Unsaflok

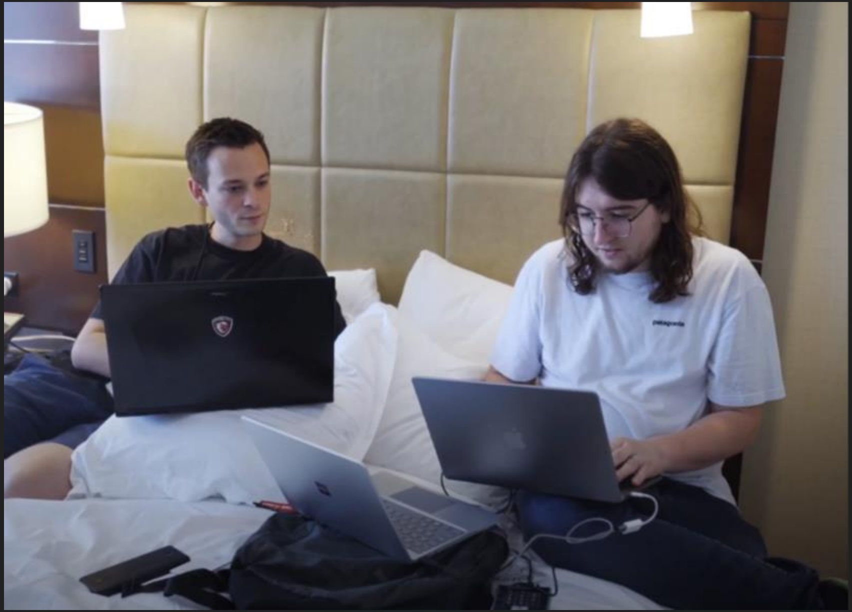
Unlocking Millions of Hotel Locks

Lennert Wouters & Ian Carroll

whoami

- **Lennert:** Hardware security researcher (COSIC, KU Leuven). @LennertWo

- **Ian:** Application security researcher, founder of Seats.aero. Formerly Red Team at Robinhood. @iangcarroll



Background

- Neither of us was particularly knowledgeable about RFID or locks.
- A group of Las Vegas hotels and casinos ran a bug bounty event alongside DEF CON 30 in 2022.
- A large group of us participated and the locks were in scope!

Related Research: Onity (2012)

- My Arduino can beat up your hotel room lock - Sera Brocious (@daeken)
  - https://daeken.dev/bhpaper.html
- Onity HT locks were introduced in 1993.
- Programming port allows to read lock memory, including the sitecode.
- Fix: mechanical cap or PCB replacement.

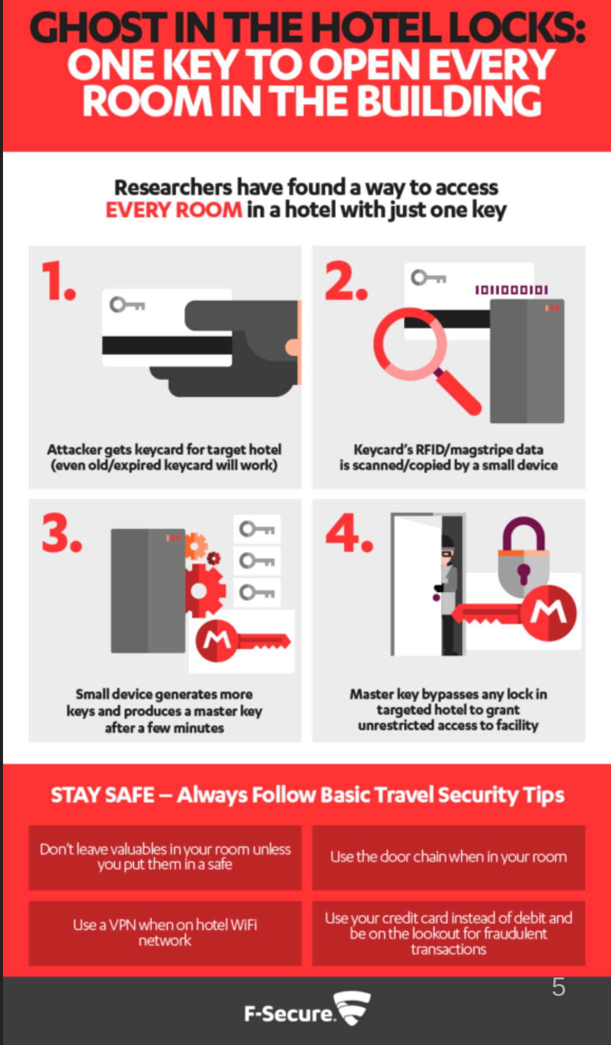
THE
HOTEL ROOM
HACKER
A global vulnerability in hotel keycard locks was a security disaster—and the opportunity of a lifetime for one burglar.
BY ANDY GREENBERG
LISK FENG
ANIMATION BY
CASEY CHIN

https://www.wired.com/2017/08/the-hotel-hacker/

Related Research: Vingcard (2018)

- Ghost In The Locks: Owning Electronic Locks Without Leaving A Trace
    - Tomi Tuominen and Timo Hirvonen
- Reading a single card allows to make a master key.
- F-Secure worked with ASSA ABLOY on a fix.
- Every lock had to be updated.

blog.f-secure.com/hotel-room-keys-can-be-hacked



GHOST IN THE HOTEL LOCKS: ONE KEY TO OPEN EVERY ROOM IN THE BUILDING

Researchers have found a way to access EVERY ROOM in a hotel with just one key

1. Attacker gets keycard for target hotel (even old/expired keycard will work)

2. Keycard's RFID/magstripe data is scanned/copied by a small device

3. Small device generates more keys and produces a master key after a few minutes

4. Master key bypasses any lock in targeted hotel to grant unrestricted access to facility

STAY SAFE – Always Follow Basic Travel Security Tips

Don't leave valuables in your room unless you put them in a safe

Use the door chain when in your room

Use a VPN when on hotel WiFi network

Use your credit card instead of debit and be on the lookout for fraudulent transactions

This talk: dormakaba Saflok

- Introduced in 1988 by Computerized Security Systems (CSS)
- Acquired by Kaba Holding AG in 2006
- 2015: merger between Dorma and Kaba
- dormakaba Holding AG

This talk: dormakaba Saflok

- Introduced in 1988 by Computerized Security Systems (CSS)
- Acquired by Kaba Holding AG in 2006
- 2015: merger between Dorma and Kaba
- dormakaba Holding AG

- 3 million doors
- 13,000 properties in 131 countries

Dormakaba Saflok architecture (offline)

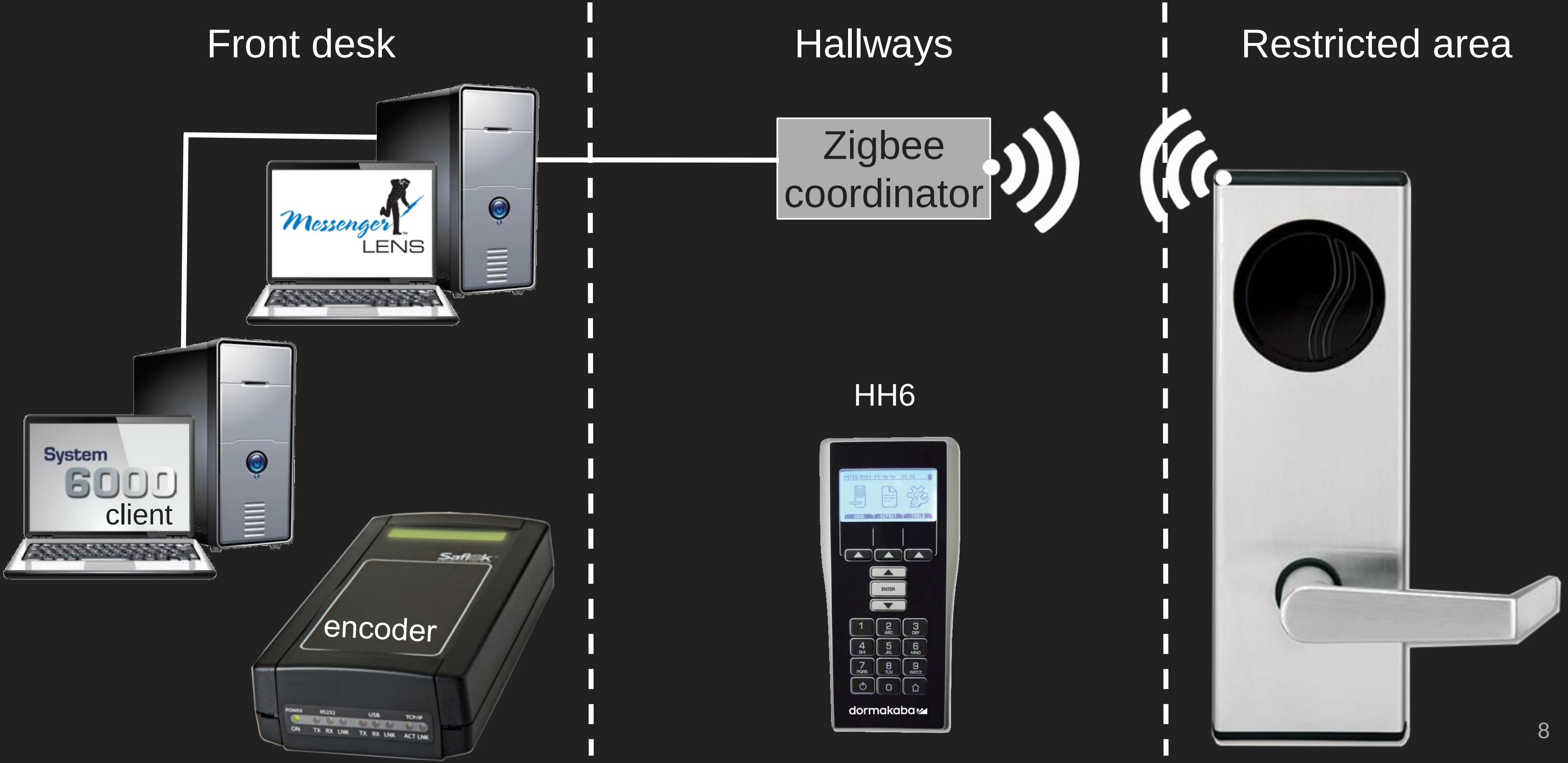
Front desk

Hallways

HH6

Restricted area

Dormakaba Saflok architecture (semi-online)



