LOCK IT AND STILL LOSE IT— ON THE (IN)SECURITY OF AUTOMOTIVE REMOTE KEYLESS ENTRY SYSTEMS

FLAVIO GARCIA, DAVID OSWALD, TIMO KASPER, PIERRE PAVLIDES

PRESENTED BY JACOB BEDNARD, WAYNE STATE UNIVERSITY
CSC5991

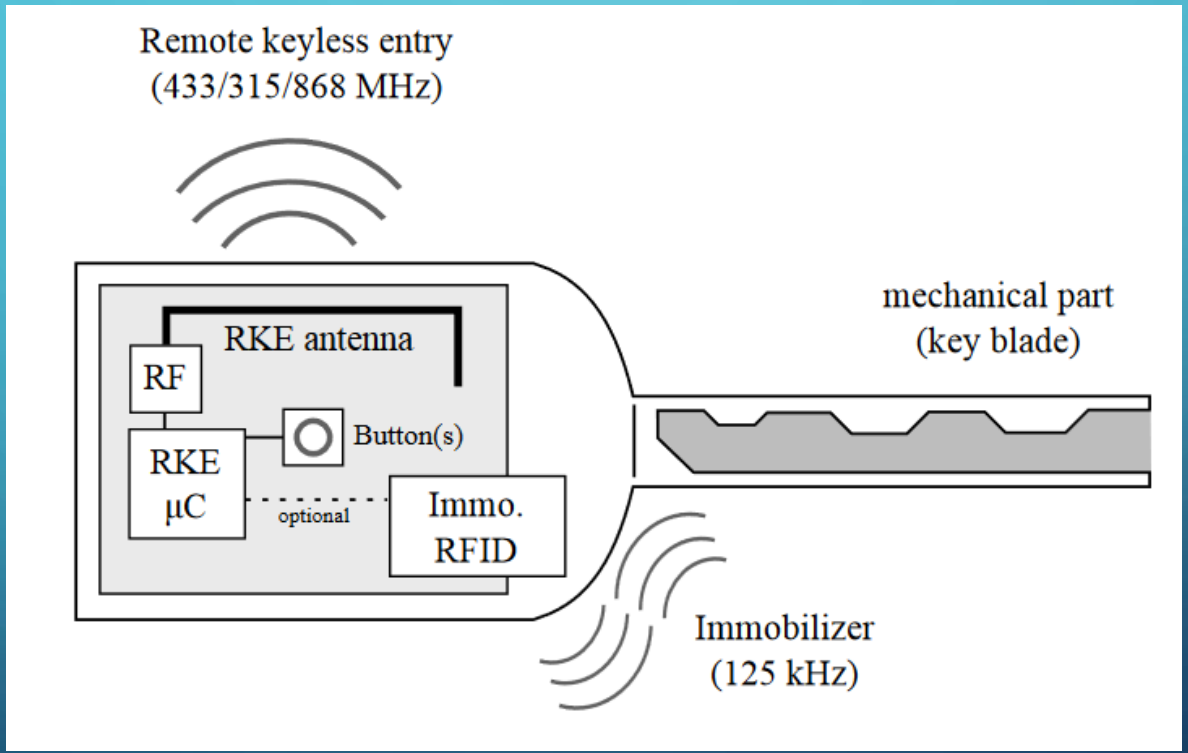
MAJOR CONTRIBUTIONS

- VW Group vehicles manufactured between 1995 and 2016 are vulnerable due to key reuse in cryptographic algorithms used for remote keyless entry (RKE)

- Correlation-based attack on Hitag2 which allows recovery of the cryptographic key used in RKE

GAINING ACCESS TO VEHICLES – PHYSICAL KEYS

GAINING ACCESS TO VEHICLES – ELECTRONIC KEYS



IMMOBILIZER VS REMOTE KEYLESS ENTRY

PASSIVE KEYLESS ENTRY AND START

- “Always on” – One Meter Radius of Vehicle
- Bidirectional Challenge Response Scheme
- Prone to relay attacks
- Blackmarket tools available

REMOTE KEYLESS ENTRY SYSTEMS (RKE)

- Used to unlock a vehicle from a distance
- Unidirectional data transmission from the remote control to vehicle
- RF Transmitter transmits in the 315Mhz (433/868Mhz - EU) Frequency Band
- Communication range: 10's-to-100's of meters
- Some use infrared
- First implementation lacked any means of security

MODERN REMOTE KEYLESS ENTRY

- Cryptography!
- Counter value that increments on each button press (i.g. Rolling Code)
- Comparison of counter between vehicle/remote
- No replay attacks

RELATED WORK

- KEELOQ (2008) – Used in Garage Door Remote Openers… Broken by Cryptoanalysis, Sidechannel Attacks.

