GSM privacy attacks

Karsten Nohl, nohl@srlabs.de

SECURITY RESEARCH LABS

Agenda

- **GSM attack history**
- GSM attack vectors
- Attacking GSM’s A5/1 encryption
- Risk scenario: GSM payment

GSM is global, omnipresent and wants to be hacked

80% of mobile phone market

200+ countries

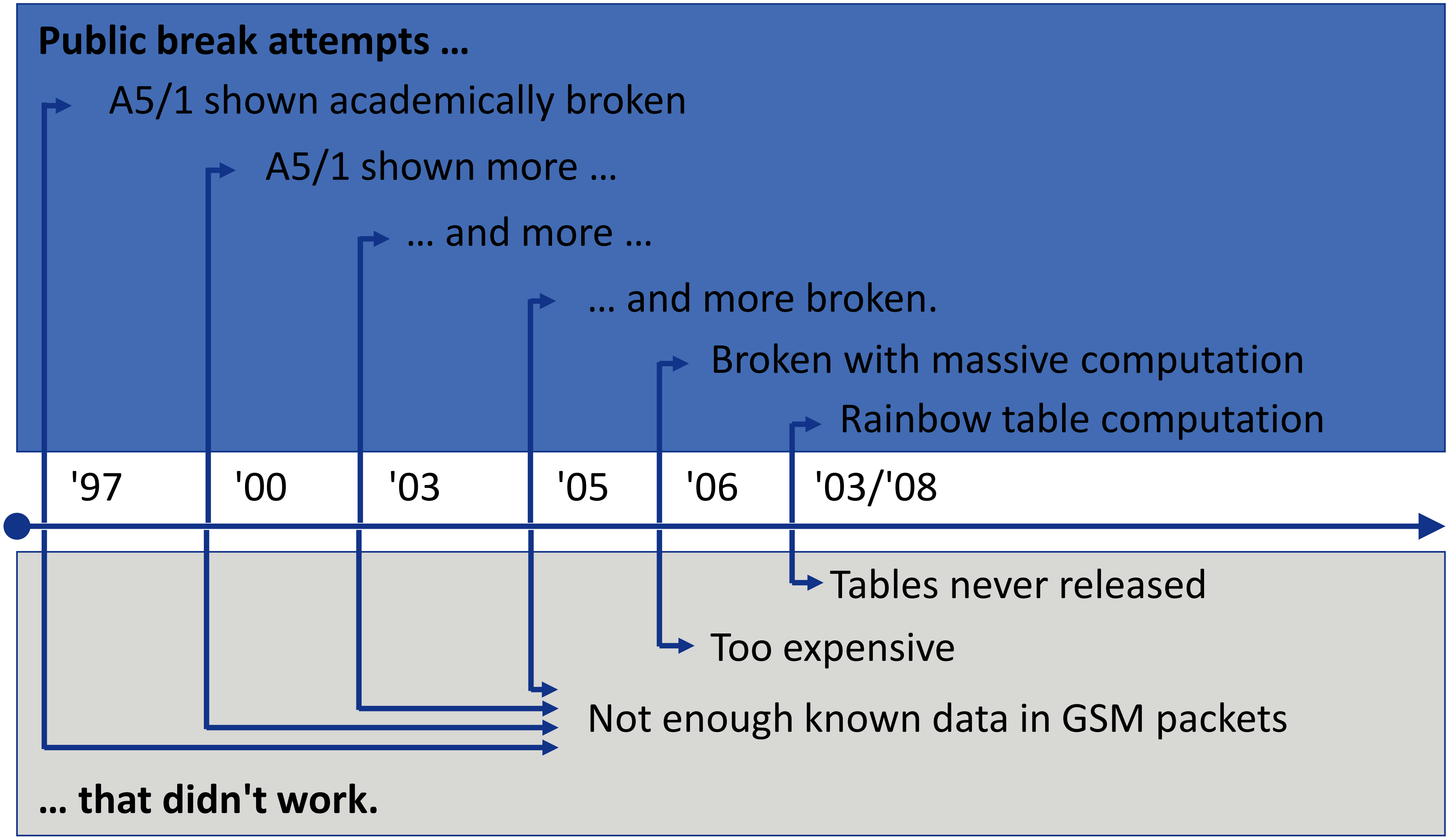
5 billion users!

GSM encryption introduced in 1987 …

… then disclosed and shown insecure in 1994

Source:Wikipedia, Bitkom press statement July 28th 2010

We wanted to publicly demonstrate that GSM uses insufficient encryption



Industry responds to GSM cracking attempts by creating new challenges

"… the GSM **call has to be** identified and **recorded** from the radio interface. […] we strongly suspect **the team** developing the intercept approach **has underestimated its practical complexity**.
A hacker would need a <u>radio receiver system</u> and the **<u>signal processing software</u>** necessary to process the raw radio data."

– GSMA, Aug. '09

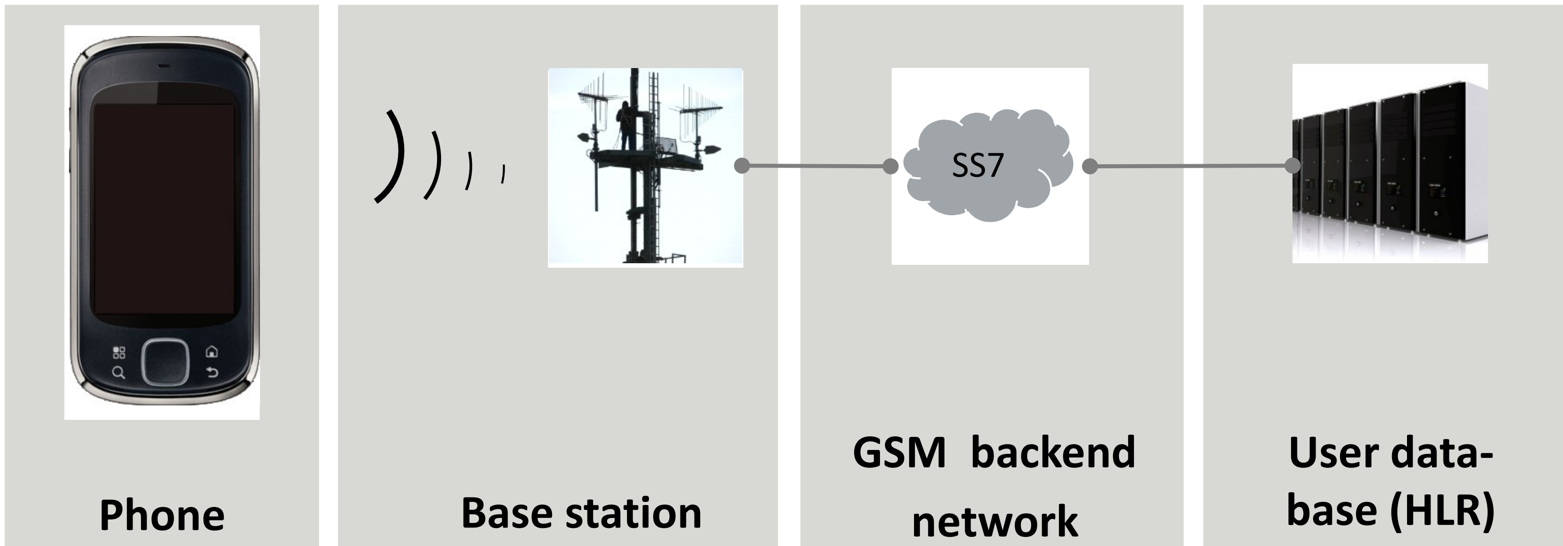
This talk introduces signal processing software to decode GSM calls

Source: GSMA press statement

Agenda

- GSM attack history
- **GSM attack vectors**
- Attacking GSM’s A5/1 encryption
- Risk scenario: GSM payment

GSM networks are victim and source of attacks on user privacy



Attack vectors

- GUI attacks, phishing
- Malware

- Weak encryption
- No network authentication

- Over-the-air software installation (security optional)

- Access to private user data

Covered in this lecture

Network operator and manufacturer can install software on a phone

Operator

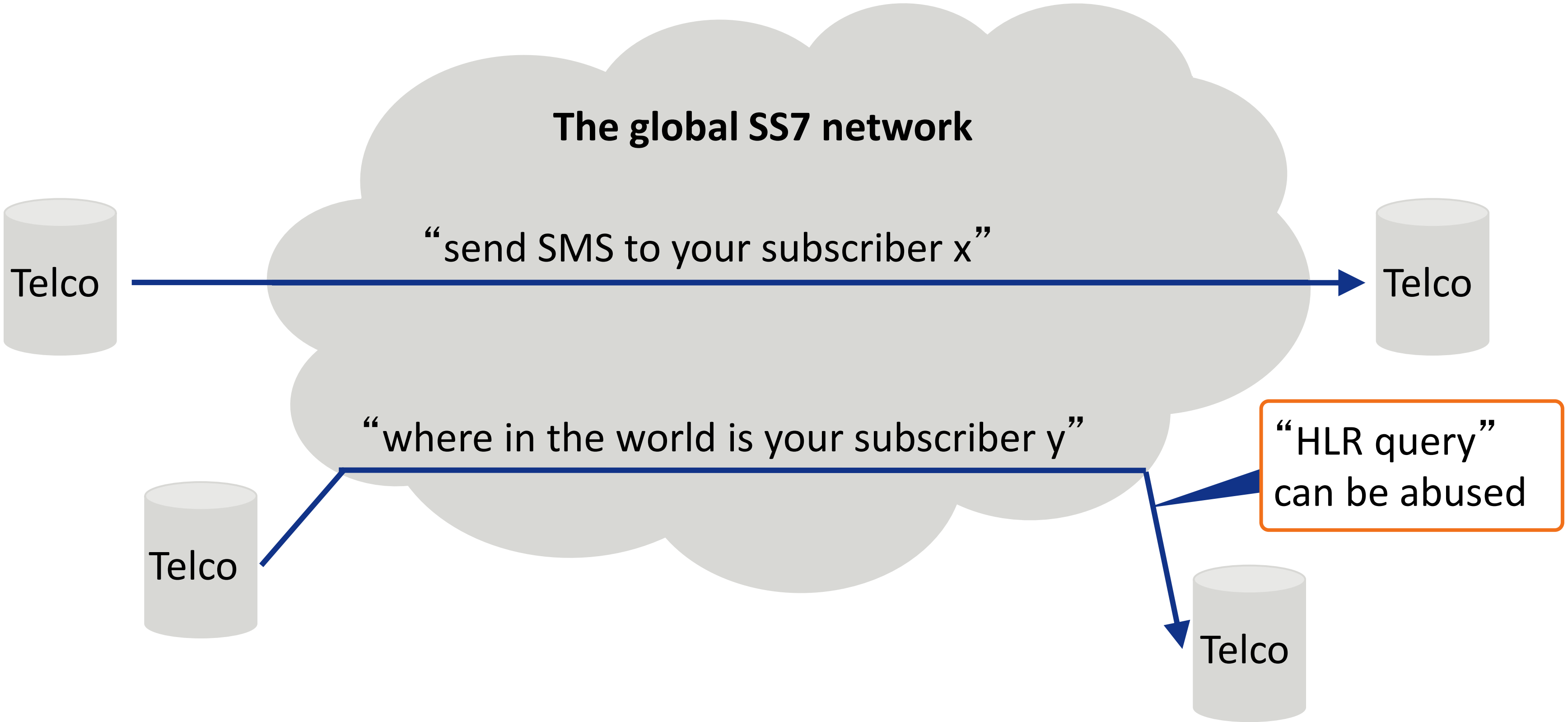
- Install or update software (SIM)
- Update service books (BlackBerry)
- Read phone book, text messages

