Shahid Beheshti University

رمزنگـاری

هادی سلیمانی

پژوهشکده فضای مجازی دانشگاه شهید بهشتی

مجوز استفاده و نشر

- اجازه‌ی ایجاد نسخه‌های دیجیتالی جدید براساس بخشی یا تمام مطالب این اسلاید بدون پرداخت هزینه اعطا می‌شود، مشروط بر این‌که:
  - فقط به‌منظور و در راستای استفاده‌ی آموزشی (شخصی و یا کلاسی) ساخته شده باشند و برای کسب هرگونه سود و یا مزیت تجاری استفاده نشوند.
  - نسخه‌های جدید حاوی ارجاع مستقیم به نام تهیه‌کننده اسلاید (هادی سلیمانی) و محل کار وی (پژوهشکده فضای مجازی دانشگاه شهید بهشتی) باشند.
- مجموعه‌ی حاضر براساس نظرات ارزشمند دانشجویان (سابق) دانشگاه شهید بهشتی و همکاران محترم تهیه شده است که از تمام آن‌ها قدردانی می‌شود؛

(به‌خصوص خانم‌ها **سارا زارعی و فاطمه عزیزی** نقش مهمی را در تهیه نسخه‌ی نهایی بر عهده داشته‌اند. خانم مهندس زارعی علاوه بر کمک در آماده‌سازی نسخه‌ی فعلی اسلایدها، در تصحیح اشتباهات نسخه‌ی قبلی و همچنین تکمیل و بازتعریف محتوای درس‌ها بسیار تاثیرگذار بوده‌اند).

- برای مشاهده‌ی اسلایدها و ویدئوهای تدریس این درس به آدرس زیر مراجعه فرمایید:

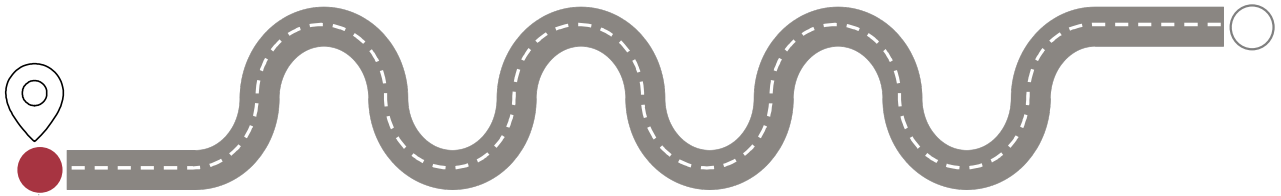
http://facultymembers.sbu.ac.ir/h_soleimany/cryptography-course/

درس چهارم

رمزهای جریانی

فهرست عناوین درس

رمزهای جریانی



مقـدمه

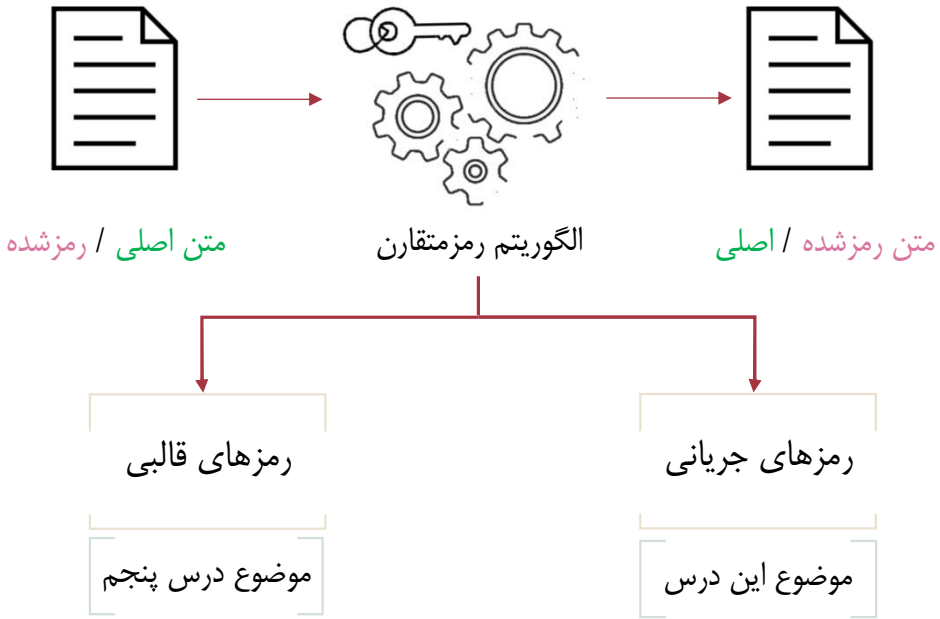
آشنایی با مفاهیم رمز متقارن و نامتقارن

- رمزنگاری متقارن:

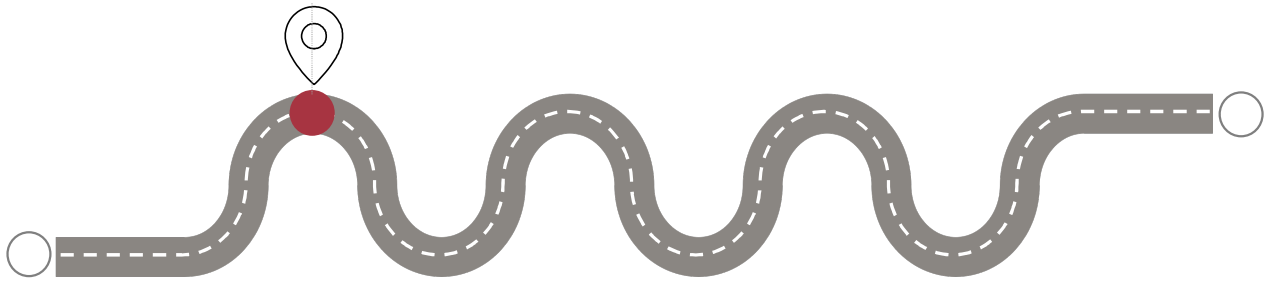
- رمزنگاری نامتقارن:

رمزنگاری متقارن

- مزیت رمزنگاری متقارن نسبت به رمزنگاری نامتقارن: سرعت بیشتر.
- چالش رمزنگاری متقارن: مسئله‌ی تبادل کلید مشترک!
- انواع رمزنگاری متقارن:



ساختار رمزهای جریانی

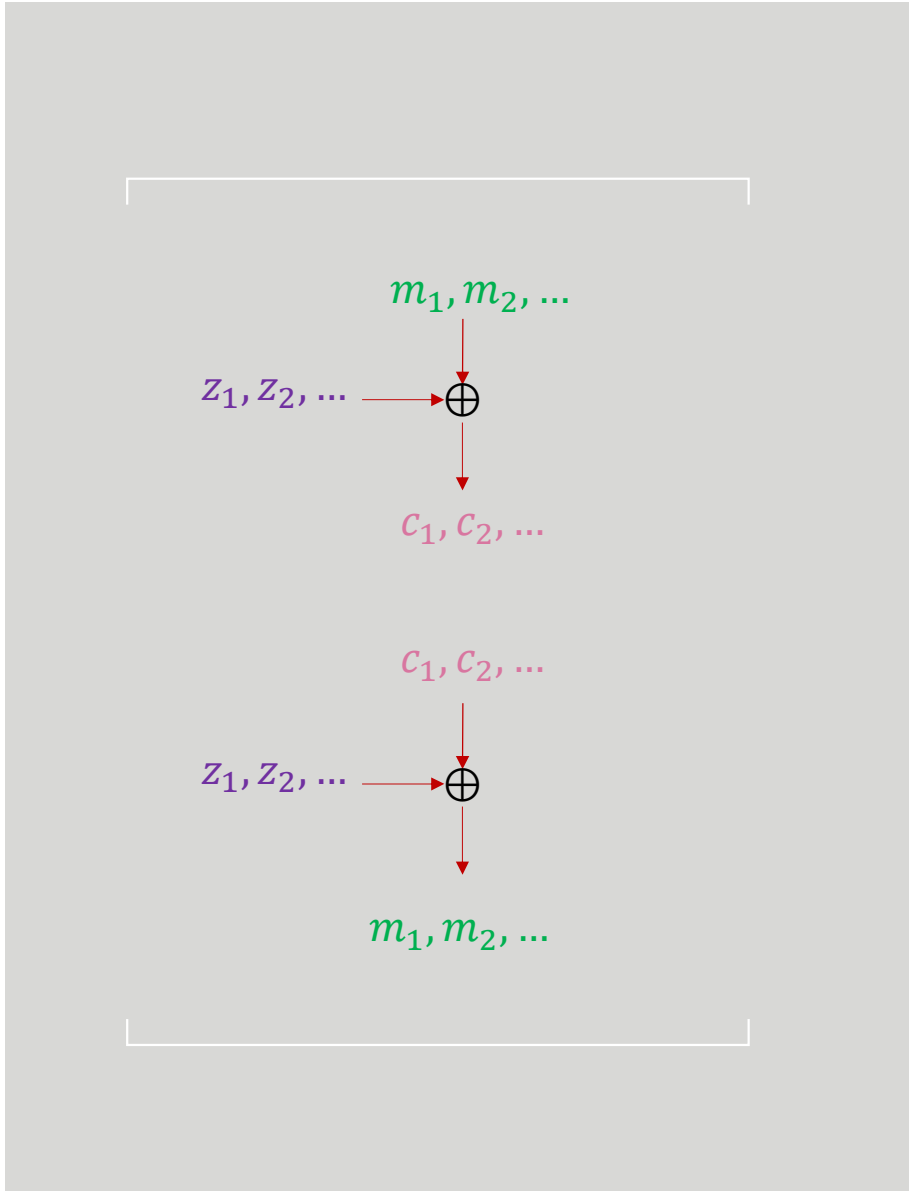


مقـدمه

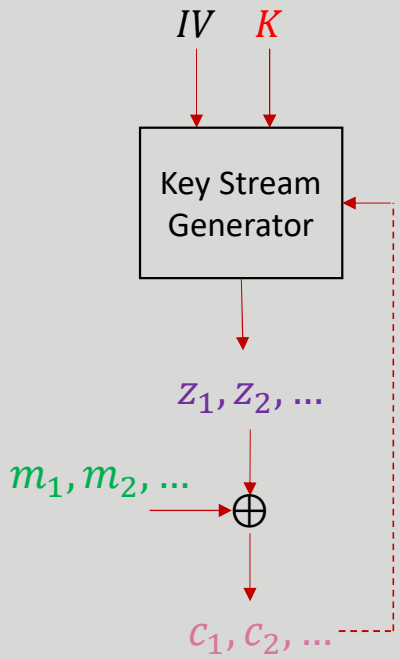
رمزهای جریانی

(Stream Ciphers)

