

SIM card exploitation

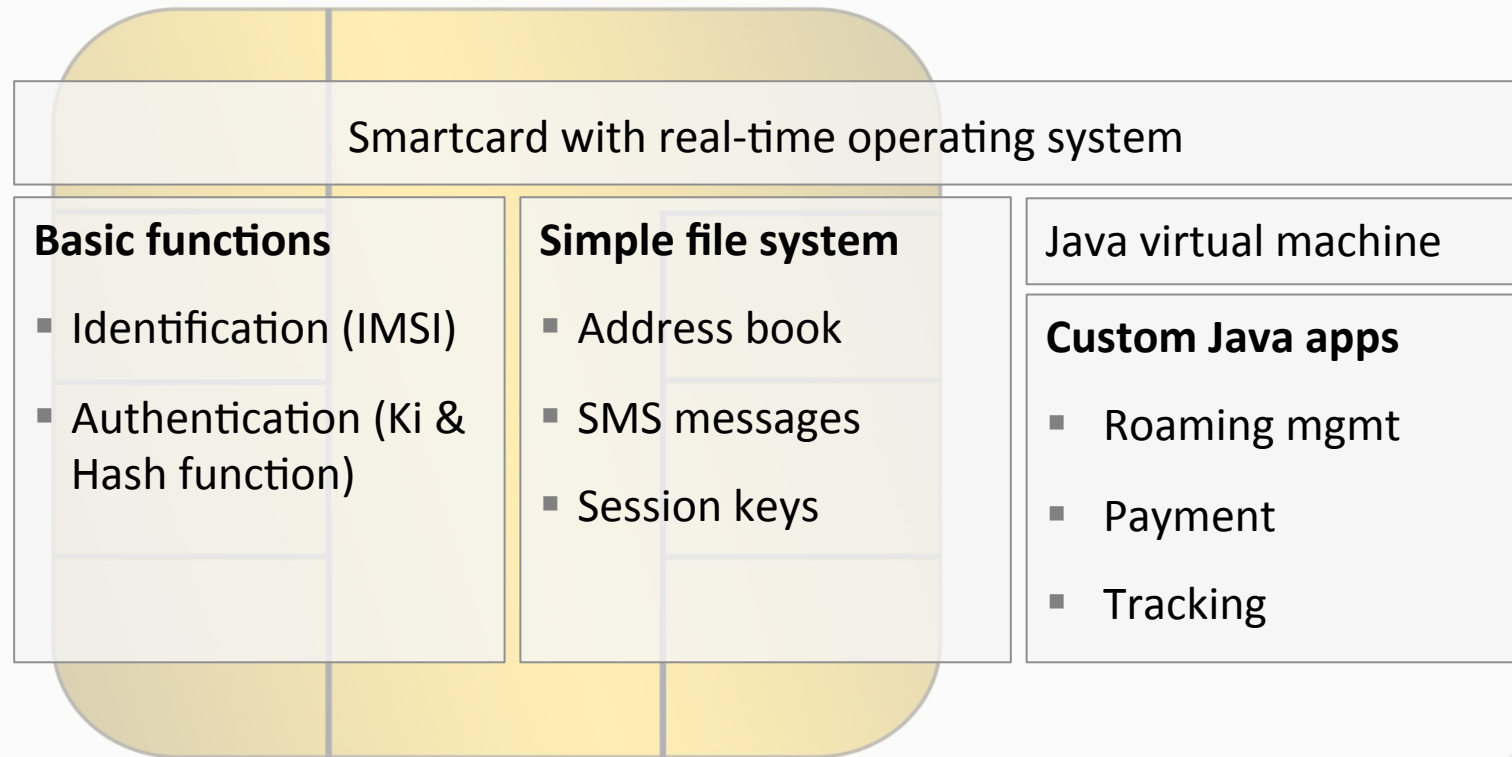
Karsten Nohl <nohl@srlabs.de>



SECURITY
RESEARCH
LABS

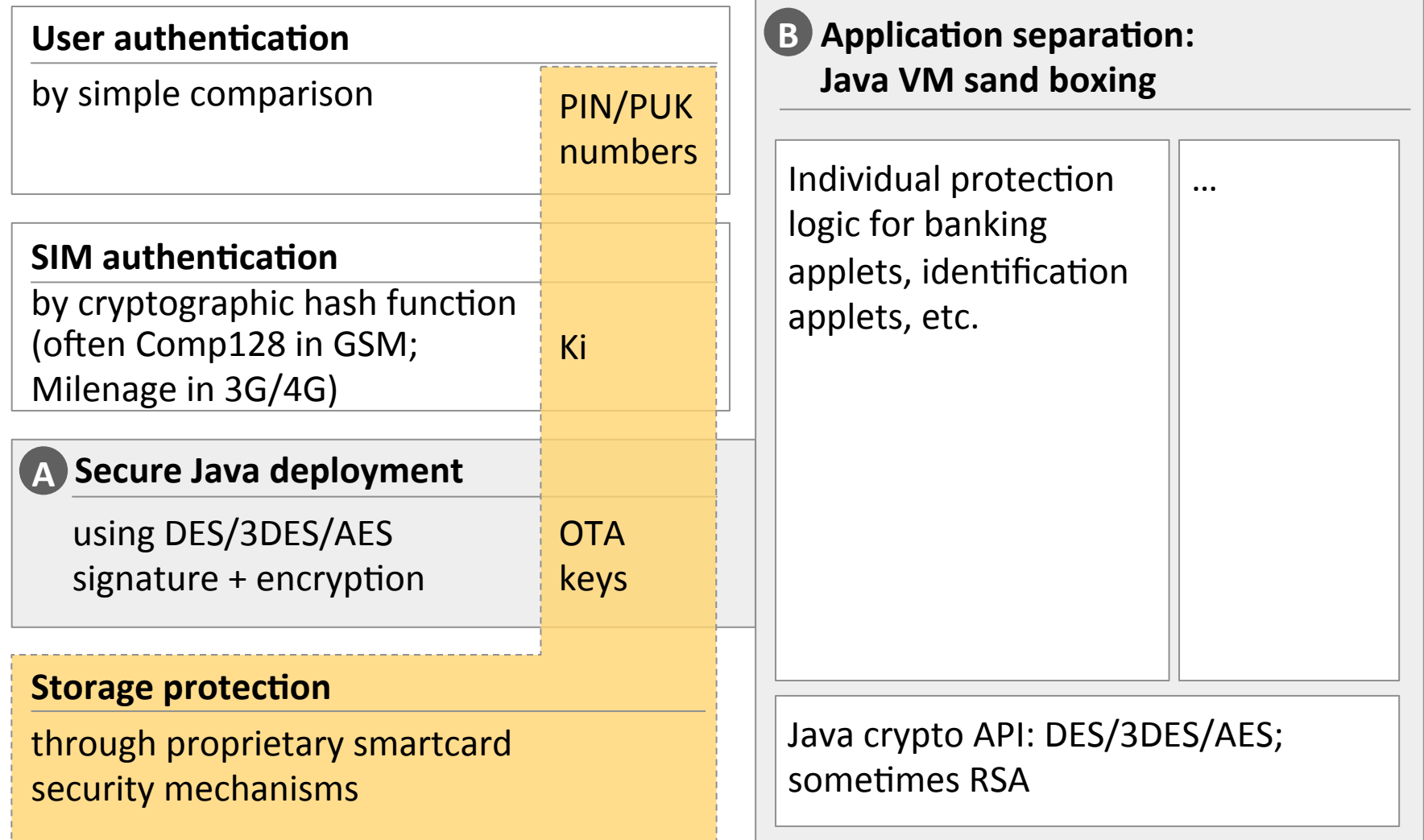
SIM cards are fully programmable computer systems

