On linear cryptanalysis of stream ciphers

Risto Hakala

December 13, 2007

Outline

- Stream ciphers
- Linear distinguishing attacks on stream ciphers
- Constructing a linear distinguisher for a filter generator
- Linear distinguishers for an LFSR-based filter generator

Stream ciphers

Stream ciphers

- Stream ciphers are symmetric encryption primitives, which are used to ensure confidentiality of messages in digital communication.
- Stream ciphers often have several advantages over block ciphers:
  - more efficient
  - smaller complexity in hardware
  - very little error propagation
- The security of stream ciphers has not been on the same level with the most secure block ciphers.

Synchronous stream ciphers

- A synchronous stream cipher generates a sequence of pseudo-random bits, called the *keystream*, which is combined with the plaintext to produce the ciphertext.
- A synchronous stream cipher can be described as a finite state machine that has an *internal state* and an *update function*.
- In addition, synchronous stream ciphers have a *keystream function* that is used to produce the keystream, and an *output function* that is used to combine the keystream with the plaintext.

Synchronous stream ciphers

- Formal definition of encryption with synchronous stream ciphers:

Internal state:	$\sigma_t = (\sigma_t^{(0)}, \dots, \sigma_t^{(l-1)})$
State update function G:	$\sigma_{t+1} = G(\sigma_t, K)$
Keystream function F:	$z_t = F(\sigma_t, K)$
Output function H:	$c_t = H(p_t, z_t)$

- Additive synchronous stream ciphers use the bitwise exclusive-or to combine the plaintext and the keystream:

$$c_t = p_t \oplus z_t, \quad t \geq 0.$$

Shift registers

- Shift registers are essential building blocks for stream ciphers.
- A shift register consists of a state and and a recurrence relation which defines how the state is updated at each time step $t \geq 0$.
- The state consists of r memory cells, each of which holds one element from the finite field $\mathbf{F}_q$, where $q = p^k$ for prime p and an integer k.
- The state is a vector $S_t = (s_t, \ldots, s_{t+r-1})$, where each $s_{t+i} \in \mathbf{F}_q$, $i = 0, \ldots, r-1$.

Shift registers

- A shift register produces a sequence $(s_t)_{t\geq 0}$, which satisfies the recurrence relation.
- A *linear feedback shift register* (LFSR) has a linear recurrence relation

$$s_{t+r} = a_0 s_t + a_1 s_{t+1} + \cdots + a_{r-1} s_{t+r-1}, \quad t \geq 0,$$

  where $a_0, \ldots, a_{r-1} \in \mathbf{F}_q$ are the feedback coefficients.
- A *nonlinear feedback shift register* (NLFSR) uses a nonlinear recurrence relation instead of a linear one.

Nonlinear filter generators

- A nonlinear filter generator consists of a shift register and a nonlinear filter (NLF) function.
- The state σ_t of the nonlinear filter generator is the state S_t of the shift register.
- The state update function G of the generator is the state update function of the shift register.
- The keystream function F is the NLF.

Linear distinguishing attacks on stream ciphers

Statistical distinguishing attacks

- The security of a stream cipher is largely dependent on how random the keystream $(z_t)_{t \geq 0}$ can be made to appear.
- Statistical distinguishing attacks aim at detecting statistical bias in the keystream using a distinguisher.
- A statistical distinguisher is a statistical hypothesis test which decides whether a sample sequence $(x_t)_{t \geq 0}$ is from the cipher or not.
- A distinguishing attack with a very high complexity indicates a weakness in the primitive.

Linear distinguishing attacks on stream ciphers

- Linear distinguishing attacks are distinguishing attacks, which make use of linear cryptanalytic techniques.
- A linear distinguisher operates in two phases: the *transformation phase* and the *statistical inference phase*.
- It is assumed that the input sequence $(x_t)_{t \geq 0}$ for the distinguisher is a sequence over the binary vector space $\mathbf{F}_2^n$.

The transformation phase

- In the transformation phase, a $\mathbf{F}_2$-linear transformation is applied to the input sequence $(x_t)_{t \geq 0}$ in order to obtain a new sequence $(\hat{x}_t)_{t \geq 0}$:

$$\hat{x}_t = \bigoplus_{j \in J} v_j \cdot x_{t+j}, \quad t \geq 0,$$

  where $v_j, x_{t+j} \in \mathbf{F}_2^n$ and $\hat{x}_t \in \mathbf{F}_2$, for all $j \in J,\ t \geq 0$.
- The set J is the index set that defines which input sequence vectors are included in the transformation.

The statistical inference phase

- In the statistical inference phase, the distribution of the sequence $(\hat{x}_t)_{t\geq 0}$ is examined in order to decide whether the input sequence $(x_t)_{t\geq 0}$ is from the stream cipher or not.
- The decision is made based on a test statistic, which is usually a function of the ratio of zeros and ones in $(\hat{x}_t)_{t\geq 0}$.
- For a random input sequence, this ratio should be close to $\frac{1}{2}$.
- The goal is usually to find such a linear transformation that the ratio of zeros and ones in $(\hat{x}_t)_{t\geq 0}$ differs from $\frac{1}{2}$ as much as possible if the input sequence has been generated by the stream cipher.