- Cesare (2014) – Showed that RKE rolling codes can be predicted by analyzing three-subsequent rolling codes.

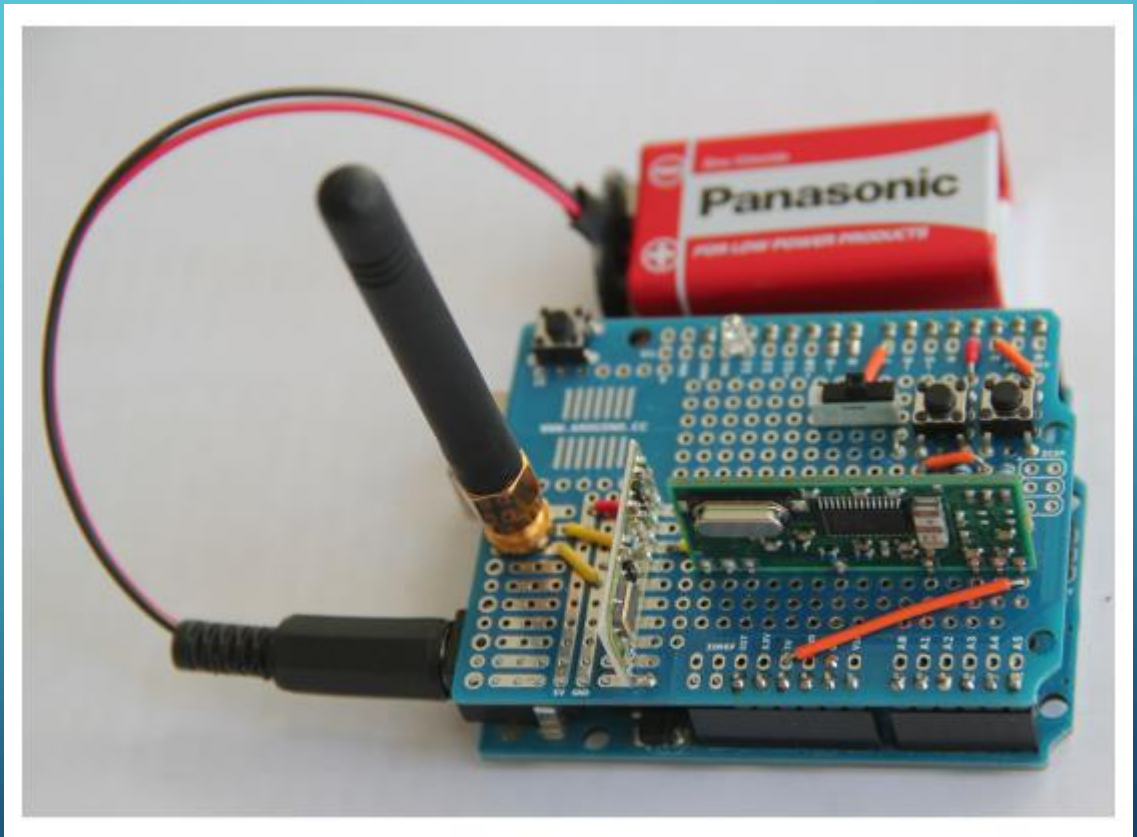
NAÏVE APPROACH

- "Selective Jamming"
- When a car owner locks their car, a malicious actor may jam the lock signal from the remote while also recording the transmission. This blocks the car from locking and the actor can utilize a replay attack to access the car.
- Not that feasible… *you recorded a lock signal (not an unlock)*

PRELIMINARY ANALYSIS OF RKE

- Bought a variety of RKE remote controls
- Analyzed their RF outputs using Software Define Radios (SDR)
- Arduino SDR Platform
- Majority used Amplitude Shift Keying (ASK)
- Others used Frequency Shift Keying (FSK)
- Manchester Encoding or Pulse-width Encoding
- Bitrate: 1-20 kilobits/second

PRELIMINARY ANALYSIS OF RKE



PRELIMINARY ANALYSIS OF RKE

- General Frame Layout:

Figure 3: General packet structure of a rolling code. Gray background indicates that the part is either encrypted or authenticated.

