MIFARE Classic Exploits

Julius Putra Tanu Setiaji

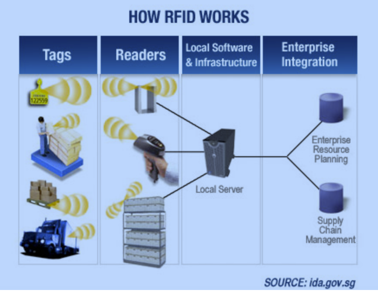
8 June 2018

Disclaimer

The author isn't responsible by the use of the presented content to do illegal activities. Do it at your own risk!

RFID (Radio-frequency identification)

- Uses electromagnetic fields to automatically identify and track tags containing electronically-store information.
- Passive tags collect energy from a nearby RFID reader's interrogating radio waves.
- Active tags have a local power source and may operate hundreds of meters from the RFID reader.



SOURCE: ida.gov.sg

NFC (Near Field Communiation)

- A subset of RFID with much shorter communication ranges.
- Unlike most RFID reader-tag pairs, they are able to function as both a reader and a tag:
  1. Card Emulation Mode (Android/Apple Pay)
  2. **Reader/Writer Mode**
  3. Peer to Peer Mode (Android Beam)

MIFARE

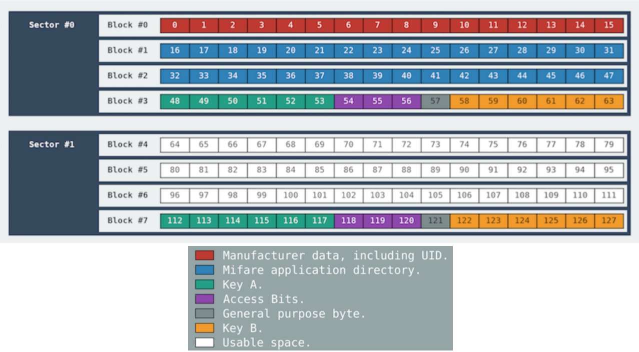
- MIFARE is a group of chips introduced by NXP Semiconductors that is used widely in contactless smart cards.
- Introduced in 1995, it is most commonly used in public transportation, access control and ticketing systems.
- There are 4 types:
  1. **MIFARE Classic**
  2. MIFARE Plus (AES-128)
  3. MIFARE Ultralight
  4. MIFARE DESFire (DES, 3DES)

MIFARE Classic

- The MIFARE Classic card is generally a memory storage device, where its memory is divided into segments and blocks.
- There are 3 types of MIFARE Classic cards:
  1. **MIFARE Classic 1K** (most common)
  2. MIFARE Classic 2K
  3. MIFARE Classic 4K
- Compliant with parts 1-3 (out of 4) of ISO/IEC 14443
- Operating at 13.56 MHz with range of up to 10 cm
- Proprietary protocol for authentication and ciphering (CRYPTO-1)
- 4 bytes UID

MIFARE Classic 1K

- 1024 bytes, split into 16 sectors (of 64 bytes each), each divided into 4 blocks (of 16 bytes each).
- Each sector is protected by two different keys, each 6-bytes long.and 4-bytes Access Condition specifier.
- Hence, effectively only 768 bytes are available.

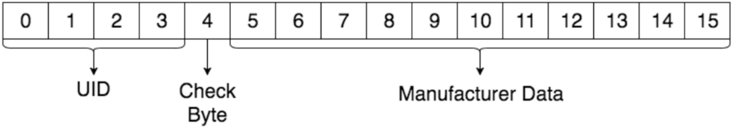


Sector #0	Block #0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	Block #1	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Block #2	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
	Block #3	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
Sector #1	Block #4	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
	Block #5	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
	Block #6	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
	Block #7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127

🟥 Manufacturer data, including UID.
🟦 Mifare application directory.
🟩 Key A.
🟪 Access Bits.
⬛ General purpose byte.
🟧 Key B.
⬜ Usable space.