Required sample size for the distinguisher

- To make the decision with high confidence level, the sample size has to be large enough.
- The required sample size depends on the chosen test statistic.
- The required sample size with the log-likelihood ratio statistic can be shown to be $\mathcal{O}(\epsilon^{-2})$, where $\Pr[\hat{x}_t = 0] = \frac{1}{2} + \epsilon$, for all $t \geq 0$.

Problems in linear distinguishing attacks

- How to determine the bias ϵ of $(\hat{x}_t)_{t\geq 0}$ if the input sequence $(x_t)_{t\geq 0}$ is from the cipher?
- How to choose the transformation

$$\hat{x}_t = \bigoplus_{j\in J} v_j \cdot x_{t+j}, \quad t \geq 0,$$

such that the bias ϵ of $(\hat{x}_t)_{t\geq 0}$ is large whenever the input sequence is from the cipher.

Constructing a linear distinguisher for a filter generator

Piling-Up Lemma

- Suppose that $X_0, \ldots, X_{N-1}$ are independent binary random variables such that $\Pr[X_i = 0] = \frac{1}{2} + \epsilon_i$, $i = 0, \ldots, N-1$.
- The Piling-Up Lemma states that

$$\Pr[X_0 \oplus \cdots \oplus X_{N-1} = 0] = \frac{1}{2} + 2^{N-1} \prod_{i=0}^{N-1} \epsilon_i .$$

Linear approximations

- A linear approximation of $f\colon (\mathbf{F}_2^n)^m \to \mathbf{F}_2^n$ is a relation of the form
$$v \cdot f(x^{(0)}, \ldots, x^{(m-1)}) = \bigoplus_{i=0}^{m-1} u^{(i)} \cdot x^{(i)},$$
where the $u^{(0)}, \ldots, u^{(m-1)} \in \mathbf{F}_2^n$ are called the *linear input masks* and $v \in \mathbf{F}_2^n$ is called the *linear output mask*.
- We use $ua \in \mathbf{F}_2^n$ to denote the linear mask which satisfies the equality
$$ua \cdot x = u \cdot ax, \quad \text{for all } x \in \mathbf{F}_2^n,$$
where the product ax is taken in $\mathbf{F}_{2^n}$.

Linear approximations

- The efficiency of a linear approximation of f is measured by its *correlation*

$$c_f(v,u) = 2\Pr\left[v \cdot f(x^{(0)},\ldots,x^{(m-1)}) = \bigoplus_{i=0}^{m-1} u^{(i)} \cdot x^{(i)}\right] - 1,$$

  where the probability is taken over uniform $x^{(0)},\ldots,x^{(m-1)} \in \mathbf{F}_2^n$.
- The *bias* of a linear approximation is defined to be $\epsilon_f(v,u) = c_f(v,u)/2$.

Linear chains

- Let $f = f_{N-1} \circ \cdots \circ f_0$ be an iterated mapping such that $f_i \colon \mathbf{F}_2^{n_i} \to \mathbf{F}_2^{n_{i+1}}$, $i = 0, \ldots, N-1$.
- Denote by $c_{f_i}(u_{i+1}, u_i)$ the correlation of a linear approximation of f_i with the output mask $u_{i+1} \in \mathbf{F}_2^{n_{i+1}}$ and the input mask $u_i \in \mathbf{F}_2^{n_i}$.
- A linear chain is a chain of approximations over the invidual components of f.
- The correlation of a linear chain is defined to be

$$c_f = \prod_{i=0}^{N-1} c_{f_i}(u_{i+1}, u_i).$$

Linear chains

- It can be shown that the the correlation of a linear approximation of f is

$$c_f(v, u) = \sum_{u_1,\ldots,u_{N-1}} \prod_{i=0}^{N-1} c_{f_i}(u_{i+1}, u_i),$$

where $v = u_N$ and $u = u_0$.

- If the sum is dominated by a single linear chain with the masks $u_0, \ldots, u_N$, one can estimate that

$$c_f(u_N, u_0) \approx \prod_{i=0}^{N-1} c_{f_i}(u_{i+1}, u_i).$$

Linear distinguishers for filter generators

- A linear distinguisher for a filter generator is constructed as follows:
  1. Several linear approximations of the nonlinear filter F are formed. These approximations involve keystream variables $(z_t)_{t\geq 0}$ and state variables S_t.
  2. Using a time-invariant relation, the state variables S_t can be canceled out so that we get an approximation which involves keystream variables only:

$$\bigoplus_{j\in J} v_j \cdot z_{t+j} = 0, \quad t \geq 0,$$

Choosing the linear transformation

- The linear transformation in the distinguisher is chosen from an approximation of the keystream $(z_t)_{t\geq 0}$ variables.

$$\bigoplus_{j\in J} v_j \cdot z_{t+j} = 0, \quad t \geq 0,$$

where $v_j \in \mathbf{F}_2^n$ is the linear mask used in the approximation of the keystream word $z_{t+j} \in \mathbf{F}_2^n$.