- دنباله‌ای شبه تصادفی تولید می‌شود که معمولا به عنوان دنباله‌ی کلید اجرایی (Key Stream) خوانده می‌شود.
- این دنباله با استفاده از کلید مخفی و یک مقدار اولیه‌ی غیرمخفی (IV) تولید می‌شود.
- دنباله‌ی کلید اجرایی با متن اصلی ترکیب می‌شود (معمولا xor بیتی) و متن رمزشده را تولید می‌کند.
- متن رمزشده و IV بر روی کانال ناامن ارسال می‌شوند.
- گیرنده نیز دنباله‌ی کلید اجرایی را به صورت مشابه، با استفاده از کلید مخفی و IV تولید کرده و متن اصلی را بازیابی می‌کند.



تولید دنباله‌ی کلید اجرایی

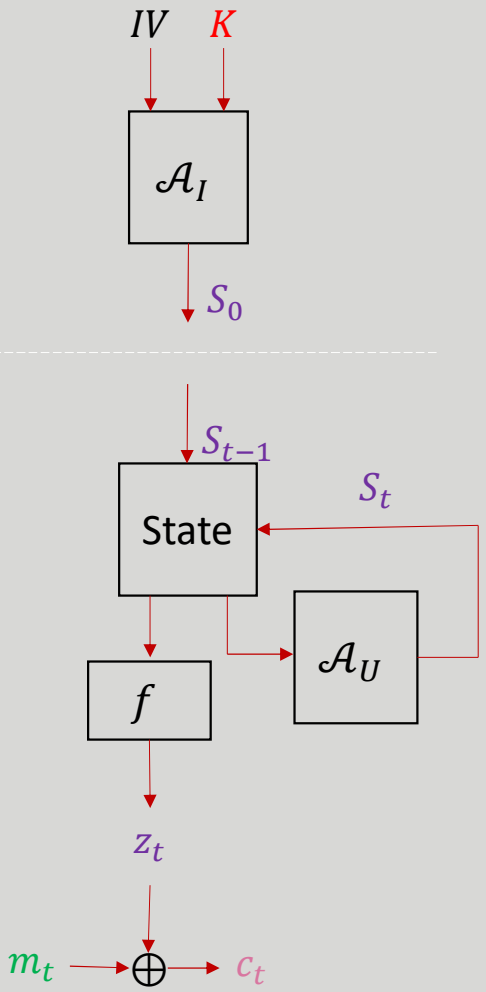


- دنباله‌ی کلید اجرایی همیشه با استفاده از کلید و یک مقدار اولیه‌ی غیرمخفی (IV) تولید می‌شود.
- امنیت رمزهای جریانی، امنیت محاسباتی است چراکه با امتحان کردن تمام حالات ممکن می‌توان کلید را بازیابی کرد.
- براساس این‌که فرآیند تولید دنباله‌ی کلید اجرایی به متن رمزشده نیز بستگی دارد یا خیر، رمزهای جریانی به دو دسته تقسیم می‌شوند:

1. رمزهای جریانی همزمان (Synchronous Stream Cipher)
2. رمزهای جریانی خود همزمان (Self-synchronizing Stream Cipher)

ساختار رمزهای جریانی هم‌زمان

(Synchronous Stream Ciphers)

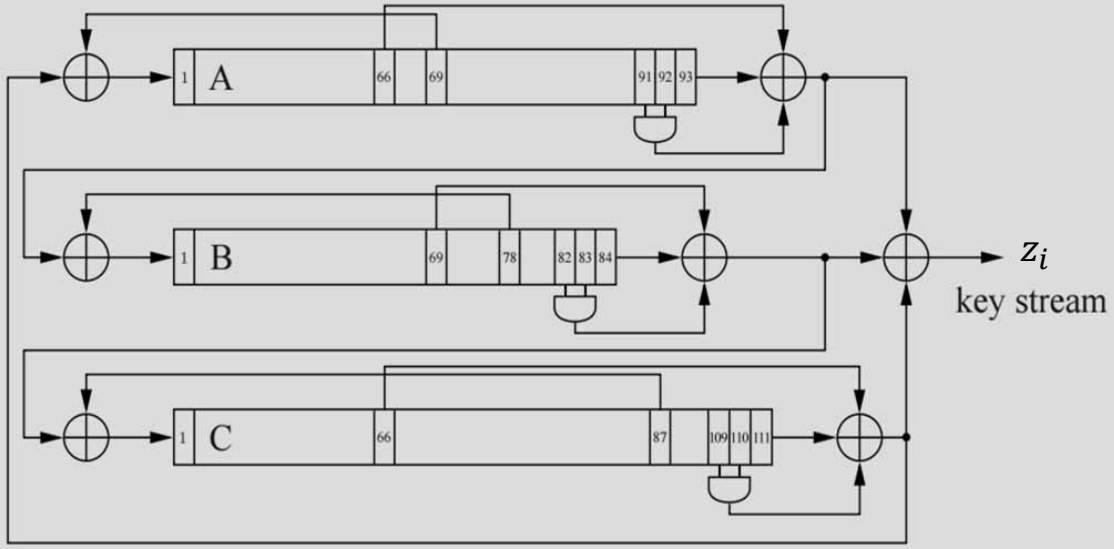


- حالت اولیه (S_0) با استفاده کلید و مقدار اولیه غیرمخفی (IV) تولید می‌شود.
- در هر مرحله (مرحله‌ی tام، پس از t کلاک)، حالت قبلی (S_{t-1}) با استفاده از الگوریتم به‌روزرسانی به حالت بعدی (S_t) تبدیل می‌شود.
- همچنین یک یا چند بیت (z_t) به عنوان دنباله‌ی کلید اجرایی تولید می‌شود.
- دنباله‌ی کلید اجرایی برای رمز کردن متن استفاده می‌شود.

ساختار رمزهای جریانی هم‌زمان

مثال : Trivium؛ الگوریتمی برای تولید حالت اولیه

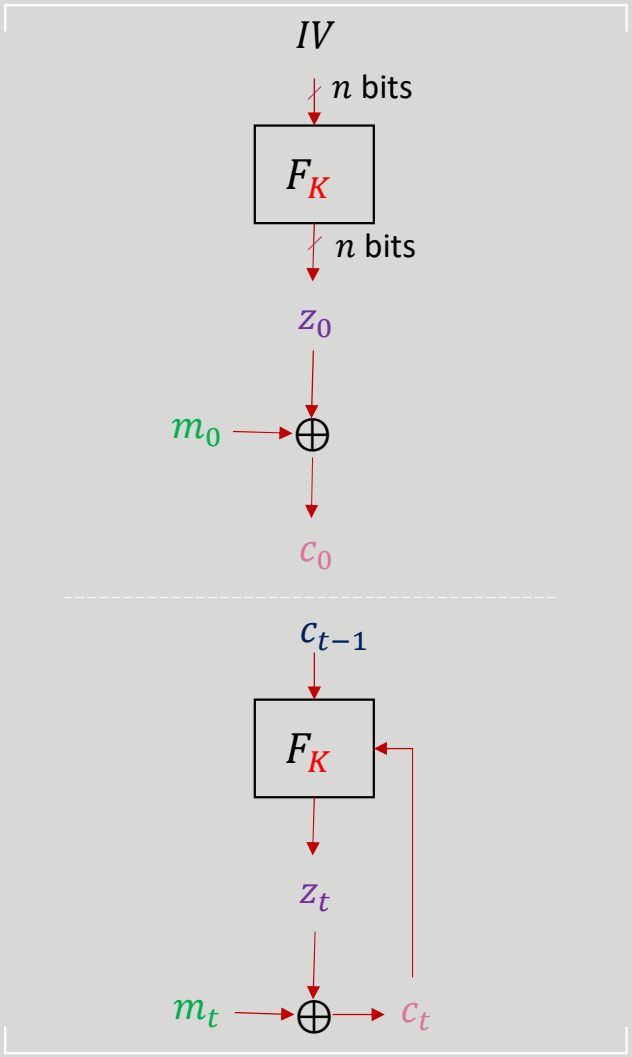
- ۸۰ بیت کلید در قسمت سمت چپ رجیستر A و ۸۰ بیت IV در قسمت سمت چپ رجیستر B قرار می‌گیرند.
- پس از هر کلاک (یک دور اجرای الگوریتم) یک بیت در خروجی تولید می‌شود.
- بیتی که پس از ۱۱۵۲ بار اجرای الگوریتم ایجاد می‌شود را به عنوان حالت اولیه‌ی رمز جریانی استفاده می‌کنند.