PRELIMINARY ANALYSIS OF RKE

- Message Authentication:

Implicit authentication: The complete payload (or part of it) is symmetrically encrypted. The receiver then decrypts the packet, and checks if the content is valid, i.e., if the UID is known to the vehicle and the counter is in its validity window. Examples for this approach can be found in Section 3.

- Payload Layout:
  - Unique Identifier (UID)
  - Rolling Counter Value
  - Button Pressed

Explicit authentication: Some form of Message Authentication Code (MAC) is computed over the data payload and then appended to the packet. An example of this approach is the Hitag2 scheme described in Section 4.

CASE STUDY 1 – VW GROUP ATTACKS

- Analyzed RKE schemes used in most VW Group vehicles manufactured between 1995 and 2016
- “How secure are modern RKE systems?”
- Utilized personal vehicles for testing

CASE STUDY 1 – VW GROUP ATTACKS

- Analyzed 7 schemes, 4 of which are discussed:

VW-1: The oldest system, used in model years until approximately 2005. The remote control transmits On-Off-Keying (OOK) modulated signals at 433.92 MHz, using pulse-width coding at a bitrate of 0.667 kBit/s.

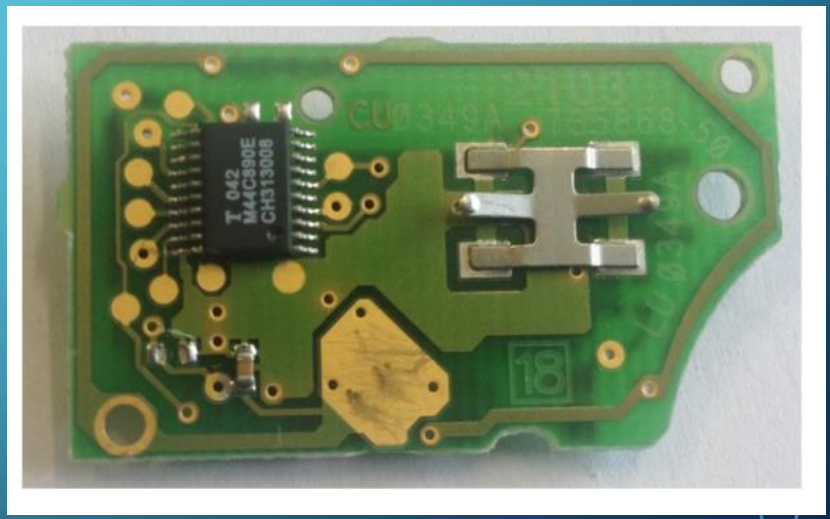
VW-2: Used from approximately 2004 onwards. The operating frequency is 434.4 MHz using OOK (same as for VW-3 and VW-4), transmitting Manchester-encoded data at a bitrate of 1 kBit/s.

VW-3: Employed for models from approximately 2006 onwards, using a frequency of 434.4 MHz and Manchester encoding at a bitrate of 1.667 kBit/s. The packet format differs considerably from VW-2.

VW-4: The most recent scheme we analyzed, found in vehicles between approximately 2009 and 2016. The system shares frequency, encoding, and packet format with VW-3, but uses a different encryption algorithm (see below).

CASE STUDY 1 – VW GROUP ATTACKS

- Initial Procedure
  - Implement likely modulation/demodulation procedure
  - Test!



- …Realized that key derivation was likely done on the engine control unit (ECU) side.

CASE STUDY 1 – VW GROUP ATTACKS

- Bought numerous ECU's for testing
- Dumped firmware for Static Analysis
- Looked for constants, lookups, ciphers, etc.