- The linear approximation of the nonlinear filter F is usually formed by forming a linear chain of approximations over the components of F.

Linear distinguishers for an LFSR-based filter generator

LFSR-based filter generators

- We suppose that the output keystream $(z_t)_{t\geq 0}$ does not depend on the key K, i.e., $z_t = F(S_t)$ and $s_{t+r} = G(S_t)$, for all $t \geq 0$.
- We also suppose that the elements in the state S_t of the LFSR are from $\mathbf{F}_{2^n}$, and that they are statistically indpendent for all $t \geq 0$.
- The recurrence relation of the LFSR can be written as

$$a_0 s_t \oplus a_1 s_{t+1} \oplus \cdots \oplus a_{r-1} s_{t+r-1} \oplus a_r s_{t+r} = 0, \quad t \geq 0,$$

  where $a_0, \ldots, a_{r-1} \in \mathbf{F}_{2^n}$, $a_r = 1$, and the product $a_i s_{t+i}$ is taken in $\mathbf{F}_{2^n}$, for $i = 0, \ldots, r$.

LFSR-based filter generators

- Let $0 \leq j \leq r$ and denote by

$$v_j \cdot z_{t+j} = \bigoplus_{i=0}^{r-1} u^{(i)} a_j \cdot s_{t+j+i} \tag{1}$$

  a linear approximation of $z_{t+j} = F(S_{t+j})$ with the output mask $v_j \in \mathbf{F}_2^n$ and the input masks $u^{(0)} a_j, \ldots, u^{(r-1)} a_j \in \mathbf{F}_2^n$.
- Summing up the approximations (1) for $j = 0, \ldots, r$ gives

$$\bigoplus_{j=0}^{r} v_t \cdot z_{t+j} = \bigoplus_{j=0}^{r} \bigoplus_{i=0}^{r-1} u^{(i)} a_j \cdot s_{t+j+i}.$$

LFSR-based filter generators

- Since $u^{(i)}a_j \cdot x = u^{(i)} \cdot a_j x$, for all $x \in \mathbf{F}_{2^n}$, it follows that

$$\bigoplus_{j=0}^{r} v_j \cdot z_{t+j} = \bigoplus_{i=0}^{r-1} u^{(i)} \cdot \left[\bigoplus_{j=0}^{r} a_j s_{t+j+i} \right] = 0.$$

- The last equivalence holds, since $\bigoplus_{j=0}^{r} a_j s_{t+j+i} = 0$ is the recurrence relation

$$a_0 s_t \oplus a_1 s_{t+1} \oplus \cdots \oplus a_{r-1} s_{t+r-1} \oplus a_r s_{t+r} = 0, \quad t \geq 0,$$

at time $t := t + i$.

LFSR-based filter generators

- Denote the correlation of the approximation of the NLF F by $c_F(v_j, u_j)$, where $u_j = (u^{(0)}a_j, \ldots, u^{(r-1)}a_j)$.
- The final approximation is formed by taking the xor of the binary random variables $v_j \cdot z_{t+j}$, $j = 0, \ldots, r$.
- The correlation c of the final approximation can be estimated with the Piling-Up Lemma as

$$c \approx \prod_{j=0}^{r} c_F(v_j, u_j),$$

which is the same value for all $t \geq 0$.

LFSR-based filter generators

- To find a good distinguisher, we need to find good approximations (v_j, u_j) for the NLF.
- Good approximations are often searched by forming a linear chain with high bias over the NLF.
- The reason for this is that it is very difficult to examine the NLF as a single function.
- The correlation of the approximation is estimated to be the correlation of the linear chain.

Discussion

- To construct a linear distinguisher for a NLFSR-based filter generator, one needs to form a linear approximation for the nonlinear recurrence relation of the NLFSR also.
- If the keystream $(z_t)_{t \geq 0}$ is dependent on the secret key K such that $z_t = F(S_t, K)$ and $S_{t+1} = G(S_t, K)$, the correlation of the linear approximation

$$\bigoplus_{j \in J} v_j \cdot z_{t+j} = 0, \quad t \geq 0,$$

  depends also on K.
- This makes it possible to gain information from K.

Discussion

- It is possible to improve a distinguishing attack by using a multidimensional transformation in the distinguisher:

$$\hat{X}_t = \begin{bmatrix} \hat{x}_{0,t} \\ \vdots \\ \hat{x}_{s-1,t} \end{bmatrix} = \begin{bmatrix} \bigoplus_{j=0}^{r} v_{0,j} \cdot x_{t+j} \\ \vdots \\ \bigoplus_{j=0}^{r} v_{s-1,j} \cdot x_{t+j} \end{bmatrix}.$$

- In this case, the distribution of the sequence $(\hat{X}_t)_{t \geq 0}$ is compared with the uniform distribution in order to decide whether the input sequence $(x_t)_{t \geq 0}$ is from the cipher or not.