Manufacturer Block

- First block of sector 0 is known as Manufacturer Block.
- First 4 bytes are the UID, next byte is Bit Count Check (XOR of the UID bytes).
- the remaining eleven bytes are used to store the manufacturer's data.
- This further reduces the available space to 752 bytes.
- This block is written to and locked in the factory, thus preventing modification.



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

UID

Check Byte

Manufacturer Data

Sector Trailer

- Block 3 of each sector is called the Sector Trailer.
- Used to store 2 secret keys, Key A and Key B of 6 bytes each.
- Bytes 6-9 are used to store the access bits meant for accessing the four blocks in each sector.

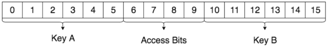
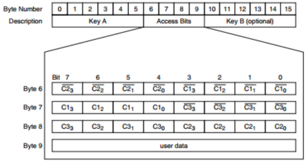


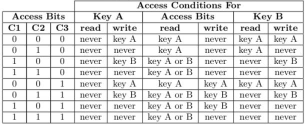
Figure 4: Sector Trailer



Byte Number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Description	Key A						Access Bits				Key B (optional)					

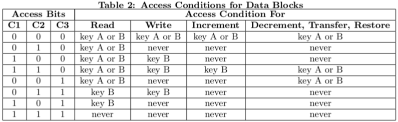
Bit	7	6	5	4	3	2	1	0
Byte 6	$\overline{C2_3}$	$\overline{C2_2}$	$\overline{C2_1}$	$\overline{C2_0}$	$\overline{C1_3}$	$\overline{C1_2}$	$\overline{C1_1}$	$\overline{C1_0}$
Byte 7	$C1_3$	$C1_2$	$C1_1$	$C1_0$	$\overline{C3_3}$	$\overline{C3_2}$	$\overline{C3_1}$	$\overline{C3_0}$
Byte 8	$C3_3$	$C3_2$	$C3_1$	$C3_0$	$C2_3$	$C2_2$	$C2_1$	$C2_0$
Byte 9	user data							

Table 1: Access Conditions for Sector Trailer



			Access Conditions For					
Access Bits			Key A		Access Bits		Key B	
C1	C2	C3	read	write	read	write	read	write
0	0	0	never	key A	key A	never	key A	key A
0	1	0	never	never	key A	never	key A	never
1	0	0	never	key B	key A or B	never	never	key B
1	1	0	never	never	key A or B	never	never	never
0	0	1	never	key A	key A	key A	key A	key A
0	1	1	never	key B	key A or B	key B	never	key B
1	0	1	never	never	key A or B	key B	never	never
1	1	1	never	never	key A or B	never	never	never

Table 2: Access Conditions for Data Blocks



Access Bits			Access Condition For			
C1	C2	C3	Read	Write	Increment	Decrement, Transfer, Restore
0	0	0	key A or B	key A or B	key A or B	key A or B
0	1	0	key A or B	never	never	never
1	0	0	key A or B	key B	never	never
1	1	0	key A or B	key B	key B	key A or B
0	0	1	key A or B	never	never	key A or B
0	1	1	key B	key B	never	never
1	0	1	key B	never	never	never
1	1	1	never	never	never	never

Short history of CRYPTO-1

- In December 2007, two German researchers, Nohl and Plötz) presented at CCC the partial reverse engineering of Crypto-1 with some weaknesses.
- They partially reverse-engineered by slicing the chip and taking pictures using a microscope.
- In March 2008, a research group from Radbond University completely reverse-engineered the Crypto-1 cipher by analysing the communication between the tag and the reader.
- They intended to publish it, however NXP tried stop the full disclosure of Crypto-1 cipher by judicial process.
- However, in July 2008 the court decides allow the publication of the paper and reject the prohibition based in freedom of speech principles.