Front desk

Hallways

Restricted area

Zigbee coordinator

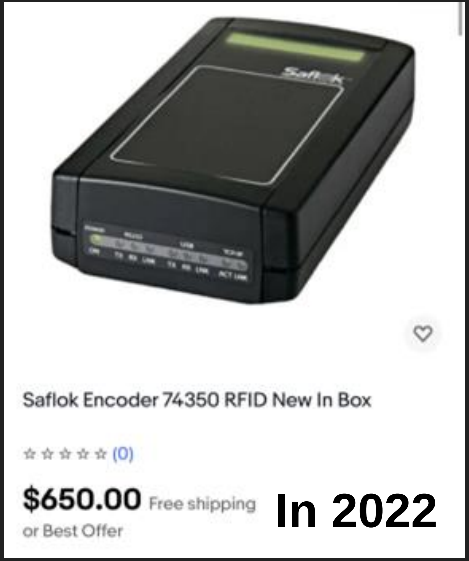
HH6

Build your own Saflok System 6000 hotel

- The System 6000 software
- A Saflok RFID encoder (74350-RP)
  - These used to be expensive! But hotels have to replace them now…
  - Alternative: ACR1281U-C8
- MIFARE Classic 1k cards
- Optional: door locks and HH6 programmer

Build your own Saflok System 6000 hotel

- The System 6000 software
- A Saflok RFID encoder (74350-RP)
  - These used to be expensive! But hotels have to replace them now…
  - Alternative: ACR1281U-C8
- MIFARE Classic 1k cards
- Optional: door locks and HH6 programmer



Saflok Encoder 74350 RFID New In Box

☆☆☆☆☆ (0)

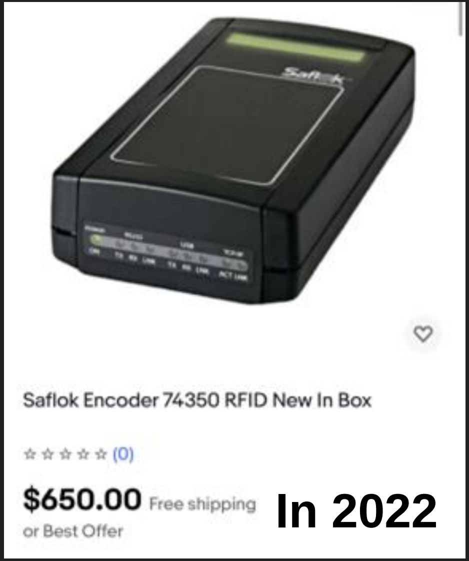
$650.00 Free shipping

or Best Offer

In 2022

Build your own Saflok System 6000 hotel

- The System 6000 software
- A Saflok RFID encoder (74350-RP)
  - These used to be expensive! But hotels have to replace them now…
  - Alternative: ACR1281U-C8
- MIFARE Classic 1k cards
- Optional: door locks and HH6 programmer



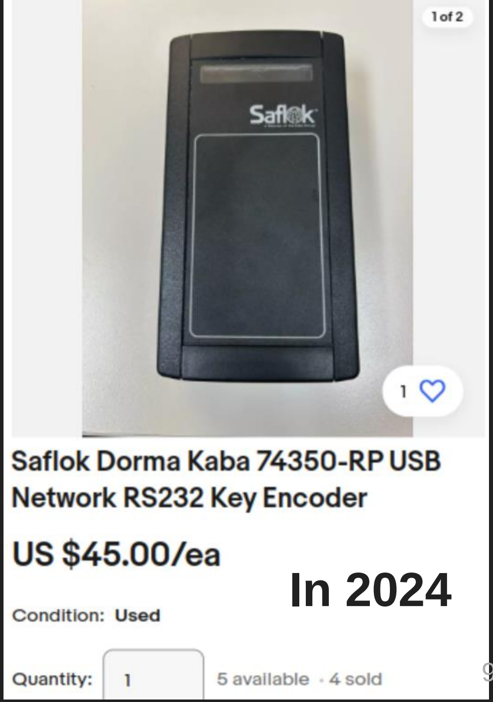
Saflok Encoder 74350 RFID New In Box

☆☆☆☆☆ (0)

$650.00 Free shipping

or Best Offer

In 2022



1 of 2

Saflok Dorma Kaba 74350-RP USB Network RS232 Key Encoder

US $45.00/ea

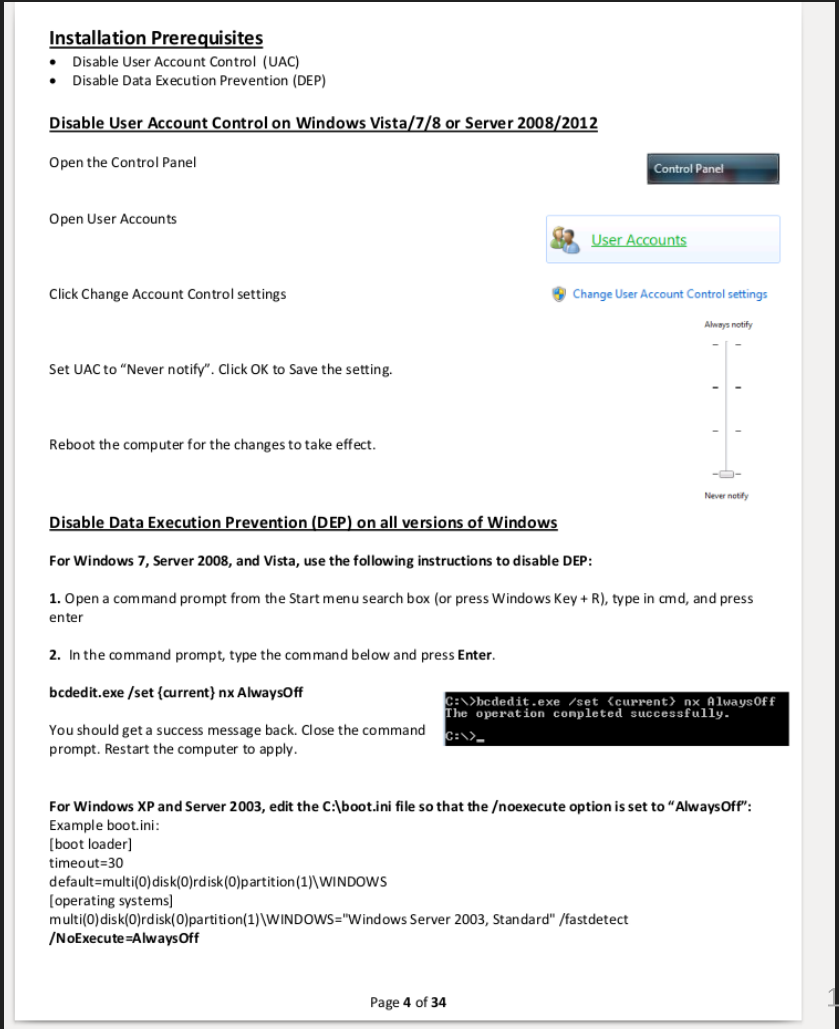
Condition: Used

Quantity: 1 5 available · 4 sold

In 2024

Setting up the software

1. Disable all security features!
2. Run the installer.
3. Place your gdb database file in the Program Files for Saflok
   a. Installer does not create one!
4. Start the software!



Installation Prerequisites

- Disable User Account Control (UAC)
- Disable Data Execution Prevention (DEP)

Disable User Account Control on Windows Vista/7/8 or Server 2008/2012

Open the Control Panel

Open User Accounts

Click Change Account Control settings

Set UAC to “Never notify”. Click OK to Save the setting.

Reboot the computer for the changes to take effect.

Control Panel
User Accounts
Change User Account Control settings
Always notify
Never notify

Disable Data Execution Prevention (DEP) on all versions of Windows

For Windows 7, Server 2008, and Vista, use the following instructions to disable DEP:

1. Open a command prompt from the Start menu search box (or press Windows Key + R), type in cmd, and press enter

2. In the command prompt, type the command below and press **Enter**.

bcdedit.exe /set {current} nx AlwaysOff

```
C:\>bcdedit.exe /set {current} nx AlwaysOff
The operation completed successfully.

C:\>_
```

You should get a success message back. Close the command prompt. Restart the computer to apply.

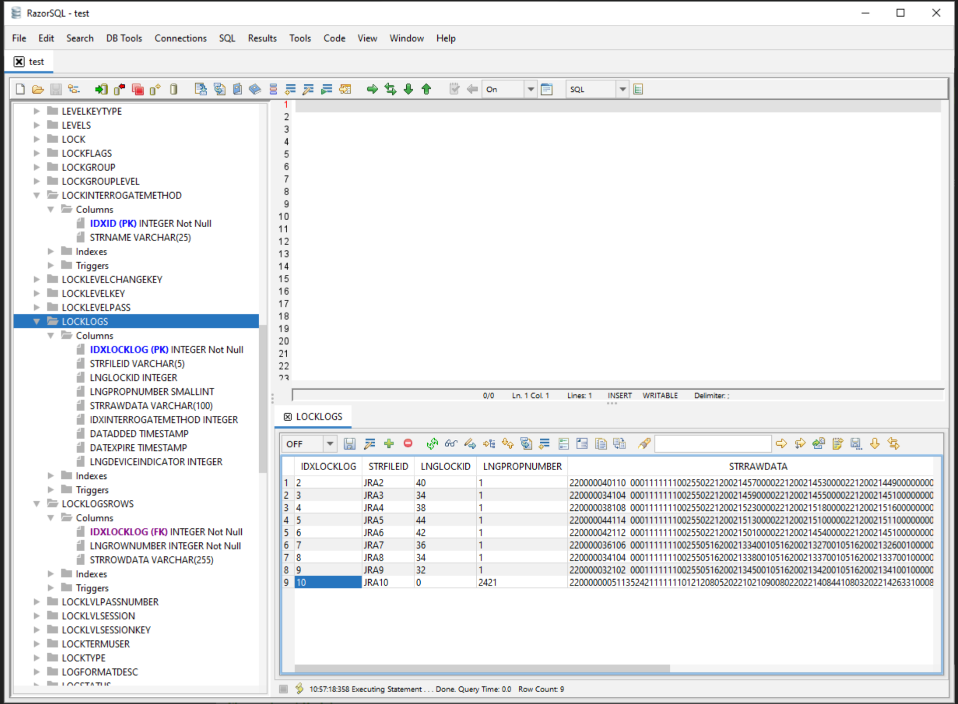
For Windows XP and Server 2003, edit the C:\boot.ini file so that the /noexecute option is set to “AlwaysOff”:
Example boot.ini:

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows Server 2003, Standard" /fastdetect
/NoExecute=AlwaysOff
```

Page **4** of **34**

System 6000 Firebird database

- System 6000 uses a local Firebird database with the hardcoded username **SYSDBA** and password **QUSOSQ**
- Database contains configuration data as well card data.



