# Ein kompletter Bruch der KeeLoq Chiffre und seine Einsatzszenarien

Andreas Gross

Sommerakademie La Colle-sur-Loup

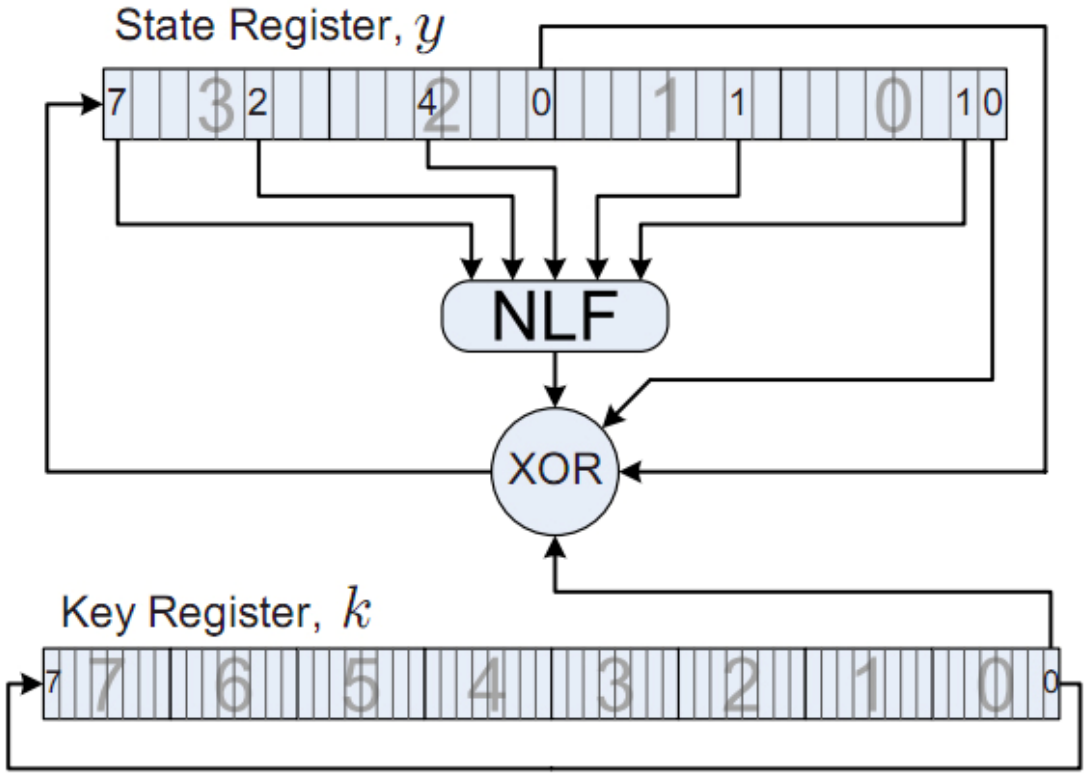
Dienstag, der 29. September 2009

Ein Vortrag über das Paper „On the power of power analysis in the real world: A complete break of the KeeLoq code hopping scheme“ von Thomas Eisenbarth, Timo Kasper, Amir Moradi, Christof Paar, Mahmoud Salmasizadeh, and Mohammad T. Manzuri Shalmani.

# Inhalt

1. Die KeeLoq Chiffre
2. Das Hopping-Code Protokoll
3. Key Derivation Varianten
4. Die Seitenkanalanalysen
   a) Technische Vorbereitungen
   b) Energiemodell
   c) Ergebnisse
5. Anwendungsszenarien
   a) Klonen eines Transmitters
   b) Erlangen eines Herstellerschlüssels
   c) Klonen eines Transmitters ohne physischen Zugang
   d) Außer Funktion setzen der Anlage