Applications on modern SIM card



SIM have many security layers from smartcards to cryptography and Java process separation

SIM card includes various protection mechanisms



Agenda

SIM card background

 **A** **Getting on to the SIM**

B Stealing SIM secrets

OTA security level is chosen by server while SIM enforces mandatory minimum level

ILLUSTRATIVE

Binary SMS communication

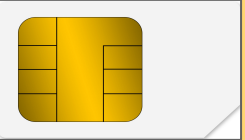
OTA server
initiates
remote
transaction

Target app / key set #

Command – possibly encrypted and/or signed	Used security level	Reque- sted security level
---------------------------------------------------------------	---------------------------	-------------------------------------

Response protected
according to request,
but not below minimum
level stored on card

SIM card stores multiple
key sets, possibly with
different protection levels



Key set 3

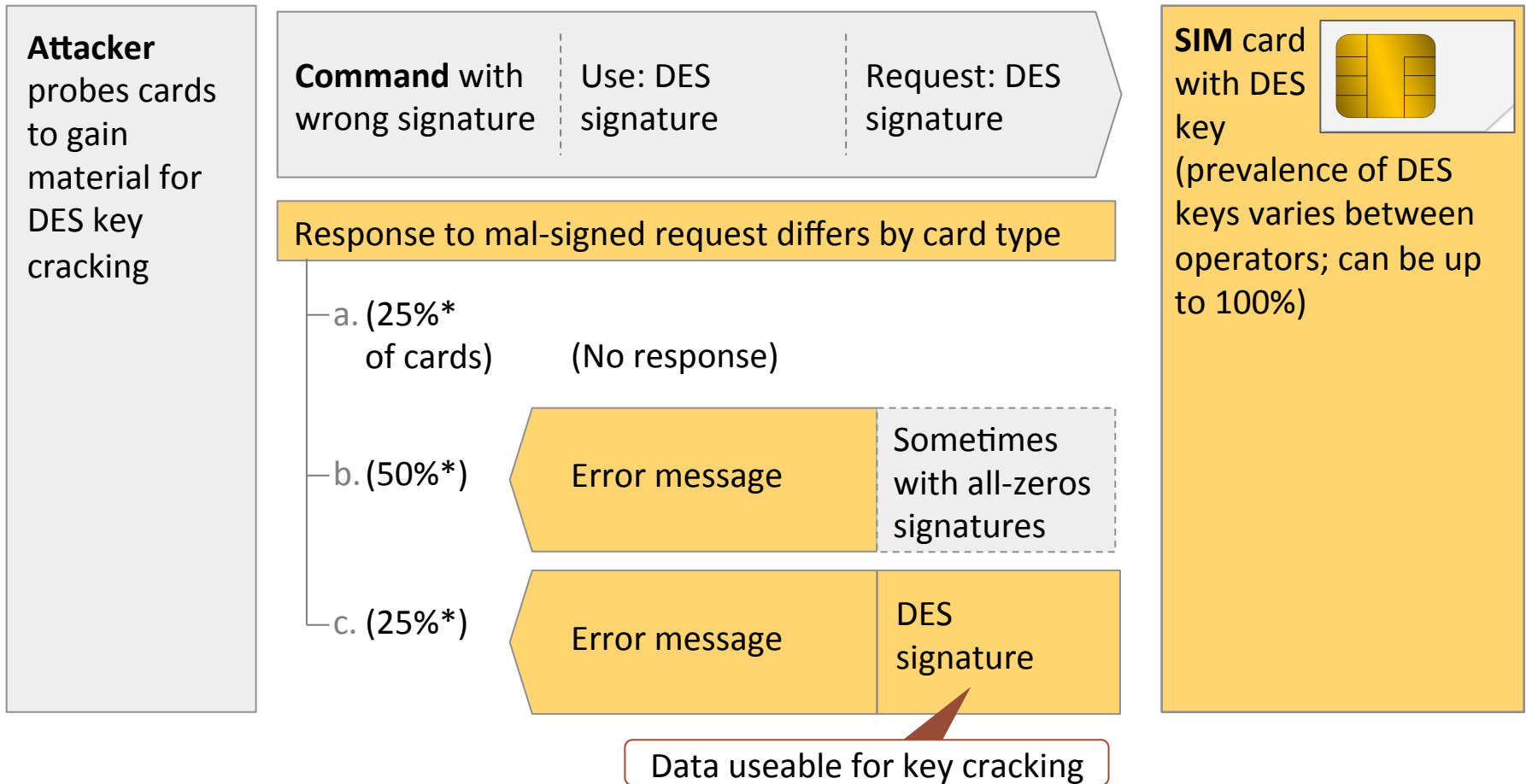
Key set 2

Key set 1

	DES	3DES	AES	Man- datory
Encry- ption	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
Signa- ture	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="checkbox"/>