```
LEVELKEYTYPE
LEVELS
LOCK
LOCKFLAGS
LOCKGROUP
LOCKGROUPLEVEL
LOCKINTERROGATEMETHOD
    Columns
        IDXID (PK) INTEGER Not Null
        STRNAME VARCHAR(25)
    Indexes
    Triggers
LOCKLEVELCHANGEKEY
LOCKLEVELKEY
LOCKLEVELPASS
LOCKLOGS
    Columns
        IDXLOCKLOG (PK) INTEGER Not Null
        STRFILEID VARCHAR(5)
        LNGLOCKID INTEGER
        LNGPROPNUMBER SMALLINT
        STRRAWDATA VARCHAR(100)
        IDXINTERROGATEMETHOD INTEGER
        DATADDED TIMESTAMP
        DATEXPIRE TIMESTAMP
        LNGDEVICEINDICATOR INTEGER
    Indexes
    Triggers
LOCKLOGSROWS
    Columns
        IDXLOCKLOG (FK) INTEGER Not Null
        LNGROWNUMBER INTEGER Not Null
        STRROWDATA VARCHAR(255)
    Indexes
    Triggers
LOCKLVLPASSNUMBER
LOCKLVLSESSION
LOCKLVLSESSIONKEY
LOCKTERMUSER
LOCKTYPE
LOGFORMATDESC
LOGSTATUS
```

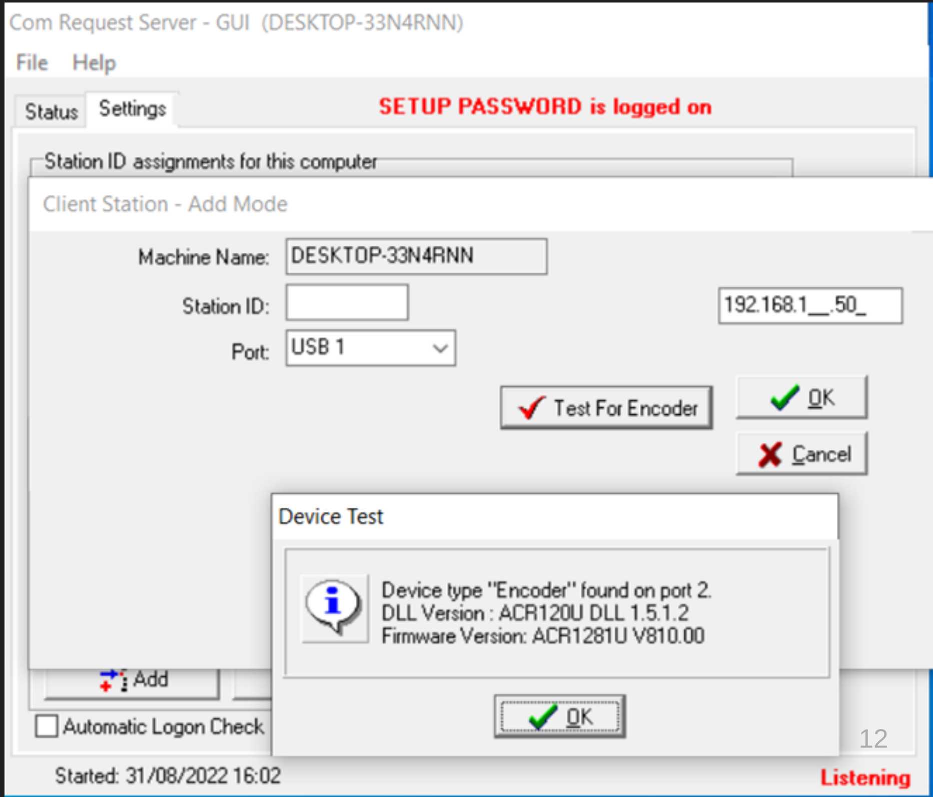
```
0/0    Ln. 1 Col. 1    Lines: 1    INSERT    WRITABLE    Delimiter: ;
```

	IDXLOCKLOG	STRFILEID	LNGLOCKID	LNGPROPNUMBER	STRRAWDATA
1	2	JRA2	40	1	220000040110 0001111111002550221200214570000221200214530000221200214490000000
2	3	JRA3	34	1	220000034104 0001111111002550221200214590000221200214550000221200214510000000
3	4	JRA4	38	1	220000038108 0001111111002550221200215230000221200215180000221200215160000000
4	5	JRA5	44	1	220000044114 0001111111002550221200215130000221200215100000221200215110000000
5	6	JRA6	42	1	220000042112 0001111111002550221200215010000221200214540000221200214510000000
6	7	JRA7	36	1	220000036106 0001111111002550516200213340010516200213270010516200213260010000
7	8	JRA8	34	1	220000034104 0001111111002550516200213380010516200213370010516200213370010000
8	9	JRA9	32	1	220000032102 0001111111002550516200213450010516200213420010516200213410010000
9	10	JRA10	0	2421	2200000005113524211111110121208052022102109008022022140844108032022142633100008

```
10:57:18:358 Executing Statement . . . Done. Query Time: 0.0   Row Count: 9
```

Using a commercial ACR RFID reader as an encoder

- System 6000 implements support for ACR RFID readers.
- Can be enabled through the Firebird DB.
- Set BHIDELEGACYENCODERS to 0.


Com Request Server - GUI (DESKTOP-33N4RNN)

SETUP PASSWORD is logged on

Station ID assignments for this computer

Client Station - Add Mode

Machine Name: DESKTOP-33N4RNN

Station ID:

192.168.1__.50_

Port: USB 1

Device Test

Device type "Encoder" found on port 2.
DLL Version : ACR120U DLL 1.5.1.2
Firmware Version: ACR1281U V810.00

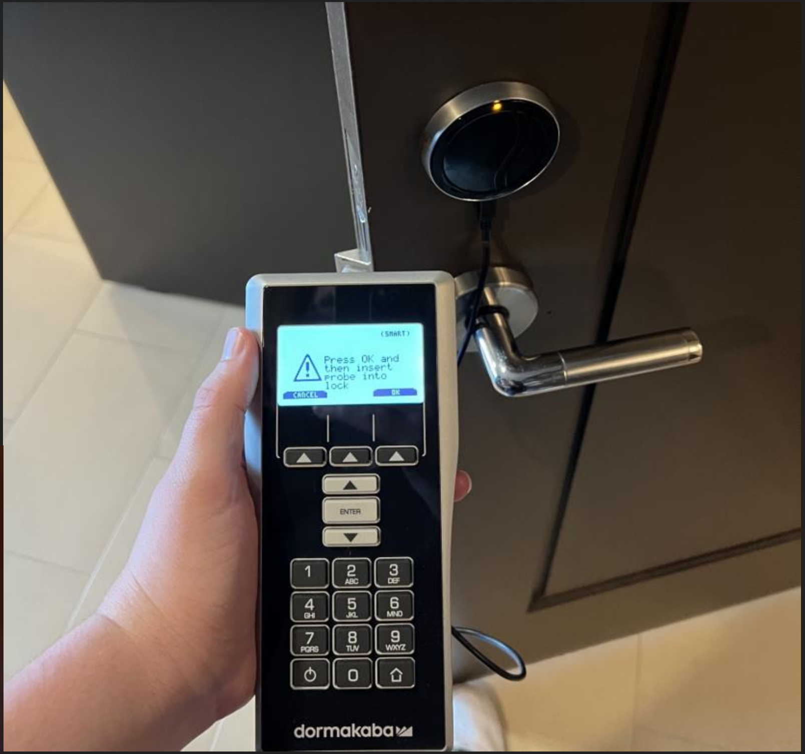
□ Automatic Logon Check

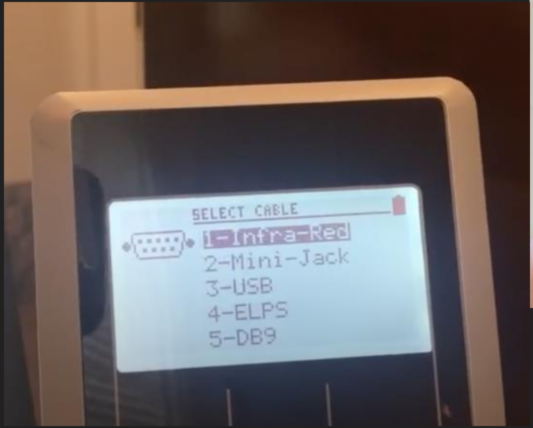
Started: 31/08/2022 16:02

Listening

Using the handheld programmer (HH6)

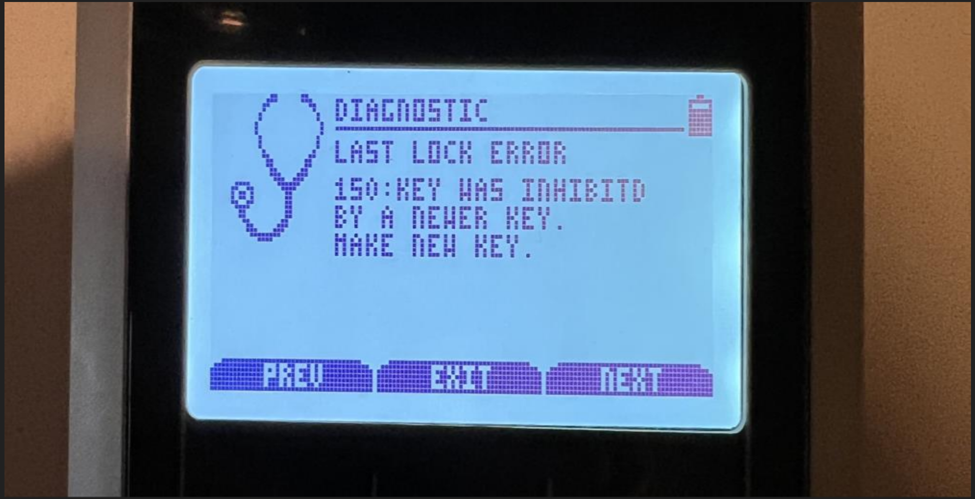
- HH6 communicates with locks through a mini-USB connector or over NFC.
- Needs to be programmed using System 6000 before it can be used.
- Errors out if the property ID of the lock is wrong…
- **…but** if you change the property ID in Firebird? 🤔





Using the handheld programmer (HH6)

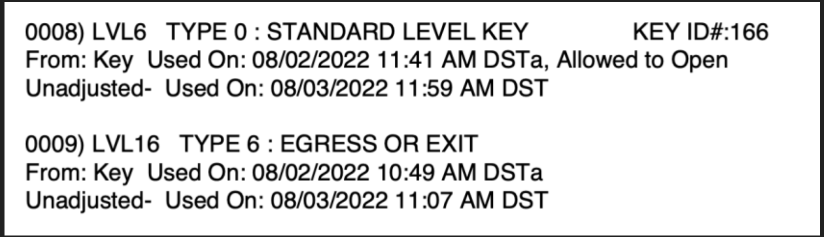
- HH6 can interrogate the lock and view all entries and exits
- Useful for debugging why a key does not open the door



DIAGNOSTIC

LAST LOCK ERROR

150:KEY WAS INHIBITD
BY A NEWER KEY.
MAKE NEW KEY.