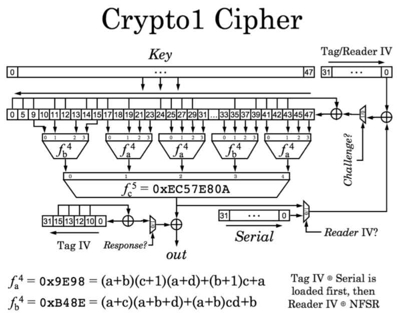
CRYPTO-1 (1/3)

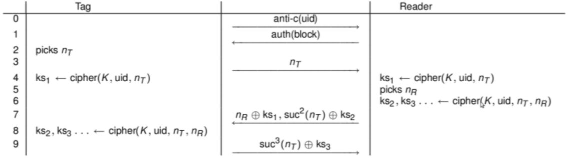
- A stream cipher that uses a 48-bit secret key.
- The card sends a challenge nonce n_T, after which the reader sends the encrypted reader nonce $n_R \oplus ks_1$ and challenge response $suc^2(n_T) \oplus ks_2$.
- The reader completes the 3-way authentication by sending the encrypted challenge response $suc^3(n_T) \oplus ks_3$.
- The 32 bit nonces are generated by a 16 bit linear feedback shift register (LSFR).
- In this case, $suc(x)$ refers to the next 32 bits generated by the LSFR after x.
- ks_1, ks_2, ks_3 are key stream generated by cipher (32 bits each.)

CRYPTO-1 (2/3)

- At the heart is a 48 bit feedback shift register which is initialized with with the secret key K, the uid and the n_T, and later n_R is fed in.
- 20 bits of the feedback shift register are used as input to a filter function to generate the keystream.
- The researchers were able to invert the filter function so as to effectively generate all the possible internal states of the feedback shift register given a partial keystream.

CRYPTO-1 (3/3)





	Tag		Reader
0		anti-c(uid) →	
1		← auth(block)	
2	picks n_T		
3		n_T →	
4	$\mathsf{ks}_1 \leftarrow \mathsf{cipher}(K, \mathsf{uid}, n_T)$		$\mathsf{ks}_1 \leftarrow \mathsf{cipher}(K, \mathsf{uid}, n_T)$
5			picks n_R
6			$\mathsf{ks}_2, \mathsf{ks}_3 \ldots \leftarrow \mathsf{cipher}(K, \mathsf{uid}, n_T, n_R)$
7		← $n_R \oplus \mathsf{ks}_1, \mathsf{suc}^2(n_T) \oplus \mathsf{ks}_2$	
8	$\mathsf{ks}_2, \mathsf{ks}_3 \ldots \leftarrow \mathsf{cipher}(K, \mathsf{uid}, n_T, n_R)$		
9		$\mathsf{suc}^3(n_T) \oplus \mathsf{ks}_3$ →	

Replay Attack and Active Sniffing

- After the 2008 publication of the full CRYPTO-1 cipher, any attacker is able to emulate any Mifare card by just sniffing the communication between the card and reader and replaying it (including the UID value).
- Also, the attacker will be able to recover all keys from sectors involved in this communication.
- However, this attack needs to sniff the communication between the card and a valid reader.
- The hardware required are also rather expensive and not easily accessible.

Darkside Attack

- Introduced in 2009 by Nicolas Courtois and implemented by Andrei Costin with the `MFCUK`.
- During the 3-step authentication, when the reader sends $n_R \oplus ks_1$ and $suc^2(n_T) \oplus ks_2$, the tag checks the 8 parity bits before checking the correctness of $suc^2(n_T) \oplus ks_2$.
- If the parity bits for these 8 bytes are correct but $suc^2(n_T) \oplus ks_2$ is wrong, the card will respond with a 4-bit encrypted error code (NACK) indicating a transmission error, 0x5 $\oplus k$ where k is the first 4 bits of ks_3.
- However, if the parity bits are wrong, the card does not respond.
- This allows the attacker to correctly guess 4 bits of the keystream after an average of 2^8 tries.

Other Weaknesses

- The keys are only 48-bits long. Can be brute-forced with FPGA, approximately 10 hours to recover one key.
- The LFSR used by the RNG is predictable (constant initial condition)
- Each random number only depends of the quantity of clock cycles between: the time when the reader was turned up and the time when the random number is requested.
- Since an attacker controls the time of protocol, one is able to control the generated random numbers and that way recover the keys from communication.