OTA error handling is underspecified, possibly opening attack surface

Binary SMS communication



OTA DES do not withstand key cracking

Challenge: Derive 56 bit DES key from OTA response signature

Cracking strategies

Investment

Cracking time

Be patient

Brute force on GPU

EUR 1.000

6 months

Throw money at it

Brute force on FPGA cluster

EUR 50.000

1 day

Ride the rainbow

Time-memory trade-off
using large hard disks & GPU

EUR 1.500 +
1 year pre-computation

1 minute
(but <100% success rate)

Only possible when OTA
response is fully predictable

Attacker SMS can request DES-signed SMS response with fully predictable content

Attack-specific features

Command packet

is sent by the attacker to provoke response

UDHI	PID	DCS	UDH	CPL	CHL	SPI	KIc	KID	TAR	CNTR	PCNTR	CC	Data
1	127	246	027000	Packet length	Header length	<ul style="list-style-type: none">No cipheringSignPoR request	No cipher	DES signature	App	01	Padding counter	Rand. invalid	Generic command

Packet details:

0 0 0 1 0 0 1 0 0 0 1 0 1 0 0 1

- No ciphering
- Cryptographic checksum
- Do not cipher PoR
- Sign PoR
- Send PoR in any case

Response packet

may offer attack surface

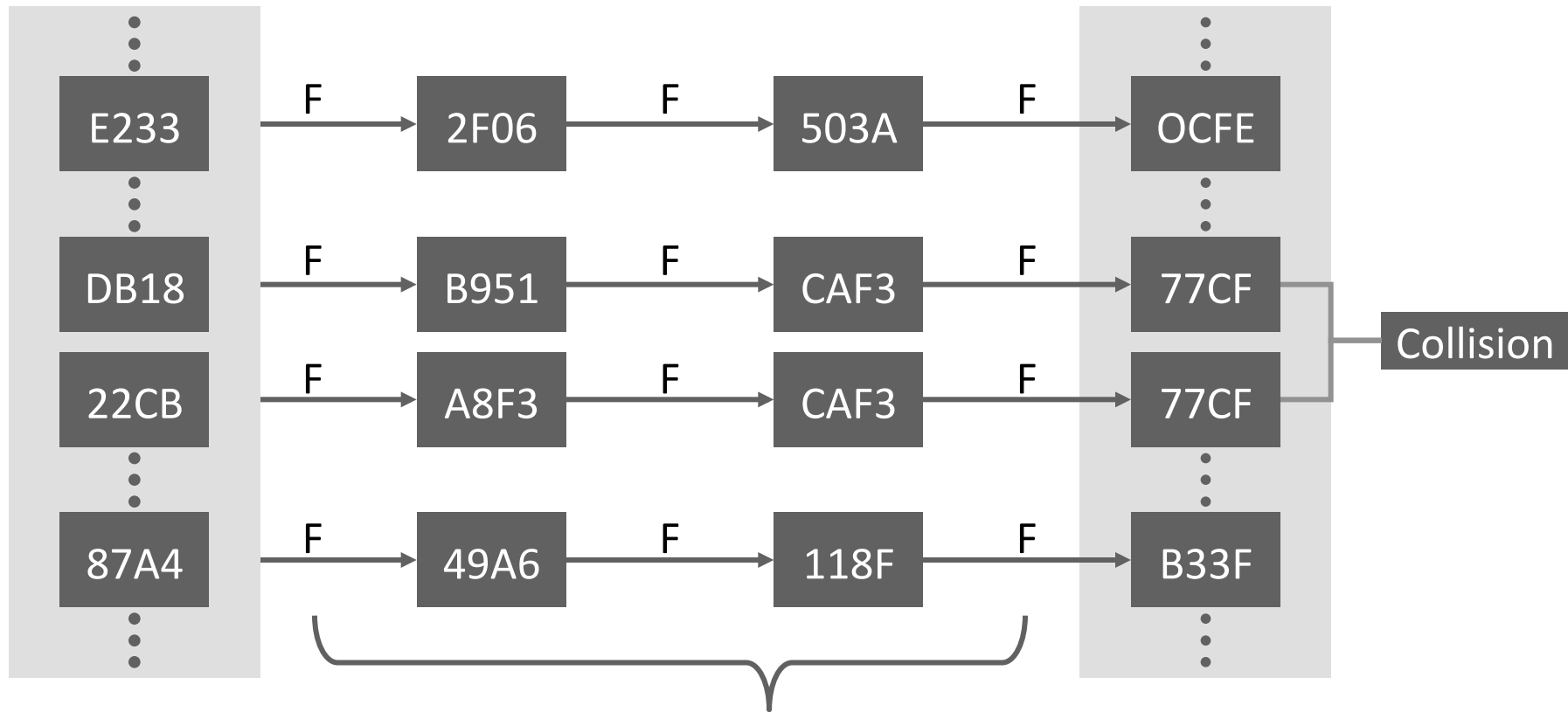
UDH	RPL	RHL	TAR	CNTR	PCNTR	Status Code	CC	Data
027100	Packet length	Header length	App	01	Padding counter	Status Code	Crypto-Checksum	Response