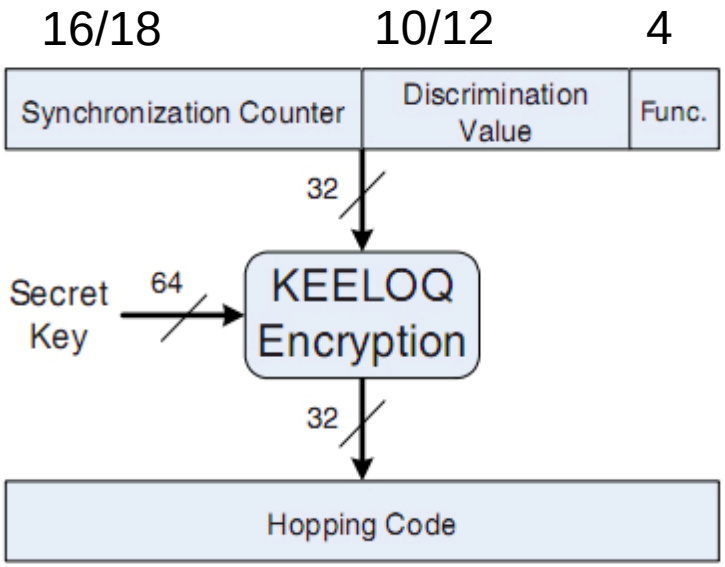
# 1. Die KeeLoq Chiffre

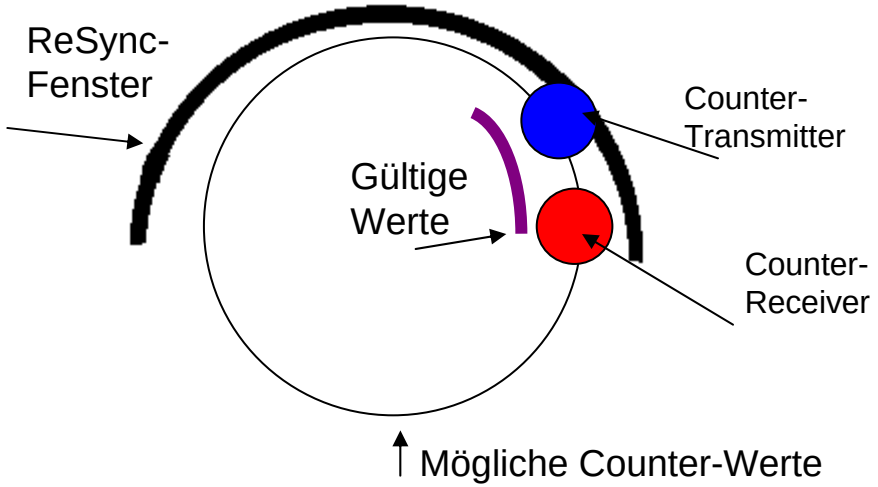



- NLF=Non-linear function
- KeeLoq ist eine Blockchiffre
- Blocklänge=32bit
- Schlüssellänge=64bit
- 528 Runden

# 2. Das Hopping-Code Protokoll

- Ziel: Authentifizierung des Transmitters beim Receiver
- Zwei Möglichkeiten:
    1. Challenge-Response-Protokoll (bidirektional) mit Nonces (Random Numbers)
    2. unidirektional mit synchronisiertem Counter oder Zeitstempel
- Bei Keeloq: unidirektional mit synchronisiertem Counter
- Hopping-Code = Der Plaintext ändert sich bei jeder Anwendung
- => Schutz vor Replay-Attacken
- Im speziellen Fall von Keeloq besteht der gesendete Plaintext aus drei Teilen, von denen sich einer verändert:
    - *Counter:* Der sich verändernde Teil
    - *Discrimination Wert:* Zur Identifizierung des Transmitters
    - *Befehl:* Gibt z.B. an welches Garagentor geöffnet werden soll




Der Counter ist sowohl im Transmitter als auch im Receiver gespeichert und erhöht sich nach jeder erfolgreich gesendeten Nachricht.

Es kann vorkommen, dass der Counter-Wert des Transmitters höher ist als der des Receivers.

Ist er um weniger als 16 höher, so funktioniert alles wie gewohnt.

Ist er um mehr als 16 höher, aber weniger als $32768=2^{15}$, so resynchronisieren sich Receiver und Transmitter. Dazu muss der Transmitter zweimal gedrückt werden.