```
0008) LVL6   TYPE 0 : STANDARD LEVEL KEY              KEY ID#:166
From: Key  Used On: 08/02/2022 11:41 AM DSTa, Allowed to Open
Unadjusted-  Used On: 08/03/2022 11:59 AM DST

0009) LVL16   TYPE 6 : EGRESS OR EXIT
From: Key  Used On: 08/02/2022 10:49 AM DSTa
Unadjusted-  Used On: 08/03/2022 11:07 AM DST
```

Reverse engineering System 6000

- Mix of Delphi executables, native and .NET
  - Delphi tooling didn’t work
  - But .NET code was easy to reverse engineer using dotPeek!

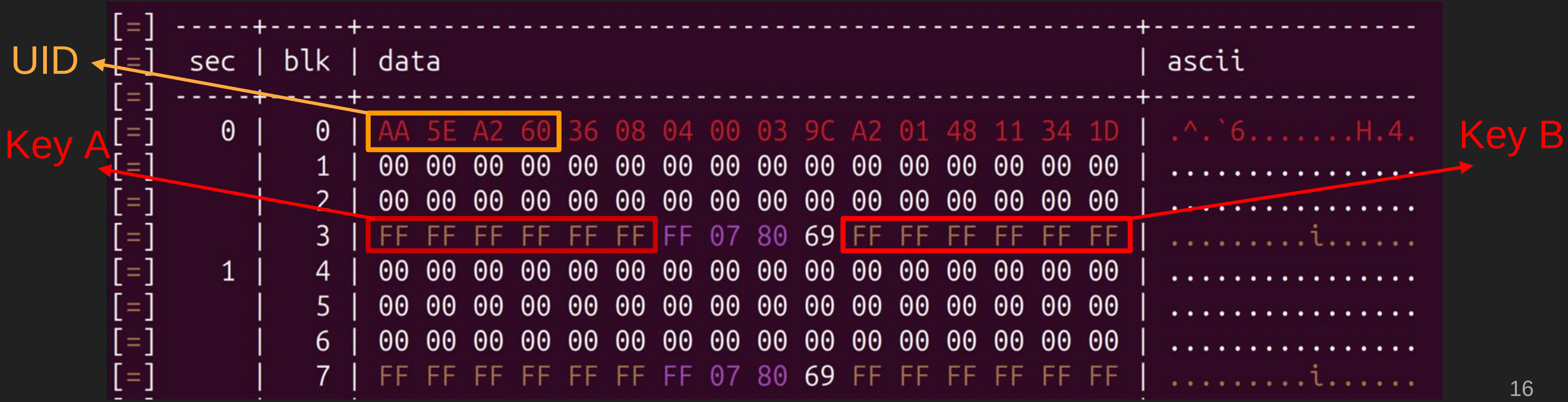
Reverse engineering System 6000

- Mix of Delphi executables, native and .NET
  - Delphi tooling didn’t work
  - But .NET code was easy to reverse engineer using dotPeek!

- Goals:
  - Understand how the sector keys are derived
  - Understand how the data on the card is encrypted
  - Understand the meaning of the card data

MIFARE Classic cards

- Most Saflok deployments used MIFARE Classic 1k cards.
  - Each card consists of 16 sectors, each containing 4 16-byte blocks of data.
  - Block 0 / manufacturer block contains the card’s UID and the manufacturer data.
  - The last block of each sector contains the keys and access conditions.
- These cards have inherent weaknesses and can be cloned.
  - This takes several seconds (less than 2 seconds now that the KDF is known).
- A cloned card has the same capabilities as the original card.

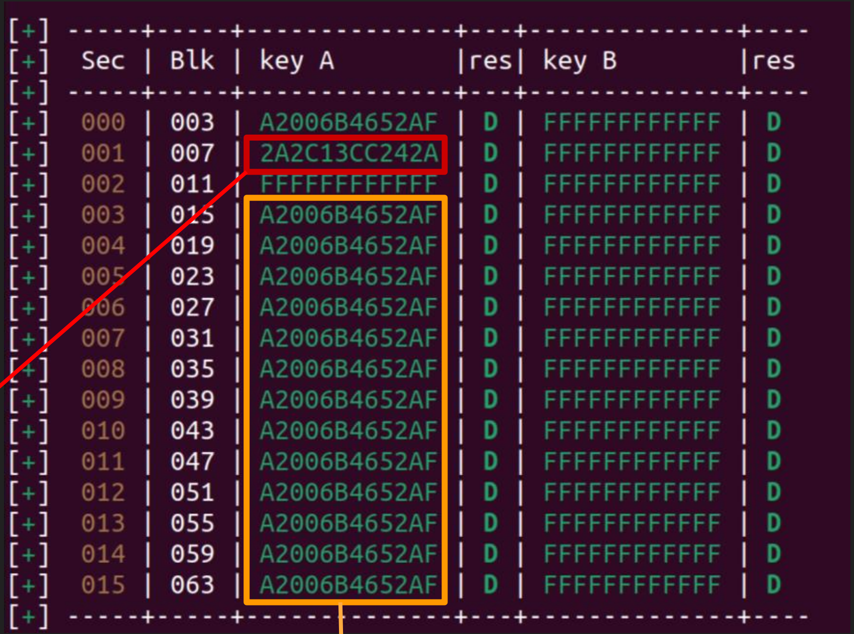


```
[=] -----+-----+-------------------------------------------------+-----------------
[=]  sec | blk | data                                            | ascii
[=] -----+-----+-------------------------------------------------+-----------------
[=]    0 |   0 | AA 5E A2 60 36 08 04 00 03 9C A2 01 48 11 34 1D | .^.`6.......H.4.
[=]      |   1 | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | ................
[=]      |   2 | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | ................
[=]      |   3 | FF FF FF FF FF FF FF 07 80 69 FF FF FF FF FF FF | .........i......
[=]    1 |   4 | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | ................
[=]      |   5 | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | ................
[=]      |   6 | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | ................
[=]      |   7 | FF FF FF FF FF FF FF 07 80 69 FF FF FF FF FF FF | .........i......
```

Saflok MIFARE Classic cards

- The Saflok data is stored in sector 0
- The key to read that sector is derived from the card's UID
    - The Key Derivation Function (KDF)
- The key for sector 1 is the same on all Saflok cards
    - Makes it easy to identify!

Proxmark3 `hf mf autopwn` result



```
[+] -----+-----+--------------+---+--------------+----
[+]  Sec | Blk | key A        |res| key B        |res
[+] -----+-----+--------------+---+--------------+----
[+]  000 | 003 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  001 | 007 | 2A2C13CC242A | D | FFFFFFFFFFFF | D
[+]  002 | 011 | FFFFFFFFFFFF | D | FFFFFFFFFFFF | D
[+]  003 | 015 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  004 | 019 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  005 | 023 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  006 | 027 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  007 | 031 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  008 | 035 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  009 | 039 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  010 | 043 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  011 | 047 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  012 | 051 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  013 | 055 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  014 | 059 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+]  015 | 063 | A2006B4652AF | D | FFFFFFFFFFFF | D
[+] -----+-----+--------------+---+--------------+----
```

Same key on all Saflok cards

Unique for every Saflok card!
But derived from the UID…

Saflok Key Derivation Function (KDF)

- We need the keys to the MIFARE sectors of the card to read the card data.
  - Proxmark3 to recover the keys or figure out how the keys are generated.

Saflok Key Derivation Function (KDF)

- We need the keys to the MIFARE sectors of the card to read the card data.
    - Proxmark3 to recover the keys or figure out how the keys are generated.

- Back in 2022 multiple people had already reverse engineered the KDF.
    - Back then the KDF was not public so we reverse engineered it ourselves.
    - Published on Gitee by user jadenwu https://gitee.com/jadenwu/Saflok_KDF/blob/master/saflok.c
    - Has since been integrated in the Proxmark3 and Flipper Zero.

Saflok Key Derivation Function (KDF)

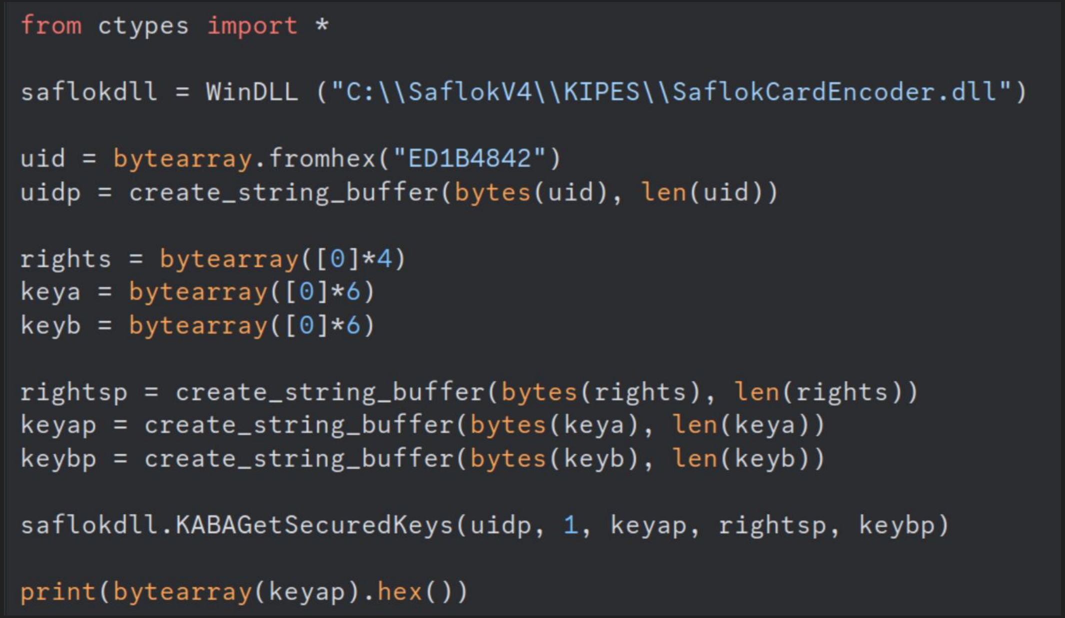
- We need the keys to the MIFARE sectors of the card to read the card data.
  - Proxmark3 to recover the keys or figure out how the keys are generated.

- Back in 2022 multiple people had already reverse engineered the KDF.
  - Back then the KDF was not public so we reverse engineered it ourselves.
  - Published on Gitee by user jadenwu
    https://gitee.com/jadenwu/Saflok_KDF/blob/master/saflok.c
  - Has since been integrated in the Proxmark3 and Flipper Zero.

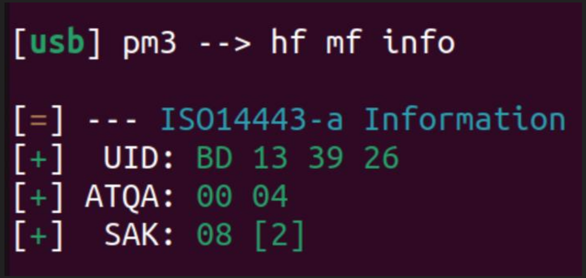
- Not strictly needed for our attack since MFC has other vulnerabilities.

Saflok Key Derivation Function (KDF)

- Implemented in SaflokCardEncoder.dll
  - KABAGetSecuredKeys()
- Started by directly using the DLL.
- Later ported the KDF to Python.



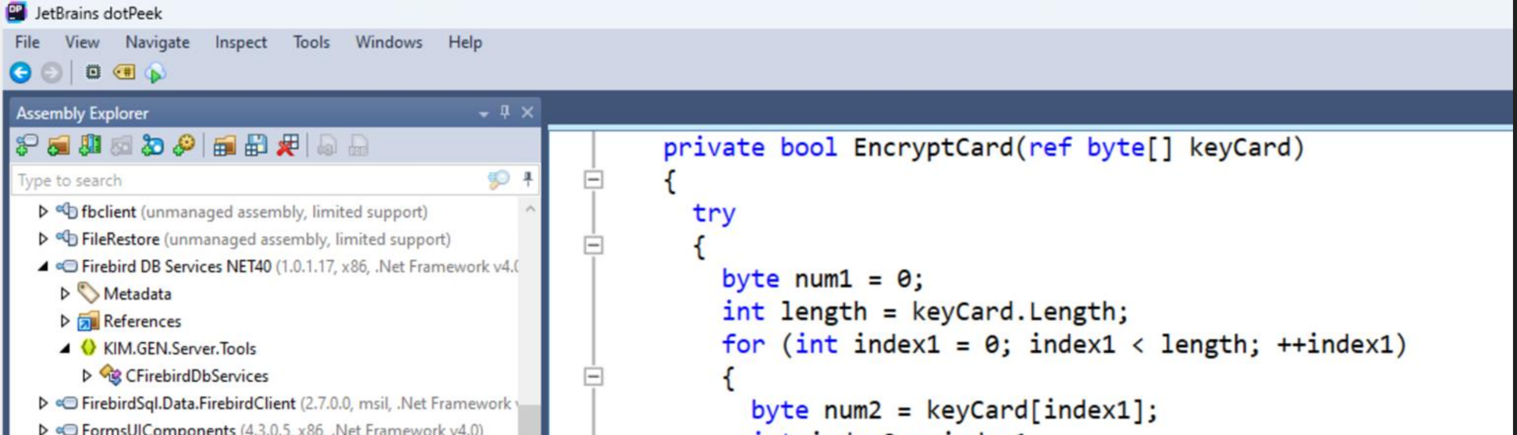
```python
from ctypes import *

