Präsentation Stream Cipher RC4

Günther Eder — Andreas Weichhart

23.05.2011

Stream Ciphers

1. Symmetrische Verschlüsselungsverfahren
2. Unterschiede zwischen Block Chipher und Stream Cipher
3. Entstehung von Stream Cipher
4. Definition von Stream Cipher
5. Aufbau eines Stream Cipher
6. Attacken

RC4

7. Entstehung/Einsatz
8. Aufbau/Funktionsweise
9. KSA und PRGA
10. Attacken
11. Laufzeitverhalten AES/RC-4
12. Quellenangaben

Symmetrische Verschlüsselungsverfahren

- Gleicher Schlüssel für Ver-/Entschlüsselung
- Schlüsselaustausch über sicheren Kommunikationskanal
- Sicherheit abhängig von:
  - kryptografischer Stärke des verwendeten Verfahren
  - Geheimhaltung des Schlüssels
- Einteilung:
  - Block Cipher
  - Stream Cipher

Unterschiede zwischen Block und Stream Ciphers

Block Cipher

- erster zivil nutzbarer Block Cipher "Lucifer", IBM (1971)
- Unterteilung in gleich lange Blöcke (64, 128, 192, 256 Bit)
- mögliche Unterschiede in der Klartext- und Chiffretext-Grösse
- Verwendung, wenn Klartext vor Beginn der Verschlüsselung vorliegt zB
  - Email
  - File, Filesystem

Unterschiede zwischen Block und Stream Ciphers

Stream Cipher

- Gilbert Sandford Vernam 1926 (One-time Pad)
- bitweise Verschlüsselung
- Klartext- und Chiffrentext-Grösse immer gleich
- Verwendung bei Echtzeitübertragung

Entstehung

Geschichtlicher Überblick

- Ende 19. Jahrhundert - Radio
  Jeder konnte *entschlüsseln*
- 1920: bereits weite Verbreitung
  - Enigma
- Rotor-basierte elektromechanische Verschlüsselung
- hohe Speicherkosten - Zeichen für Zeichen

Defenition von Stream Cipher

Ein synchronisierter Stream Cipher besteht aus einem internen Zustand $x \in X$, einer Updatefunktion $L : X \rightarrow X$ und einer Outputfunktion $f : X \rightarrow Z$, wobei Z das Keystream-Alphabet ist.

Der Output $z \in Z$ zur Zeit t wird produziert nach $z^t = f(x^t)$, wobei $x^t = L^t(x)$ und x der Initialisierungszustand ist.

Der Initialisierungszustand x wird durch eine Initialisierungsfunktion des geheimen Keys K und einem Initialisierungsvektor IV produziert.

Defenition von Stream Cipher

Der Outputstream $z^0, z^1, \ldots$ wird als Keystream bezeichnet. Jedes Outputsymbol wird mit einem entsprechendem Plaintextsymbol verknüpft, um das Ciphertextsymbol zu generieren.

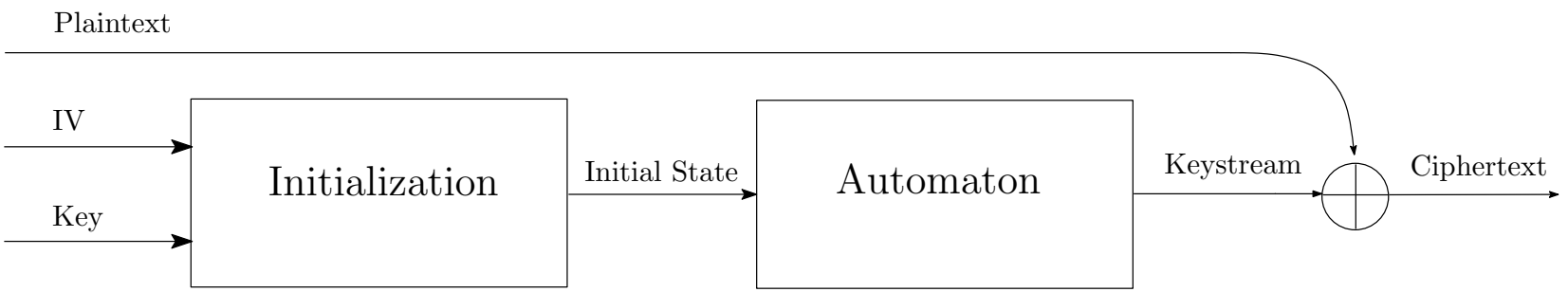


Abbildung: Stream Cipher

Aufbau von Stream Ciphers

Automaton für Stream Cipher

- Feedback Shift Register
- T-Function

Definition Feedback Shift Register

Ein binäres lineares Feedback Shift Register (LFSR) mit der Grösse von n ist ein finiter State Automaton mit einem internen Zustand von n Bits. In jedem Taktzyklus verschiebt die Updatefunktion L den Zustand der Bits um eine Position, wobei das Input-Bit eine lineare Funktion des vorherigen Bits ist.