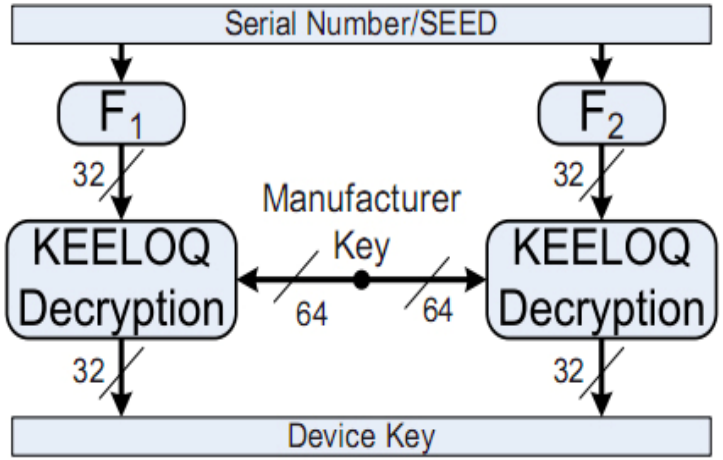
Andernfalls passiert nichts. (Siehe Denial of Service Scenario 5d)

Bei Verwendung von KeeLoq im Hopping-Code Modus passiert also Folgendes:

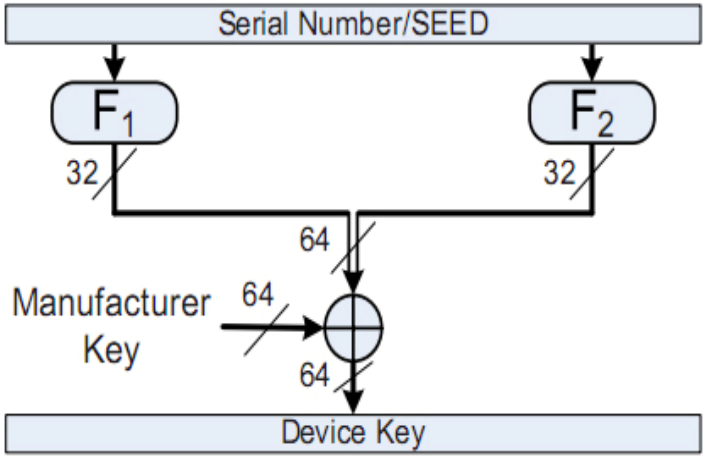
1. Transmitter verschlüsselt Nachricht
2. Transmitter sendet Nachricht zusammen mit unverschlüsselter Seriennummer
3. Receiver empfängt Nachricht und entschlüsselt sie
4. Receiver überprüft ob Discrimination-Wert mit gespeichertem übereinstimmt und ob Counter-Wert im gültigen Bereich liegt
5. Eventuell wartet der Receiver auf eine weitere Nachricht des Transmitters zwecks Resynchronisierung

# 3. Key Derivation Varianten

- Jeder Hersteller hat soweit wie bekannt einen eigenen Hersteller-schlüssel.
- Schlüssel eines Geräts wird erzeugt aus Herstellerschlüssel und Seriennummer/Seed.
- Herstellerkey in jedem Receiver gespeichert
- Für diesen Vorgang gibt es vier bekannte Varianten, wie im Folgenden dargestellt:




(a) Starke KDF

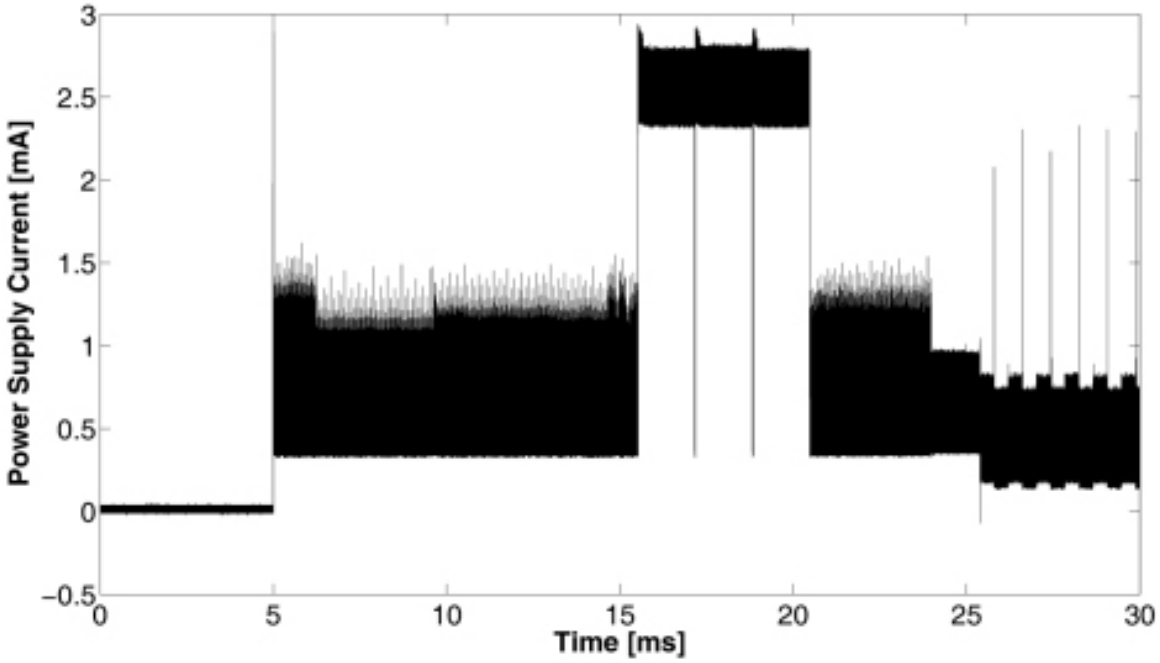



(b) Schwache KDF

# 4. Die Seitenkanalanalyse

## a) Technische Vorbereitungen

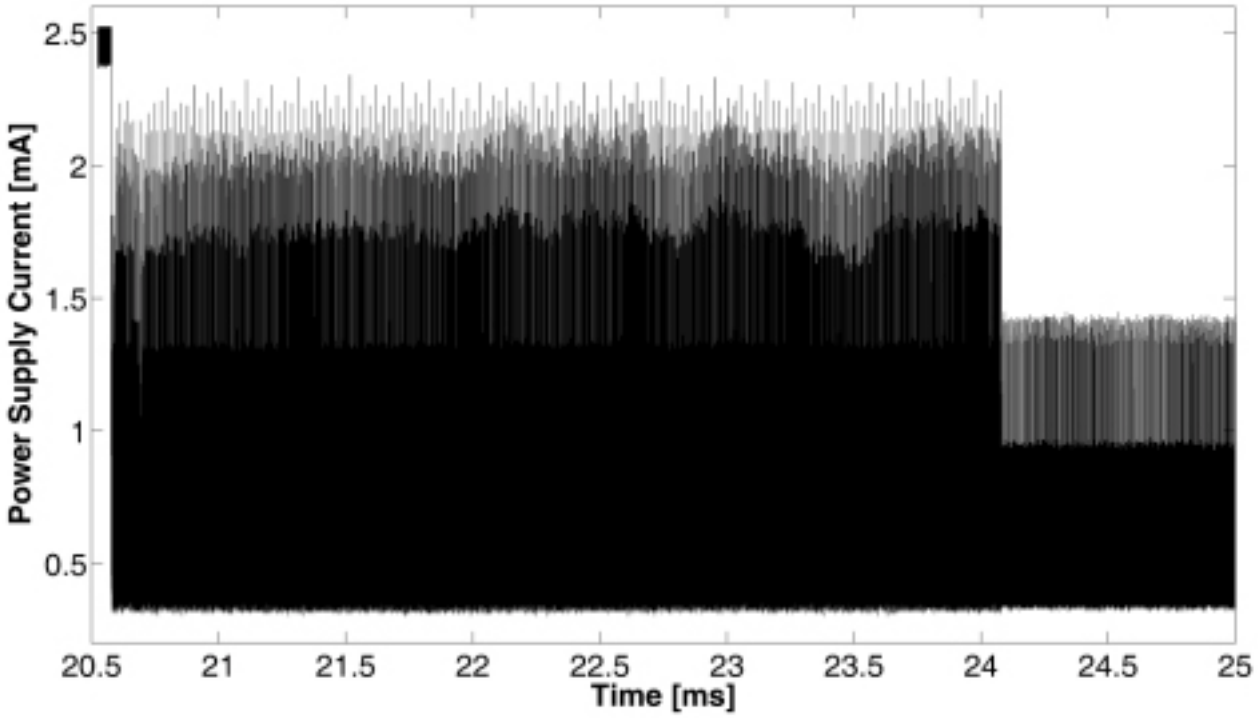
Leistungsbedarfmessung bei KeeLoq im Transmitter mittels Oszilloskop:



Durch ein Trial-and-Error Verfahren fand man heraus, dass der größte Energiebedarf entsteht während das Gerät in den EEPROM schrieb.

Die Verschlüsselung findet danach statt.

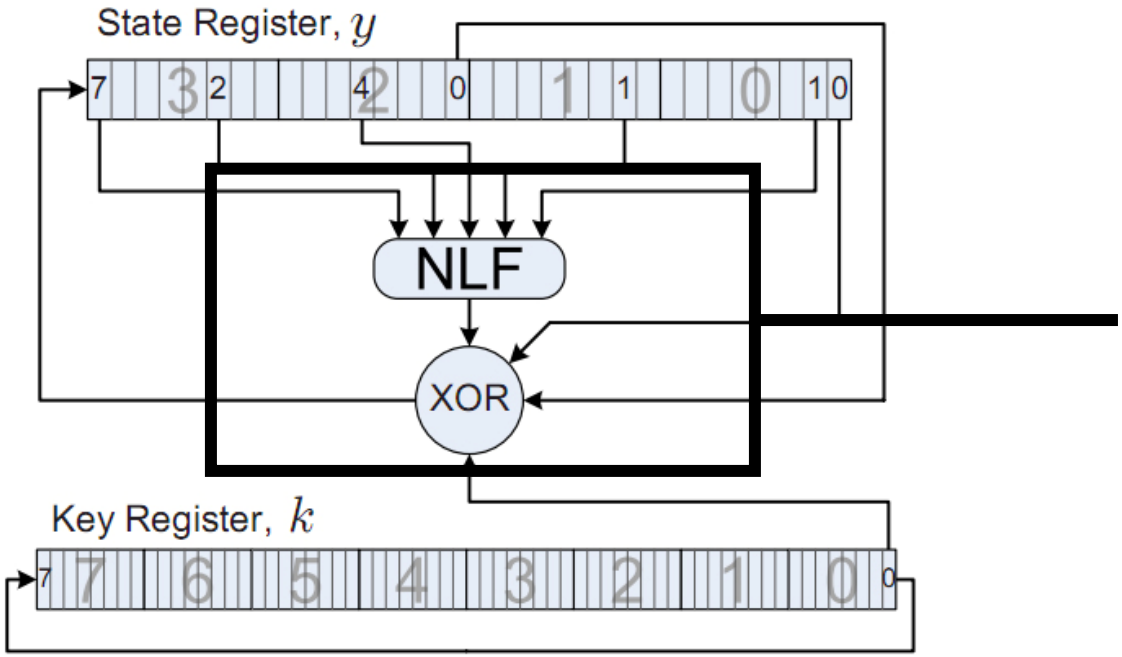
Hier nochmal der Teil in dem die Verschlüsselung stattfindet:



# 4. Die Seitenkanalanalyse

## b) Energiemodell

Um eine DPA anwenden zu können braucht man nun noch ein Energiemodell:




vernachlässigbarer Energiebedarf

Bleibt noch der Bedarf der Shift-Register zu modellieren.

Uns interessieren nur die Stellen, an denen die Belegungen der Shift-Register geändert werden.