saflokdll = WinDLL ("C:\\SaflokV4\\KIPES\\SaflokCardEncoder.dll")

uid = bytearray.fromhex("ED1B4842")
uidp = create_string_buffer(bytes(uid), len(uid))

rights = bytearray([0]*4)
keya = bytearray([0]*6)
keyb = bytearray([0]*6)

rightsp = create_string_buffer(bytes(rights), len(rights))
keyap = create_string_buffer(bytes(keya), len(keya))
keybp = create_string_buffer(bytes(keyb), len(keyb))

saflokdll.KABAGetSecuredKeys(uidp, 1, keyap, rightsp, keybp)

print(bytearray(keyap).hex())
```



```
[usb] pm3 --> hf mf info

[=] --- ISO14443-a Information
[+]  UID: BD 13 39 26
[+] ATQA: 00 04
[+]  SAK: 08 [2]
```

Proprietary Saflok card data encryption

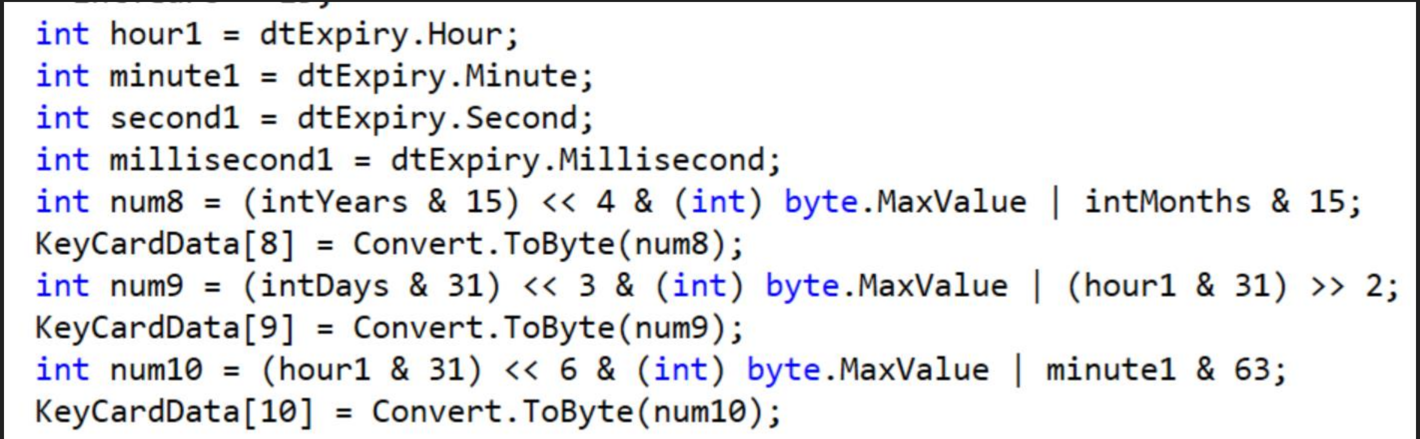
- Functionality of the card is determined by 17-bytes of encrypted data.
  - 16 bytes from block 1
  - First byte from block 2
- encryptCard() and decryptCard() in Firebird DB Services NET40.dll
  - Can be decompiled using dotPeek and is easy to translate to Python.
- Security through obscurity
  - Bit manipulations and a substitution table.
  - The secret substitution table is the same for every Saflok installation.


JetBrains dotPeek
File
View
Navigate
Inspect
Tools
Windows
Help
Assembly Explorer
Type to search
fbclient (unmanaged assembly, limited support)
FileRestore (unmanaged assembly, limited support)
Firebird DB Services NET40 (1.0.1.17, x86, .Net Framework v4.(
Metadata
References
KIM.GEN.Server.Tools
CFirebirdDbServices
FirebirdSql.Data.FirebirdClient (2.7.0.0, msil, .Net Framework
FormsUIComponents (4.3.0.5, x86, .Net Framework v4.0)

```
private bool EncryptCard(ref byte[] keyCard)
{
  try
  {
    byte num1 = 0;
    int length = keyCard.Length;
    for (int index1 = 0; index1 < length; ++index1)
    {
      byte num2 = keyCard[index1];
```

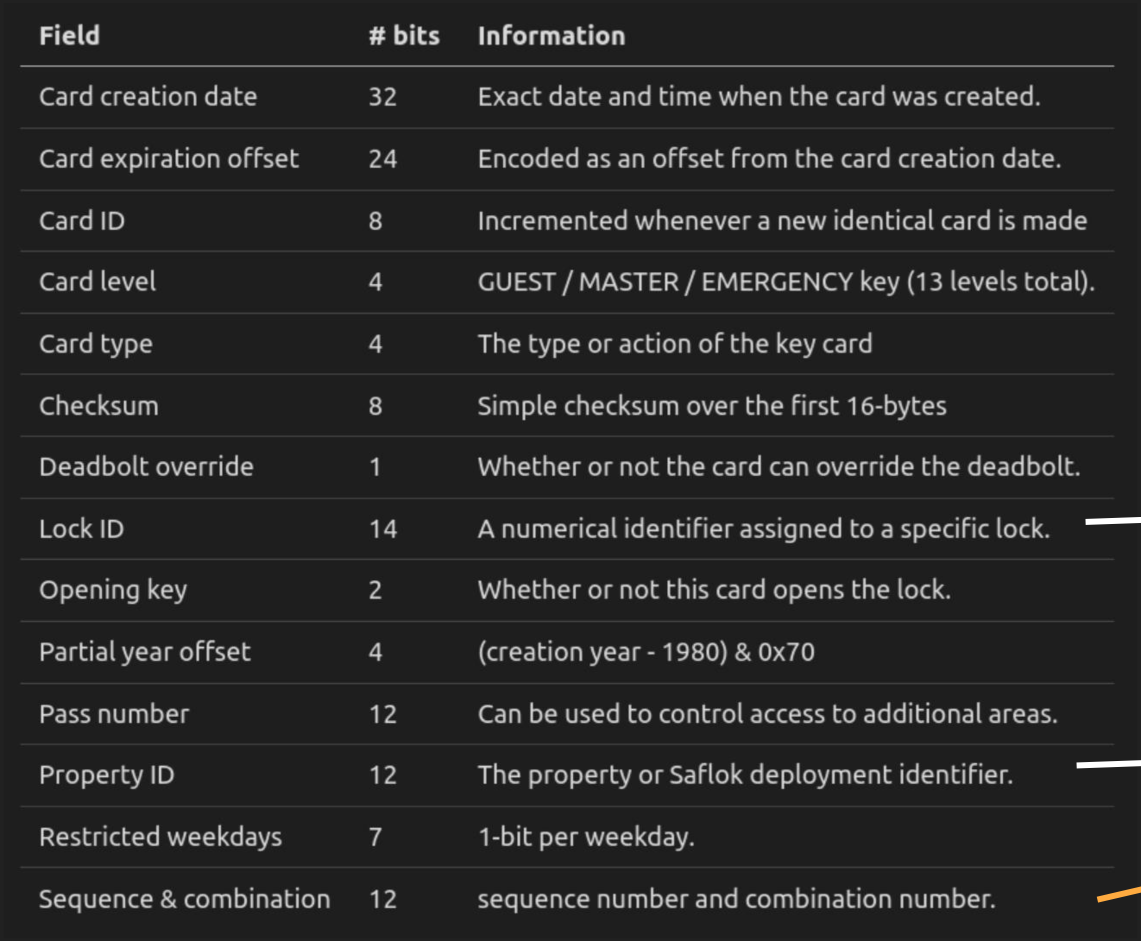
The Saflok card data format

- GetEmergencyCardInfo() implemented in Firebird DB Services NET40.dll
  - Not called to create cards using the normal GUI.
  - Useful to understand how the different data fields are serialised into the 17-byte structure.
- Being able to create cards helps a lot!
  - Look at the database entries.
  - Read the data from the card and decrypt it.
- Slowly developed our understanding of the format, field by field.



