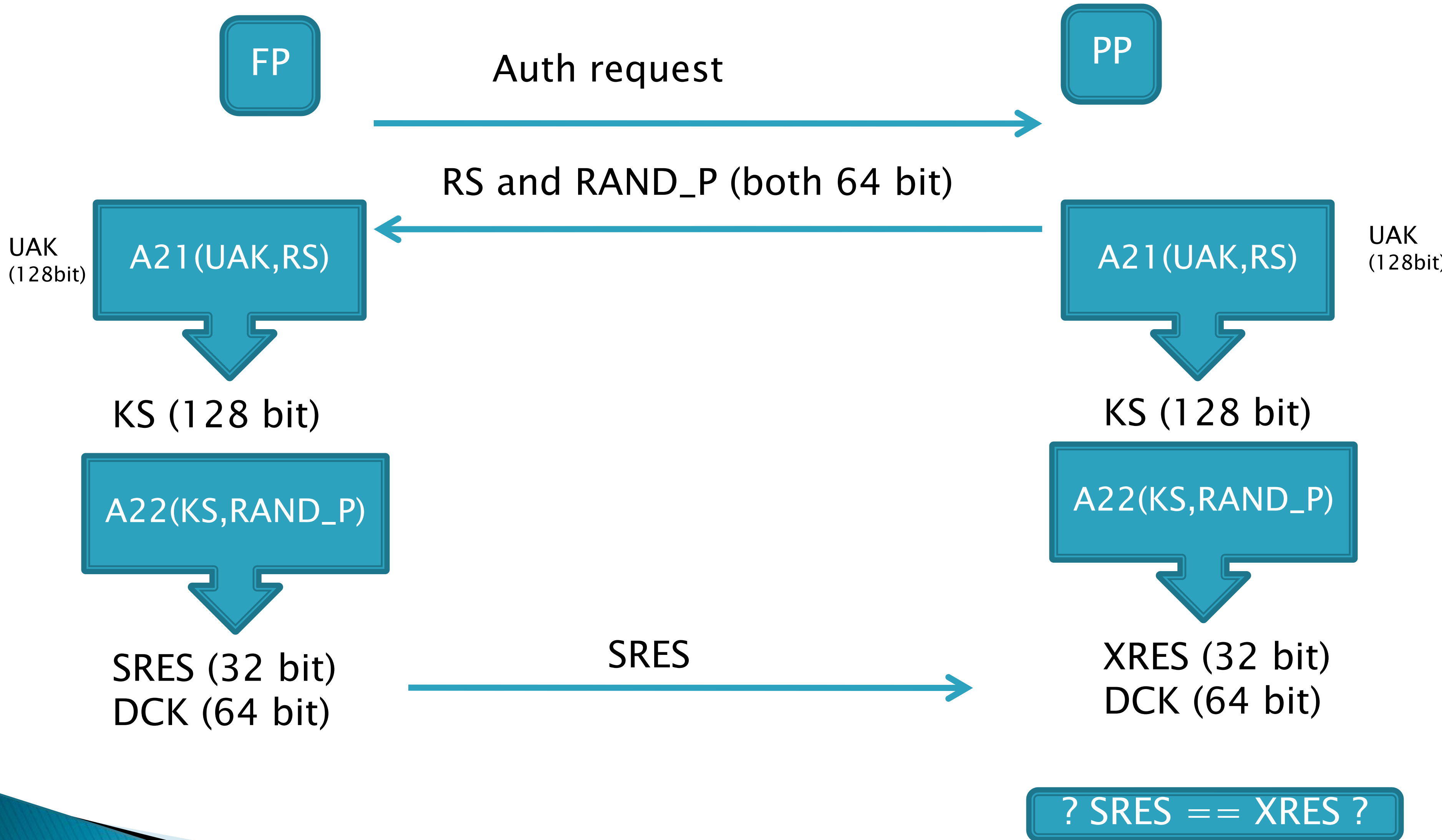
And:

- Algorithms were a secret

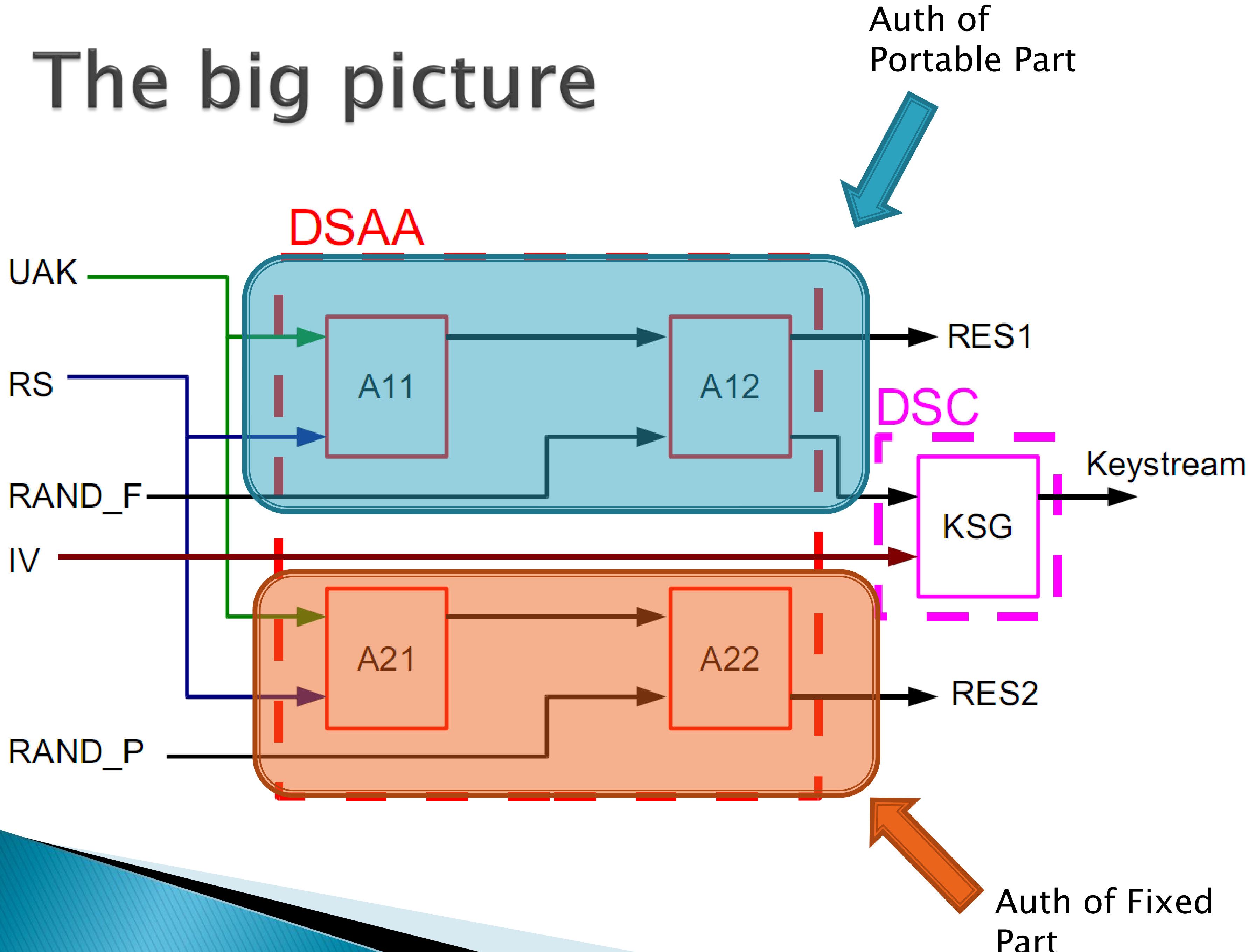
PP Authentication



FP Authentication

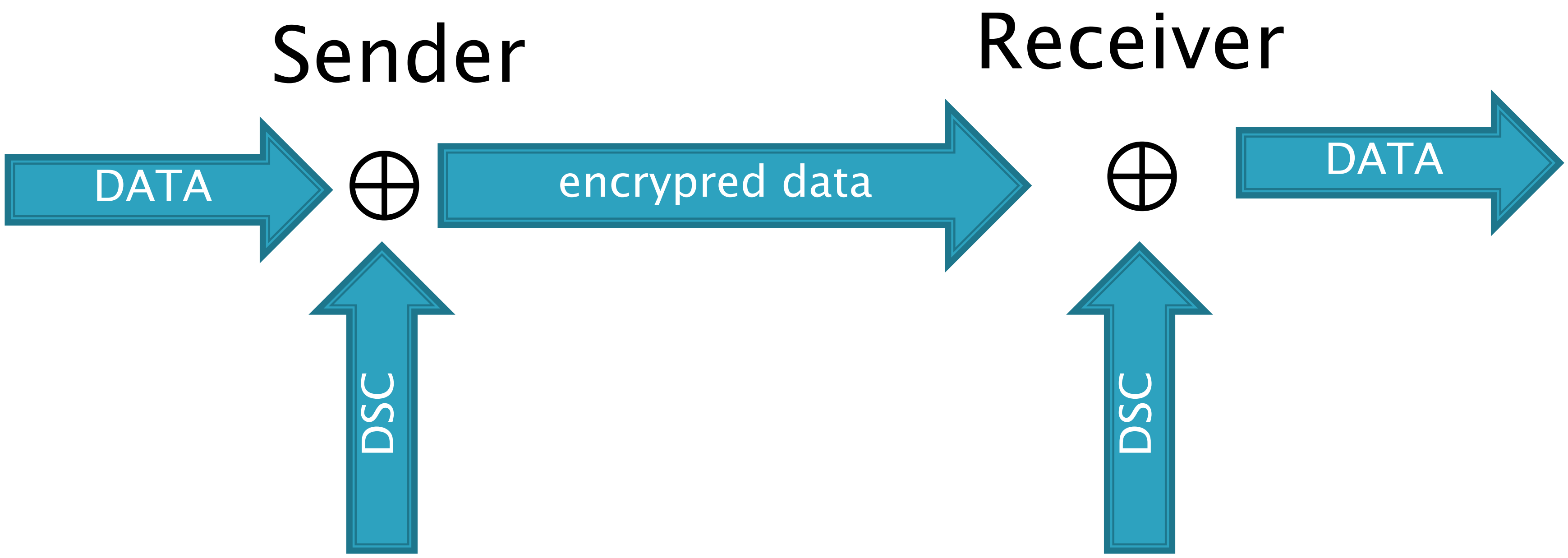


The big picture



DSC: DECT Standard Cipher

- If encryption is enabled, signaling and data will be XOR'ed with the output of the DSC Streamcipher



DeDECTed

Who?

At this moment, members of the the project are people of the following entities:

- Chaos Computer Club (Munich, Trier)
- TU-Darmstadt Germany