ویژگی‌های رمزهای جریانی هم‌زمان

1. باید هم‌زمانی کامل وجود داشته باشد.
   - اضافه یا کم کردن حتی یک بیت به متن رمزشده (توسط مهاجم یا به صورت غیرعمد)، می‌تواند منجر به اخلال کامل عمل رمزگشایی شود.
2. انتشار خطا ندارد؛ به این معنی که تغییر یک بیت از متن رمزشده، تنها منجر به خطا در رمزگشایی همان بیت خواهد شد و به بیت‌های دیگر سرایت نمی‌کند.
   - از منظری مناسب است: بازیابی متن اصلی با کم‌ترین خطا انجام می‌شود.
   - از طرفی نشان‌دهنده‌ی این است که احراز اصالت پیام در اینجا بسیار مهم است: ممکن است مهاجم بتواند پیام را به صورت معنی‌داری تغییر دهد!

ساختار رمزهای جریانی خود همزمان



- برخلاف آن‌چه از نام‌گذاری استنباط می‌شود، این دسته از اولیه‌های رمزنگاری بیشتر شبیه رمزنگاری قالبی هستند.
- رمزهای جریانی ناهمزمان عموما از یک جایگشت کلیددار n بیتی (F_K) استفاده می‌کنند.
- با استفاده از IV، یک قالب n بیتی (z_0) از دنباله‌ی کلید اجرایی تولید می‌شود.
- یک قالب n بیتی متن اصلی (m_0) با استفاده z_0 رمز می‌شود.
- در مرحله tام، $z_t = F_K(c_{t-1})$ محاسبه شده و برای رمزگذاری m_t استفاده می‌شود.

ویژگی‌های رمزهای جریانی خود هم‌زمان

1. به هم‌زمان بودن نیاز ندارند.
   - اضافه یا کم شدن بیت‌های رمزشده، تنها منجر به اخلال در بخشی از عملیات رمزگشایی می‌شود.
2. انتشار خطای محدود دارد؛ یعنی تغییر یک بیت از متن رمزشده، منجر به خطا در رمزگشایی تعدادی از بیت‌های بعدی نیز خواهد شد.
   - جعل معنی‌دار پیام توسط مهاجم، در مقایسه با رمزهای جریانی هم‌زمان سخت‌تر است.

دلیل استفاده از IV

- چرا در رمزهای جریانی هم‌زمان نیاز به استفاده از IV داریم، درحالی‌که یک مقدار غیرمخفی است؟
- اگر دنباله‌ی اجرایی صرفا تابعی از کلید مخفی باشد، بدین معنی است که برای رمز کردن متن‌های مختلف از کلید اجرایی ثابت استفاده می‌شود!

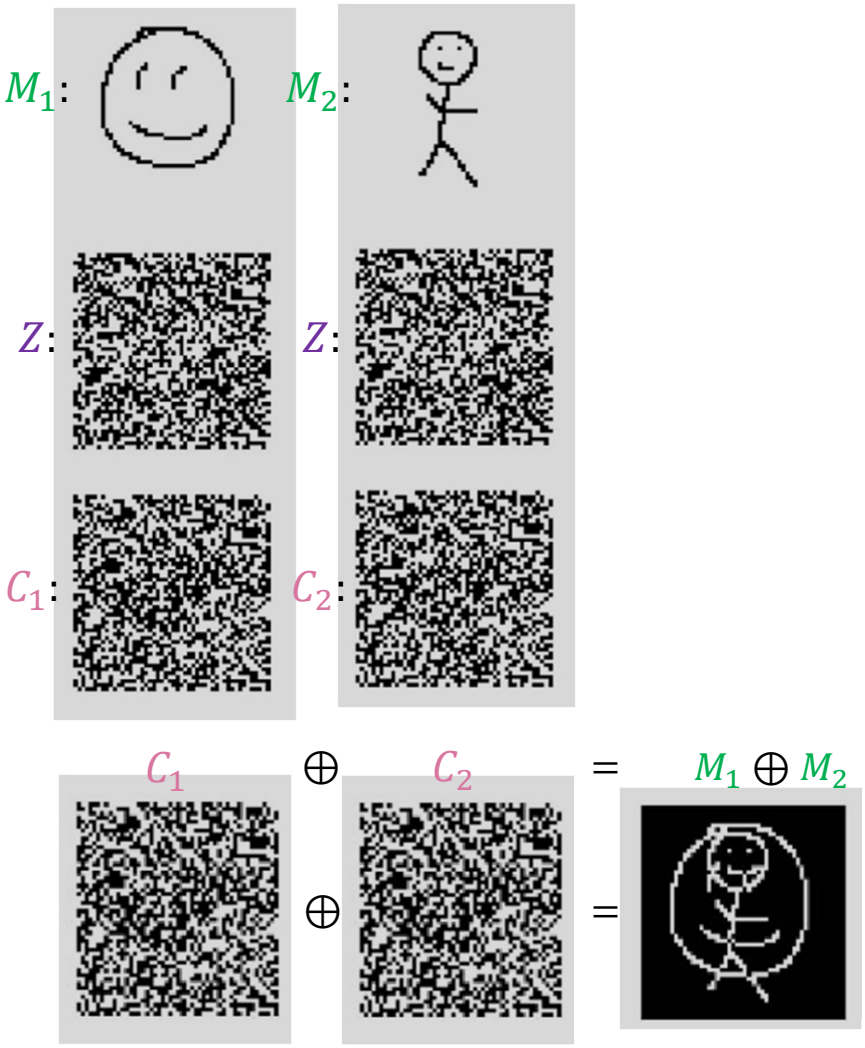
$$\begin{matrix} C_1 = M_1 \oplus Z \\ C_2 = M_2 \oplus Z \end{matrix} \Rightarrow C_1 \oplus C_2 = (M_1 \oplus Z) \oplus (M_2 \oplus Z) = M_1 \oplus M_2$$

تفاضل بین متون رمزشده با تفاضل بین متون اصلی معادل، ارتباط مستقیم دارد.

دلیل استفاده از IV

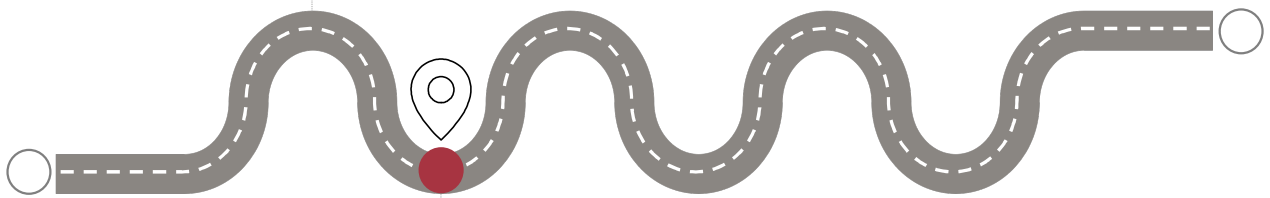
... ادامه

- استفاده‌ی نامناسب از RC4 (یک رمز جریانی پرکاربرد) در Office 2003 سبب ضعف امنیتی در این نرم‌افزار شد.
- مقدار اولیه‌ی S_0 تابعی از حالت اولیه و پسورد بود که با تغییر نسخه‌های مختلف فایل تغییر نمی‌کرد.



Example's source: Michal Dobes

ساختار رمزهای جریانی



مقـدمه

بررسی اولیه‌ی امنیت رمزهای جریانی

اهداف امنیتی در رمزهای جریانی

- نتوان از متن‌های رمزشده، اطلاعاتی در خصوص متن‌های اصلی به دست آورد.
- با دانستن بخشی از دنباله‌ی کلید اجرایی، نتوان درباره سایر بیت‌های دنباله‌ی کلید اجرایی اطلاعاتی به دست آورد.
  - حدس زدن ℓ بیت از دنباله کلید اجرایی با احتمال بیشتر از $2^{-\ell}$ ممکن نباشد.
- دنباله‌ی کلید اجرایی را نتوان از یک دنباله‌ی کاملا تصادفی تمایز داد.
  - هیچ‌گونه مشخصه‌ی آماری غیرتصادفی نداشته باشد.
- با دانستن بخشی از دنباله‌ی کلید اجرایی، نتوان اطلاعاتی درباره کلید مخفی به دست آورد.
  - بازیابی کلید با روشی سریع‌تر از جست‌وجوی جامع امکان‌پذیر نباشد.
  - سناریوی ممکن برای حمله در رمزهای جریانی هم‌زمان: متن اصلی معلوم
  - سناریوهای ممکن برای حمله در رمزهای جریانی خودهم‌زمان: متن اصلی معلوم، متن اصلی منتخب و متن رمزشده منتخب.
- در ادامه، تمرکز ما بر روی بررسی معیارهای امنیتی رمزهای جریانی هم‌زمان خواهد بود که کاربرد بیشتری دارند.