Smart phone manufacturer

- Install, delete, update any software
- Read all data

Source:DSTK standards WiP and S@T

Telcos do not authenticate each other but leak private user data



- All telcos trust each other on the global SS7 network
- SS7 is abused for security and privacy attacks; currently for SMS spam
- SMS messages and caller ID can be spoofed

Information leaked through SS7 network disclose user location

Query	Accessible to	Location granularity
HLR query	Anybody on the Internet	General region (rural) to city part (urban)
Anytime interogation	Network operators	Cell ID: precise location

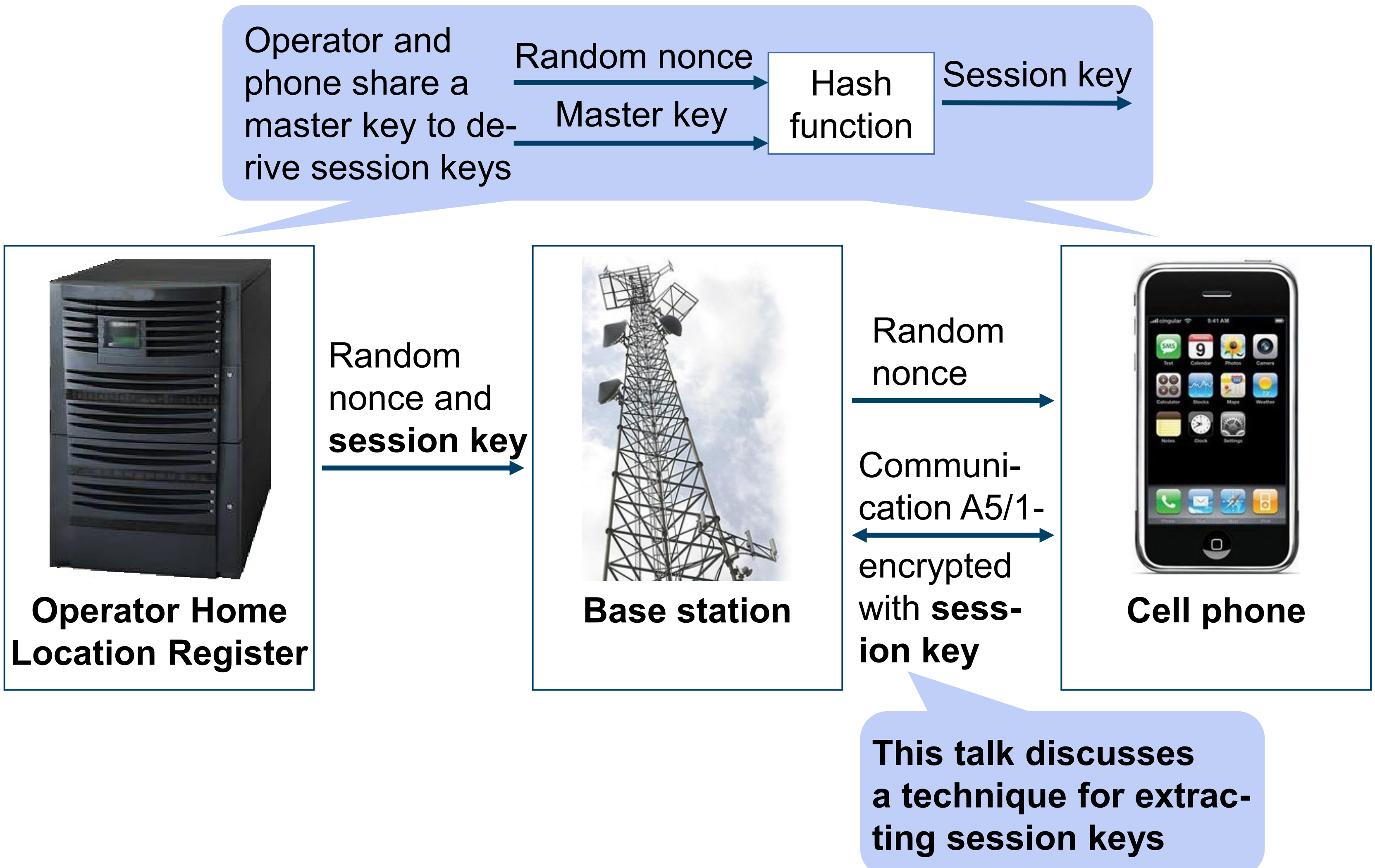
	T-Mobile Germany First digit of area code	Vodafone Germany First digit of ZIP code
Berlin	+4917103$\boxed{3}$60000	+49172001$\boxed{1}$2097
Hamburg	+4917104$\boxed{4}$00000	+49172002$\boxed{2}$2097
Frankfurt	+4917106$\boxed{6}$50000	+49172006$\boxed{6}$1097

-SMSC granularity accessible from the Internet-

Agenda

- GSM attack history
- GSM attack vectors
- **Attacking GSM's A5/1 encryption**
- Risk scenario: GSM payment

GSM uses symmetric A5/1 session keys for call privacy



A5/1 is vulnerable to pre-computation attacks

Code book attacks

- Code books break encryption functions with small keys

Secret state	**Output**
A52F8C02	52E91001
62B9320A	52E91002
C309ED0A	52E91003

- Code book provides a mapping from known output to secret state
- An A5/1 code book is 128 Petabyte and takes 100,000+ years to be computed on a PC

This talk revisits techniques for computing an A5/1 code book fast and storing it efficiently

Optimized A5/1 attack pre-computation takes just a few GPU-months

Time on single threaded CPU: **100,000+ years**

1 Parallelization
- Bitslicing increases already large number of parallel computations by a factor of 256

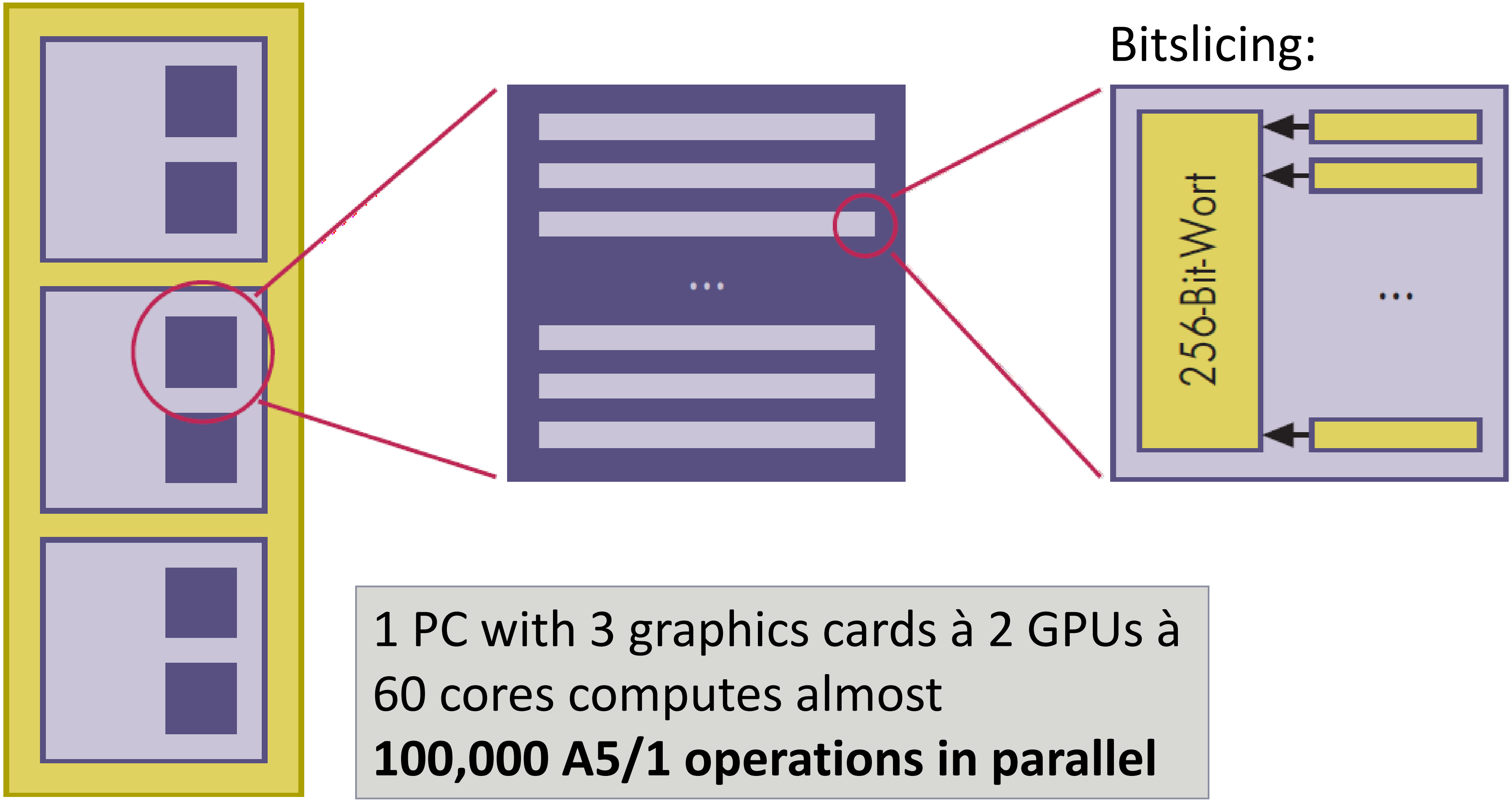
2 Algorithmic tweaks
- Compute 4 bits at once