- University of Luxembourg
- Bauhaus-Universität Weimar Germany

and some individuals:

- krater Andreas Schuler
- mazzoo Matthias Wenzel
- Erik Tews
- Ralf-Philipp Weinmann (University of Luxembourg)
- kaner Christian Fromme
- H. Gregor Molter
- Harald Welte

Sniffing

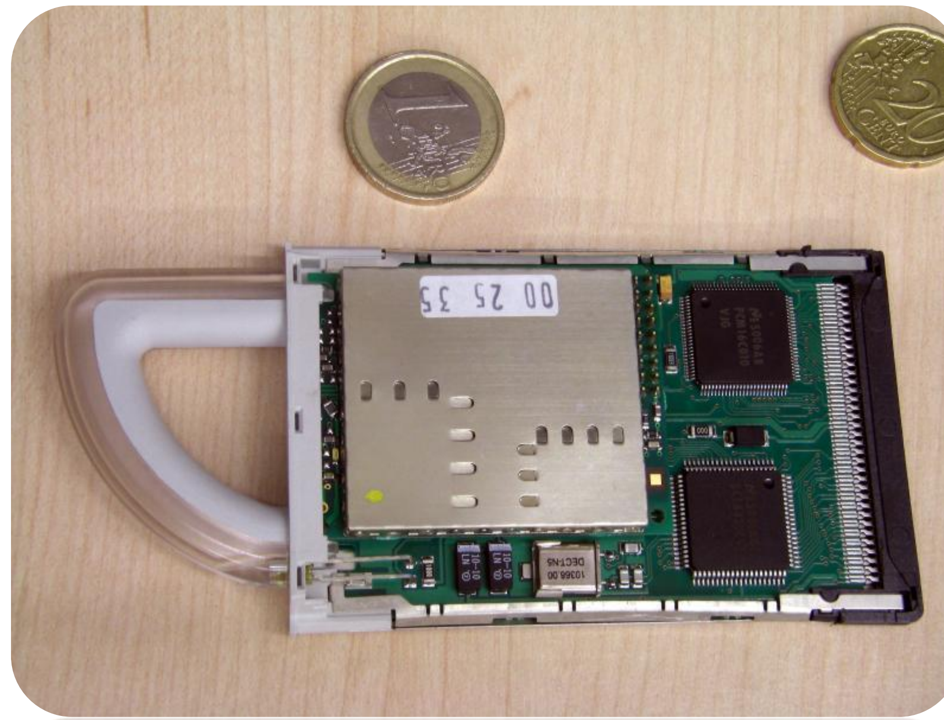
- Problems:
  - Stations not synced
  - No Source/Dest Fields in Packets
  - No Information when PP opens connection
  - Descrambling requires Framenumber