Genauer gesagt, sei $x = (x_0, ..., x_{(n-1)})$ *der Ausagangszustand, dann ist die Ausgabesequenz* $X = (x_0, x_1, ...)$ *bestimmt durch die Rekursion*

$$x_t = (c_1 x_{t-1} \oplus ... \oplus c_n x_{t-n})$$

für $t \geq n$ *, wobei alle* c_i *fixe Elemente in* $\{0, 1\}$

Feedback Shift Register

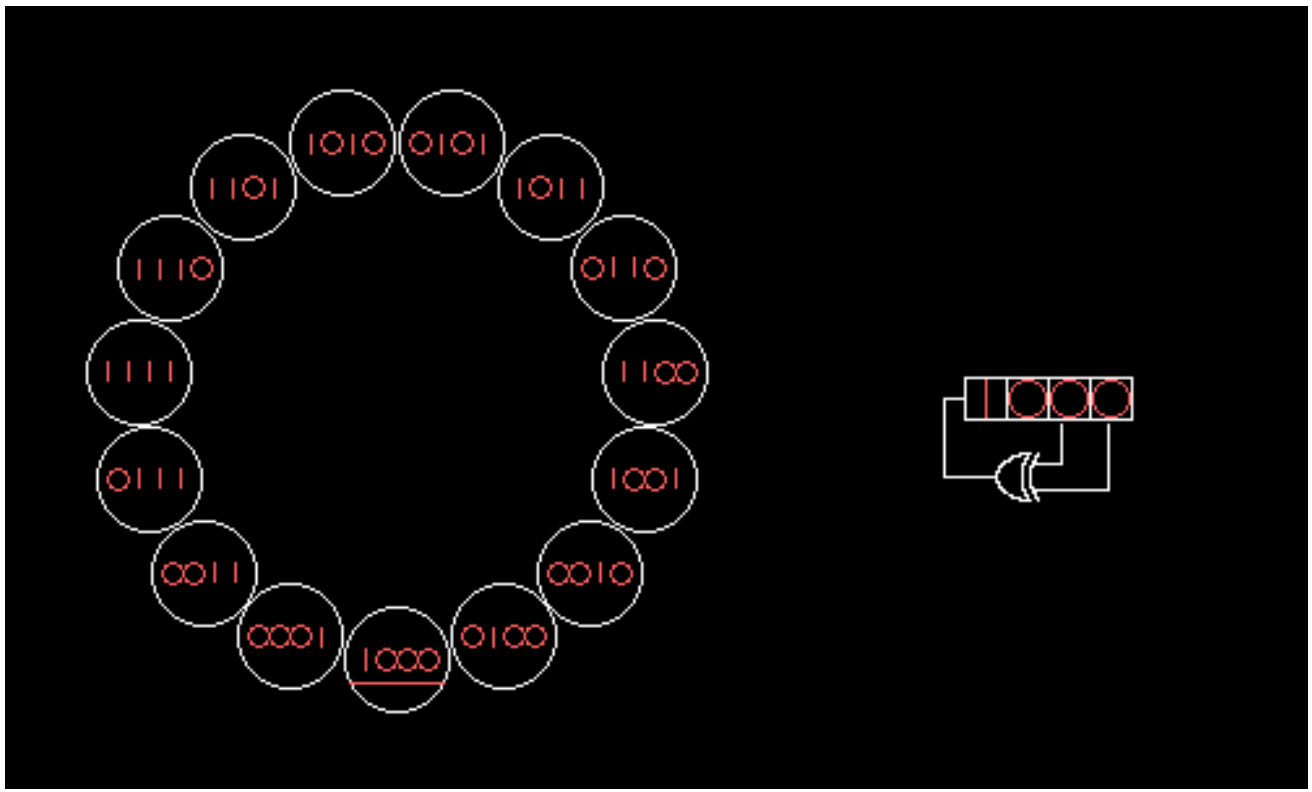


Abbildung: Wie funktioniert ein Feedback Shift Register?