–or–

No response

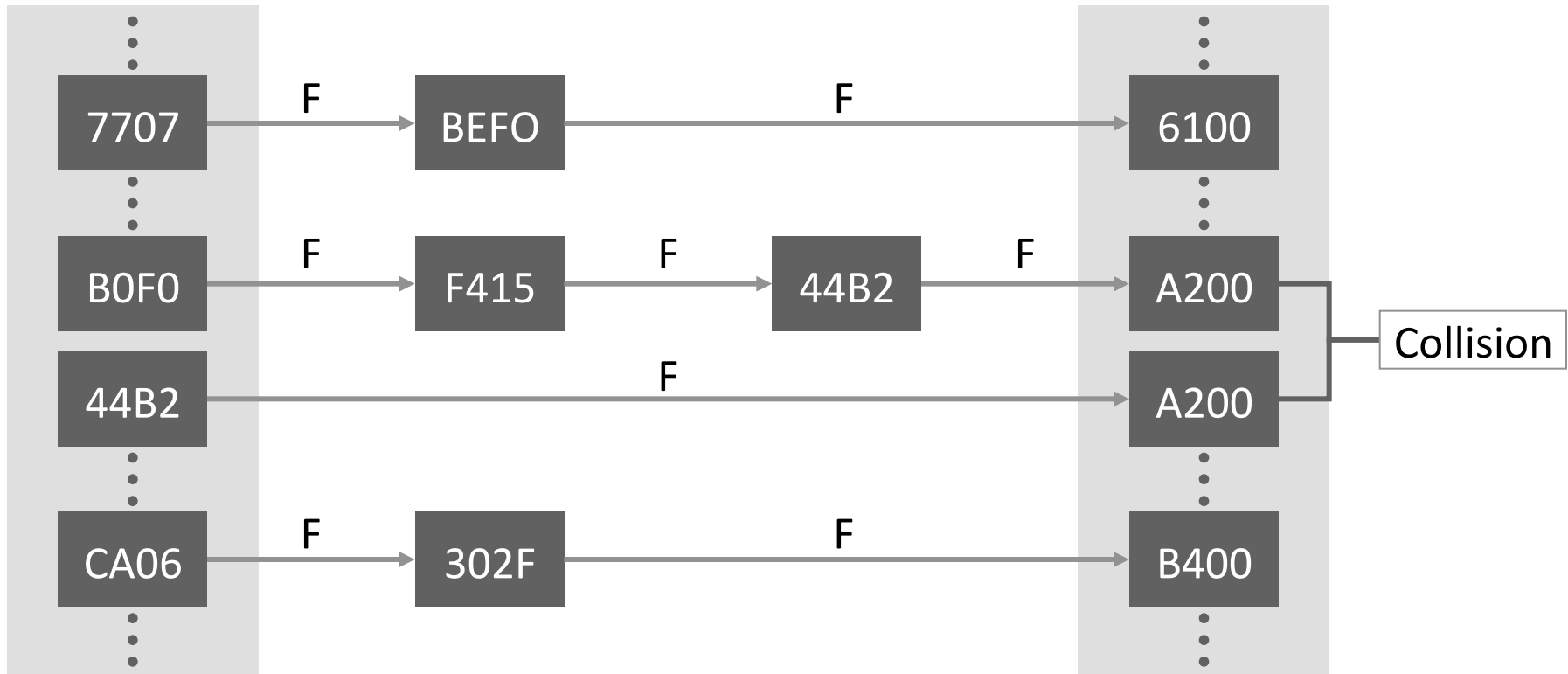
Signature over predictable data useable for rainbow table key cracking

Pre-computation tables store DES code book in condensed form



The uncondensed code book is 100's of Petabyte. Tables provide a **trade-off**: Longer chains := a) less storage, b) longer attack time

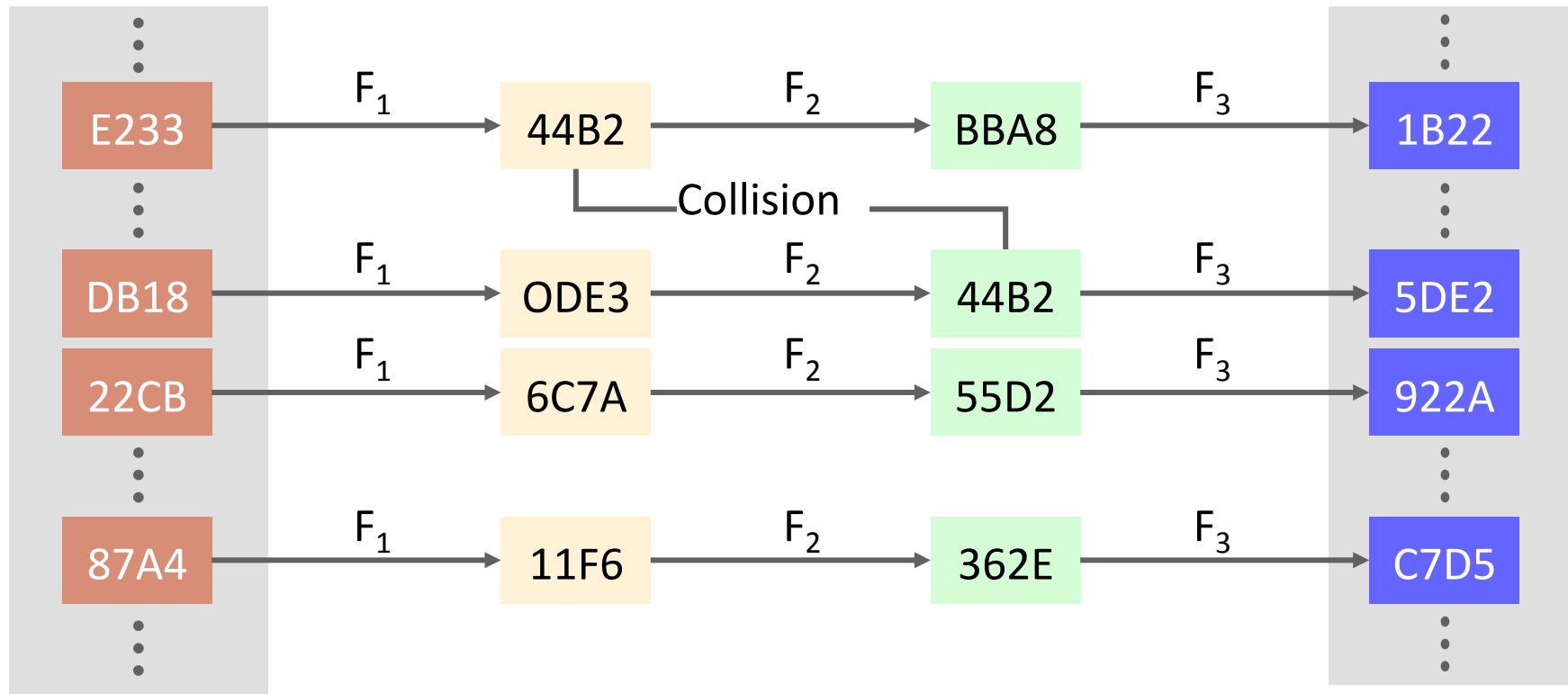
Table optimization 1: Distinguished point tables save hard disk lookups