First try: USRP

- Can capture all packets on a channel
- CPU requirements are high (2 GHz+ CPU required)
- Time multiplexing is difficult to handle
- Sending frames is not supported
- Costs : 1000 EUR

Second try: ComOnAir

- Can capture all packets on a channel
- Can scan for stations or active calls
- Can sync on stations and dump active calls
- CPU requirements low
- Sending frames supported soon
- Costs : 23 EUR



Problem 1: Windows only

- Solution: reverse engineer:
  - Removing case
  - Searching datasheets
  - Reversing Windows driver
  - Find firmware image
  - Try to activate hardware
  - Upload firmware to chip
  - Wait for interrupts

Result 1: Linux Driver

commit b2185f943fd642bd46ca4e13f87d3fce374fbe69
Author: Andreas Schuler krater@badterrorist.com
Date: Wed Dec 3 23:59:21 2008 +0000
WE HAVE INTERRUPTS cat /proc/interrupts ! :))

Hack 1: passive sniffing

- If there is no ciphering
  ➔ capture and record audio data
- Userspace utility scans for an active call and tracks the first one found
- Packets are recorded to a pcap file
- The file can later be played with an audio player

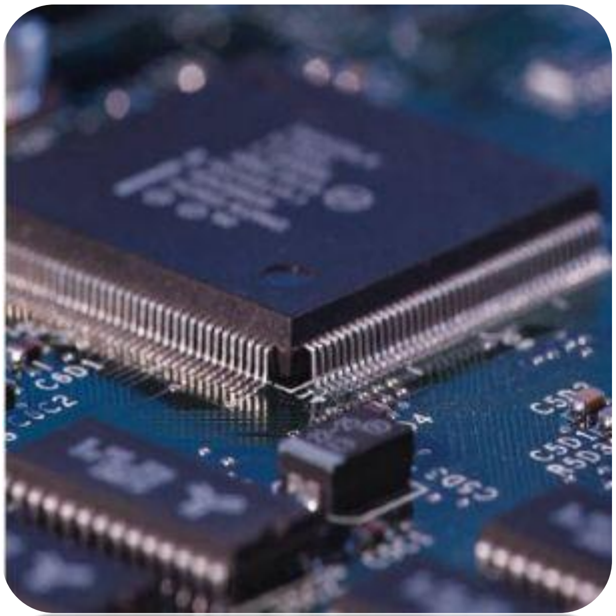
- Total costs for the attack: 23 EUR.

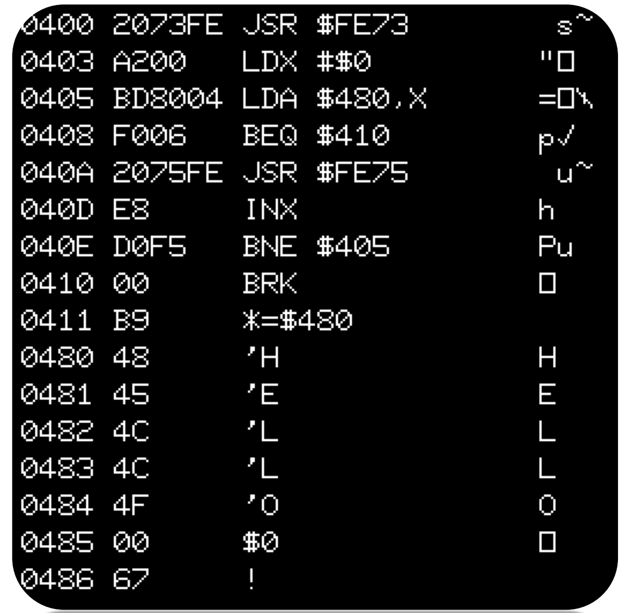
Hack 2: impersonating a basestation

- Even when a phone supports encryption, most phones will not abort connection if base station does not
- Calls can be rerouted (and recorded)
- Implementation requires attacker to enter RFPI of base station to impersonate and IPUI of phone to accept
- Total costs for this attack: 23 EUR.