تاثیر طول دوره‌ی تناوب بر امنیت رمز جریانی

- هر مولد کلید اجرایی قطعا دارای یک دوره‌ی تناوب است.
- کوچک بودن دوره‌ی تناوب معادل استفاده از کلیدهای (کوچک‌تر و) برابر است.
- در این حالت، با دانستن بخشی از متن اصلی (یا به صورت معادل بخشی از کلید اجرایی)، اطلاعاتی در خصوص سایر بخش‌های متن اصلی نیز به دست می‌آید.

$$C_1 = M_1 \oplus Z_1$$

$$C_2 = M_2 \oplus Z_2$$

If $Z_1 = Z_2$

$$\Rightarrow \quad M_2 = M_1 \oplus C_1 \oplus Z_1$$

تصادفی بودن دنباله‌ی تولیدی

- متن اصلی معنی‌دار است.
  - به عبارت دیگر احتمال رخ دادن حالات ممکن برای متن اصلی یکسان نیست.
  - توزیع متن اصلی یکنواخت نیست.
- درصورتی‌که دنباله‌ی کلید اجرایی تصادفی باشد، متن رمزشده نیز تصادفی خواهد شد.
  - تصادفی بودن کلید اجرایی یعنی:

$$\Pr[z_i = 0] = \Pr[z_i = 1] = 1/2$$

$$\Pr[c_i = 0] = \Pr[m_i \oplus z_i = 0]$$

$$= \Pr[m_i, z_i = 0] + \Pr[m_i, z_i = 1]$$

$$= \Pr[m_i = 0] \times \Pr[z_i = 0] + \Pr[m_i = 1] \times \Pr[z_i = 1]$$

$$= \Pr[m_i = 0] \times \frac{1}{2} + \Pr[m_i = 1] \times \frac{1}{2}$$

$$= \frac{1}{2}(\Pr[m_i = 0] + \Pr[m_i = 1]) = \frac{1}{2}$$

هدف: بررسی معیارهایی برای تصادفی بودن (تصادفی به نظر رسیدن) دنباله‌ی کلید اجرایی

معیارهای Golomb برای بررسی تصادفی بودن یک دنباله

معیار اول

برای یک دوره‌ی تناوب از یک دنباله، اختلاف تعداد بیت‌های 1 و تعداد بیت‌های 0 حداقل باشد.

- در صورت زوج بودن طول دنباله: تعداد بیت‌های 1 و تعداد بیت‌های 0 برابر باشند.

0101101100

- در صورت فرد بودن طول دنباله: اختلاف تعداد بیت‌های 1 و تعداد بیت‌های 0، یک باشد.

00101101011

- مفهوم: احتمال رخ دادن 1 و یا 0 برابر است.

معیارهای Golomb برای بررسی تصادفی بودن یک دنباله

معیار دوم

مقدمه:

- تعریف run: مجموعه‌ای از بیت‌های یکسان که با بیت قبل و بعد خود متفاوت باشند.

0001101100

- به یک run با بیت‌های 1، block و با بیت‌های 0، gap می‌گویند.

معیار دوم:

- برای یک دوره تناوب از یک دنباله، نصف runها طول 1 داشته باشند.
- تعداد $\frac{1}{4}$ از runها طول 2 داشته باشند.
- به طور کلی $\frac{1}{2^i}$ از runها، طول i داشته باشند (تا زمانی‌که محاسبه تعداد runها براساس این فرمول ممکن باشد).
- مفهوم: بیت تولیدی در مرحله tام هیچ ارتباطی به بیت‌های قبلی نداشته و کاملا مستقل باشد.

تابع خودهمبستگی

(Autocorrelation Function)

- به منظور فهم راحت‌تر، مفهوم خودهمبستگی را ابتدا از طریق یک مثال شرح می‌دهیم و در ادامه تعریف دقیق‌تر را نیز ارائه خواهیم داد.
- دنباله را به اندازه k بیت شیفت می‌دهیم و آن را با دنباله‌ی اصلی مقایسه می‌کنیم:

$$C(k) = \frac{\text{تعداد دفعات نابرابری} - \text{تعداد دفعات برابری}}{\text{طول دوره‌ی تناوب}}$$

$k = 2$:

$$\begin{array}{r} \mathbf{1011001} \\ 10110011\mathbf{10011}011001 \end{array} \quad \Rightarrow \quad C(k) = \frac{1-6}{7} = \frac{-5}{7}$$

Example's source: Michal Dobes

تابع خودهمبستگی

... ادامه

- **تعریف** (ریاضیاتی دقیق): تابع خودهمبستگیِ یک دنباله‌ی متناوب با طول N، به صورت زیر تعریف می‌شود:

$$C(k) = \frac{1}{N}\sum_{i=0}^{N-1}(2s_i - 1)(2s_{i+k} - 1)\,, k \in N$$

- اگر بیت‌های s_i و s_{i+k} برابر باشند (نباشند)، مقدار $(2s_i - 1)(2s_{i+k} - 1)$ برابر 1 (-1) می‌شود.
- مفهوم خودهمبستگی: معیاری برای فهم تعداد دفعاتی که بیت‌های دنباله‌ی شیفت‌یافته با بیت‌های متناظرِ در دنباله‌ی اصلی برابر هستند (نیستند).
- می‌توان نشان داد که همیشه: $C(k) = C(N - k)$
- مستقل از ویژگی‌های دنباله همیشه $C(0) = \frac{N}{N} = 1$ است.

Example's source: Michal Dobes

معیارهای Golomb برای بررسی تصادفی بودن یک دنباله

معیار سوم

- تابع خودهمبستگی دنباله، دو مقدار داشته باشد:

$$C(k) = \begin{cases} 1, \text{if } k = 0 \\ \text{Fixed value} \end{cases}$$

معیارهای Golomb برای بررسی تصادفی بودن یک دنباله

- دنباله‌هایی که معیارهای سه‌گانه‌ی Golomb را محقق کنند، اصطلاحا pseudo-noise و یا pn-sequence نامیده می‌شوند.
- مثال: دنباله‌ی زیر معیارهای سه‌گانه را برآورد می‌کند (بررسی کنید!):

011001000111101

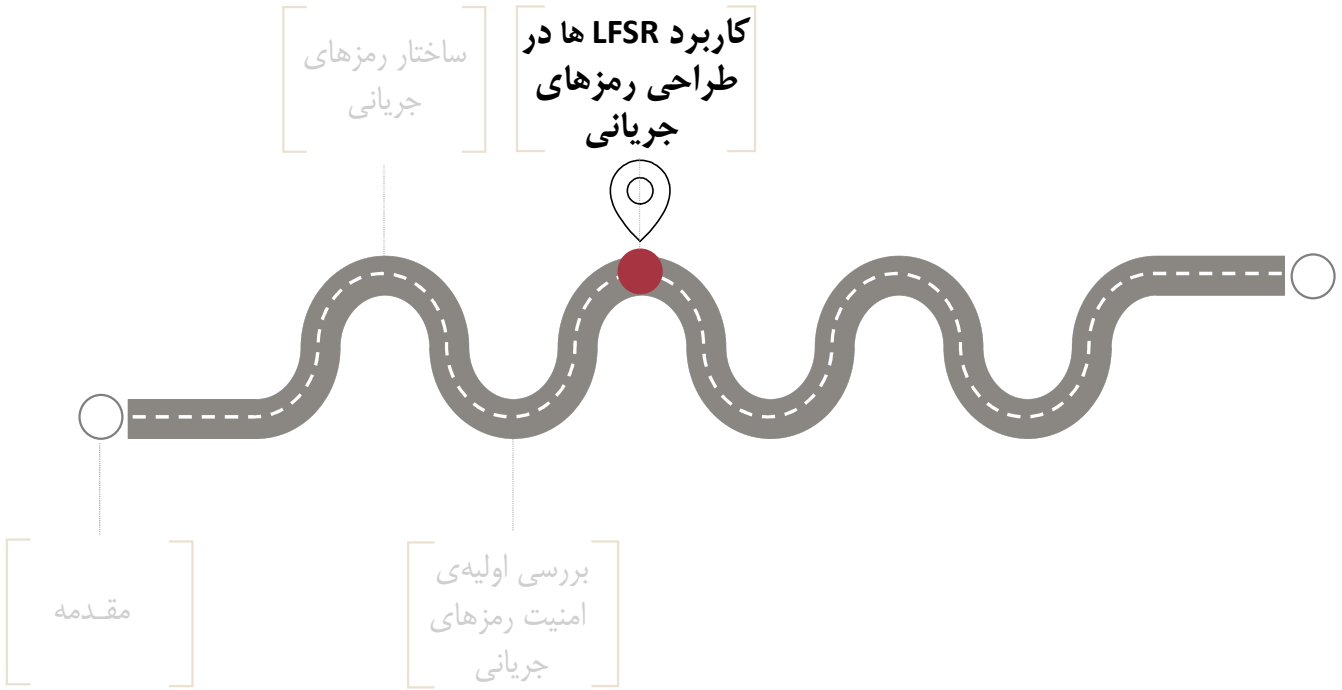
الزامات اولیه‌ی دنباله‌ی کلید اجرایی

دوره‌ی تناوب

- به اندازه کافی بزرگ باشد، به‌گونه‌ای که در عمل قابل مشاهده نباشد.

معیارهای Golomb
(جنبه‌ی تئوری)

- توازن بین تعداد 1 ها و 0 ها وجود داشته باشد.
- $\frac{1}{2^i}$ از runها، طول i داشته باشند.
- تابع خودهمبستگی تنها دو مقدار داشته باشد.