Attacken

Wichtigste Attacken gegen LFSR:

- LFSR-Synthese
- algebraische Attacken
- Korrelationen und lineare Attacken
- Differentielle Attacken
- Tradeoff Attacken

Wichtigste Attacken gegen Stream Ciphers:

- Attacken bei wiederverwendetem Key
- bit-flipping Angriff

Attacken bei wiederverwendetem Key

Vorraussetzungen:

- Nachricht A und B
  $\longrightarrow$ selbe Länge
- selber Verschlüsselungskey K

Der Streamcipher produziert einen String von Bits C(K) mit derselben Länge von A und B. Durch bitweises XOR erhalten wir:

$$E(A) = A \oplus C$$

$$E(B) = B \oplus C$$

Attacken bei wiedererwendetem Key

- Eve fängt E(A) und E(B) ab
- XOR ist kommutative mit der Eigenschaft $X \oplus X = 0$

daraus folgt:

$$E(A) \oplus E(B) \Leftrightarrow (A \oplus C) \oplus (B \oplus C)$$

$$\Leftrightarrow$$

$$A \oplus B \oplus C \oplus C \Leftrightarrow A \oplus B$$

John Tiltman konnte den Lorenz Cipher im 2. Weltkrieg auf diese Weise entschlüsseln

RC-4 Enstehung/Einsatz

Enstehehung

- 1987 von Ron Rivest (RSA Securty)
- geheim bis 1994
- Veröffentlicht durch Cypherpunks mailing list

Einsatz bei oft genutzten Verschlüsselungsprotokollen wie:

- WEP
- WPA / TKIP
- SSL

Implementierung hardware- und softwareseitig effizient

Aufbau/Funktionsweise

- pseudorandom Bit-Stream (Keystream)
- Ver/Entschlüsselung → bitweises XOR

geheimer Internal State → Keystream

- Permutation aller 256 möglichen Bytes (S)
  → Key-Scheduling Algorithm (KSA)
  - 2 8-Bit Index Pointer (i, j)
  - variable Key-Länge zwischen 40 und 256 Bits

Bitstreamproduktion

- Pseudo-random Generator Algorithmus (PRGA)

KSA

- Permutation eines Arrays S durch Vertauschen von Bits
- Key-Länge definiert durch $1 \leq$ Key-Länge ≤ 256 Bits

Listing 1: Pseudo KSA Code

```
for i from 0 to 255
  S[i] := i
endfor
j := 0
for i from 0 to 255
  j := (j + S[i] + key[i mod keylength]) mod 256
  swap values of S[i] and S[j]
endfor
```

Funktionsweise PRGA

Zweck

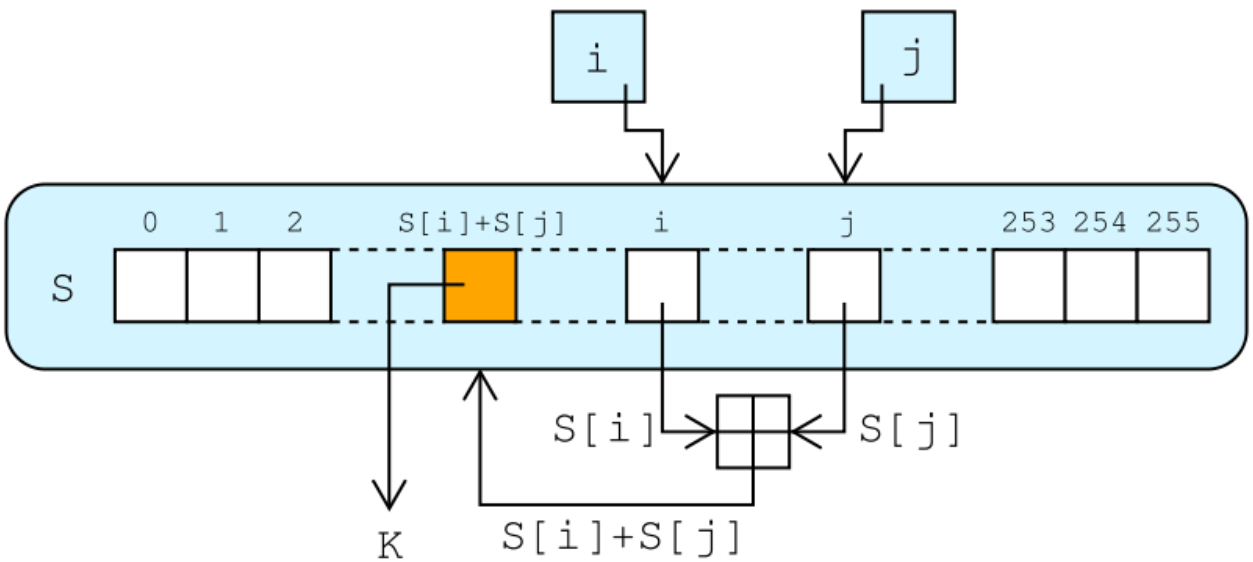
- Modifiziert State
- 1 Byte pro Iteration

Funktionsweise

- Pro Iteration wird i erhöht, addiert die Werte von S[i] und S[j] und schreibt sie wieder in S.
- Vertauscht S[i] und S[j].
- Gibt Element an der Stelle S[i]+S[j] modulo 256.
- In 256 Iterationen wird ein Element min. einmal vertauscht.