Next step: Reversing DSAA

DSAA = software!





DSAA

- A12, A21, and A22 are just simple wrappers around A11
  - A11 just returns the whole output of DSAA, without any further modification.
  - A21 behaves similar to A11, but here, every second bit of the output is inverted, starting with the first bit of the output.
  - A22 just returns the last 4 bytes of output of DSAA as RES.
  - A12 is similar to A22, except here, the middle 8 bytes of DSAA are returned too, as DCK.

- A11 takes a 128 bit key and a 64 bit random number to generate a 128 bit output
- A11 uses four different block ciphers we call *cassable* to generate the output

Sbox – a digital smoke grenade

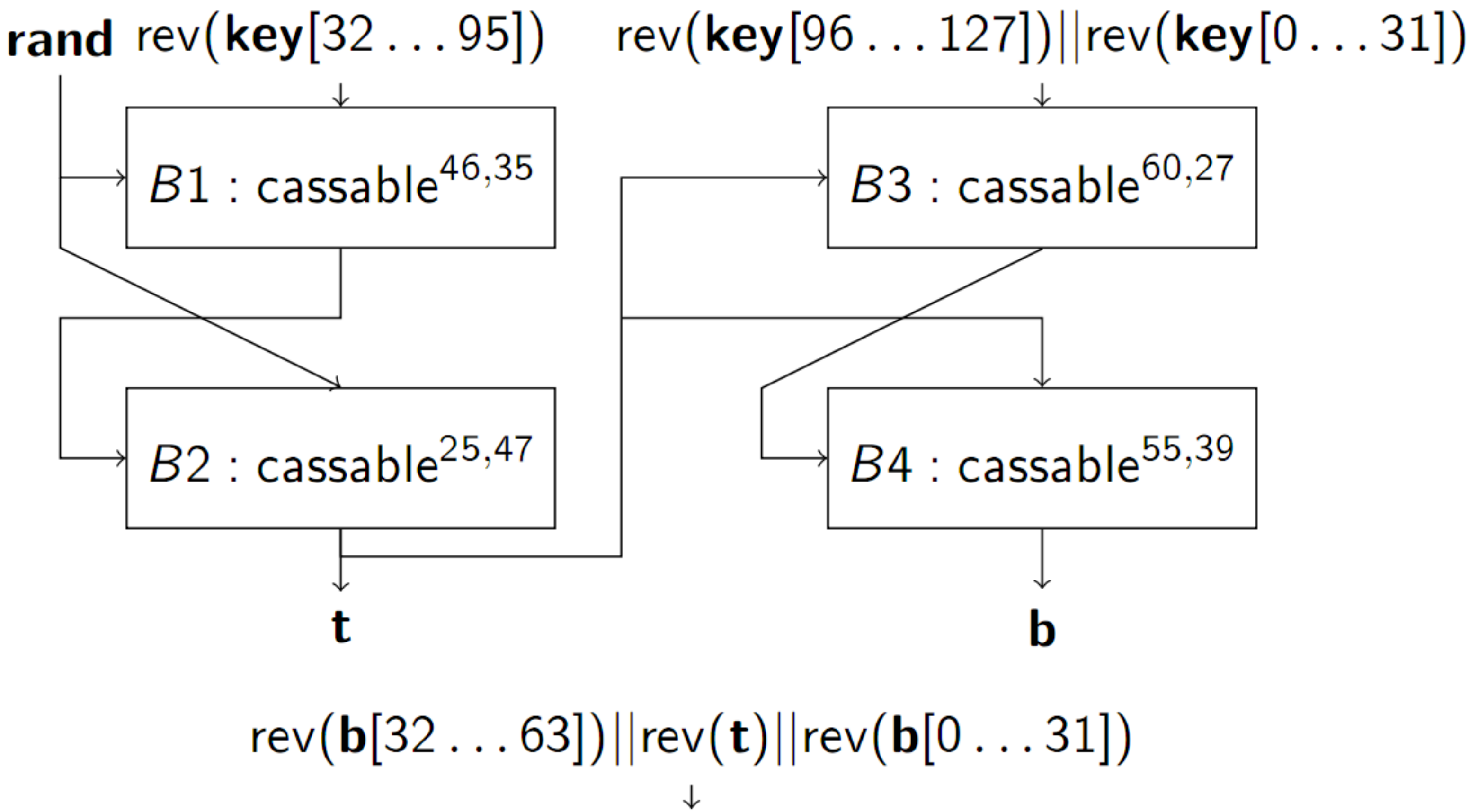
- Grepping for XORs in firmware files

→ 256 unique bytes in all of them

```
b0 68 6f f6 7d e8 16 85 39 7c 7f de 43 f0 59 a9
fb 80 32 ae 5f 25 8c f5 94 6b d8 ea 88 98 c2 29
cf 3a 50 96 1c 08 95 f4 82 37 0a 56 2c ff 4f c4
60 a5 83 21 30 f8 f3 28 fa 93 49 34 42 78 bf fc
61 c6 f1 a7 1a 53 03 4d 86 d3 04 87 7e 8f a0 b7
31 b3 e7 0e 2f cc 69 c3 c0 d9 c8 13 dc 8b 01 52