```
int hour1 = dtExpiry.Hour;
int minute1 = dtExpiry.Minute;
int second1 = dtExpiry.Second;
int millisecond1 = dtExpiry.Millisecond;
int num8 = (intYears & 15) << 4 & (int) byte.MaxValue | intMonths & 15;
KeyCardData[8] = Convert.ToByte(num8);
int num9 = (intDays & 31) << 3 & (int) byte.MaxValue | (hour1 & 31) >> 2;
KeyCardData[9] = Convert.ToByte(num9);
int num10 = (hour1 & 31) << 6 & (int) byte.MaxValue | minute1 & 63;
KeyCardData[10] = Convert.ToByte(num10);
```

The Saflok card data format



Field	# bits	Information
Card creation date	32	Exact date and time when the card was created.
Card expiration offset	24	Encoded as an offset from the card creation date.
Card ID	8	Incremented whenever a new identical card is made
Card level	4	GUEST / MASTER / EMERGENCY key (13 levels total).
Card type	4	The type or action of the key card
Checksum	8	Simple checksum over the first 16-bytes
Deadbolt override	1	Whether or not the card can override the deadbolt.
Lock ID	14	A numerical identifier assigned to a specific lock.
Opening key	2	Whether or not this card opens the lock.
Partial year offset	4	(creation year - 1980) & 0x70
Pass number	12	Can be used to control access to additional areas.
Property ID	12	The property or Saflok deployment identifier.
Restricted weekdays	7	1-bit per weekday.
Sequence & combination	12	sequence number and combination number.

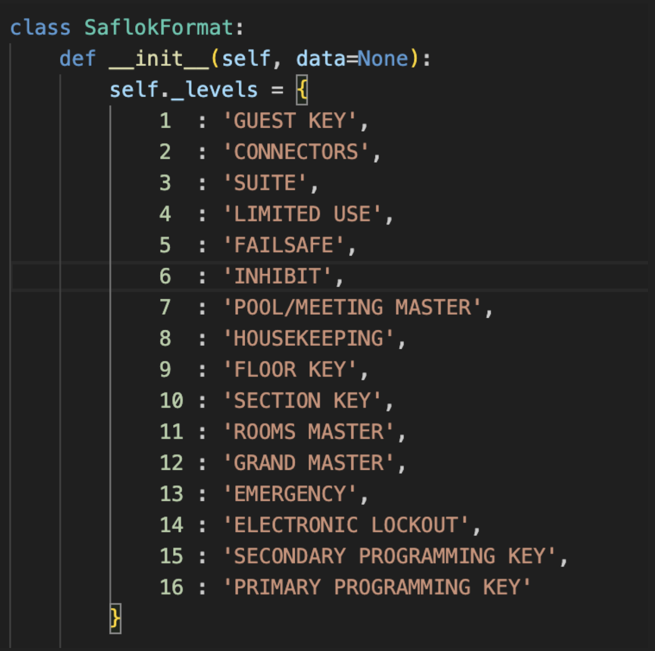
Typically sequential but not necessarily related to the room number

We need to know this value for a given hotel/property to mint valid cards

The bane of our existence for a few weeks

Different card levels and card types

- Most keys are level 1-3 (guest keys), opening one room
- Housekeeping may use level 8, opening a range of rooms or all rooms
- Emergency keys open all rooms and override the deadbolt!
  - The deadbolt may look mechanical, but is controlled by software on most hotel locks.
- PPK/SPK for programming the lock



```python
class SaflokFormat:
    def __init__(self, data=None):
        self._levels = {
            1  : 'GUEST KEY',
            2  : 'CONNECTORS',
            3  : 'SUITE',
            4  : 'LIMITED USE',
            5  : 'FAILSAFE',
            6  : 'INHIBIT',
            7  : 'POOL/MEETING MASTER',
            8  : 'HOUSEKEEPING',
            9  : 'FLOOR KEY',
            10 : 'SECTION KEY',
            11 : 'ROOMS MASTER',
            12 : 'GRAND MASTER',
            13 : 'EMERGENCY',
            14 : 'ELECTRONIC LOCKOUT',
            15 : 'SECONDARY PROGRAMMING KEY',
            16 : 'PRIMARY PROGRAMMING KEY'
        }
```

The Saflok card data format: sequence & combination

- Each lock has its own combination value.
    - A random number between 0 and 4095.
    - You might be able to guess a lock ID, but guessing the combination is difficult.
- Each card level has a sequence associated to it.
    - Allows to invalidate older cards.
- (encrypt(sequence) + combination) & 0xFFF
- This is the only field that prevents us from easily making a valid GUEST key for another lock.

Secret combination numbers and resequencing cards

- At first we tried to brute force the combination field…
    - Very painful and not successful!

Secret combination numbers and resequencing cards

- At first we tried to brute force the combination field…
  - Very painful and not successful!
- Later discovered that the card type field can enable resequencing!
  - Lock will set its internal sequence to what it calculates from the resequencing card.
- Resequence the targeted level -> use a forged card with the same sequence & combination field.

Building a full PoC

- Obtain any card of the hotel to read the property ID.
  - Can be your own card, or can be an expired card (express checkout, floor, eBay).

Building a full PoC

- Obtain any card of the hotel to read the property ID.
  - Can be your own card, or can be an expired card (express checkout, floor, eBay).
- Forge a resequencing key for a privileged level (i.e. emergency/grand master) and a fake opening key.
  - Don’t need to know the specific lock IDs for emergency/grand master keys!

Building a full PoC

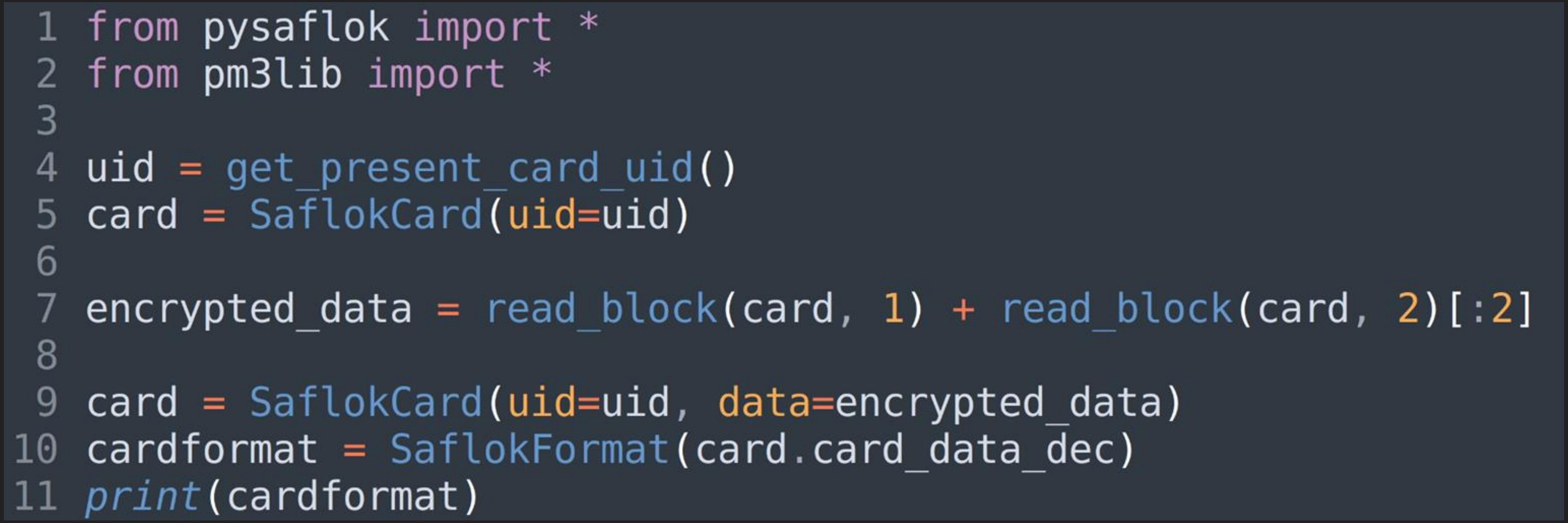
- Obtain any card of the hotel to read the property ID.
    - Can be your own card, or can be an expired card (express checkout, floor, eBay).
- Forge a resequencing key for a privileged level (i.e. emergency/grand master) and a fake opening key.
    - Don’t need to know the specific lock IDs for emergency/grand master keys!
- Tap the resequencing key to resequence the lock, then open the door with the opening key!

Building a full PoC

- Obtain any card of the hotel to read the property ID.
    - Can be your own card, or can be an expired card (express checkout, floor, eBay).
- Forge a resequencing key for a privileged level (i.e. emergency/grand master) and a fake opening key.
    - Don’t need to know the specific lock IDs for emergency/grand master keys!
- Tap the resequencing key to resequence the lock, then open the door with the opening key!
- This pair of cards works on every door in the property

Proof-of-Concept: Proxmark3

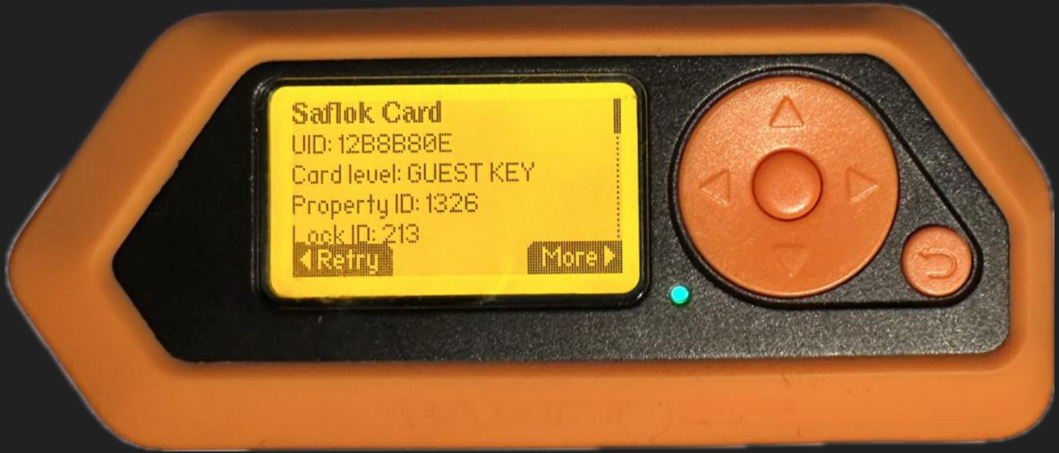
- We already had a pySaflok module
  - Supports KDF, decryption, deserializing, modifying, encrypting and serializing Saflok card data
- Instead of porting all of this over to the Proxmark3 project we wrote simple wrapper functions