Explanation

- Recover all keys after at least one key has been found, taking advantage of the weakness in the RNG.
- Introduced in 2009 by Nijmegan Oakland and Implemented by Nethemba with the `mfoc` tool.
- When attempting to authenticate to another sector, the card will send $n_T \oplus k'$ where n_T is the nonce and k' is the keystream generated by the key of the new sector.
- Hence, by correctly guessing the nonces, a partial keystream can be found.
- From the invertibility of the filter function, each correctly guessed nonce will result in a set of possible candidate keys. An intersection of these sets will quickly find the required secret key.

Less cheem explanation

- Authenticate to a block with known key and read n_T (determined by LFSR)
- Authenticate to the same block again with the default key and read n'_T (determined by LFSR)
- Compute the number of LFSR shifts (”timing distance”)
- Guess the next n_T value, calculate ks_1, ks_2, ks_3 and try authenticating to a different block.

Hardened MIFARE Classic Cards

- In light of this, many manufactures and system integrators started to deploy "fixed" mifare Classic cards which are resilient to such vulnerabilities.
- However, these countermeasures are inadequate for a cryptographically insecure cipher such as CRYPTO-1.
- Instead of taking advantage of the MIFARE protocol and its implementations (non-cryptographically related implementation flaws), researchers look into breaking the CRYPTO-1 cipher itself.

Collecting nonces stage (1/2)

- The information obtained allows an attacker to drop the computational complexity from 2^{48} to approximately 2^{30}
- Retrieve encrypted nonces n_T using the nested authentication, i.e. by authenticating for a sector with a known key, followed by an authentication request for the request for the target sector.
- Given the set of encrypted nonces obtained so far, determine sum property of the cipher's initial state S_e and of the cipher's state after byte b is fed, S_b for all 256 possible first input bytes b.
- Depending on the probability that we guessed S_b correctly (using a probability threshold value), incorporate byte b in the differential analysis, and incorporate all first nonce bytes for which the filter flip property holds.

Collecting nonces stage (2/2)

- Given the information determined from the set of encrypted nonces, we determine the size of the leftover search space.
- The leftover search space shrinks as the number of harvested encrypted nonces increases since more nonces allows us to more accurately guess sum properties and observe filter flip properties.
- When the search space is sufficiently small, we construct a candidate list for $a_{[9,55]}$, extended to $a_{[8,55]}$, then performing an LFSR-rollback to transform them into candidates for $a_{[0,47]}$, i.e. the secret key.

Brute-force Stage

- This candidates list can then be used for offline brute force attack (which can be parallelised!)
- Parity bits are computed over plaintext byte XOR-ed with the next keystream bit. This property can be exploted to verify whether a candidate key is the correct key.
- Given an encrypted nonce obtained through a nested authentication attempt, the attacker can attempt to "decrypt" the nonce using the candidate key.
- In case the candidate is the correct key, the parity bits will be correct. However, in case a wrong key was used, a parity bit will be correct with probability $\frac{1}{2}$
- If the key is not found, revert to Stage 2 optionally with an increased probability threshold. However, gathering of more nonces increases the certainty and reduces the number of candidate keys.

Further Reading

- The offline brute-forcing part can be improved by using bit-slicing, achieving 8-10 times speedup. (https://github.com/aczid/crypto1_bs)
- Details about this attack is available on this paper: http://www.cs.ru.nl/~rverdult/Ciphertext-only_Cryptanalysis_on_Hardened_Mifare_Classic_Cards-CCS_2015.pdf

Closing Statements

- MIFARE Classic practically offers no security all, just like WEP for the Wi-Fi standard.
- Moreover, in reality, there are other ways to defeat MIFARE Classic security system.
- For example, some MIFARE Classic cards from China allows first block of sector 0 (Manufacturer Block) to be rewritten. This defeats some systems that bases identification on UID.

Live Demonstration

The part y'all have been waiting for! Thank you very much.