c1 48 ef af 73 dd 5c 2e 19 91 df 22 d5 3d 0d a3
58 81 3e fd 62 44 24 2d b6 8d 5a 05 17 be 27 54
5d 9d d6 ad 6c ed 64 ce f2 72 3f d4 46 a4 10 a2
3b 89 97 4c 6e 74 99 e4 e3 bb ee 70 00 bd 65 20
0f 7a e9 9e 9b c7 b5 63 e6 aa e1 8a c5 07 06 1e
5e 1d 35 38 77 14 11 e2 b9 84 18 9f 2a cb da f7
a6 b2 66 7b b1 9c 6d 6a f9 fe ca c9 a8 41 bc 79
db b8 67 ba ac 36 ab 92 4b d7 e5 9a 76 cd 15 1f
4e 4a 57 71 1b 55 09 51 33 0c b4 8e 2b e0 d0 5b
47 75 45 40 02 d1 3c ec 23 eb 0b d2 a1 90 26 12
```

A11

Thanks to the software implementations, it is now known that:



Cassable block cipher:

Other things we learned:

- cassable is a substitution permutation type network
- input is 64 bit
- key is 64 bit
- output is 64 bit
- internal state also has 64 bit
- for key scheduling, a bit permutation is used
- each variant of cassable only differs in this bit permutation
- to add the round key, ⊕ is used
- a single cassable invocation does 6 rounds in total
- each round consists of
  - a key addition (⊕)
  - S-box application
  - one of three different mixing functions
  - No final key addition (→ only 5 relevant rounds)