کاربرد LFSR ها در طراحی رمزهای جریانی
ساختار رمزهای جریانی
مقـدمه
بررسی اولیه‌ی امنیت رمزهای جریانی

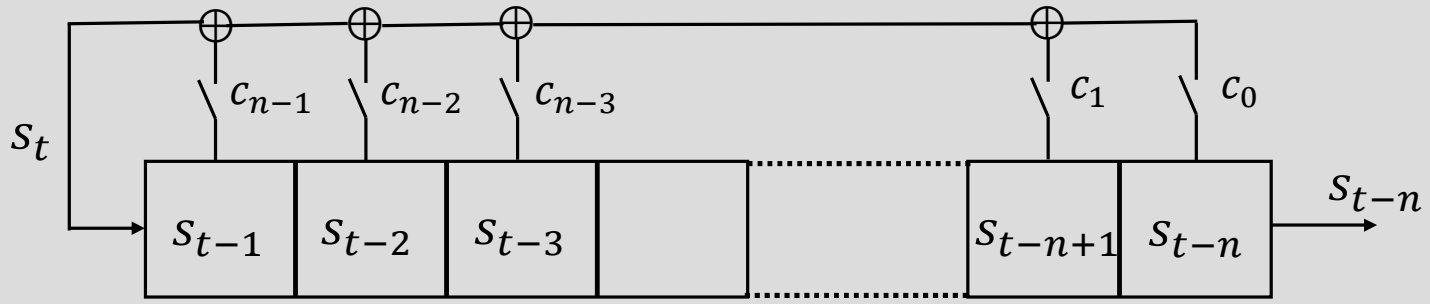
LFSR

(Linear Feedback Shift Register)

$$s_t = \sum_{i=1}^{n} c_{n-i} s_{t-i} = c_{n-1}s_{t-1} + c_{n-2}s_{t-2} + \cdots + c_0 s_{t-n}$$

- بر اساس ضرائب c_i، چندجمله‌ای فیدبک به شکل زیر تعریف می‌شود:

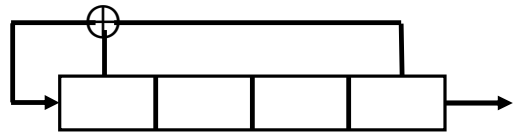
$$f(x) = x^n + c_{n-1}x^{n-1} + c_{n-2}x^{n-2} + \cdots + c_1 x + c_0$$



مثال

$$f(x) = x^4 + x^3 + 1$$

$$\Rightarrow c_0 = c_3 = 1 \ \ and \ \ c_1 = c_2 = 0$$



0	0	1	1	حالت اولیه:
1	0	0	1	حالت بعدی:
0	1	0	0	ادامه‌ی حالت ها به همین ترتیب:
0	0	1	0	
0	0	0	1	
1	0	0	0	
...	...	...	...	
0	1	1	0	حالت پانزدهم:
0	0	1	1	حالت شانزدهم:

دنباله‌ی خروجی (در یک تناوب): 110010001111010

برخی از ویژگی‌های LFSR

- هر حالت به صورت یکتا به یک حالت دیگر تبدیل می‌شود.
- حالت تمام 0 به تمام 0 می‌رود.
- در نتیجه حداکثر مقدار دوره‌ی تناوب برای LFSR برابر است با: $2^n - 1$.
- اگر تمام ضرایب تابع فیدبک صفر باشند، بیت جدید حالت بعدی همیشه 0 است.
  - در این صورت پس از n پالس زمانی حالت تمام 0 خواهیم داشت.
  - واضح است که این حالت کاربردی نیست؛ بنابراین هیچ‌وقت تمام ضرایب به صورت هم‌زمان 0 نیستند.
- منطقا همواره $c_0 = 1$ است.
  - استفاده از طول حداکثری LFSR

LFSR با طول حداکثری

- **قضیه:** اگر $f(x)$ یک چندجمله‌ای اولیه از درجه‌ی n در $GF(2)$ باشد، در این صورت دنباله‌ی غیر صفر حاصل از یک LFSR با تابع فیدبک $f(x)$، دارای حداکثر طول دوره‌ی تناوب $(2^n - 1)$ است.
- بنابراین می‌توان به راحتی یک LFSR با دوره‌ی تناوب حداکثری $(2^n - 1)$ ساخت.

بررسی معیارهای Golomb برای LFSR با دوره‌ی تناوب حداکثری

معیار اول

- تعداد 1ها در هر دوره تناوب: $\frac{2^n}{2}$
- تعداد 0ها در هر دوره تناوب: $\frac{2^n}{2} - 1$

بررسی معیارهای Golomb برای LFSR با دوره‌ی تناوب حداکثری

معیار دوم

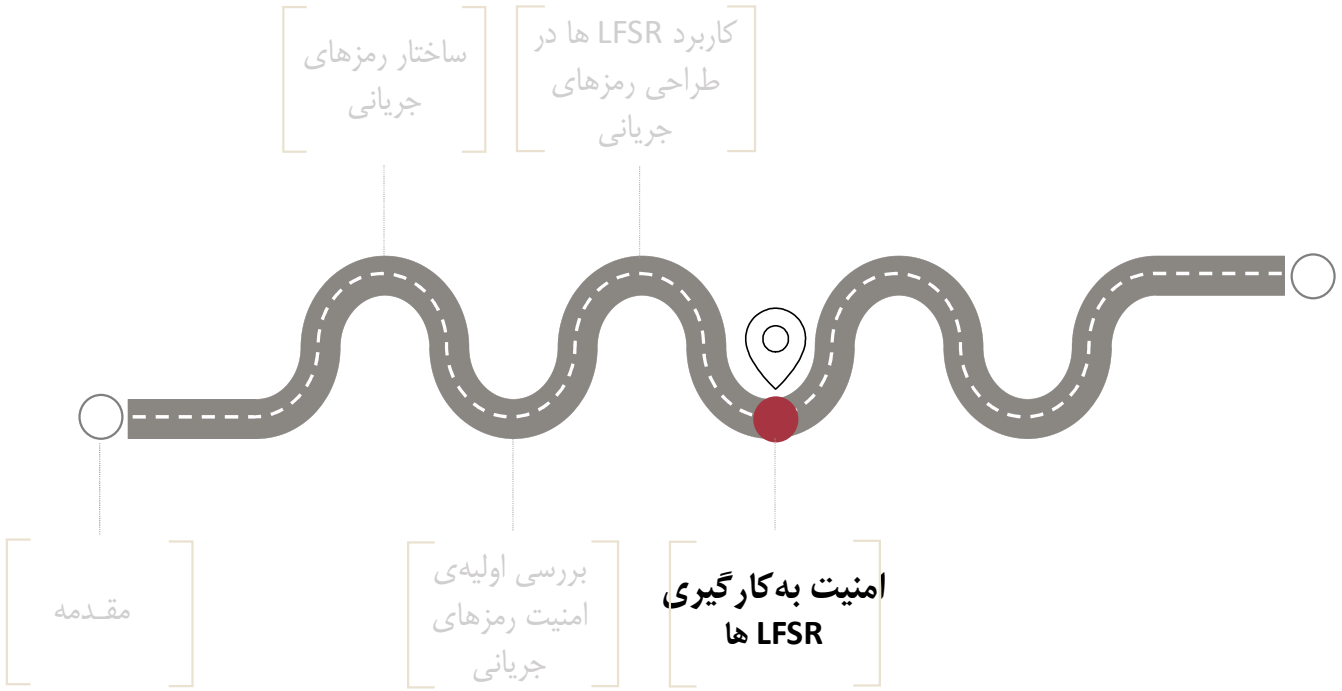
- تعداد gap با طول n و بزرگتر از n: 0
- تعداد block با طول n: 1
- تعداد block با طول بزرگتر از n: 0
- تعداد gap با طول $n-1$: 1
- تعداد block با طول $n-1$: 0
  - وجود یک block به طول n: $11\ldots11 \longrightarrow 11\ldots10$
برای $1 \le r \le n-2$:
- تعداد gap با طول r: 2^{n-r-2}
  - 10…01
- تعداد block با طول r: 2^{n-r-2}

بررسی معیارهای Golomb برای LFSR با دوره‌ی تناوب حداکثری

معیار سوم

- $\{s_t\} \oplus \{s_{t+k}\} = \{s_{t+p}\}$
- تعداد دفعاتی که در دنباله‌ی $\{s_{t+p}\}$، 0 وجود دارد: $2^{n-1} - 1$
- تعداد دفعاتی که در دنباله‌ی $\{s_{t+p}\}$، 1 وجود دارد : 2^{n-1}
- $C(k) = \frac{-1}{2^{n-1}}$

نتیجه‌ی بررسی هر سه معیار: LFSR ویژگی‌های بسیار مناسبی دارد که ظاهرا الزامات اولیه برای استفاده در ساخت رمزهای جریانی را برآورده می‌سازد!


ساختار رمزهای جریانی
کاربرد LFSR ها در طراحی رمزهای جریانی
مقـدمه
بررسی اولیه‌ی امنیت رمزهای جریانی
امنیت به‌کارگیری LFSR ها

امنیت LFSR به عنوان یک رمز جریانی

- یک LFSR را به عنوان رمز جریانی را در نظر می‌گیریم که ضرایب تابع فیدبک آن کلید مخفی هستند.
- فرض کنید $2n$ بیت **متوالی** از دنباله‌ی کلید اجرایی $(s_r, \dots, s_{r+2n-1})$ را در اختیار داشته باشیم.
- در این صورت می‌توان کلید اصلی را به دست آورد!
- دلیل: ضعف ذاتی LFSRها، خطی بودن آن‌ها است.

$$s_{t+n} = \sum_{i=0}^{n-1} c_i s_{t+i}, \qquad t = r, \dots, n+r-1$$

$$\begin{bmatrix} s_{r+n} \\ s_{r+n-1} \\ \vdots \\ s_{r+2n-1} \end{bmatrix} = \begin{bmatrix} s_r & s_{r+1} & \cdots & s_{r+n-1} \\ s_{r+1} & s_{r+2} & & s_{r+n} \\ \vdots & \vdots & \vdots & \vdots \\ s_{r+n-1} & s_{r+n} & \cdots & s_{r+2n-2} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix}$$

امنیت LFSR به عنوان یک رمز جریانی

... ادامه

سناریوی مشابه:

- فرض کنید به جای $2n$ بیت، $2n-m$ بیت **متوالی از** دنباله‌ی کلید اجرایی در اختیار باشد.
- حتی در این صورت نیز حمله به صورت زیر قابل اجرا است:
  - m بیت را حدس می زنیم و حمله قبلی را 2^m بار تکرار می‌کنیم.
  - اگر 2^m کوچکتر از 2^n باشد، حمله بهتر از جست‌وجوی کامل است.