Only one hard disk access needed instead of one for each chain link

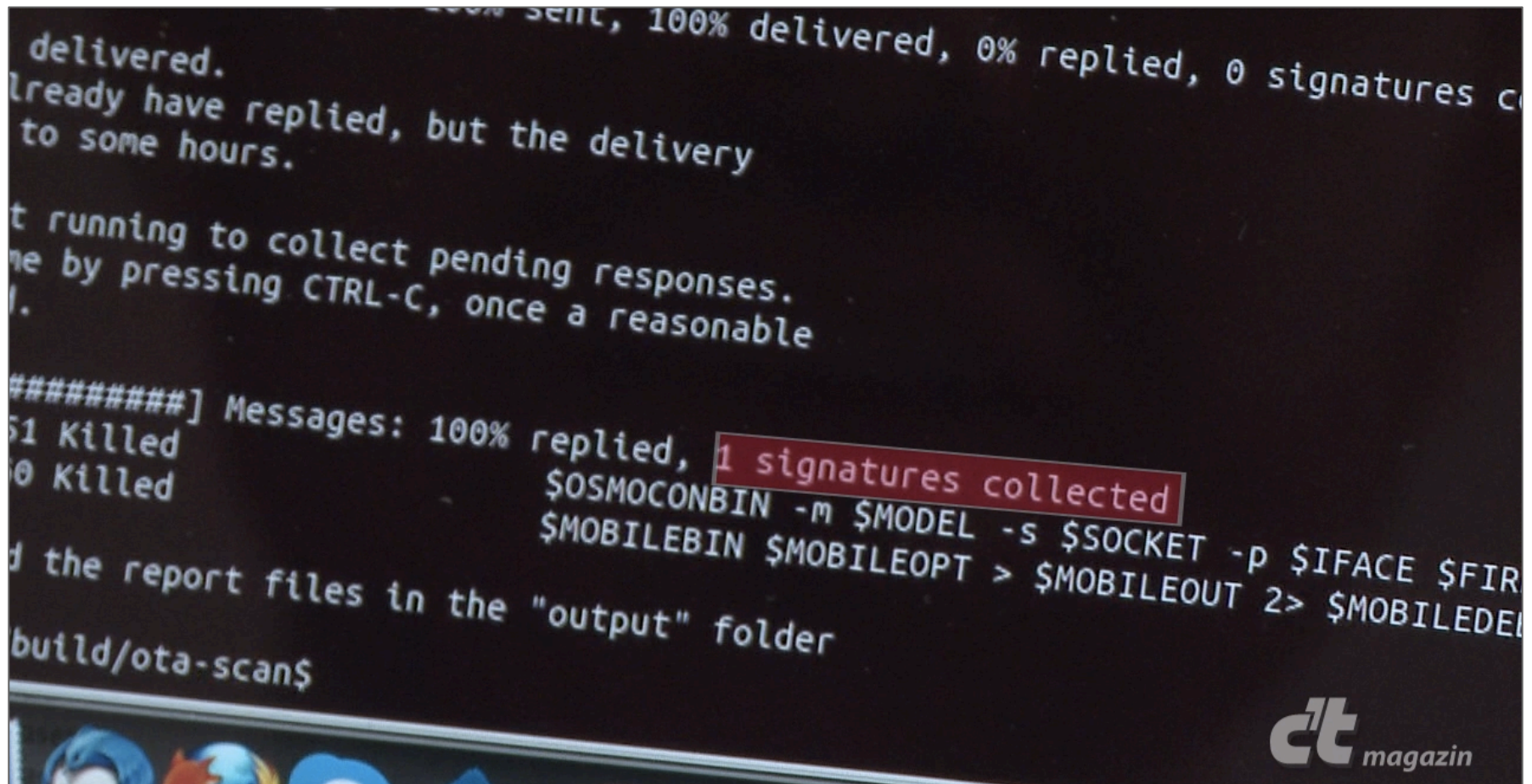
Table optimization 2:

Rainbow tables mitigate the effect of collisions



Rainbow tables have no mergers, but quadratically growing attack time

Video: Remotely cracking OTA key



OTA attacks extend beyond DES signatures

Many mobile operators responded to the looming SIM hacking risk considerate and faster than we could have wished for. Others quickly concluded they were not affected:

Operator statements

We use encryption instead of signatures; the attack does not apply here

We don't even use OTA

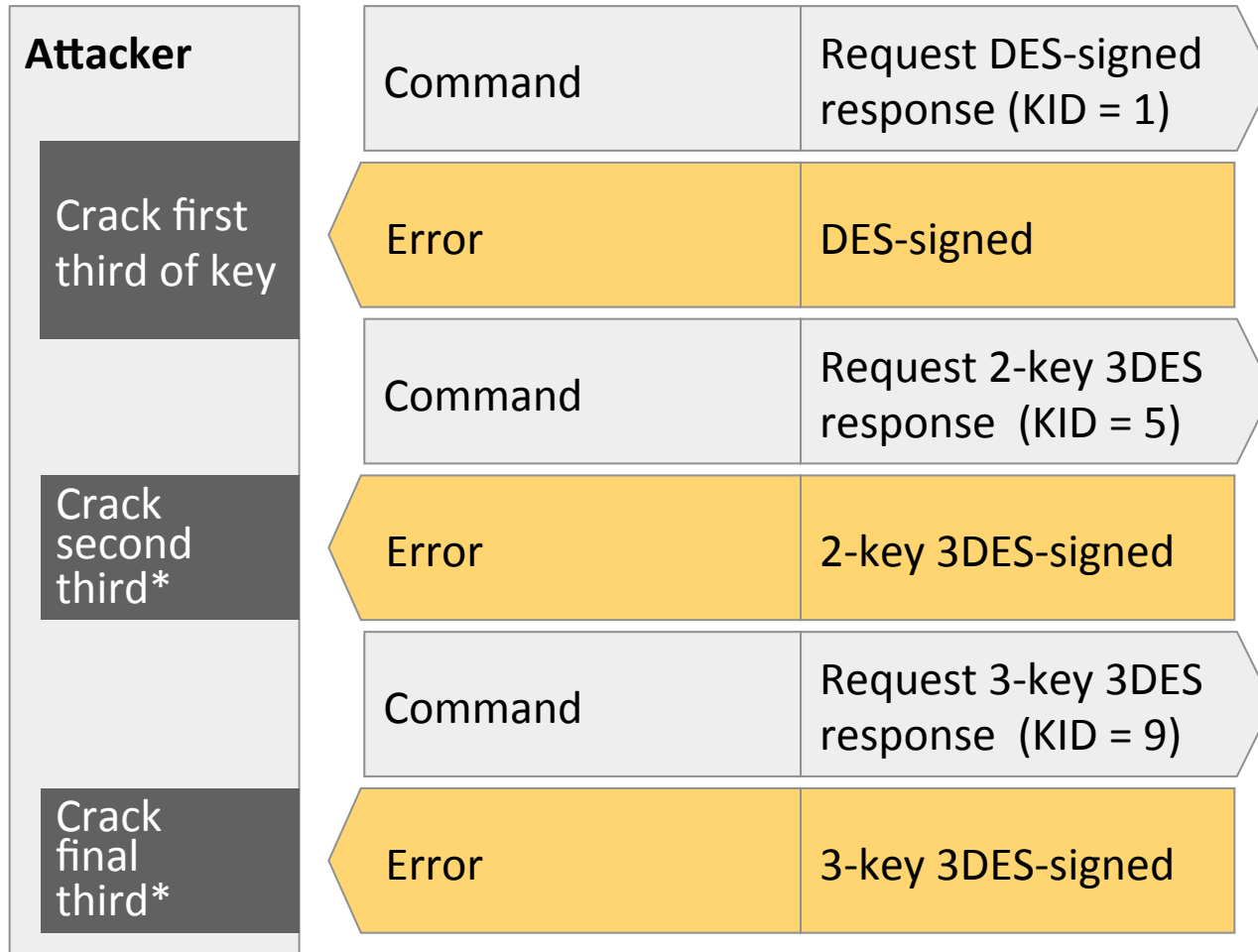
We only use 3DES

Does it make sense?

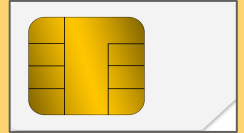
- **No.** Encrypting a known plaintext with DES is as bad as signing it. Even when both are required, the attack still applies (but needs two rainbow tables)
-
- **No.** virtually all SIMs are Java cards. Even if you are not using those capabilities, an attacker may (and will probably find that you never cared to update the keys of this virtual waste land)
-
- **Maybe.** 3DES is good, but have you ...
 - ... made sure to use full entropy 112/168 bit keys instead of multiple copies of a 56 bit key?
 - ... changed all the standard keys?
 - ... heard of *downgrade attacks*?

For some cards, even 3DES keys are crackable

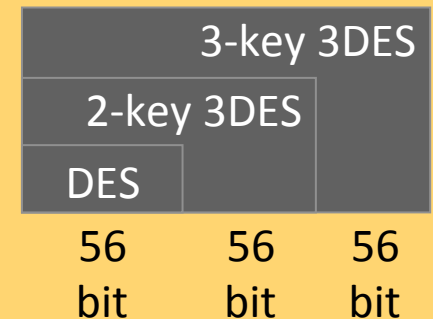
Downgrade attack flow



Some **SIM** cards with **3DES** key



use lower signature schemes when requested (in violation of the standard)

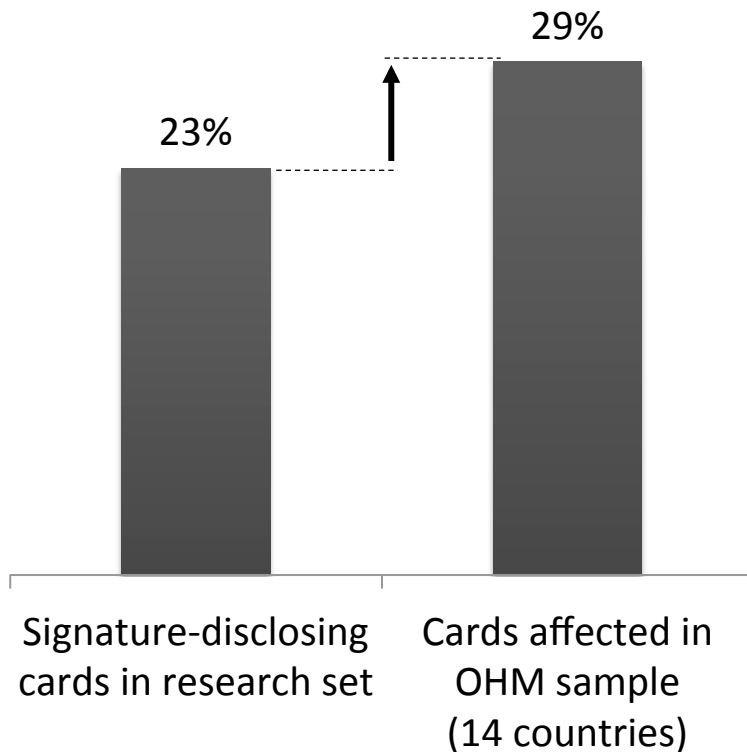


OHM 2013 workshops confirm vulnerability and point to next issues

Signature disclosure

Cards still disclose signatures even though:

- Many cards got patched over last 2 weeks
- OHM participants use newer SIM cards than average



Key entropy is low (measured at OHM)

OTA DES key	Entropy
000028b000208002	~16bit
3088802602104804	~24bit
2cee2212443ae0a6	~44bit
aa7890c234ae0e28	~50bit
96a6141aaa5ef0ee	~52bit

⋮

Agenda

SIM card background

A Getting on to the SIM

 **B Stealing SIM secrets**

Java virus does not automatically have access to all SIM assets

Java sand box
should protect
critical data on
SIM

OTA-deployed SIM virus can access SIM Toolkit API

Standard STK function

Abuse potential

Send SMS

- Premium SMS fraud

Dial phone numbers, send DTMF tones

- Circumvent caller-ID checks
- Mess with voice mail

Send USSD numbers

- Redirect incoming calls;
sometimes also SMS
- Abuse USSD-based payment
schemes

Query phone location and settings

- Track victim

Open URL in phone browser

- Phishing
- Malware deployment to phone
- Any other browser-based attack

Data access on SIM would enable further abuse

Protected function

Abuse potential

Read Ki

- SIM cloning
- Decrypt all 2G/3G/4G traffic

Read hash function

- Reverse-engineer proprietary
authentication functions;
perhaps find weaknesses

Read OTA keys

- Lateral attacks

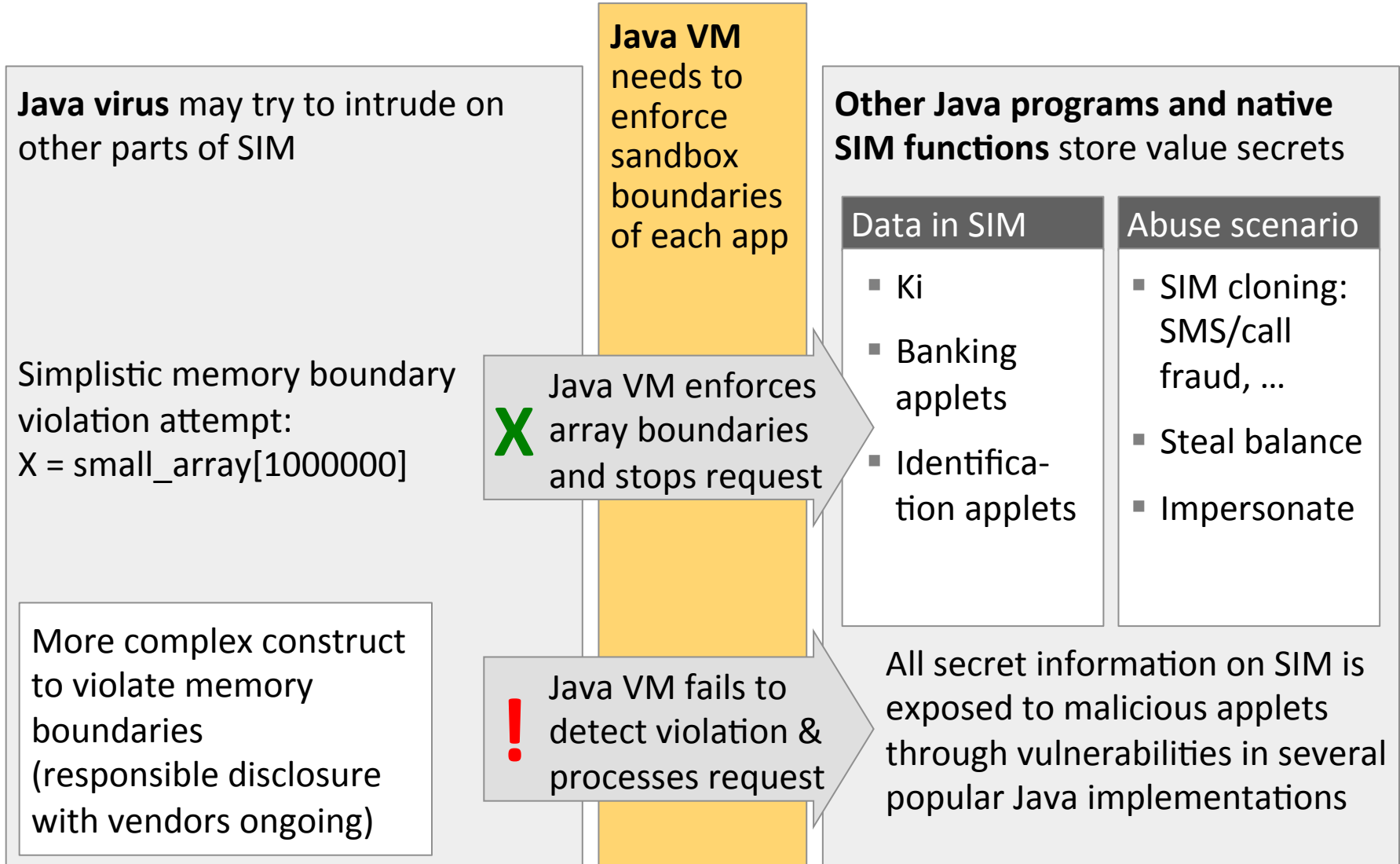
Read Java processes

- Clone NFC payment takers
and other future SIM
applications

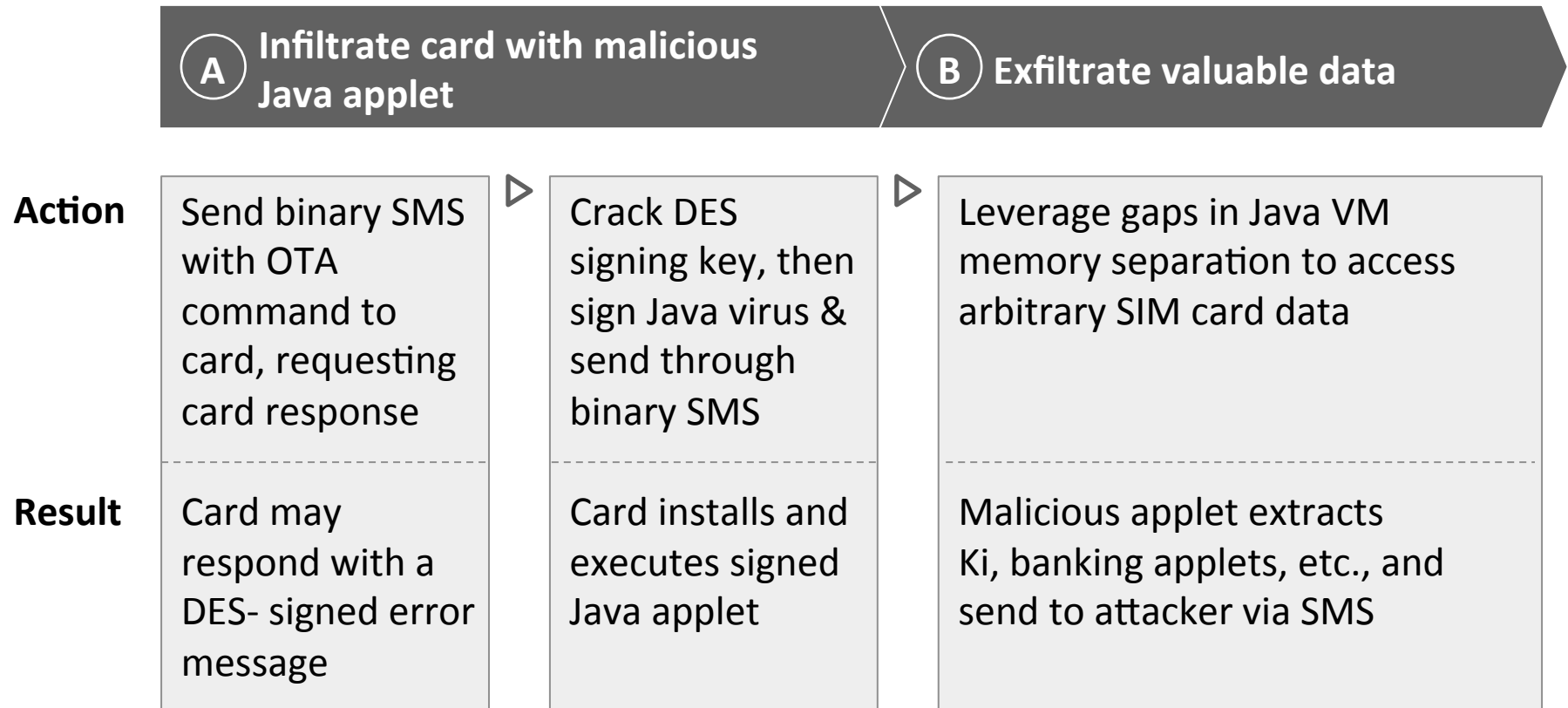
Write to Flash or EEPROM

- Alter OS to prevent
vulnerability patching

Java VM on many SIMs fails to confine malicious applets










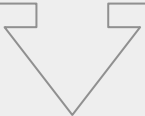




Putting it all together – Remote SIM cloning



Wide-scale SIM hacking risk must be mitigated on several layers

 Low  High

Mitigation layer for OTA hacking risk

	Effectiveness	Cost	
Filter OTA messages from unapproved sources	 Prevents probing in home network; leaves SIMs exposed when roaming, to fake base stations, and to phone malware	 Functionality readily available in most SMSCs	Network operators short-term mitigation option
Deactivate OTA on card	 Prevents attack (but also any future use of OTA w/ DES key)	 Can be done through SMS	