(Can't really tell us much because of disclosure policy)

CASE STUDY 1 – VW GROUP ATTACKS

- VW-1 Scheme:
  - Security by Obscurity
  - First four bytes hold XOR'ed UID
  - Linear Feedback Shift Register (LSFR) – Unencrypted Counter
  - The button pressed
  - Modified Replay Attacks! (Increment Counter)

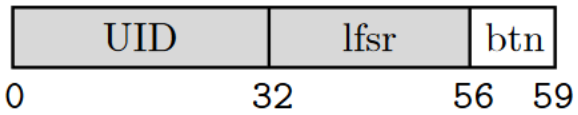


Figure 6: Packet structure of a rolling code for VW-1. Gray background indicates that the part is obfuscated or holds the LFSR state. The start pulse is not shown.

CASE STUDY 1 – VW GROUP ATTACKS

- VW-2, VW-3 Schemes:
  - Preamble
  - 8-byte encrypted payload
  - Button Pressed

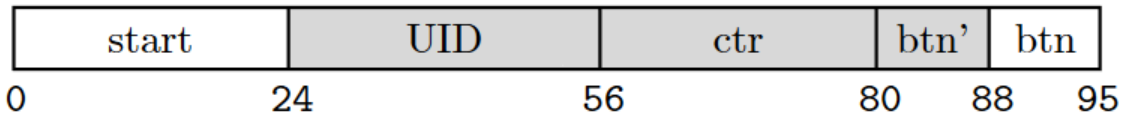


Figure 7: Packet structure of a rolling code for VW-2–4. Gray background indicates that the part is encrypted. Note that the fixed start pattern is shorter for VW-2.

CASE STUDY 1 – VW GROUP ATTACKS

- VW-2, VW-3 Schemes:
  - Preamble
  - 8-byte encrypted payload
  - Button Pressed
  - AUT64 Encryption – Round-cipher
  - 91.55 bit key size
  - GLOBAL MASTER KEY is REUSED
  - …ACROSS EVERY VEHICLE

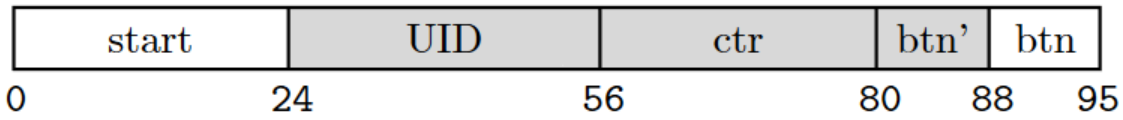


Figure 7: Packet structure of a rolling code for VW-2–4. Gray background indicates that the part is encrypted. Note that the fixed start pattern is shorter for VW-2.