```python
from pysaflok import *
from pm3lib import *

uid = get_present_card_uid()
card = SaflokCard(uid=uid)

encrypted_data = read_block(card, 1) + read_block(card, 2)[:2]

card = SaflokCard(uid=uid, data=encrypted_data)
cardformat = SaflokFormat(card.card_data_dec)
print(cardformat)
```

Proof-of-Concept: Flipper Zero

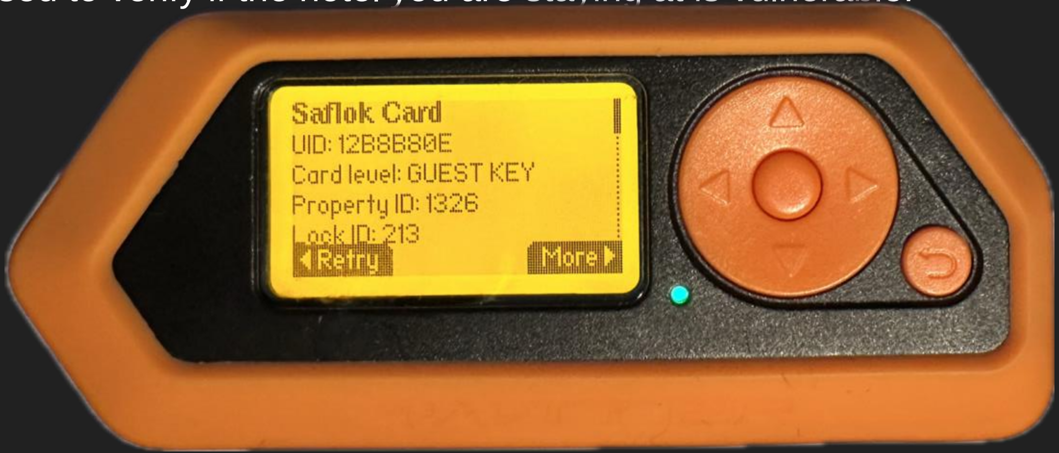
- Straightforward to setup the build environment.
- NFC supported card plugin:
    - Verify() → is it a Saflok card?
    - Read() → derive key and read relevant blocks
    - Parse() → decrypt and parse the data, generate a resequence and emergency card



Saflok Card
UID: 12B8B80E
Card level: GUEST KEY
Property ID: 1326
Lock ID: 213

Proof-of-Concept: Flipper Zero

- Straightforward to setup the build environment.
- NFC supported card plugin:
  - Verify() → is it a Saflok card?
  - Read() → derive key and read relevant blocks
  - Parse() → decrypt and parse the data, generate a resequence and emergency card
- We are NOT publishing this plugin.
- KDF only plugin by noproto: https://github.com/noproto/flipper_kdf/
  - Can be used to verify if the hotel you are staying at is vulnerable!



Saflok Card
UID: 12B8B80E
Card level: GUEST KEY
Property ID: 1326
Lock ID: 213

An overview of the almost 2 year long disclosure process

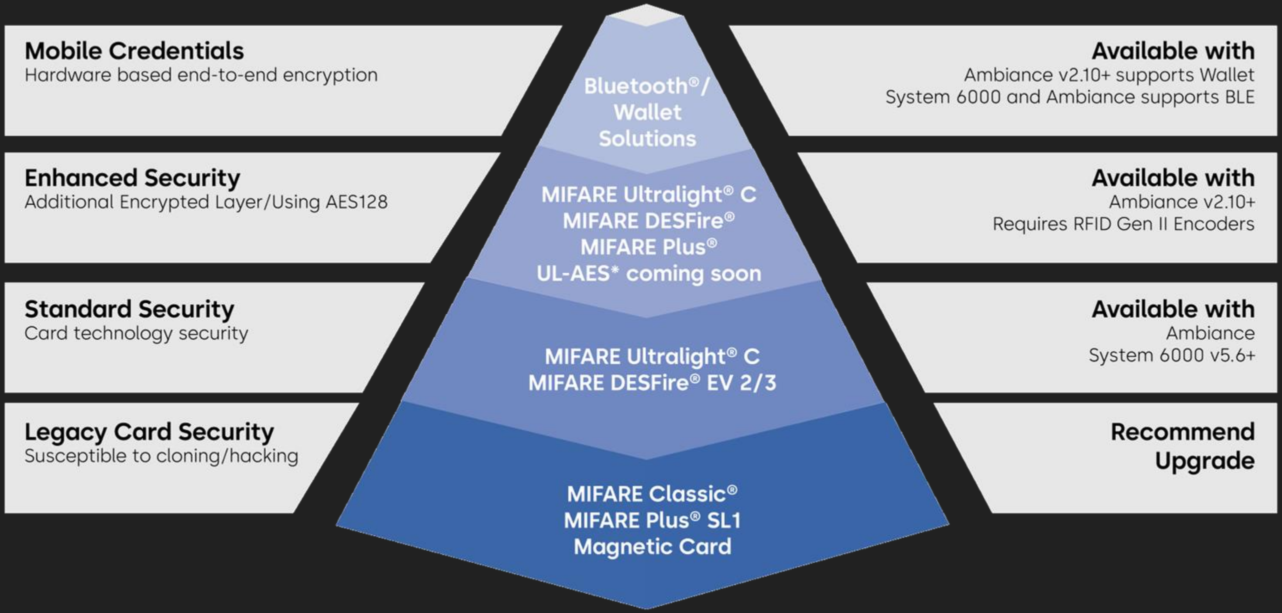
- 08/2022: Our research began on Saflok locks
- 09/2022: We completed a proof of concept exploit and contacted dormakaba
  - Initially we had to resort to LinkedIn to find a suitable point of contact
  - This has since been resolved! https://go.dormakaba.com/security-support
- 10/2022: We had our first meeting with dormakaba about the issues
- 2022 - 2024: During this time we had more than 10 meetings with dormakaba
- 11/2023: First hotels upgraded to resolve vulnerability

An overview of the almost 2 year long disclosure process

- 08/2022: Our research began on Saflok locks
- 09/2022: We completed a proof of concept exploit and contacted dormakaba
    - Initially we had to resort to LinkedIn to find a suitable point of contact
    - This has since been resolved! https://go.dormakaba.com/security-support
- 10/2022: We had our first meeting with dormakaba about the issues
- 2022 - 2024: During this time we had more than 10 meetings with dormakaba
- 11/2023: First hotels upgraded to resolve vulnerability
- 03/2024: Coordinated disclosure of the vulnerability's high-level details
    - https://unsaflok.com/
    - https://www.wired.com/story/saflok-hotel-lock-unsaflok-hack-technique/
    - At this time 36% of locks had been upgraded
- Today: DEF CON talk!
    - Currently the majority of locks have been upgraded
    - Nearly all Las Vegas properties are in the process of being mitigated or have been mitigated.

Remediation from dormakaba

- Enhanced security mode includes:
  - A new KDF
  - Card data encryption based on AES
  - MIFARE Ultralight-C for guest cards
  - A new encoder, performs the cryptographic operations in a secure element



Mobile Credentials
Hardware based end-to-end encryption

Bluetooth®/
Wallet
Solutions

Available with
Ambiance v2.10+ supports Wallet
System 6000 and Ambiance supports BLE

Enhanced Security
Additional Encrypted Layer/Using AES128

MIFARE Ultralight® C
MIFARE DESFire®
MIFARE Plus®
UL-AES* coming soon

Available with
Ambiance v2.10+
Requires RFID Gen II Encoders

Standard Security
Card technology security

MIFARE Ultralight® C
MIFARE DESFire® EV 2/3

Available with
Ambiance
System 6000 v5.6+

Legacy Card Security
Susceptible to cloning/hacking

MIFARE Classic®
MIFARE Plus® SL1
Magnetic Card

**Recommend
Upgrade**

Why did it take so long?

- A new solution had to be implemented and tested.
- 3rd party integrations may need to adapt (MFC → ULC).
  - Parking garages, elevators, kiosks, payment solutions, and even towel machines!

HOW TO USE:

STEP 1: Swipe key card to access clean towels

STEP 2: Open door and remove clean towels

STEP 3: Make sure door is closed properly

STEP 4: Return towels to the return chute of the Towel Tracker. Each towel is individually chipped and is recorded when returned to this chute.

THANK YOU!

TOWELS NOT RETURNED WILL BE CHARGED TO YOUR ROOM. EACH TOWEL IS TRACKED BY THIS SYSTEM.

THE FEE FOR UNRETURNED TOWELS IS $25 EACH.

Why did it take so long?

- A new solution had to be implemented and tested.
- 3rd party integrations may need to adapt (MFC → ULC).
    - Parking garages, elevators, kiosks, payment solutions, and even towel machines!
- Most locks have to be updated manually.
    - 2 minutes per lock x 3 million locks.
- Property owners have to be convinced to perform the upgrade and may require assistance in doing so.
    - Peak conversion rate was about 500 properties per week.

HOW TO USE:

STEP 1: Swipe key card to access clean towels

STEP 2: Open door and remove clean towels

STEP 3: Make sure door is closed properly

STEP 4: Return towels to the return chute of the Towel Tracker. Each towel is individually chipped and is recorded when returned to this chute.

THANK YOU!

TOWELS NOT RETURNED WILL BE CHARGED TO YOUR ROOM. EACH TOWEL IS TRACKED BY THIS SYSTEM.

THE FEE FOR UNRETURNED TOWELS IS $25 EACH.