PRGA

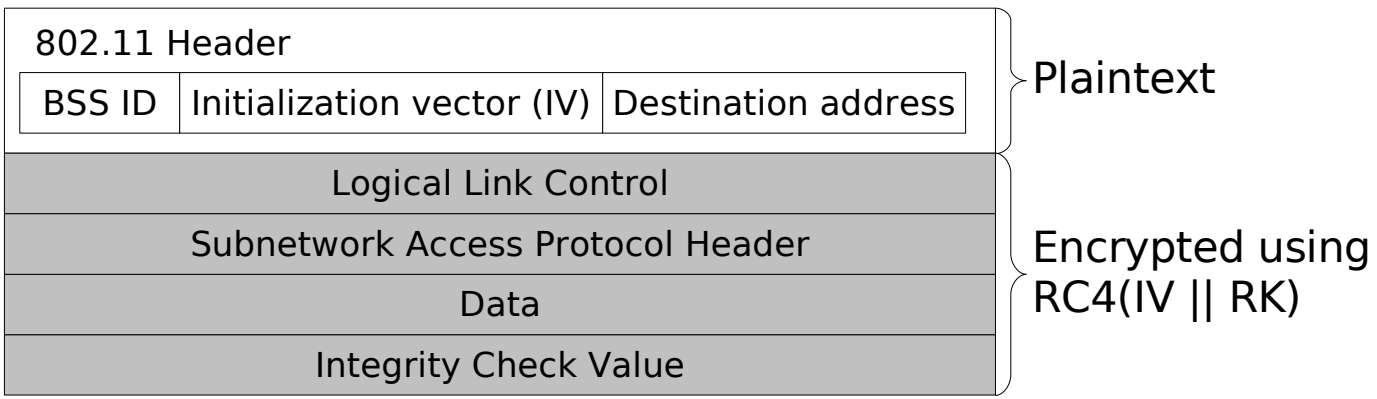
```
i := 0     j := 0
while GeneratingOutput:
  i := (i + 1) mod 256
  j := (j + S[i]) mod 256
  K := S[(S[i] + S[j]) mod 256]
  output K
endwhile
```



Attacken

- 1995: Rekonstruktion durch Permutation (Andrew Roos)
- 2001: *Fluher Mantin and Shamir-Attacke*
- 2005: *Klein's attack*
- 2007: *Breaking 104 bit WEP in less than 60 seconds*
- 2008: *New State Recovery Attack on RC4*

Klein's Attack



Vorraussetzung

- *WEP Header Frame* ist Plaintext
- alte *IVs* wiederverwendbar

Verbesserungen der Attacken

Kleins Attacke

- 1.000.000 session

Breaking 104 bit WEP in less than 60 seconds

- Bei 40.000 Paketen, 50% Erfolg

New State Recovery Attack on RC4

- RC4 - 256
  $\rightarrow$Komplexität $2^{241.7}$

WEP vs. WPA

Sicherheit/Verwendung

- 2007 Deutschland 21% unverschlüsslet 46% WEP
- WPA und RC4
  $\rightarrow$ TKIP

Laufzeitverhalten AES/RC4

lt. Recherchen: RC-4 ca. 2x AES

Vorführung einer Implementierung.

Quellenangaben

- Fischer, Simon, 2008, Analysis of Lightweight Stream Ciphers, PhD. Thesis, Universität Bern, EPFL Report Nr 4040 (2008).
- Mitchell, Chris und Dent, Alexander, 2004, International standards for stream cipher: A progress report, University of London.
- Pommerening, Klaus, 2009, Bitstrom-Verschlüsselung, Fachbereich Mathematik der Johannes-Gutenberg-Universität.
- Robshaw, Matt, 1995, Stream Ciphers, RSA Laboratories Report Nr TR.701, Version 2.0.
- Erik Tews et al., Breaking 104 bit WEP in less than 60 seconds, TU Darmstadt, FB Informatik.

Fragen?

Haben Sie Fragen zum Thema?

Ende

Vielen Dank für Ihre Aufmerksamkeit!