معادل خطی

(Linear Equivalent)

- چند LFSR وجود دارند که می‌توانند دنباله $\{s_t\}_p$ را تولید کند؟
  - بی نهایت!
- به یک LFSR با حداقل طول که می‌تواند دنباله‌ی $\{s_t\}_p$ را تولید کند، معادل خطی گویند.
- الگوریتم Berlekamp-Massey می‌تواند به صورت بهینه معادل خطی و همچنین حالت اولیه‌ی LFSR که دنباله‌ی مورد نظر را تولید می‌کند، به دست آورد.
  - پیچیدگی زمانی الگوریتم: L^2
- یک ساختار غیرخطی با معادل خطی L، با پیچیدگی $2L$ شکسته می‌شود.
- اگر L به اندازه کافی بزرگ باشد، هیچگاه این تعداد از دنباله‌ی کلید اجرایی در اختیار مهاجم قرار نمی گیرد.
- باید پیچیدگی خطی به اندازه‌ی کافی بزرگ باشد.

کاربرد LFSRها در رمزنگاری

مزایا

- دوره‌ی تناوب زیاد (تقریبا حداکثری)
- برآورده کردن معیارهای Golomb

معایب

- چالش خطی بودن

الزامات اولیه‌ی دنباله‌ی کلید اجرایی

دوره‌ی تناوب

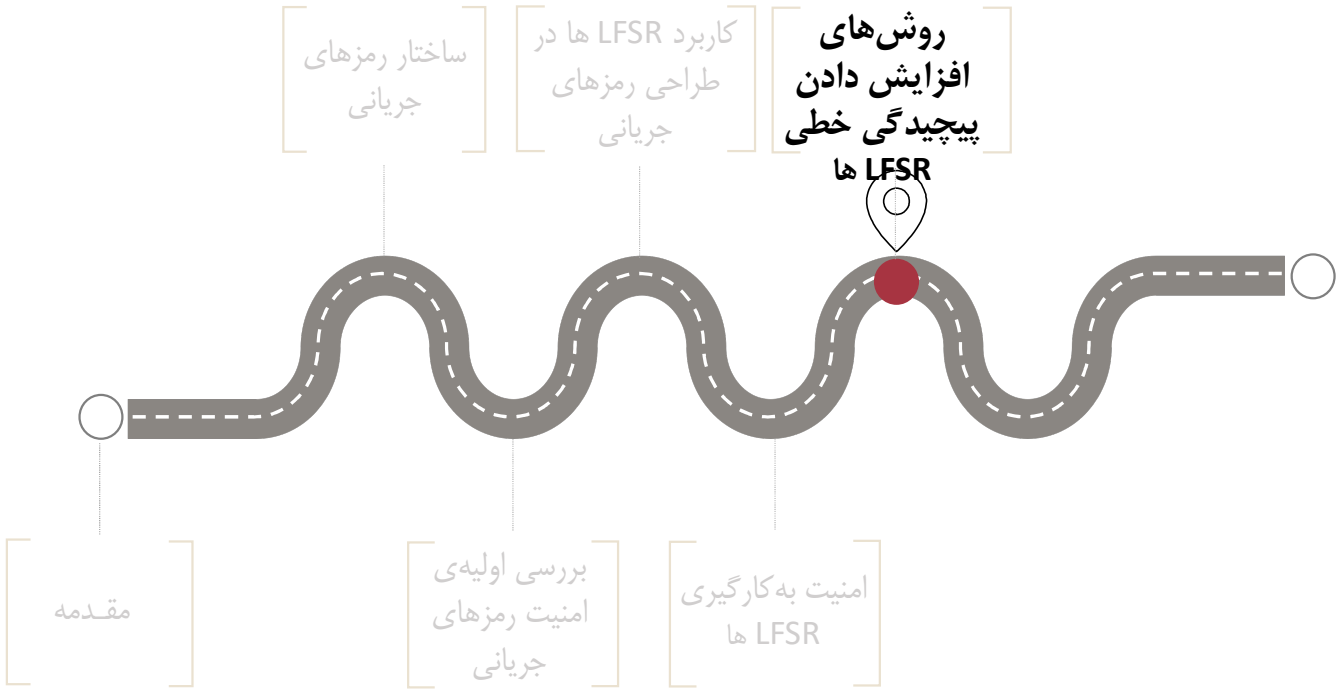
- به اندازه کافی بزرگ باشد، به‌گونه‌ای که در عمل قابل مشاهده نباشد.

معیارهای Golomb (جنبه‌ی تئوری)

- توازن بین تعداد 1 ها و 0 ها وجود داشته باشد.
- $\frac{1}{2^i}$ از runها، طول i داشته باشند.
- تابع خودهمبستگی تنها دو مقدار داشته باشد.

غیرخطی بودن

- راهکار عملی؟
- آیا می‌شود از LFSR به نحوی غیرخطی استفاده کرد و از مزایای آن بهره برد؟


روش‌های افزایش دادن پیچیدگی خطی LFSR ها
کاربرد LFSR ها در طراحی رمزهای جریانی
ساختار رمزهای جریانی
مقـدمه
بررسی اولیه‌ی امنیت رمزهای جریانی
امنیت به‌کارگیری LFSR ها

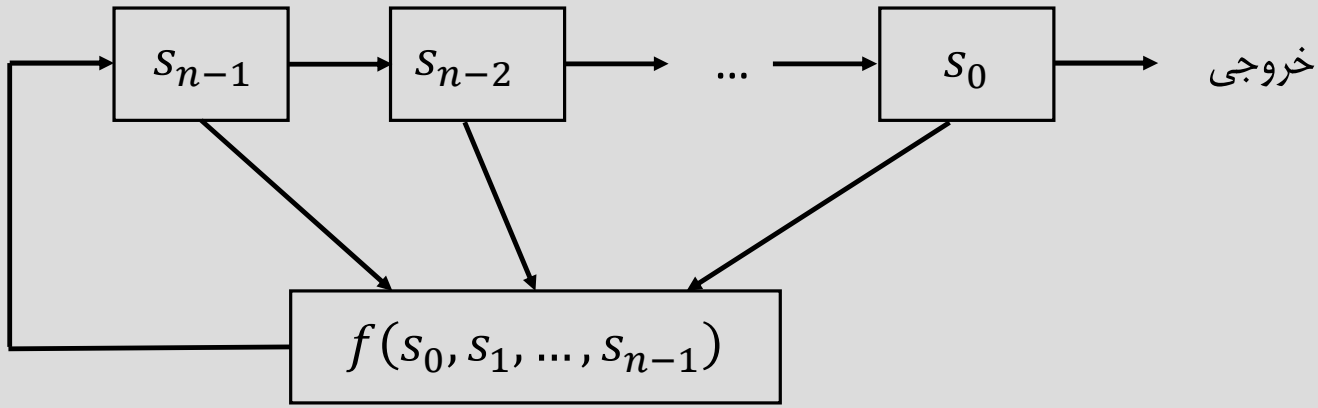
روش‌های افزایش دادن پیچیدگی LFSR ها

- استفاده از شیفت رجیستر با تابع فیدبک غیرخطی (Nonlinear FSR).
- استفاده از یک مولد فیلتر غیرخطی (Nonlinear filter generator).
- ترکیب غیرخطی خروجی چند LFSR.
- استفاده از پالس‌های نامنظم:
  - گام‌های متناوب (Alternating Steps).
  - کاهش غیرمنظم خروجی (Shrinking).
- ...
- ترکیبی از روش‌های فوق!

استفاده از شیفت رجیستر با تابع فیدبک غیرخطی

(Nonlinear FSR)

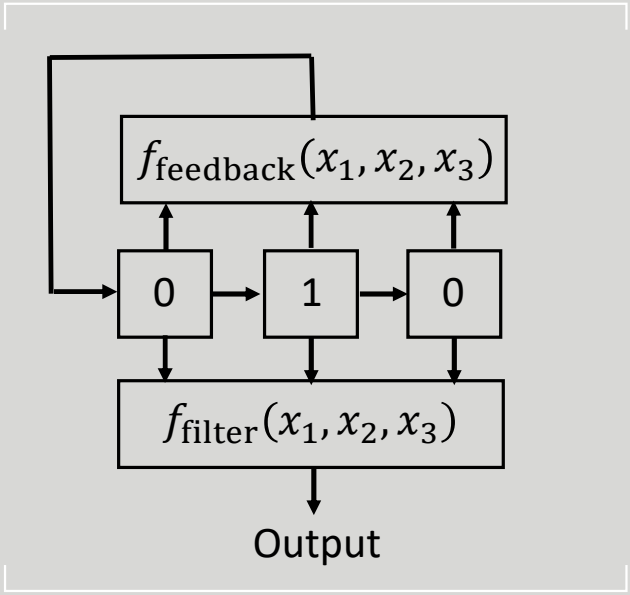
- از شیفت رجیستری استفاده کنیم که تابع فیدبک آن **غیرخطی** باشد.