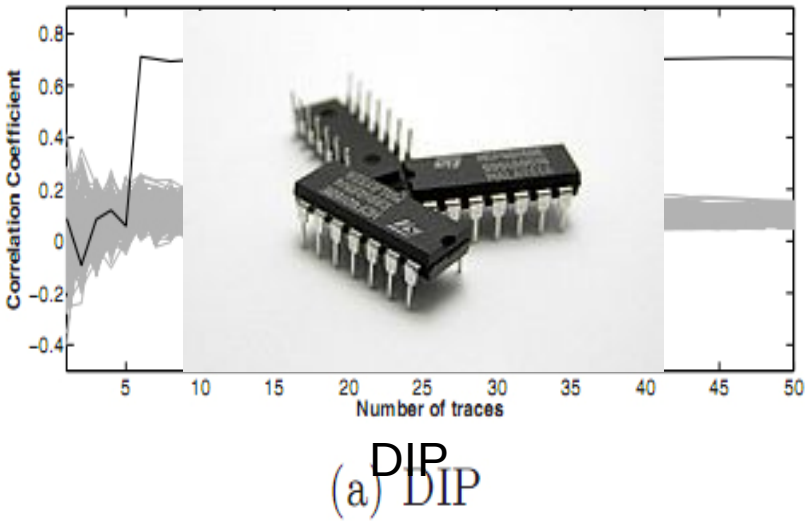
Wir nehmen an, dass der Strombedarf der Shift-Register während der Neubelegung proportional zur Hamming-Distanz zwischen alter und neuer Belegung ist. (Hamming-Distanz-Modell)

=> Key-Register hat bei den Wechseln konstanten Strombedarf

# 4. Die Seitenkanalanalyse

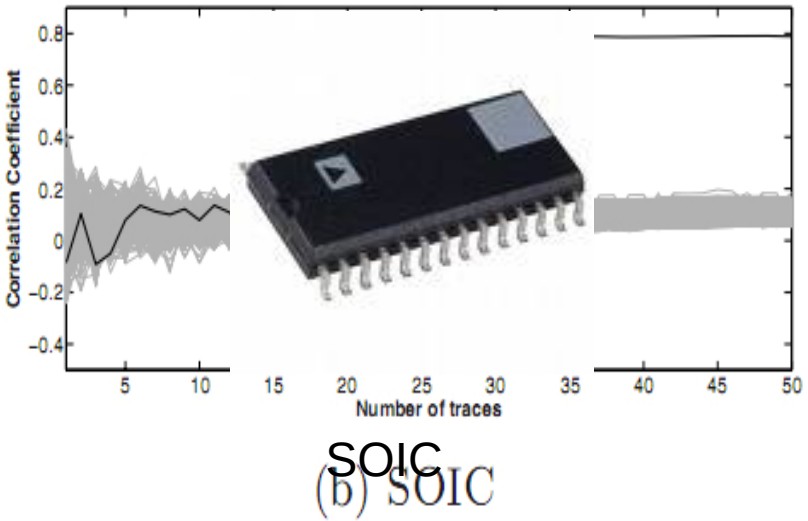
## c) Ergebnisse

- Sowohl DPA als auch DEMA lieferten bei den Transmittern hervorragende Ergebnisse.
- Man benötigt maximal 10 Stromverläufe bei Dual Inline Package (DIP) und maximal 30 bei Small Outline Integrated Circuit (SOIC).




DIP

(a) DIP




SOIC

(b) SOIC

- Um den Herstellerschlüssel aus dem Receiver zu bekommen wurden etwa 1000 Stromverläufe benötigt.
- Verwendung von DPA/CPA
- Für SCA bei Transmittern genügt billige Ausstattung:
- Mit einem Oszilloskop mit einer Abtastrate von nur 50 MS/s benötigt man etwa 60 Stromverläufe (DIP). Die Kosten für die Ausrüstung belaufen sich dann auf unter 200 Euro.