CASE STUDY 1 – VW GROUP ATTACKS

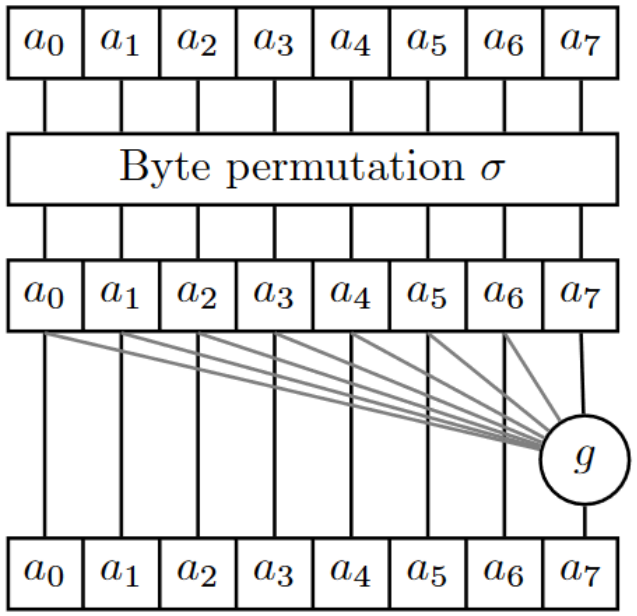
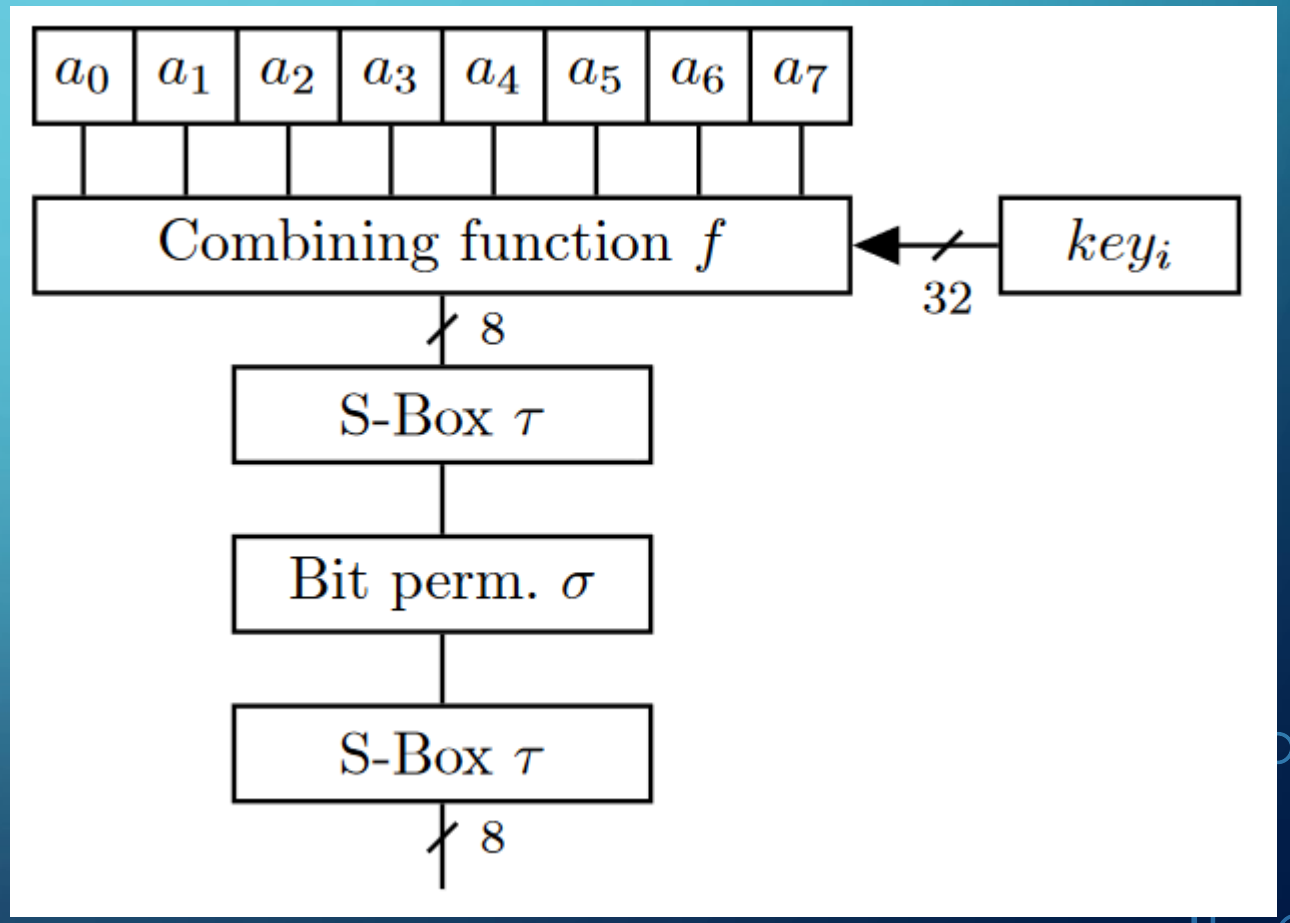


Figure 8: One round i of the AUT64 block cipher as used in VW-2 and VW-3. $a_0, \ldots, a_7$ is the 8-byte state of the cipher, $g(a_0, \ldots, a_7, key_i)$ the round function.



CASE STUDY 1 – VW GROUP ATTACKS

- VW-4 Scheme:
  - Same frame format as VW-3
  - XTEA-cipher
  - 64 Round Feistel Structure, 64-bit block size, 128-bit key
  - Well suited for low-powered remotes
  - Again… GLOBAL MASTER KEYS

CASE STUDY 1 – VW GROUP ATTACKS

- Miscellaneous stuff about Counter:
  - Using counter more than 2 increments behind disable remote entry. Must manually be reset
  - 2 or Less Increments behind places remote out of step. Button pressed must happen twice to successfully work.