Use 3DES or AES OTA keys	 Prevents attack (except for where downgrade attack works)	 Some cards need replacing,  others updates	 Network operators mid-term mitigation option
Use cards that do not disclose crypto texts	 Prevents the attack	 Some cards need to be replaced	
Filter suspicious messages on phone base band	 Prevents the attack	 New software function for future phones	Complimentary mitigation option for phone manufacturers

Industry response was encouraging for responsibly disclosing hacking research

The **responsible disclosure** went surprisingly well and is worth mentioning

- We disclosed several months ahead of the release to trusted contacts made around previous releases
- Experts from a few large companies verified the results and created best practice responses
- Industry associations disseminated guidelines to all other operators
- Many networks are now well underway implementing filtering and reconfiguring cards
- Only a single lawyer stumbled into the interaction, but quickly left

Take aways from a number of responsible disclosure that all went well (except for one)

- Find **constructive partners** in the industry; ask other hackers for their recommendations
- **Disclose early** and don't be surprised if even the most motivated disclosure partner takes months to distribute the information confidentially in their industry
- Bring someone with **disclosure experience** to meetings
- **Expect friendliness** and remind your partner of the required etiquette should they ever act rude or arrogant
- Help your technical contacts win the internal battles: **Refuse** to speak to their **lawyers**; never sign an **NDA** prior to your disclosure
- Be extremely careful accepting money; and only ever to help with mitigation

Take aways

A Some DES-secured SIM-cards allow for remote key cracking and applet installation

B Java vulnerabilities enable attacker to remotely extract Ki, banking applet data

- Mitigation options exist on network, baseband, and SIM card level

Questions?

 Karsten Nohl <nohl@srlabs.de>