استفاده از یک مولد فیلتر غیرخطی

(**Nonlinear filter generator**)

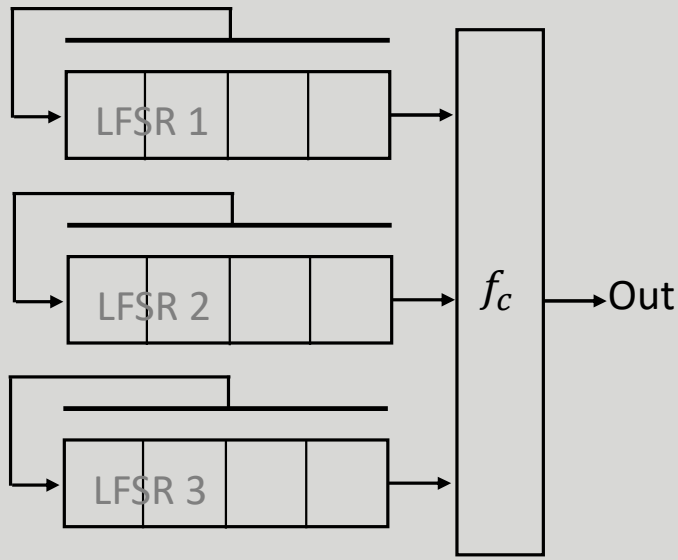
- از یک LFSR استفاده می‌کنیم.
- در هر مرحله، یک تابع فیلتر **غیرخطی** با ورودی حالت فعلی LFSR یک بیت تولید می‌کند.



Figure's source: Michal Dobes

ترکیب غیرخطی خروجی چند LFSR

- تابع ترکیب می‌تواند حافظه داشته باشد.
- خروجی‌هایی که هر بار تولید می‌شوند، ترکیبی از ورودی‌های جدید و خروجی‌های قبلی باشند.

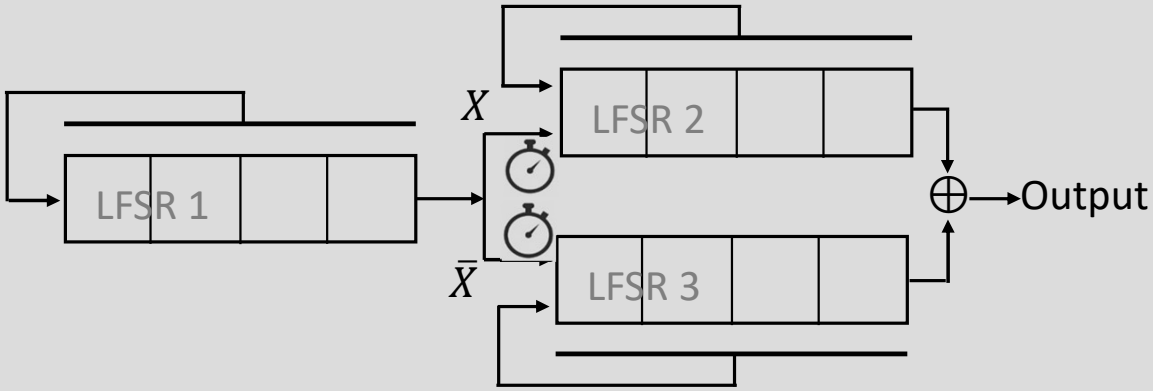


Figure's source: Michal Dobes

استفاده از پالس‌های نامنظم: گام‌های متناوب

(Alternating Steps)

- در هر مرحله براساس خروجی LFSR1، تصمیم گرفته می‌شود که کدام LFSR اجرا شود.

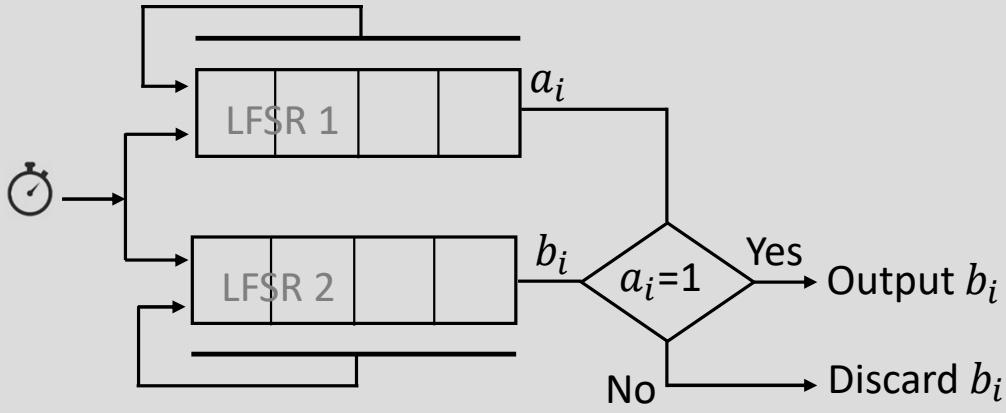


Figure's source: Michal Dobes

استفاده از پالس‌های نامنظم: کاهش غیرمنظم خروجی

(Shrinking)

- براساس خروجی LFSR1، تصمیم گرفته می‌شود که خروجی LFSR2 به عنوان خروجی الگوریتم استفاده شود یا خیر.
- برخی شرایط:
  - $\gcd(L_1, L_2) = 1$
  - $L_1 \simeq L_2$

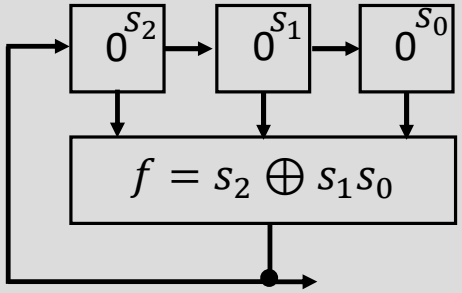


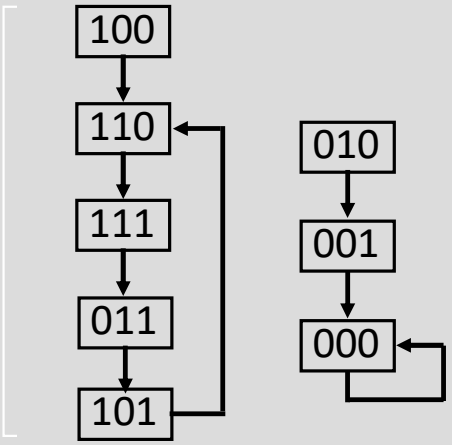
Figure's source: Michal Dobes

چالش احتمالی روش‌های افزایش پیچیدگی

وجود شاخه (Branching)

- تعریف: حالت یک شیفت رجیستر با بیش از یک حالت ماقبل را نقطه‌ی شاخه‌ای گویند (Branching Point).
- وجود شاخه موجب کوتاه شدن دوره‌ی تناوب می‌شود.





Example's source: Michal Dobes

چالش احتمالی روش‌های افزایش پیچیدگی

ضعف‌های آماری (پنهان)

- عملگرهای غیرخطی ممکن است که در مقابل حملات آماری ضعف ایجاد کنند.
  - حمله‌ی همبستگی

(از موضوعات درس رمزنگاری پیشرفته!)