CASE STUDY 1 – VW GROUP ATTACKS

- Implications:
  - If you successfully can obtain the master key (ECU dump, Bruteforce, etc), you can decrypt the current counter and UID values
  - Access Gained… unforcefully
  - Nearly 20 years worth of Volkswagen vehicles vulnerable

CASE STUDY 1 – VW GROUP ATTACKS

- Counter Measures:
  - Physical Locks
  - Seriously… that's it.

CASE STUDY 2 – HITAG2 SYSTEM

- Rolling code system
- Example of RKE Scheme
- Designed by NXP
- Not known to use Global Master Keys

CASE STUDY 2 – HITAG2 SYSTEM

- Rolling code system
- Example of RKE Scheme
- Designed by NXP
- Not known to use Global Master Keys
- Researchers can still crack after 4-8 button presses

CASE STUDY 2 – HITAG2 SYSTEM

- Hitag2 Scheme
  - UID (32-bit) + button(4-bit) + counter(10 of 28 LSB) + checksum(8-bit)

CASE STUDY 2 – HITAG2 SYSTEM

- Hitag2 Stream Cipher
  - 48-bit LSFR
  - Non-Linear Filter Function
  - For each clock cycle:
    - 20-bits are put through filter function ➔ 1-bit Key Stream
    - LSFR << 1
    - Feedback polynomial used to generate new bit on right of LSFR

Definition 4.1 *The feedback function* $L\colon \mathbb{F}_2^{48} \to \mathbb{F}_2$ *is defined by* $L(x_0 \dots x_{47}) := x_0 \oplus x_2 \oplus x_3 \oplus x_6 \oplus x_7 \oplus x_8 \oplus x_{16} \oplus x_{22} \oplus x_{23} \oplus x_{26} \oplus x_{30} \oplus x_{41} \oplus x_{42} \oplus x_{43} \oplus x_{46} \oplus x_{47}$.

The filter function f consists of three different circuits f_a, f_b and f_c, which output one bit each. The circuits f_a and f_b are employed more than once, using a total of twenty input bits from the LFSR. Their resulting bits are used as input for f_c. The circuits are represented by three Boolean tables that contain the resulting bit for each input.

Definition 4.2 (Filter function) *The filter function* $f\colon \mathbb{F}_2^{48} \to \mathbb{F}_2$ *is defined by*

$$\begin{aligned} f(x_0 \dots x_{47}) = f_c(&f_a(x_2 x_3 x_5 x_6), f_b(x_8 x_{12} x_{14} x_{15}), \\ &f_b(x_{17} x_{21} x_{23} x_{26}), f_b(x_{28} x_{29} x_{31} x_{33}), \\ &f_a(x_{34} x_{43} x_{44} x_{46})), \end{aligned}$$

where $f_a, f_b\colon \mathbb{F}_2^4 \to \mathbb{F}_2$ *and* $f_c\colon \mathbb{F}_2^5 \to \mathbb{F}_2$ *are*

$$\begin{aligned} f_a(i) &= (\texttt{0xA63C})_i \\ f_b(i) &= (\texttt{0xA770})_i \\ f_c(i) &= (\texttt{0xD949CBB0})_i. \end{aligned}$$

Because $f(x_0 \dots x_{47})$ *only depends on* $x_2, x_3, x_5 \dots x_{46}$ *we shall define* $f_{20}\colon \mathbb{F}_2^{20} \to \mathbb{F}_2$, *writing* $f(x_0 \dots x_{47})$ *as* $f_{20}(x_2, x_3, x_5 \dots x_{46})$.