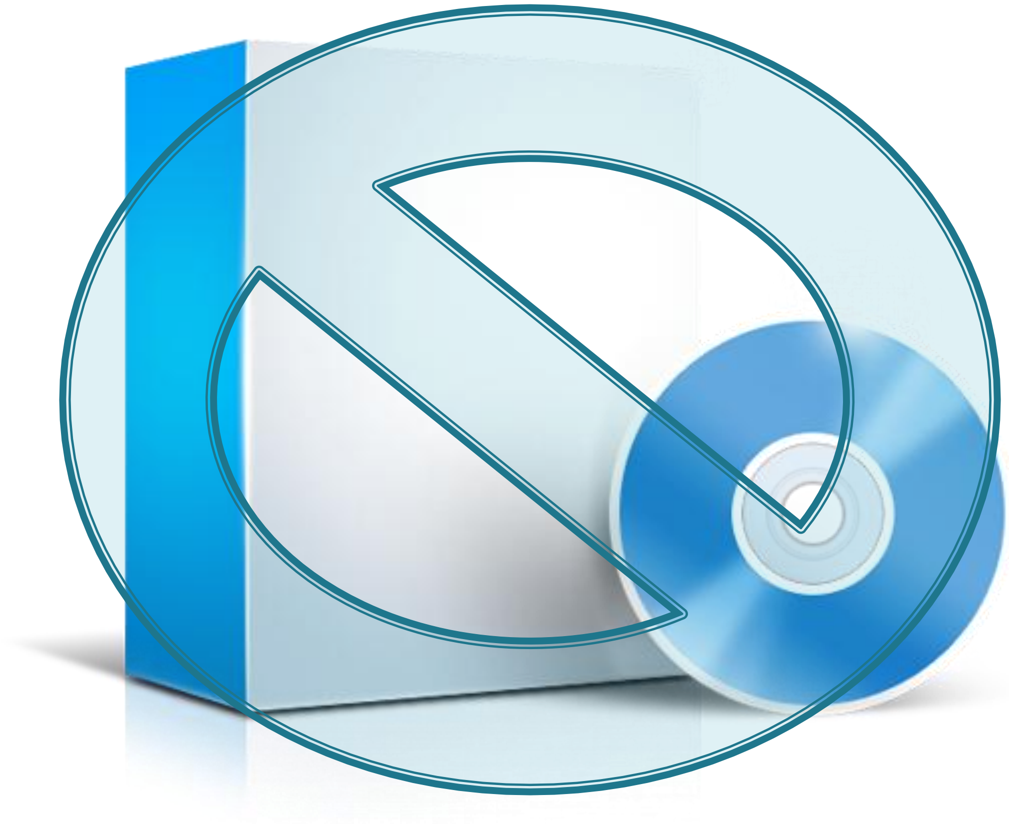
Cassable Cryptoanalysis

- No final key addition at the end, reduces strength to five effective rounds
- At first look, full diffusion after three rounds
- However, full diffusion only after four rounds
- Attacks:
  - S-Box allows linear cryptanalysis for 2-3 rounds versions
  - Practical algebraic attacks possible up to 3 rounds version of cassable
  - A differential attack possible on the full cipher with about 16 chosen input-output pairs and computational effort compareable to 2^37 invocations of cassable (before: 2^65)
- However, this has no direct impact on DSAA so far

Next step: The DSC

Problem

- No software implementation



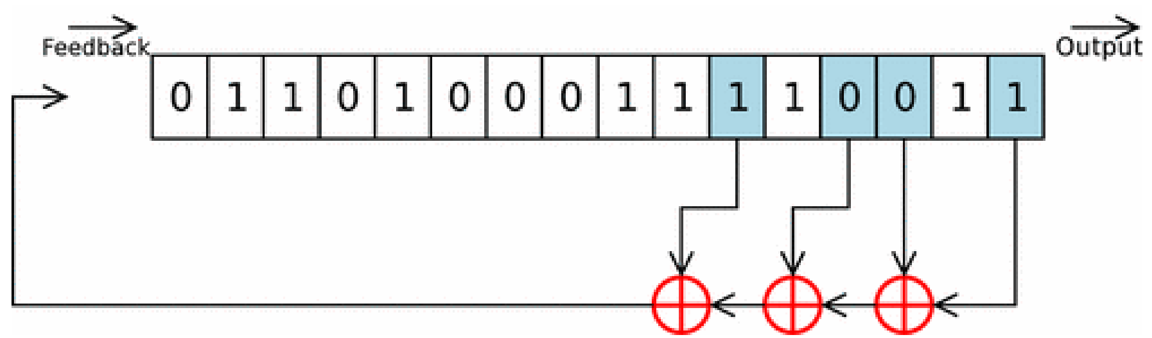
The DSC patent

- From the ETSI non-disclosure agreement for the DSC:
  - Not to register, or attempt to register, any IPR (patents or the like rights) relating to the DSC and containing all or part of the INFORMATION."
- U.S. Patent 5,608,802, registered by Alcatel, originally registered in Spain in 1993:
  - A data ciphering device that has special application in implementing Digital European Cordless Telephone (DECT) standard data ciphering algorithm [...]"