3 Cryptographic tweaks
- Executing A5/1 for 100 extra clock cycles decreases key space by 85%

Result: **1 month** on **4 ATI GPUs**

*NVidia CUDA and ATI Brook GPUs are supported

1 GPUs allow for massive parallelization of code book computation

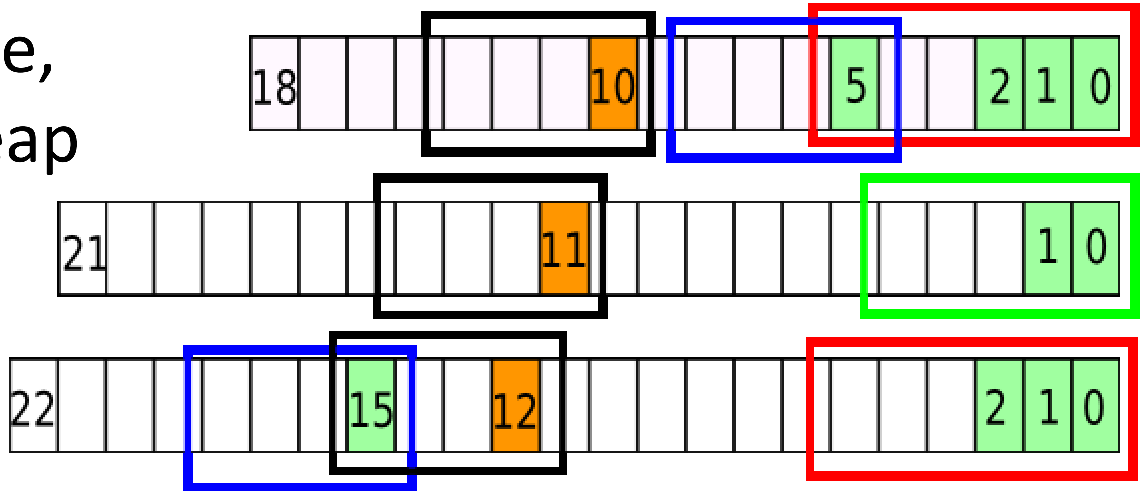


1 PC with 3 graphics cards à 2 GPUs à 60 cores computes almost **100,000 A5/1 operations in parallel**

Source:iX 5/2010: Geheimnislos

2 Algorithmic tweaks accelerate CUDA A5/1 engine significantly

- Shift registers are expensive in software, while memory is cheap
- Only a few state bits determine round function
- Trade table lookups for shifts; optimal for CUDA: 4 shifts at once



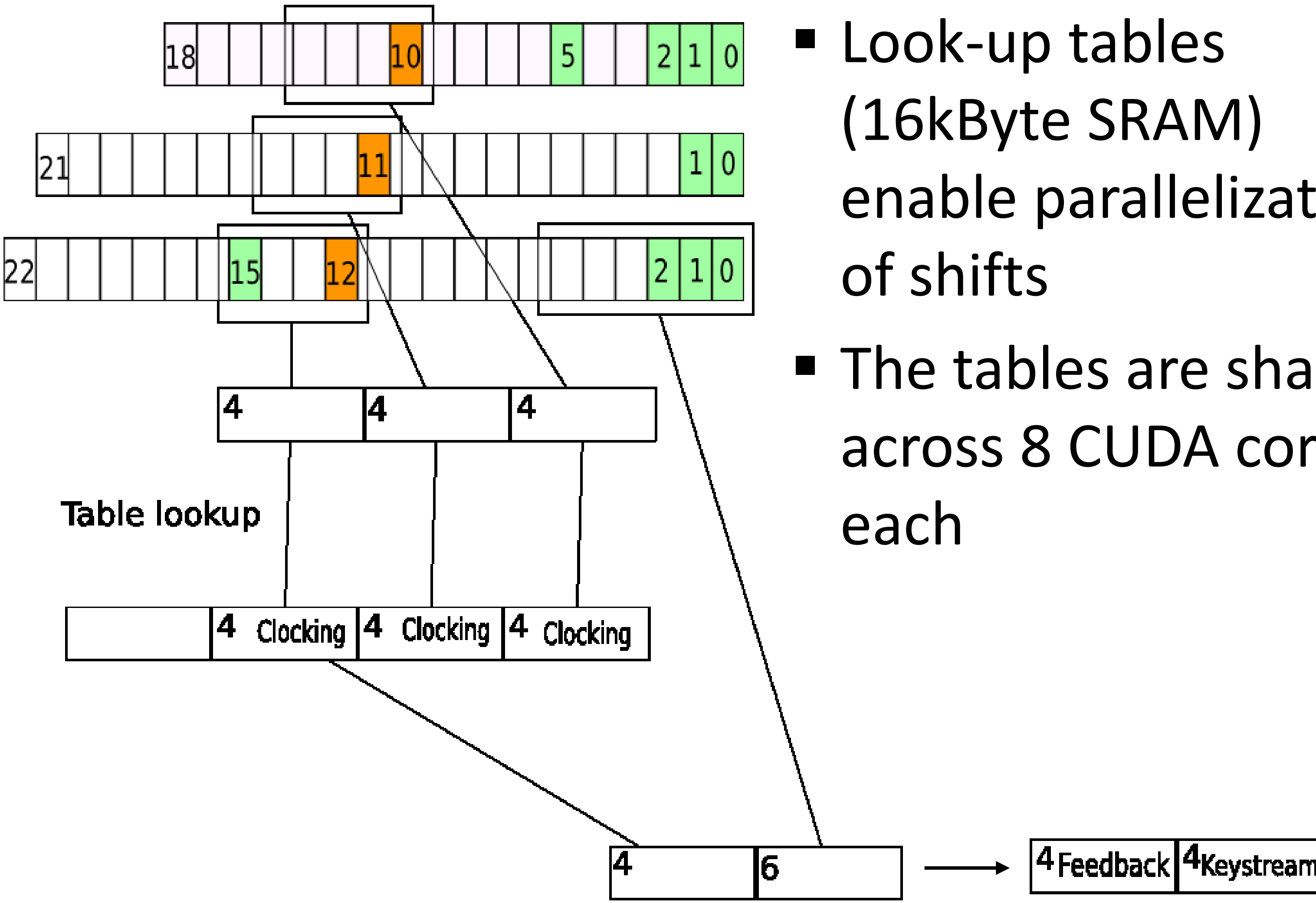
Clocking Table: 4096 x 16 bit

Table 1: 1024 x 8 bit

Table 2: 512 x 8 bit

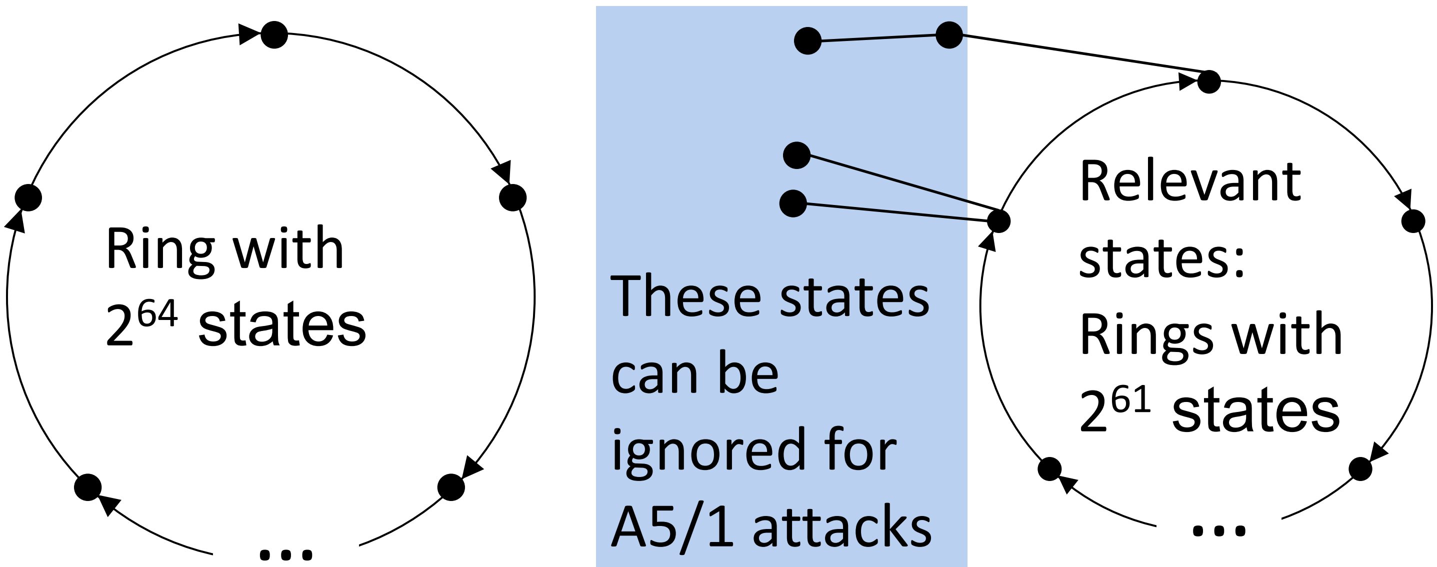
Table 3: 256 x 8 bit

❷ Balancing memory lookups and computation maximizes throughput



- Look-up tables (16kByte SRAM) enable parallelization of shifts
- The tables are shared across 8 CUDA cores each