Definition 4.4 *Given a key* $k = k_0 \dots k_{47} \in \mathbb{F}_2^{48}$, *an identifier* $id = id_0 \dots id_{31} \in \mathbb{F}_2^{32}$, *a counter* $ctr = ctr_0 \dots ctr_{27} \in \mathbb{F}_2^{28}$, *a button identifier* $btn_0 \dots btn_3 \in \mathbb{F}_2^4$ *and keystream* $ks = ks_0 \dots ks_{31} \in \mathbb{F}_2^{32}$, *we let the initialization vector* $iv \in \mathbb{F}_2^{32}$ *be defined as*

$$iv = ctr || btn.$$

Furthermore, the internal state of the cipher at time i *is* $\alpha_i := a_i \dots a_{47+i} \in \mathbb{F}_2^{48}$. *Here the* $a_i \in \mathbb{F}_2$ *are given by*

$$\begin{aligned} a_i &:= id_i && \forall i \in [0,31] && (1) \\ a_{32+i} &:= k_i && \forall i \in [0,15] && (2) \\ a_{48+i} &:= k_{16+i} \oplus iv_i \oplus f(a_i \dots a_{i+47}) && \forall i \in [0,31] && (3) \\ a_{80+i} &:= L(a_{32+i} \dots a_{79+i}) && \forall i \in \mathbb{N}\,. && (4) \end{aligned}$$

Furthermore, we define the keystream bit $ks_i \in \mathbb{F}_2$ *by*

$$ks_i := f(a_{32+i} \dots a_{79+i}) \qquad \forall i \in [0,31]. \quad (5)$$

Note that the a_i, α_i, *and* ks_i *are formally functions of* k, id, *and* iv. *Instead of making this explicit by writing, e.g.,* $a_i(k, id, iv)$, *we just write* a_i *where* k, id, *and* iv *are clear from the context.*

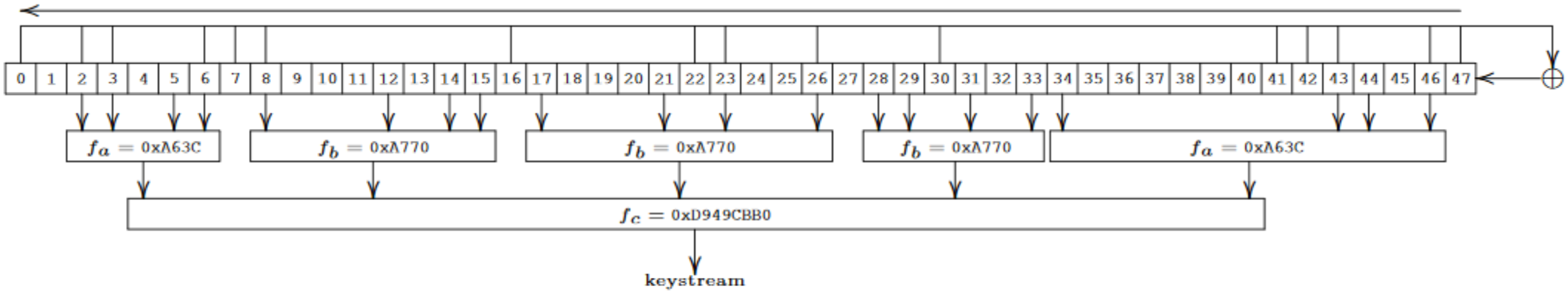


Figure 10: Structure of the Hitag2 stream cipher, based on [35]

CASE STUDY 2 – HITAG2 SYSTEM

- Hitag2 Correlation Attack

1. The adversary first guesses a 16-bit window corresponding to LFSR stream bits $a_{32} \dots a_{47}$. Observe that $a_{32} \dots a_{47} = k_0 \dots k_{15}$ and together with the UID, this gives the adversary LFSR bits $a_0 \dots a_{47}$, see Definition 4.4. Also note that $a_0 \dots a_{47}$ is constant over all traces. The adversary can now compute $b_0 = f(a_0 \dots a_{47})$.

2. The adversary will then shift this 16-bit window to the left of the LFSR, until bits $a_{32} \dots a_{47}$ are on the very left of the LFSR. This is the point when the cipher starts outputting ks, see Equation 5.

3. Next, the adversary will compute a correlation score for this guess. The window determines 8 input bits $x_0 \dots x_7$ to the filter function f_{20} (see Figure 10) while the remaining 12 inputs remain unknown. This correlation is taken as the ratio of those 2^{12} input values $x_8 \dots x_{19}$ that produce the correct keystream bit (ks_0). Furthermore, shifting our window further to the left allows the adversary to perform tests on multiple keystream bits $(ks_0 \dots ks_{15})$. Although, with every bit shift, the window becomes smaller as the leftmost bits will fall outside the LFSR, meaning that more input bits are unknown.