# 5. Anwendungsszenarien

## a) Klonen eines Transmitters

- Physischer Zugang
- Erlange Device-Key durch DPA
- Entschlüssele eine Transmitternachricht
- Klone damit einen Transmitter
- Original und Klon nun ununterscheidbar für Receiver

Folgende Szenarien wären denkbar:

- Klonen eines Transmitters durch nicht vertrauenswürdige Personen mit Zugang zu Transmitter
- Auto mieten → Transmitter klonen
- Wiederholter Zugriff auf Gebäude/Auto möglich
- → Keine Spuren
- → Probleme mit Versicherung möglich

# 5. Anwendungsszenarien

## b) Einen Herstellerschlüssel erlangen

- Hier unterscheiden wir zwei Fälle:
  1. Schwache Key Derivation Variante wird verwendet
  2. Starke Key Derivation Variante wird verwendet.
- 1. Fall: physischer Zugang zum Transmitter benötigt
- ➔ Extrahiere Device-Key ➔ Berechne Herstellerschlüssel
- 2. Fall: physischer Zugang zu einem Receiver benötigt
- ➔ Extrahiere Hersteller-Key durch DPA/CPA

Dies kann folgenderweise für kriminelle Zwecke eingesetzt werden:

- KeeLoq-System erwerben ➔ Herstellerschlüssel extrahieren ➔ KeeLoq-System zurückgeben
- Neue Transmitter herstellen ➔ kompatibel mit Geräten der Firma
- Firmenprodukt also anfällig gegen Marktpiraterie
- Klonen von Transmittern ohne physischen Zugang (siehe 5d)
- Herstellerschlüssel im Internet publizieren/verkaufen

# 5. Anwendungsszenarien

## c) Klonen eines Transmitters ohne physischen Zugang

- Max. 2 Nachrichten werden benötigt.
- Das Verfahren hängt von der Key Derivation Variante ab.
- 1. Key-Derivation aus Seriennummer ➔ Eine Nachricht ausreichend.
- Berechne Device-Key aus Seriennummer ➔ Entschlüssele Nachricht ➔ Klone Transmitter
- 2. Key-Derivation aus Seeds ➔ Zwei Nachrichten benötigt
- Probiere alle Seeds durch
- Berechne Testschlüssel aus Seed ➔ Entschlüssele Nachrichten
- Richtiger Seed gefunden, wenn Discrimination-Werte der beiden Plaintexte gleich sind und die Counter-Werte sich ähneln

- Enormes Potential für Angreifer
- Beschaffung des Herstellerschlüssels kann (kriminellen) Experten überlassen werden
- ➔ Laien können einen Einbruch in die Garage/in das Auto bewerkstelligen.
- Keine Spuren bei Einbruch
- KDF-Variante mit Seeds findet in Praxis kaum Einsatz, obwohl sicherer
- 64 bit Seed ➔ Rechenzeit so hoch, dass Klonen von Schlüsseln extrem schwierig/langwierig wird

# 5. Anwendungsszenarien

## d) Außer Funktion setzen der Anlage.

- Nehmen an, dass Transmitterdaten bekannt, z.B durch a) oder d)
- Setze Counter auf größten Wert im Resynchronisierungs-Bereich und betätige Transmitter zweimal.
- Receiver und geklonter Transmitter sind nun synchronisiert und der Counter-Wert des echten Transmitters ist ungültig.
- Der rechtmäßige Besitzer muss über 30000 mal, nämlich genau $2^{15}$ mal, den Transmitter betätigen, bis er sich wieder Zugang verschaffen kann.
- Dies ist das vom Hersteller am meisten befürchtete Szenario

- Experten entwickeln Programm ➔ Angriff auch durch Laien durchführbar
- Falls Programm sich weit verbreitet ➔ Verlässlichkeit von Remote Keyless Entry (RKE)-Systemen stark eingeschränkt.
- Durch Schädigung des Rufs der Hersteller können große wirtschaftliche Kosten entstehen.
- Vermehrte Schwierigkeiten mit RKE-Systemen ➔ Höhere Kosten der Kunden für Wartung und höhere Kosten der Unternehmen für Kundenservice

# Quellen


- Thomas Eisenbarth, Timo Kasper, Amir Moradi, Christof Paar, Mahmoud Salmasizadeh, and Mohammad T. Manzuri Shalmani. On the power of power analysis in the real world: A complete break of the KeeLoq code hopping scheme. In D. Wagner, editor, *Proceedings of CRYPTO 2008*, volume 5157 of *Lecture Notes in Computer Science*, pages 203–220. Springer-Verlag, 2008.