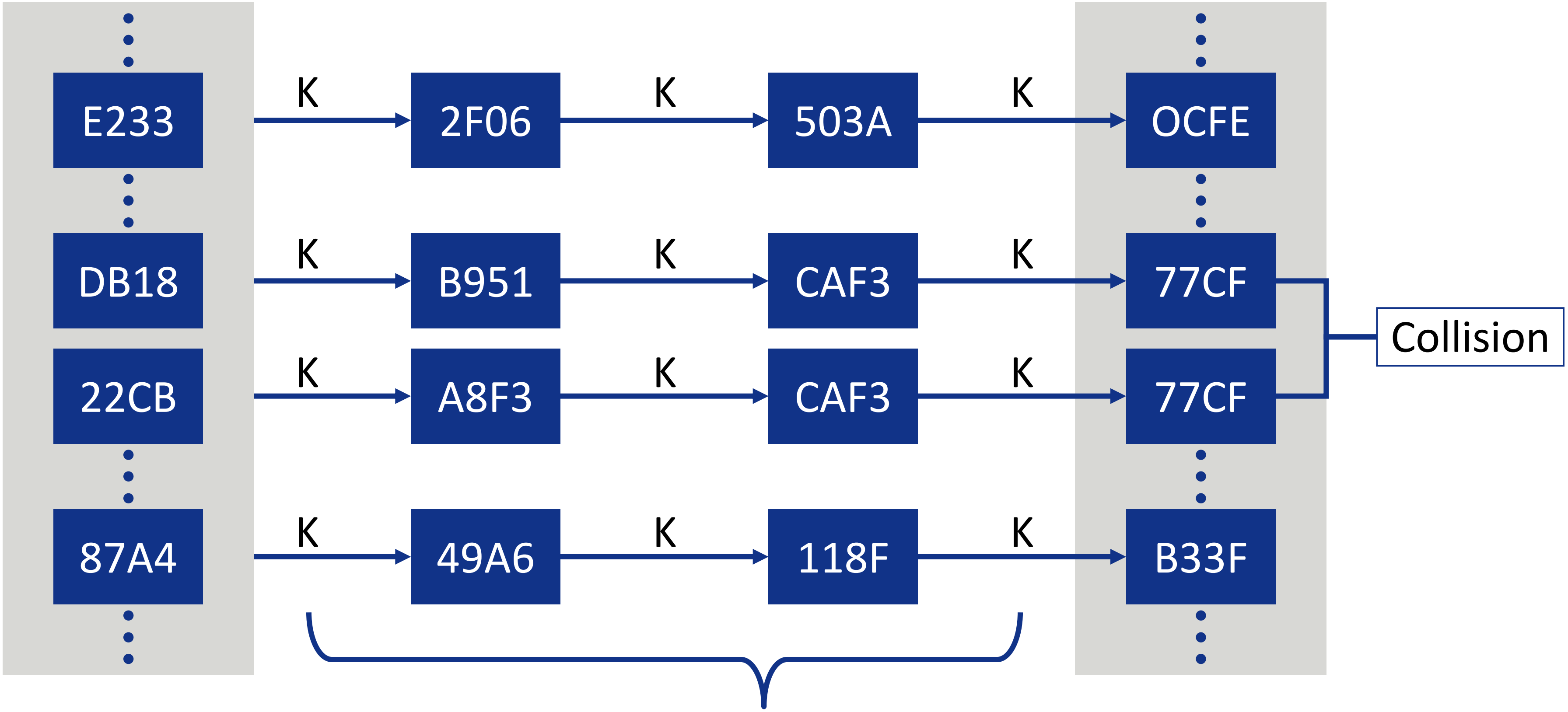
3 A5/1 key space shrinks to 2^{61} secret states



- LFSR used in older stream ciphers preserve the full output space of a function
- However, they have statistical weaknesses

- Newer stream ciphers therefore use NLFRs
- The output space of NLFSR slowly collapses
- The 100 extra A5/1 clocks in GSM shrink the output space by 85%

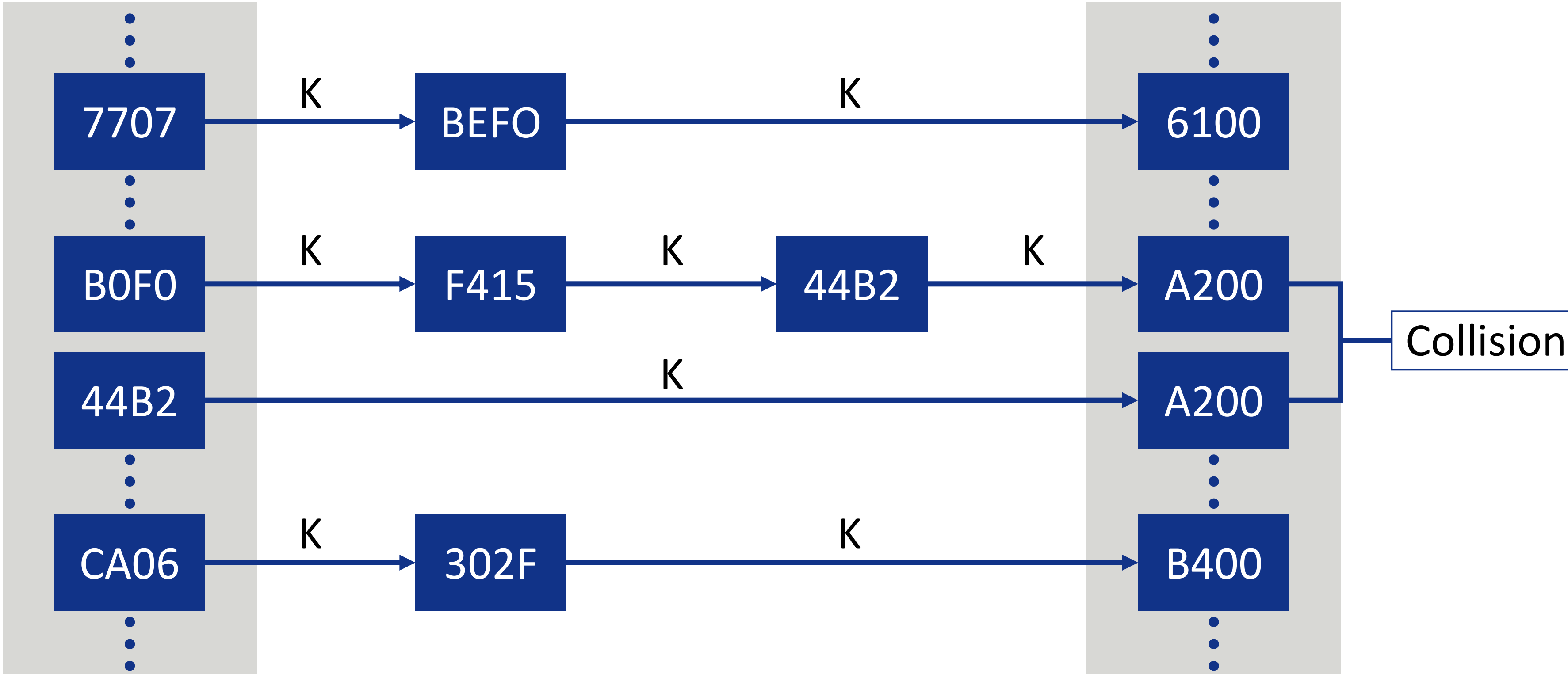
Pre-computation tables store the code book condensed



Longer chains := a) less storage, b) longer attack time

Source:c’t

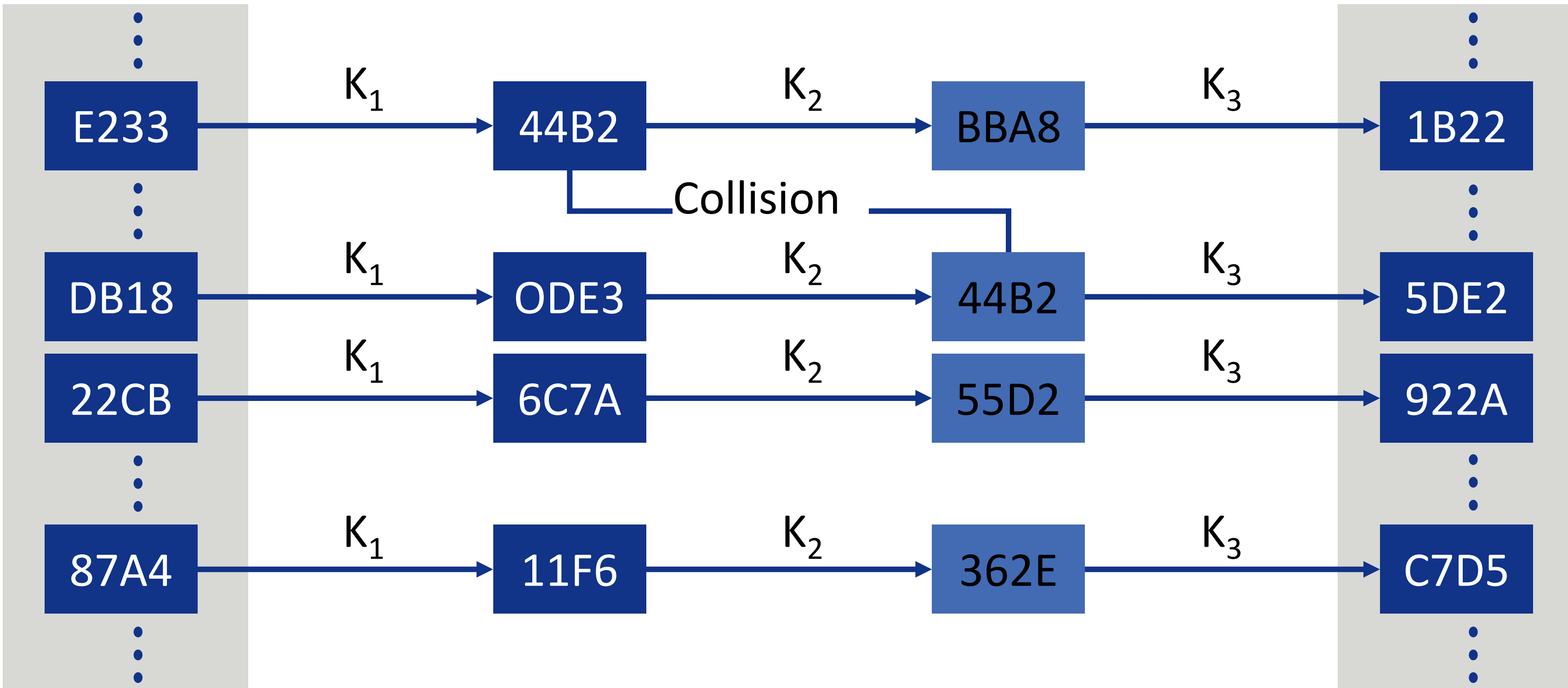
Distinguished point tables save hard disk lookups