CASE STUDY 2 – HITAG2 SYSTEM

- Hitag2 Correlation Attack

Definition 4.5 *We define the single-bit correlation score as:*

$$bit\_score(x_0 \dots x_{n-1}, b) = \frac{\#(b = f_{20}(y_0 \dots y_{19}))}{2^{19-n}}$$

where $y_0 \dots y_{n-1} = x_0 \dots x_{n-1}, n < 20$ *(at the first iteration of Step 3, n=8). We define the multiple-bit correlation score as:*

$$score(x_0, ks_0) = bit\_score(x_0, ks_0)$$

$$score(x_0 \dots x_{n-1}, ks_0 \dots ks_{n-1}) = bit\_score(x_0 \dots x_{n-1}, ks_{n-1}) * score(x_0 \dots x_{n-2}, ks_0 \dots ks_{n-2})$$

for $n < 20$.

Definition 4.5 *We define the single-bit correlation score as:*

$$bit\_score(x_0 \dots x_{n-1}, b) = \frac{\#(b = f_{20}(y_0 \dots y_{19}))}{2^{19-n}}$$

where $y_0 \dots y_{n-1} = x_0 \dots x_{n-1}, n < 20$ *(at the first iteration of Step 3, n=8). We define the multiple-bit correlation score as:*

$$score(x_0, ks_0) = bit\_score(x_0, ks_0)$$

$$score(x_0 \dots x_{n-1}, ks_0 \dots ks_{n-1}) = bit\_score(x_0 \dots x_{n-1}, ks_{n-1}) * score(x_0 \dots x_{n-2}, ks_0 \dots ks_{n-2})$$

for $n < 20$.

4. The adversary will now sort all guesses according to their score and store them in a table of fixed size, discarding the guesses with lowest scores when needed. Experiments show that a table of size 400,000 guesses is usually sufficient.

5. For each guess in the table, the adversary goes back to Step (1) and proceeds as before, except that she will now extend the window size by one (to size $17, \dots, 32$), guessing the next LFSR stream bit $(a_{48}, \dots, a_{51})$. The bigger window allows the adversary to test on an additional bit of keystream, giving her more meaningful correlation information each time. Special care needs to be taken at Step (3) while scoring multiple traces, since $a_{48} = k_{16+i} \oplus iv_i \oplus b_0$ (see Eq. 3) and the iv will be different in each trace. This is not a problem since in the previous Step (1) we had computed the corresponding keystream bit b_i, and iv_i is sent in clear. Therefore key bits k_{16+i} can be computed for $i \in [0, 31]$.

CASE STUDY 2 – HITAG2 SYSTEM

- Results:
  - ~1-Minute Average to crack with typical Laptop
  - Maximum Crack time: ~10-Minutes
  - Issue does arise when guessing the 18-MSBs of counter
    - Not a big deal though. Counter MSBs can be predicted by model year of car

OVERALL

- VW RKEs are vulnerable because of Master Key reuse
  - Only takes a recording of 1-button press transmit to crack
- Hitag2 RKEs are vulnerable due to flaw in cryptography
  - Takes 4-8 Button Presses to crack

CONCLUSION

"Lock it or Lose it" is no longer a valid statement (in some cases)

DISCLOSURE

Regarding the vulnerabilities of VW Group systems, we contacted VW Group first in November 2015. We discussed our findings in a meeting with VW Group and an affected sub-contractor in February 2016, before submitting the paper. VW Group received a draft version of this paper and the final version. VW Group acknowledged the vulnerabilities. As mentioned in the paper, we agreed to leave out amongst others the following details: cryptographic keys, part numbers of vulnerable ECUs, and the used programming devices and details about the reverse-engineering process.

For Hitag2, we notified NXP in January 2016. NXP received a version of this paper before submission. We would like to mention that the fact that Hitag2 is cryptographically broken has been publicly known for several years and NXP has already informed their customers back in 2012. We would further like to highlight that for several years, NXP offers newer, AES-based RKE ICs that are not affected by the vulnerabilities described in this paper. Furthermore, many car manufacturers have already started using the more secure chips for new designs.

Questions / Comments / Discussion