Why did it take so long?

- A new solution had to be implemented and tested.
- 3rd party integrations may need to adapt (MFC → ULC).
  - Parking garages, elevators, kiosks, payment solutions, and even towel machines!
- Most locks have to be updated manually.
  - 2 minutes per lock x 3 million locks.
- Property owners have to be convinced to perform the upgrade and may require assistance in doing so.
  - Peak conversion rate was about 500 properties per week.
- The cost of ULC cards has come down a lot, but some hotels maybe have a lot of MIFARE Classic cards in stock.

towel tracker™
GET TOWELS HERE
HOW TO USE:
STEP 1: Swipe key card to access clean towels
STEP 2: Open door and remove clean towels
STEP 3: Make sure door is closed properly
STEP 4: Return towels to the return chute of the Towel Tracker. Each towel is individually chipped and is recorded when returned to this chute.
THANK YOU!
TOWELS NOT RETURNED WILL BE CHARGED TO YOUR ROOM. EACH TOWEL IS TRACKED BY THIS SYSTEM.
THE FEE FOR UNRETURNED TOWELS IS $25 EACH.

Estimating RFID card costs

- Occupancy rate: ~65%
- Average length of stay: 1.8 nights
- Average number of cards per stay: 2
- Card return rate (NA): < 25%
- Big Las Vegas properties 3,000 - 7,000 rooms

https://en.wikipedia.org/wiki/List_of_largest_hotels
https://hoteltechreport.com/news/hospitality-statistics
https://www.ahla.com/sites/default/files/SOTI_report_Oxford_Data_Occupancy.pdf

Estimating RFID card costs

- Occupancy rate: ~65%
- Average length of stay: 1.8 nights
- Average number of cards per stay: 2
- Card return rate (NA): < 25%
- Big Las Vegas properties 3,000 - 7,000 rooms

- Number of stays per year at a big property: 5000 * 365 / 1.8 * 0.65 = 659,027
- 1.3 million cards, of which 325k can be reused
- That’s roughly one million cards per year
- Let’s assume bulk pricing is $0.10 per card, that’s $100k USD per year for just RFID cards

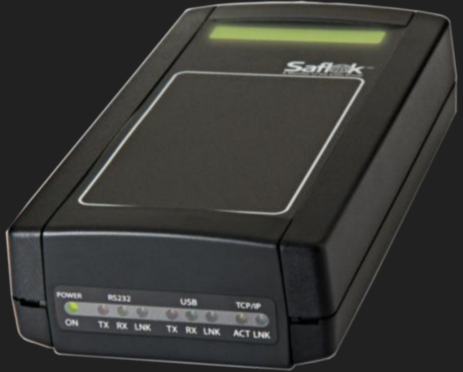
https://en.wikipedia.org/wiki/List_of_largest_hotels
https://hoteltechreport.com/news/hospitality-statistics
https://www.ahla.com/sites/default/files/SOTI_report_Oxford_Data_Occupancy.pdf

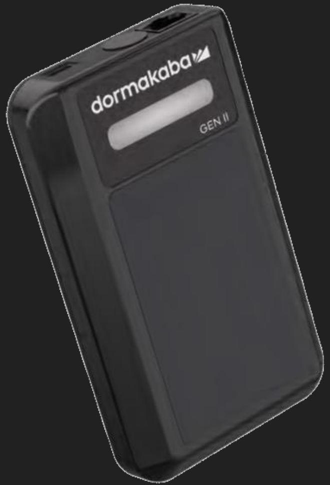
How to detect if the hotel you are staying at is fixed

1. Is the hotel using Saflok?
   a. Encoders are often visible during check-in (older style encoder → vulnerable).
   b. Saflok Quantum, MR and RT are the most common locks.

How to detect if the hotel you are staying at is fixed

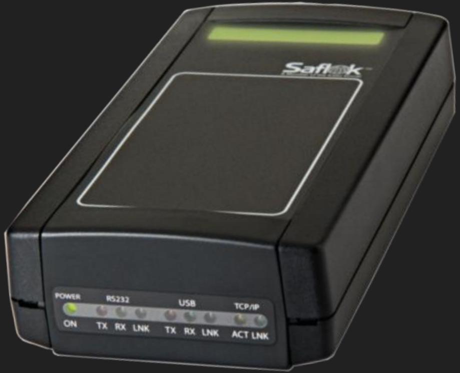
1. Is the hotel using Saflok?
   a. Encoders are often visible during check-in (older style encoder → vulnerable).
   b. Saflok Quantum, MR and RT are the most common locks.
2. Did you get a MIFARE Classic or Ultralight C card?
   a. Use your favorite RFID tool (we all know it is the Flipper Zero with Iceman firmware).
   b. Use an Android phone with the NFC TagInfo app by NXP.

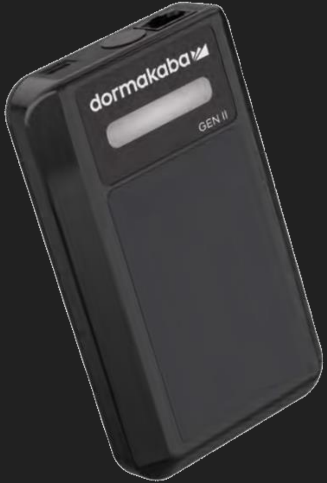




How to detect if the hotel you are staying at is fixed

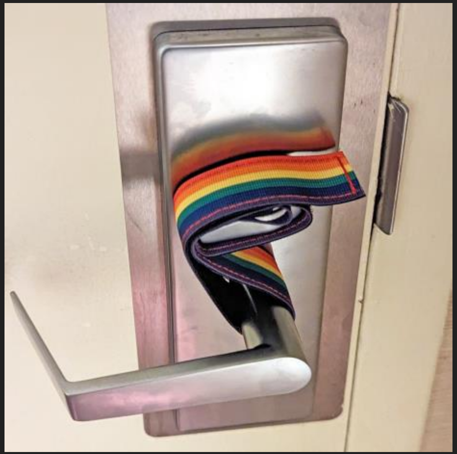
1. Is the hotel using Saflok?
   a. Encoders are often visible during check-in (older style encoder → vulnerable).
   b. Saflok Quantum, MR and RT are the most common locks.
2. Did you get a MIFARE Classic or Ultralight C card?
   a. Use your favorite RFID tool (we all know it is the Flipper Zero with Iceman firmware).
   b. Use an Android phone with the NFC TagInfo app by NXP.
3. If it is a MIFARE Classic card it is a vulnerable Saflok deployment (or not a Saflok deployment).



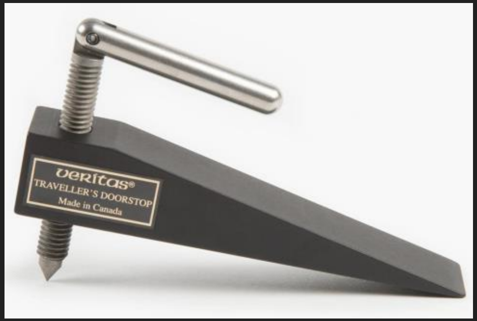


How to protect yourself in a hotel (that wasn't fixed)

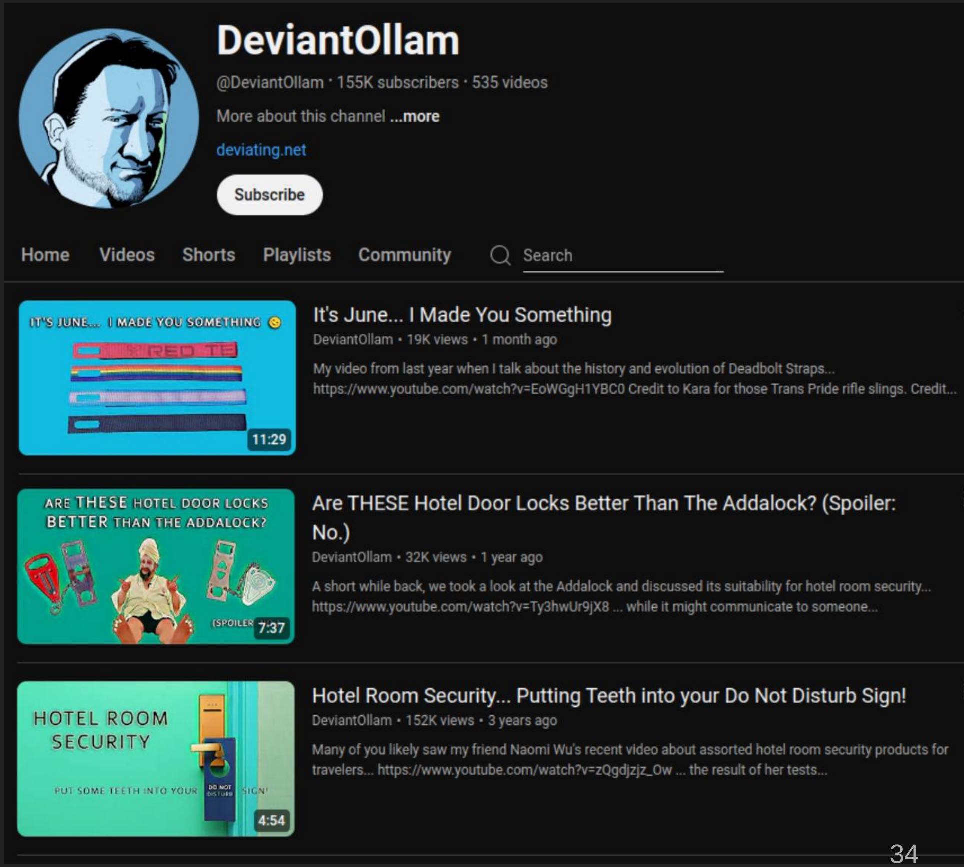
- Deadbolt velcro strap
- Under door wedge



redteamtools.com/strap
Or stop by TOOOL!



Veritas Traveller's doorstop



DeviantOllam

@DeviantOllam · 155K subscribers · 535 videos

More about this channel ...more

deviating.net

It's June... I Made You Something

DeviantOllam • 19K views • 1 month ago

My video from last year when I talk about the history and evolution of Deadbolt Straps... https://www.youtube.com/watch?v=EoWGgH1YBC0 Credit to Kara for those Trans Pride rifle slings. Credit...

Are THESE Hotel Door Locks Better Than The Addalock? (Spoiler: No.)

DeviantOllam • 32K views • 1 year ago

A short while back, we took a look at the Addalock and discussed its suitability for hotel room security... https://www.youtube.com/watch?v=Ty3hwUr9jX8 ... while it might communicate to someone...

Hotel Room Security... Putting Teeth into your Do Not Disturb Sign!

DeviantOllam • 152K views • 3 years ago

Many of you likely saw my friend Naomi Wu's recent video about assorted hotel room security products for travelers... https://www.youtube.com/watch?v=zQgdjzjz_Ow ... the result of her tests...

Summary and conclusions

- Reading a single card allows us to open any door at that property.
- This system had been vulnerable since 1988.
    - The magstripe cards were using the same format.
- Clearly many elements about these systems have not been scrutinised, and more vulnerabilities may exist.
- The cost of secure cards has come down a lot in recent years.
- Overall we had a positive experience disclosing the vulnerability to dormakaba!