Hard disk access only needed at distinguished points

Source:c’t

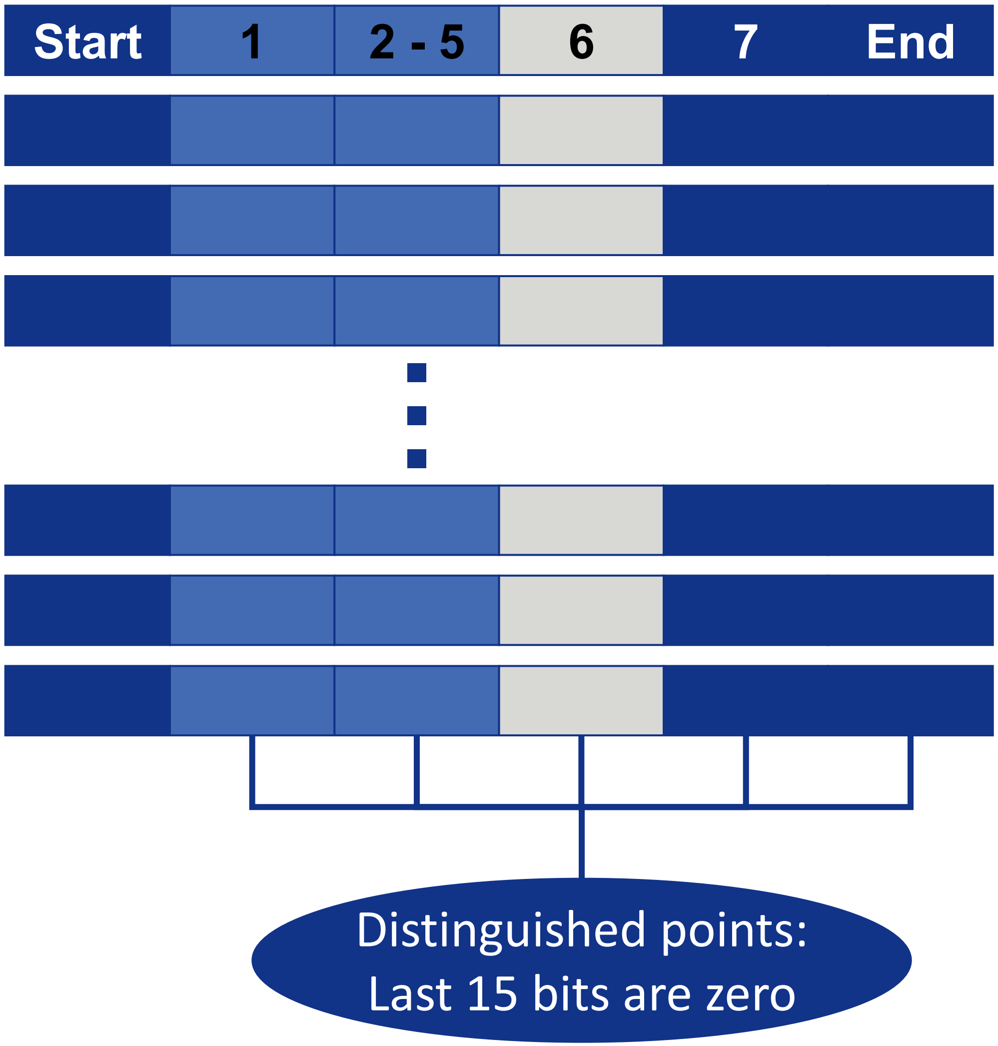
Rainbow tables mitigate collisions



Rainbow tables have no mergers, but an exponentially higher attack time

Source:c’t

The combination of both table optimizations provides best trade-off



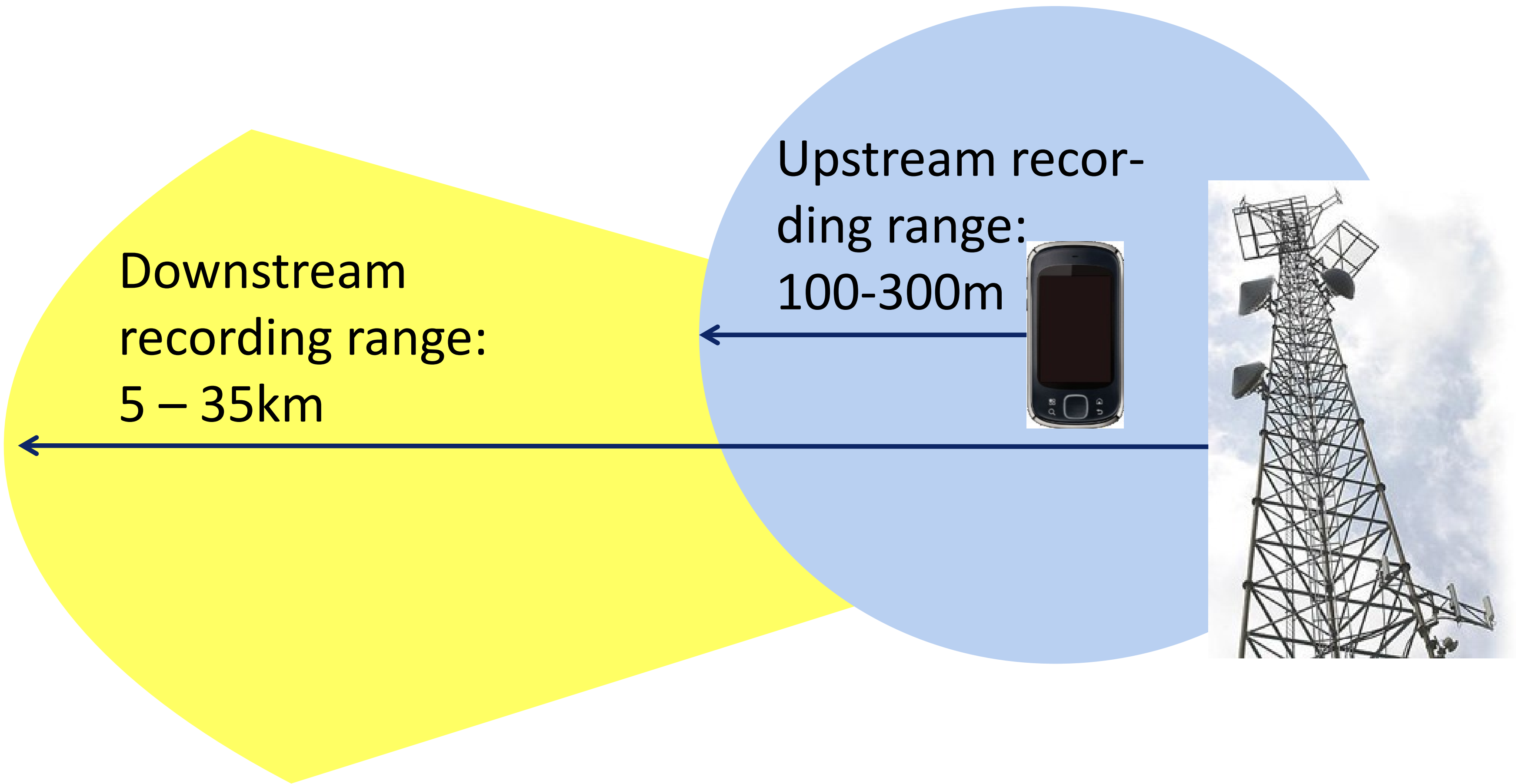
Open source components fit together in analyzing GSM calls

GnuRadio records data from air → Airprobe parses control data → Kraken cracks A5/1 key → Airprobe decodes voice

Requires
- Software radio, ie. USRP
- Recommended for up-stream: BURX board

Requires
- 2TB of rainbow tables
- CPU or ATI graphics card
- SSD/RAID for fast cracking

Downstream can be recorded from large distances



GSM discloses more known keystream than assumed in previous attacks

● Known Channel ○ Unknown Channel

	Frame with known or guessable plaintext	Assignment: Very early	Assignment: Early	Assignment: Late	Timing known through
Mobile terminated calls	1. Empty Ack after 'Assignment complete'	●	○	○	
	2. Empty Ack after 'Alerting'	○	○	○	
	3. 'Connect Acknowledge'	○	○	○	**"Stealing bits"**
	4. Idle filling on SDCCH (multiple frames)			●	
	5. System Information 5+6 (~1/sec)	◐	○	◐	
	6. LAPDm traffic	●	●	●	**Counting**
Network terminated calls	1. Empty Ack after 'Cipher mode complete'	●	●	●	**Counting frames**
	2. 'Call proceeding'	●	●	●	
	3. 'Alerting'	●	○	●	
	4. Idle filling (multiple frames)			●	**"Stealing bits"**
	5. 'Connect'	●	○	○	
	6. System Information 5+6 (~1/sec)	◐	○	◐	
	7. LAPDm	●	●	●	**Counting**

Source:GSM standards

Randomized padding would mitigate attack potential

SDCCH trace	
238530	03 20 0d 06 35 11 **2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b**
238581	03 42 45 13 05 1e 02 ea 81 5c 08 11 80 94 03 98 93 92 69 81 **2b 2b 2b**
238613	00 00 03 03 49 06 1d 9f 6d 18 10 80 00 00 00 00 00 00 00 00 00 00 00
238632	01 61 01 2b **2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b**
238683	01 81 01 2b **2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b**
238715	00 00 03 03 49 06 06 70 00 00 00 00 00 04 15 50 10 00 00 00 00 0a a8
238734	03 84 21 06 2e 0d 02 d5 00 63 01 **2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b**
238785	03 03 01 **2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b**

Padding in GSM has traditionally been predictable (2B)

Every byte of randomized padding increasing attack cost by two orders of magnitude!

Randomization was specified in 2008 (TS44.006) and should be implemented with high priority

Additionally needed: randomi-zation of system information msg.