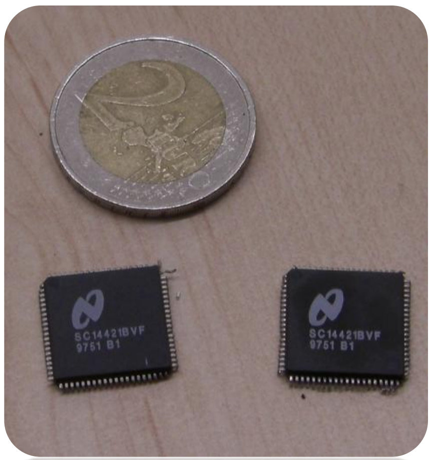
Information from the patent

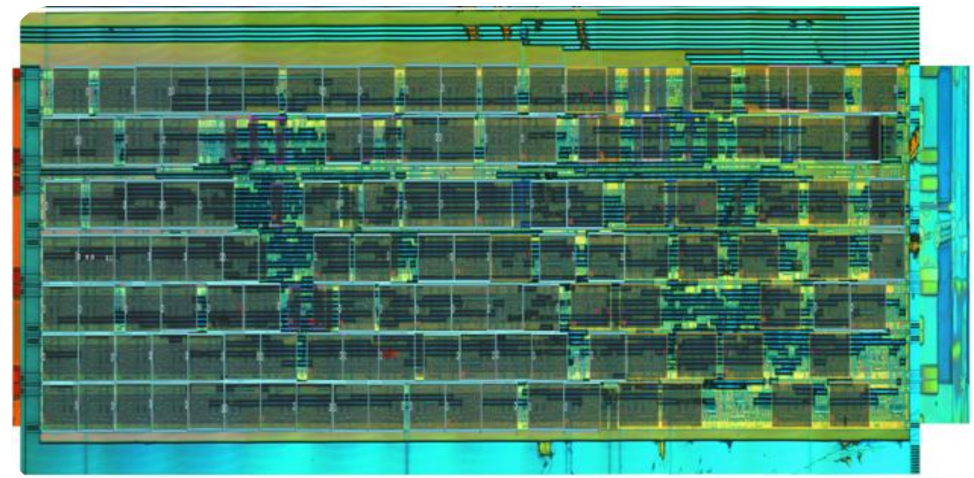
- 3 irregularly clocked LFSRs (2 or 3) of length 17,19,21
- 1 regularly clocked LFSR (3) of length 23
- key setup: load key, then 40 blank steps (irregularly clocked)
- check whether register is zero after 11 steps, load 1 into every zero register

LFSR:



Hardware reversing





Result: feedback tap positions

„Software“ reversing

- NSC/SiTel SC144xx CPUs have commands to save internal state in DIP memory (11 bytes)
- DIP memory can be read from host
- Can load/save state after and before pre-ciphering (D LDS; D WRS)
- Single-step through key loading to determine feedback taps
- Isolate subset of bits determining clocking differentially in pre-ciphering
- Interpolate clocking function (it's linear actually, could've seen that with bare eyes)
- Output combiner is still missing at the moment

DSC so far:

- Looks like A5
- Attacks not directly transferable
- Not attack available yet, looking pretty good though

Next step: Attacking the UAK

Attacking the UAK

- Reminder:
  - UAK = initial shared secret exchanged while pairing
- Impact:
  - impersonate handsets
  - decrypt encrypted calls
  - etc.

Entropy

„Randomness“

```
uint16_t counter ;
uint8_t xorvalue ;
void next_rand ( uint8_t  *rand )
{
  int i;
  for (i = 0; i < 8; i ++) {
       rand [i] = ( counter >>i) ^ xorvalue ;
  }
xorvalue += 13;
}
```

And that means...

- Grab two challenge-response „pairs“ (RS,RAND_F,RES)

- Iterate over all 4-digit PINs:
  3 * 2^35 DSAA operations
- Assume 0000 PIN:
  2^24 DSAA operations
  (50 secs on an Intel C2D 2.4GHz)

Sources

BAD:

Jabra: “DECT provides high protection against unauthorized access” Whitepapaer

OK:

dect.org

Good:

dedected.org

„Attacks on the DECT authentication mechanisms“
Stefan Lucks, Andreas Schuler, Erik Tews, Ralf-Philipp Weinmann, and Matthias Wenzel

Chaosradio Express Folge 102 : Der DECT Hack: http://chaosradio.ccc.de/cre102.html

25C3 Talk :https://dedected.org/trac/wiki/25C3

BSI: Drahtlose lokale Kommunikationssysteme und ihre Sicherheitsaspekte