×	0	1
0	0	0
1	0	1

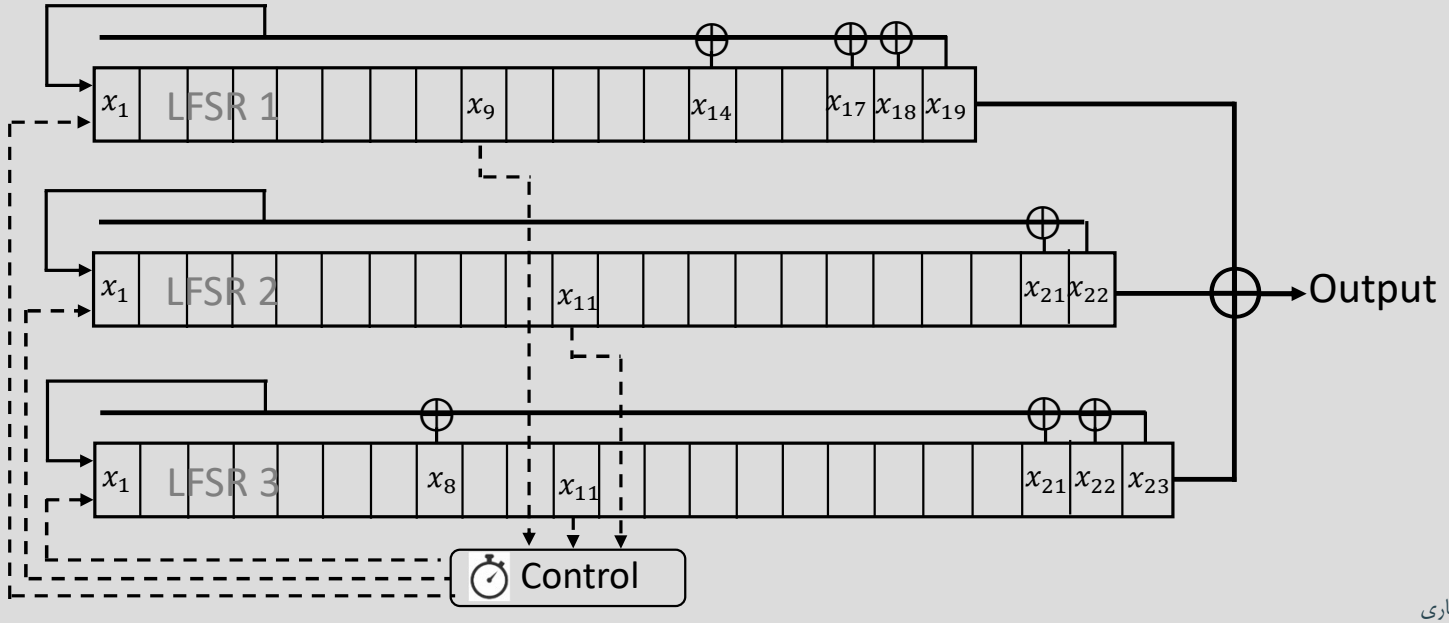
یک 1 و سه 0

⊕	0	1
0	0	1
1	1	0

دو 1 و دو 0

مثال این بخش : رمز جریانی A5/1

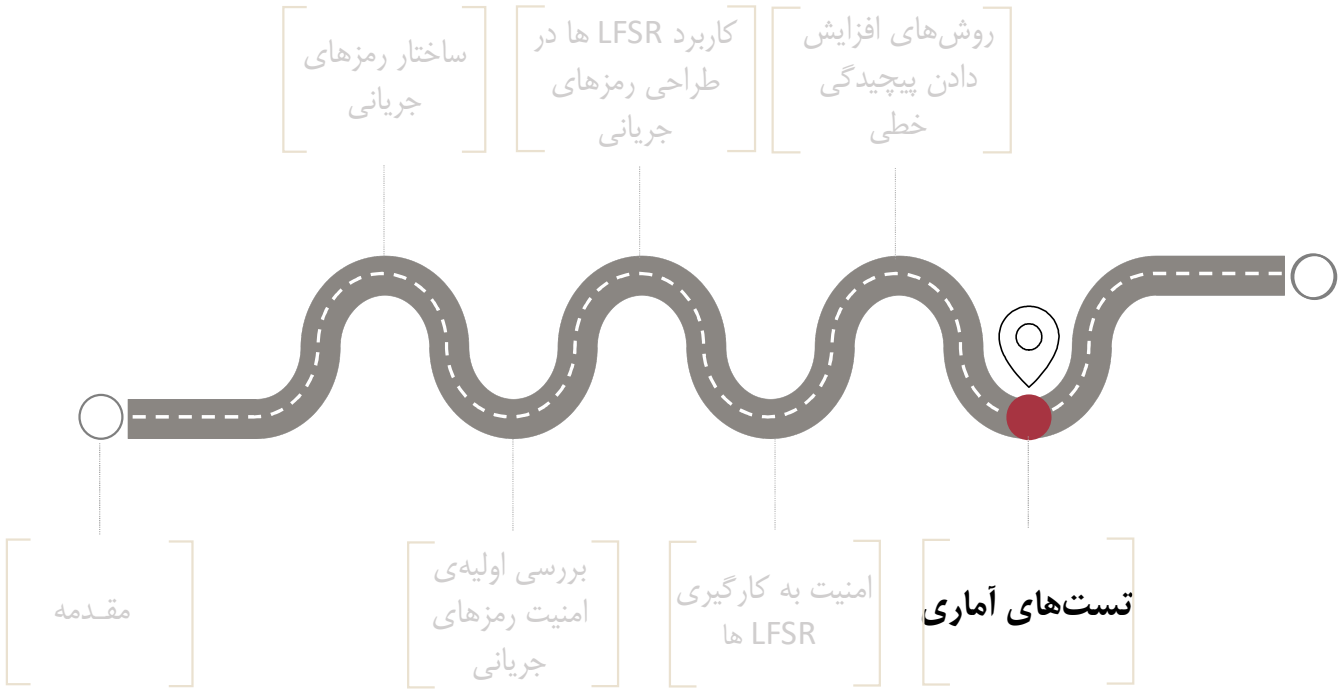
- این رمز جریانی که با استفاده از روش‌های افزایش پیچیدگی ساخته شده است، در ماژول‌های GSM کاربرد دارد.
- $\gcd(L_1,L_2) = \gcd(L_1,L_3) = \gcd(L_2,L_3) = 1$
- به خاطر طول کلید کوتاه (۶۴ بیت) در معرض حملات مبادله‌ی زمان، حافظه و داده قرار دارد!



Example's source: Michal Dobes

مقایسه LFSR و روش‌های افزایش پیچیدگی خطی

LFSR	روشهای افزایش پیچیدگی خطی
وجود قضایای ریاضی متعدد	تئوری ضعیف‌تر
فهم دقیق از عملکرد تابع	سخت‌تر شدن تجزیه و تجلیل الگوریتم
خواص مطلوب همچون دوره‌ی تناوب و یا معیارهای Golomb	می‌تواند خواص مشابه و یا بهتر داشته باشد، می‌تواند (به شدت) ضعیف‌تر شود.
ضعف خطی بودن (شکسته شدن سریع سیستم)	غیرخطی

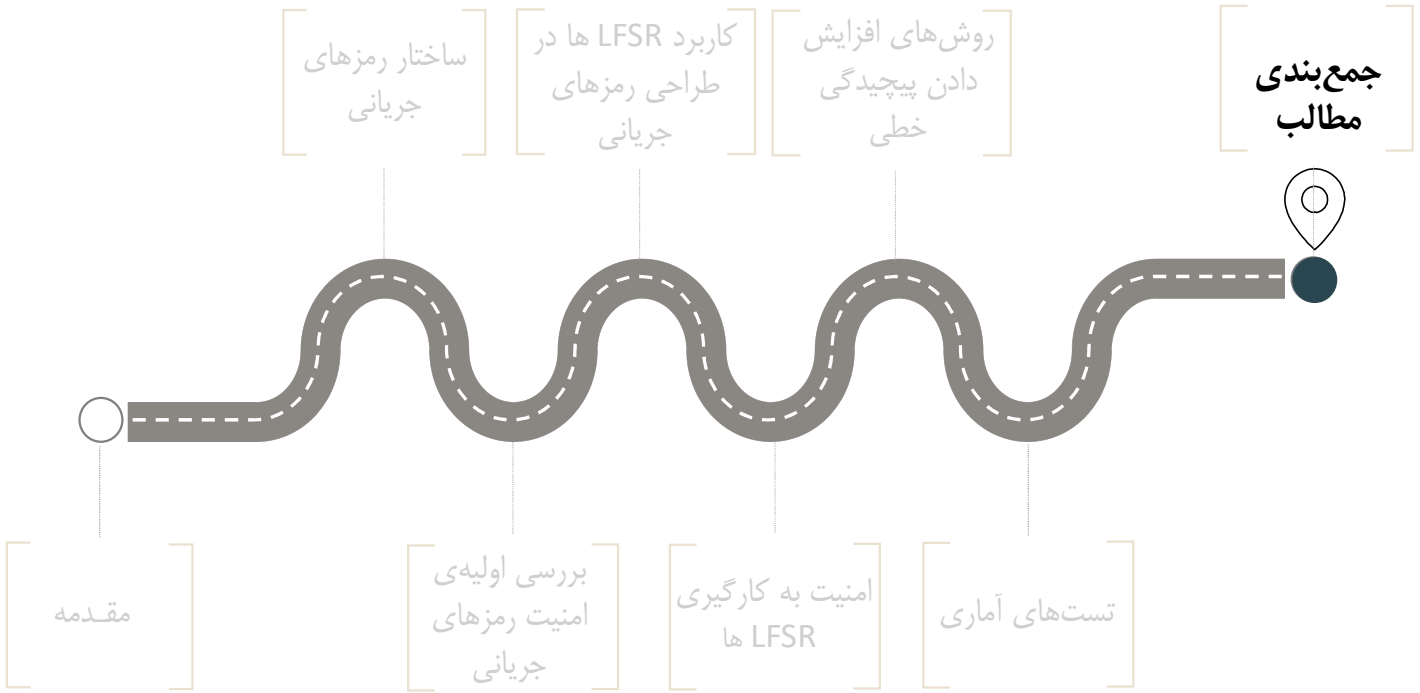

روش‌های افزایش دادن پیچیدگی خطی
کاربرد LFSR ها در طراحی رمزهای جریانی
ساختار رمزهای جریانی
تست‌های آماری
امنیت به کارگیری LFSR ها
بررسی اولیه‌ی امنیت رمزهای جریانی
مقـدمه

تست‌های آماری

- معمولا در عمل سنجش معیارهای Golomb برای دنباله‌ها امکان‌پذیر نمی‌باشد!
- چرا که برای یک رمز جریانی، دوره‌ی تناوب بسیار بزرگی داریم که نمی‌توان صحت معیارهای Golomb را برای آن تحقیق کرد.
- بنابراین برای تست اولیه‌ی امنیت رمزهای جریانی، از تست‌های آماری (نظیر مجموعه‌ی تست‌های پیشنهادی NIST) استفاده می‌شود.

الزامات اولیه‌ی دنباله‌ی کلید اجرایی

دوره‌ی تناوب	• به اندازه کافی بزرگ باشد، به‌گونه‌ای که در عمل قابل مشاهده نباشد.
معیارهای Golomb (جنبه‌ی تئوری)	• توازن بین تعداد 1 ها و 0 ها وجود داشته باشد. • $\frac{1}{2^i}$ از runها، طول i داشته باشند. • تابع خودهمبستگی تنها دو مقدار داشته باشد.
غیرخطی بودن	• راهکار عملی؟ • آیا می‌شود از LFSR به نحوی غیرخطی استفاده کرد و از مزایای آن بهره برد؟
تست‌های آماری	• تست‌های آماری معرفی شده توسط NIST.


جمع‌بندی مطالب
روش‌های افزایش دادن پیچیدگی خطی
کاربرد LFSR ها در طراحی رمزهای جریانی
ساختار رمزهای جریانی
تست‌های آماری
امنیت به کارگیری LFSR ها
بررسی اولیه‌ی امنیت رمزهای جریانی
مقـدمه

جمع‌بندی مطالب

- آشنایی با مفاهیم مرتبط با رمز جریانی و جایگاه آن‌ها
- آشنایی با الزامات اولیه طراحی رمزهای جریانی
- آشنایی با ویژگی‌های مناسب LFSR در طراحی رمزهای جریانی
- آشنایی با ایده‌های اولیه به منظور غیرخطی‌سازی رمزهای جریانی و چالش‌های آن