GSM's security must be overhauled

Short term

Configuration tweaks and small standard modifi-cations render some GSM crackers useless, but do not prevent cracking using newer tools.

Long term

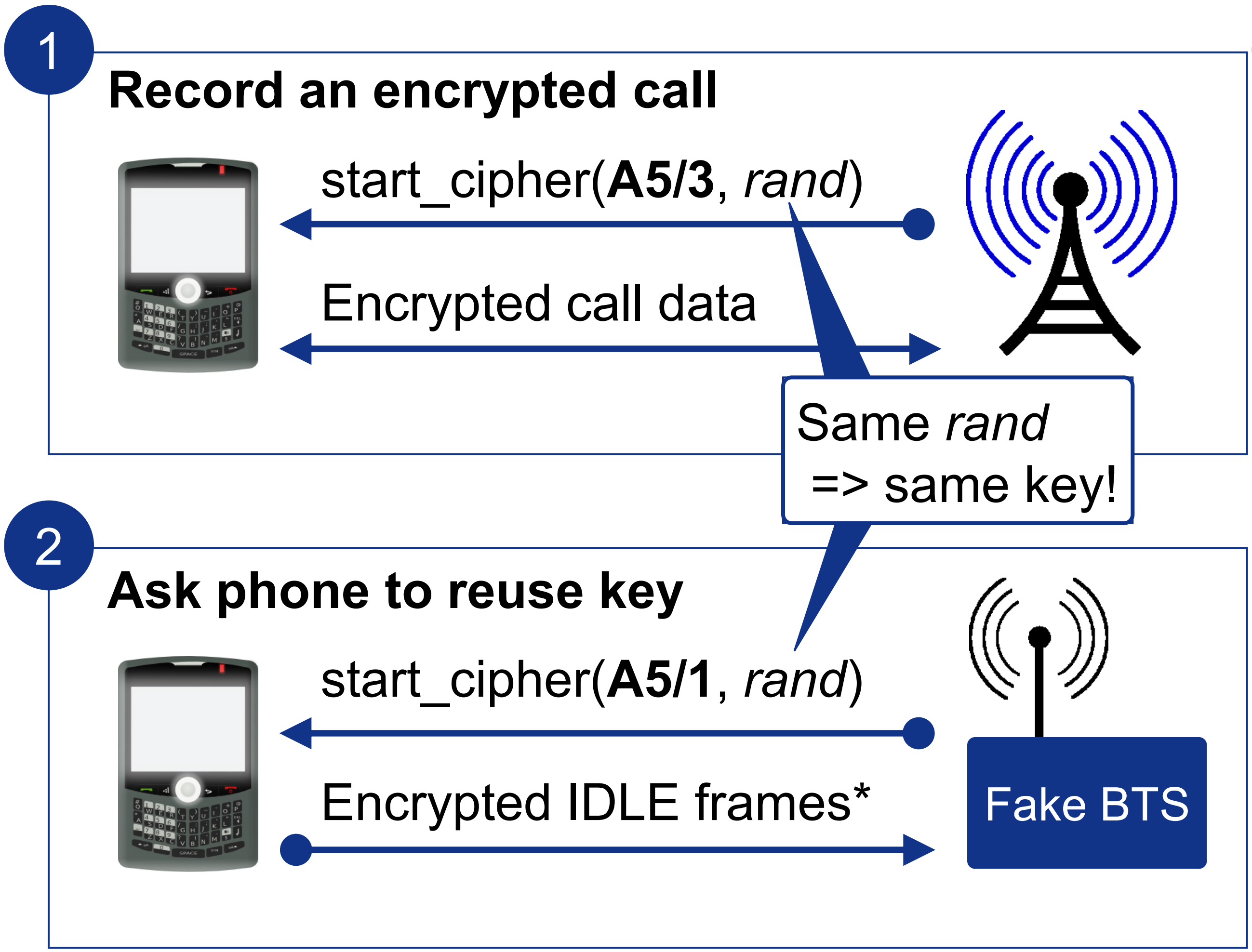
Upgrading GSM's encryption function should be a mandatory security patch

A5/1 ▶ A5/?

Replacing A5/1 with A5/3 may not be enough:

- The A5/3 cipher is academically broken
- The same master keys are used for A5/1 and A5/3 (weakest link security)

A5/3 can be cracked in a semi-active attack



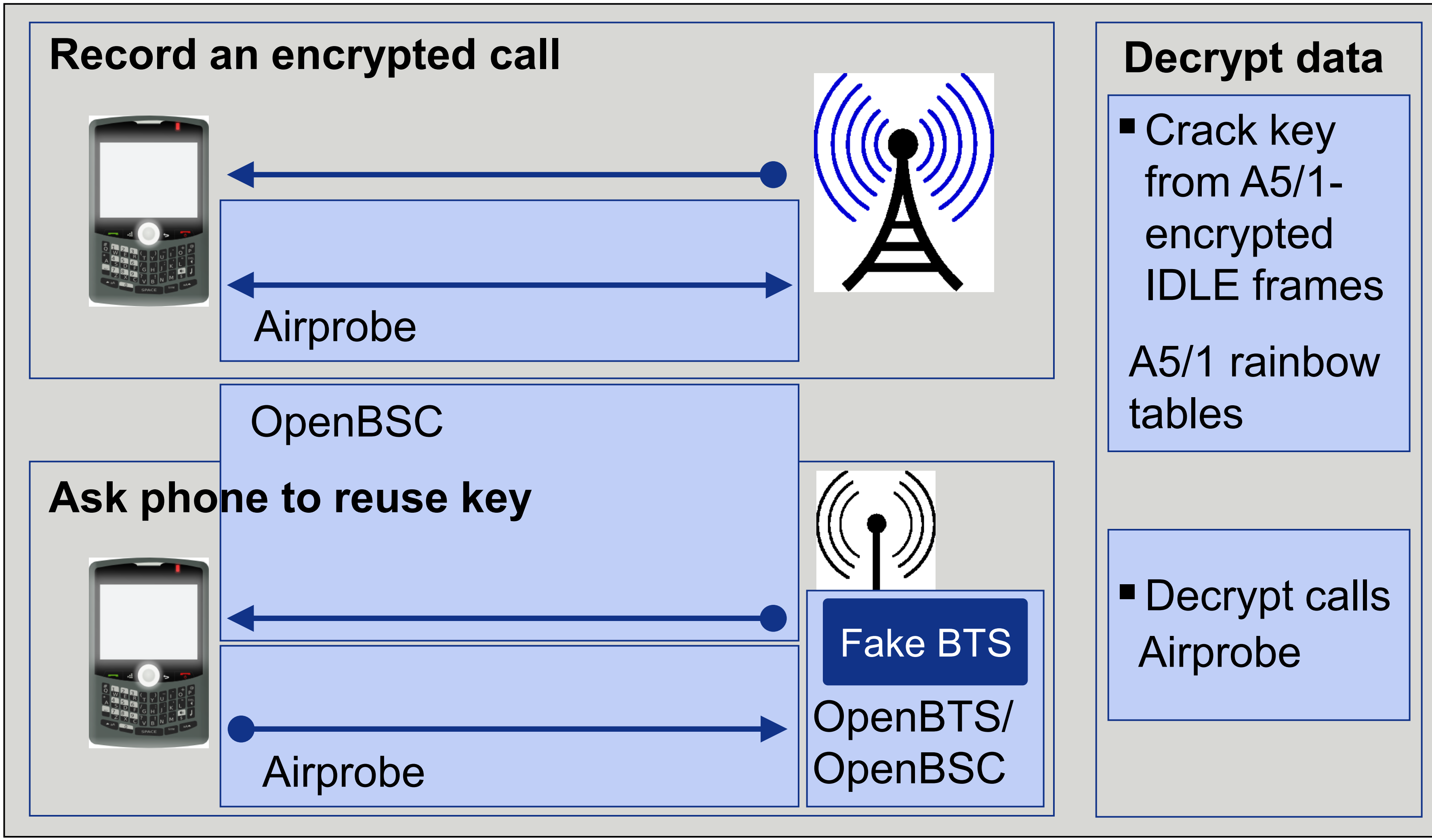
3

Decrypt data

- Decrypt call or SMS with key cracked from A5/1 trans-actions
- A5/3 cracker done ✓ (but not passive and not realtime)

*IDLE frames contain known plaintext

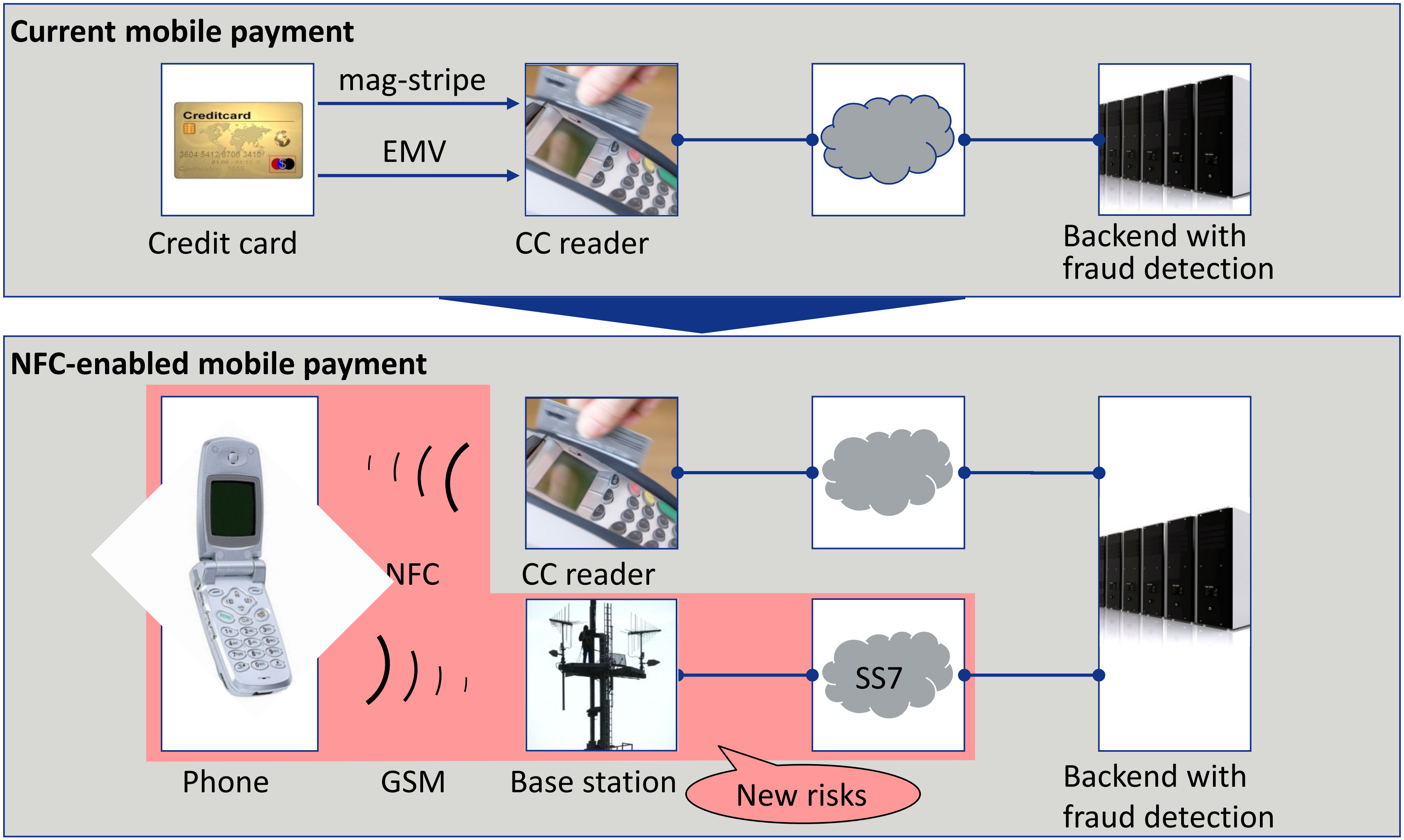
All tools needed for the semi-active attack are openly available



