From talk on youtube:

Reverse engineered Mifare crypto to evaluate security.

They reconstructed circuit from photos of chip, and sniffed reader-tag communication.

Chip is 1 square mm, in layers. They sliced layer and photographed each layer. Classified logic gates and used certain features (XOR, flip-flops) to locate crypto regions. Reconstructed circuit.

16 bit random number generator, blown up to 32b. Value of random number depends on time between powering up chip and challenging it, which could in theory be controlled. Nohl and Plotz were able to generate the same number repeatedly.

This 16 bit random number is used to create the entropy in a poor pseudo-random number generator based on 48b shift Linear Feedback Shift Register (known technique), values of which feed into a function to output a key stream, one bit per cycle. No nonlinear component to LFSR loop so no forward secrecy (can not only predict future steps but also reconstruct earlier steps)

So essentially all the randomness can be controlled by an attacker as long as attacker can control timings.

Output key bit depends on fixed subset of LFSR bits, implying non-optimal avalanche properties (small changes of the input may not result in significant changes in the output) so there may be a sub-exponential attack on the key. But brute force is actually sufficient due to low number of bits in initial PSR.

Takes 1 week on $100 device to brute force, 1 day on $700 device etc. so a certain amount of investment is required. Mifare may be sufficient to protect low-value things e.g. small payments but not to protect high-value targets e.g. access control, car theft protection, credit cards.

So, security relied on obscurity of the implementation details.

Could generate pairs of cards with same initial cipher state.

From Wikipedia

Security Flaw in MIFARE Classic paper:

MIFARE Classic smartcard was developed in mid 90s. Not programmable. Crypto operations implemented in hardware using LFSR and “filter function”. This implements proprietary algorithm CRYPTO1, which is NXP trade secret.

Card’s memory is in sectors, each protected by 2 crypto keys. Typically all cards protected by the same keys in an access control system. (alternatively, reader must look up keys associated with encountered card)

Attacked Dutch “ov-cihpkaart” system, used in dutch public transportation.

Found weaknesses in authentication:

* Working of the CRYPTO1 encryption algorithm was reconstructed by Nohl and Plotz.
* Easy method to retrieve cryptographic keys, doesn’t require expensive equipment.

Used flawed authentication attempts to reverse engineer CRYPTO1. Once known, find keys with brute-force attack. 48b keys, required 9 hours on advanced equipment. But instead, carry out many failed auth attempts, store results in table, look for match. Table only has to be computed once, prepared by running CRYPTO1 on a fixed input.

Recovery of key makes system that reuses the key vulnerable. Attack can be performed with standard reader using libnfc.

Differs from entirely hardware-based approach of Nohl and Plotz, exploit protocol weakness.

“A practical attack on the MIFARE classic”:

Uses weakness of pseudo-random generator to recover the CRYPTO1 keystream. Exploit malleability of stream cipher to read memory blocks. Can even modify memory blocks.

“Wirelessly pickpocketing a Mifare classic card”:

Proposes 4 attacks that require adversary only having wireless reader and accessing a card. One recovers key in <1s.

1. ISO 14443-A requires every byte sent has parity bit. Mifare classic computes parity bits over plantext, not ciphertext. Exploit weakness of parity bits to mount offline 48-bit brute-force attack. Due to weakness, only around 1500 authentication attempts necessary.
2. Also exploits parity bit weakness. Adaptive chosen ciphertext attack. Approx. 28500 auth attempts. Adaptively choose challenge to card, eventually obtaining a challenge that guarantees 436 possibilities for odd-numbered bits of the internal state of the cipher. Reduces offline search space to 33 bits.
3. Attacker keeps challenge constant, varies the tag’s challenge until a special internal state of the cipher is obtained. These states are stored in 384GB table, and adversary looks up card responses in the table. Takes around 2^12=4096 auth attempts on average.
4. Assume attacker already recovered one sector key. Authenticate for that sector then a different sector. Challenge nonce of new sector is encrypted with that sector key. Random number generator has 16b state, parity bits leak information, and random number generator is synched with communication timing, so attacker can guess plaintext tag nonce, and hence 32b of keystream. Compute 2^16 candidate keys, checked offline. Only requires 3 short authentication attempts, offline search is under a second.

There are numerous attacks that are variations on the ones mentioned so far.

“Conditional Multiple Differential attack on MIFARE Classic”  
few hundred queries, no pre-computation, very fast speed. Card-only attack. Finds key.

NXP replaced MIFARE classic with backwards compatible Mifare classic EV1, around 2011, resistant to known card-only attacks. In 2015m a new card-only attack was found.

MIFARE DESFire cards were proven insecure in 2010 when it was shown they could be cloned for about $25 in cheap hardware. (cite)

Other problems discovered.

MIFARE Ultralight also has some attacks against it.