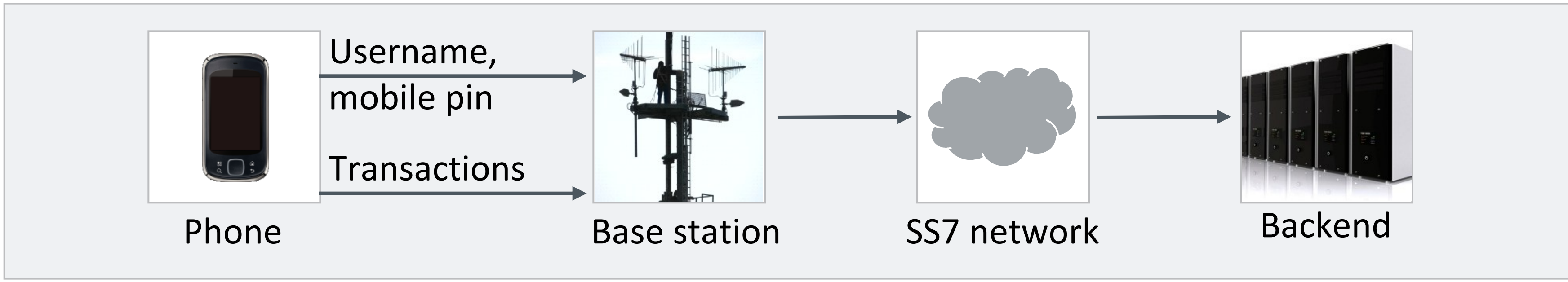
Agenda

- GSM attack history
- GSM attack vectors
- Attacking GSM's A5/1 encryption
- **Risk scenario: GSM payment**

New applications like GSM payment extend the attack incentives against GSM



GSM payment carries large risk



Easiest attack: **Break encryption**

- USSD data and sometimes SMS are weakly encrypted on the air interface.
- Attack limit: The data can only be intercepted in the vicinity of the phone, up to one mile. Therefore, attacks are location-limited.

GSM weaknesses pose a manage-able fraud risk but large publicity risk through script-kiddie attacks

Scalable attack: **Network sniffing**

•USSD data and SMS traverse networks, operator systems and the USSD provider **unencrypted**

•In low-income markets where GMS payment is popular, the cost of "buying" an insider are relatively low

Wide distribution of unencrypted login data poses an unmanageable risk of a wide-scale incident

Even legacy phones with current SIM cards can execute strong cryptography

	3DES	Software ECC	Hardware RSA
Availability	● In almost all SIM cards	◕ Deployable through OTA	◔ In high-security SIM cards
Implementation cost	○ Small applet (<5k)	◑ Large applet (>10k)	● New SIM cards
Cryptographic strength	◕	●	●
Resistance to side-channel & fault injection	◕	◔	◑

The available 3DES encryption is acceptably strong for micro-payment. Better protection requires better SIM cards

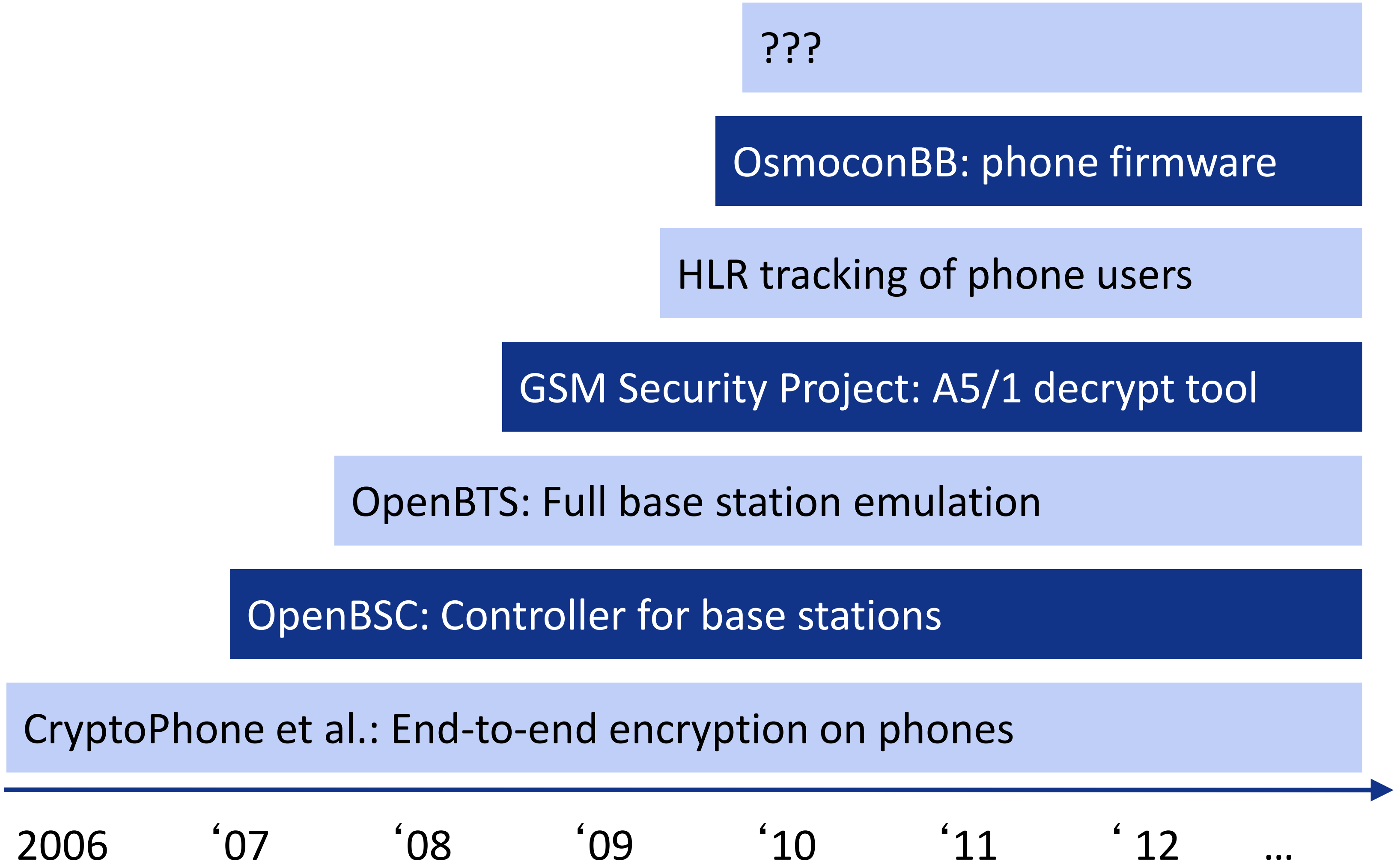
GSM should currently be used as an untrusted network, just like the Internet

Threat	Investment	Scope	Mitigation
Fake base station	Low	Local	Application encryption & trust anchor
Passive intercept voice + SMS	Low	Local	
Passive intercept data	Currently not possible		
Phone virus / malware	Medium to high	Large	Trust anchor
Phishing	High	Large	

Cell phone networks do not provide state-of-the art security.

Protection must be embedded in the phones and locked away from malware.

Open research into GSM security grows exponentially and so will the attacks



Deepsec slides

Workshop Agenda

Day 1

- 9:30 – 13:00: GSM theory
- Lunch
- 14:00 – 17:00: GSM crypto attacks
  - Airprobe + Kraken
  - A5/3 downgrade
- 17:30 – 18:30: SIM card attacks
  - SIM sniffing
  - Over-the-air updates

Day 2

- 9:30 – 11:30: GSM advanced theory
- 12:00 – 16:00: Active attacks
  - [Lunch]
  - Uplink/downlink fuzzing
  - IMSI catching
- 16:15 – 17:30: Tracking attacks
  - SS7, RRLP, HLR
- Open lab

Workshop Agenda

Day 1

9:30 – 13:00	GSM theory
	Lunch
14:00 – 17:00	GSM crypto attacks Airprobe + Kraken A5/3 downgrade
17:30 – 18:30	SIM card attacks • SIM sniffing • Over-the-air updates

Day 2

9:30 – 11:30	GSM advanced theory
12:00 – 16:00	Active attacks [Lunch] • Uplink/downlink fuzzing • IMSI catching
16:15 – 17:30	Tracking attacks • SS7, RRLP, HLR
	Open lab

Workshop Agenda

Day 1

9:30 – 13:00	GSM theory
	Lunch
14:00 – 17:00	GSM crypto attacks • Airprobe + Kraken • A5/3 downgrade
17:30 – 18:30	SIM card attacks SIM sniffing Over-the-air updates

Day 2

9:30 – 11:30	GSM advanced theory
12:00 – 16:00	Active attacks [Lunch] • Uplink/downlink fuzzing • IMSI catching
16:15 – 17:30	Tracking attacks • SS7, RRLP, HLR
	Open lab

Workshop Agenda

Day 1

9:30	GSM theory
13:00	
	Lunch
14:00	GSM crypto attacks • Airprobe + Kraken • A5/3 downgrade
17:00	
17:30	SIM card attacks • SIM sniffing • Over-the-air updates
18:30	

Day 2

9:30	GSM advanced theory
11:30	
12:00	Active attacks [Lunch] • Uplink/downlink fuzzing • IMSI catching
16:00	
16:15	Tracking attacks • SS7, RRLP, HLR
17:30	
	Open lab

Workshop Agenda

Day 1

9:30 GSM theory

13:00

Lunch

14:00 GSM crypto attacks
- Airprobe + Kraken
- A5/3 downgrade

17:00

17:30 SIM card attacks
- SIM sniffing
- Over-the-air updates

18:30

Day 2

9:30 GSM advanced theory

11:30

12:00 Active attacks

[Lunch]

Uplink/downlink fuzzing

IMSI catching

16:00

16:15 Tracking attacks
- SS7, RRLP, HLR

17:30

Open lab

Workshop Agenda

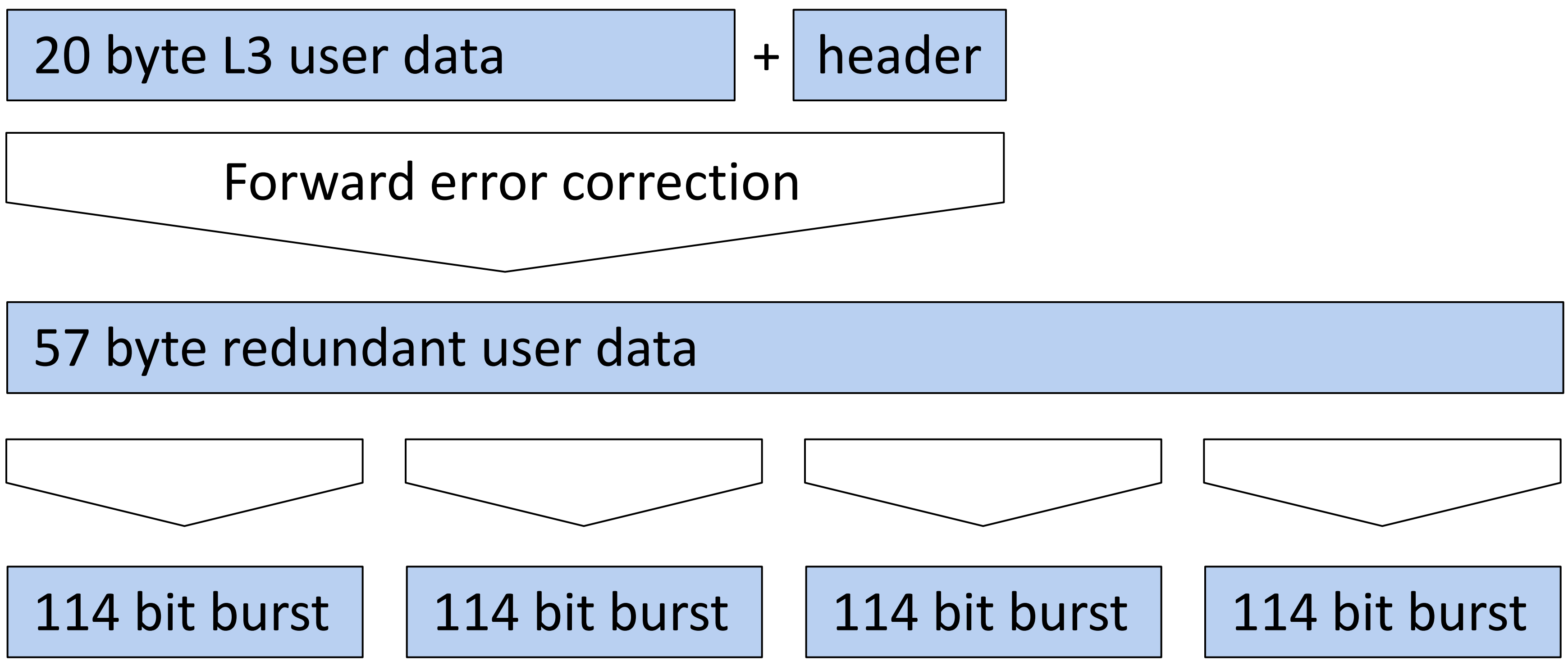
Day 1

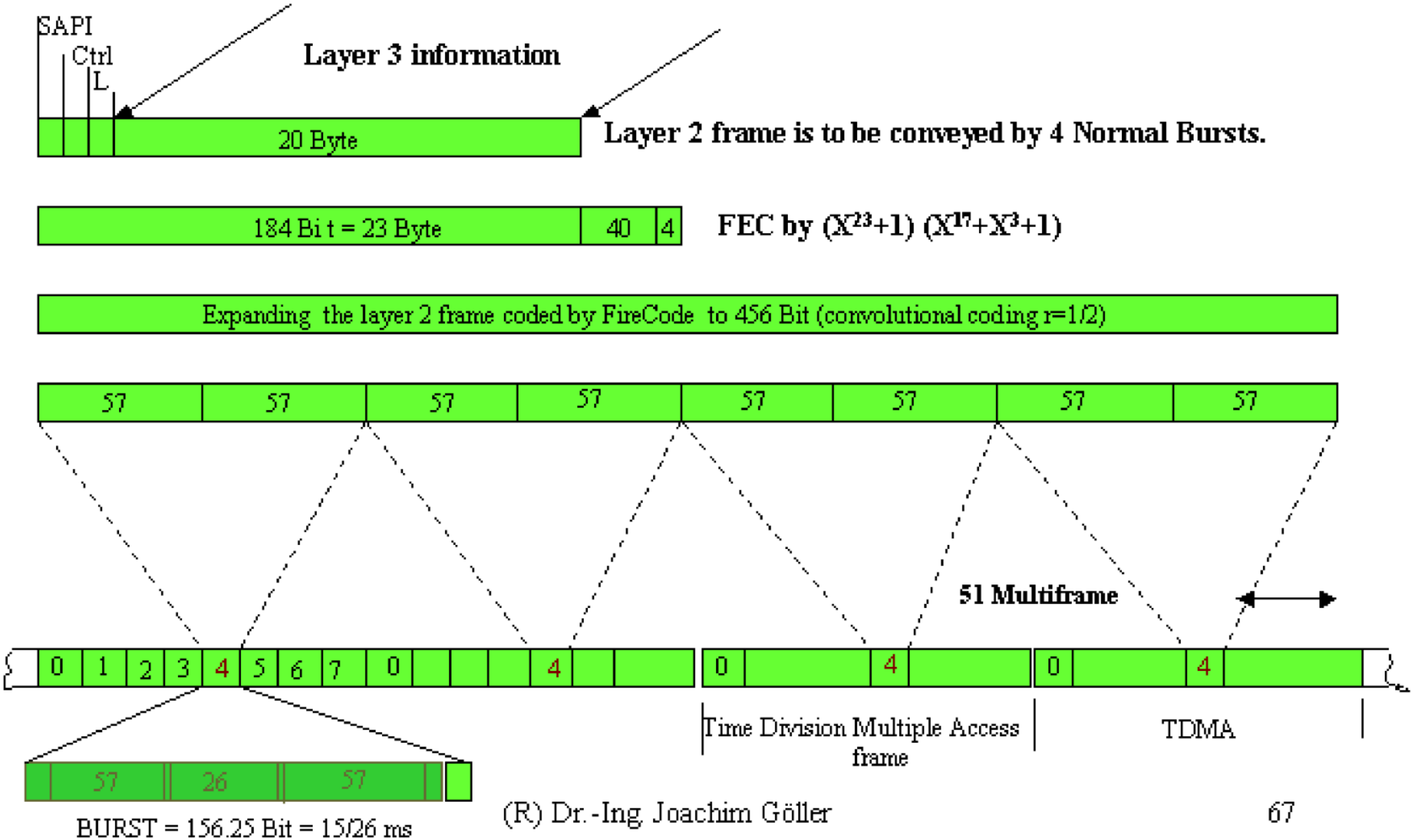
9:30	GSM theory
13:00	
	Lunch
14:00	GSM crypto attacks • Airprobe + Kraken • A5/3 downgrade
17:00	
17:30	SIM card attacks • SIM sniffing • Over-the-air updates
18:30	

Day 2

9:30	GSM advanced theory
11:30	
12:00	Active attacks [Lunch] • Uplink/downlink fuzzing • IMSI catching
16:00	
16:15	Tracking attacks SS7, RRLP, HLR
17:30	
	Open lab

GSM packets are expanded and spread over four frames




SAPI
Ctrl
L
Layer 3 information
20 Byte
Layer 2 frame is to be conveyed by 4 Normal Bursts.
184 Bi t = 23 Byte
40
4
FEC by $(X^{23}+1)(X^{17}+X^{3}+1)$
Expanding the layer 2 frame coded by FireCode to 456 Bit (convolutional coding r=1/2)
57
57
57
57
57
57
57
57
51 Multiframe
0
1
2
3
4
5
6
7
0
4
0
4
0
4
Time Division Multiple Access frame
TDMA
57
26
57
BURST = 156.25 Bit = 15/26 ms

Questions?

GSM project supported by

Tables, Airprobe, Kraken	srlabs.de
GSM Project Wiki	reflextor.com/trac/a51

Karsten Nohl	**nohl@srlabs.de**

Message Title

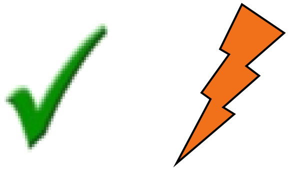
- ffff
  - fff
    - fff
      - ffff

A Heading

- Text
- Text
  - …

ToDos

- Box for internal ToDo in document



Textbox for additional important thoughts