فصلنامه پدافند غیرعامل

سال دوم، شماره ۳، تابستان و پاییز ۱۳۹۰، (پیاپی ۷): صص ۴۷-۵۲

پدافند غیرعامل در شبکه‌های ارتباطات زیرساخت با تأکید بر ارزیابی امنیتی[1] الگوریتم‌های رمز جریانی

احمدرضا ویزندان[2]، عبدالرسول میرقدری[2]، جواد شیخ‌زادگان[3]

تاریخ دریافت: ۹۰/۰۸/۱۰
تاریخ پذیرش: ۹۰/۱۰/۲۶

چکیده

شبکه‌های ارتباطی، به‌عنوان یکی از زیرساخت‌های مهم کشور، بستری برای بهره‌برداری‌های گوناگون ارتباطی می‌باشد و روز به روز بر کاربردهای آن افزوده می‌شود. عدم توجه به امنیت شبکه‌های ارتباطی در کنار مزایای غیرقابل انکار حاصل از آن، می‌تواند معضلات مهم و غیرقابل جبرانی را در سطح کشور ایجاد نماید. شناخت جامع مشکلات امنیتی حاصل از این شبکه‌ها و اقدام جهت مرتفع نمودن معضلات عمده این حوزه، نقش قابل ملاحظه‌ای در ارتقاء امنیت، ایمنی و پایداری زیرساخت ارتباطات ایفا می‌نماید و گامی در راستای تامین اهداف پدافند غیرعامل در این خصوص به‌شمار می‌آید. رمزهای جریانی، یکی از مهم‌ترین نوع از الگوریتم‌های رمزنگاری متقارن می‌باشند که به‌لحاظ قابلیت‌های ویژه و مناسب در برخی از کاربردها مانند امنیت شبکه و زیرساخت مخابراتی، ارزیابی امنیتی آن‌ها در حوزه پدافند غیرعامل از اهمیت به‌سزایی برخوردارند و پروژه بین‌المللی eSTREAM در راستای افزایش فعالیت در این شاخهٔ رمزنگاری نقش به‌سزایی ایفا نمود. این مقاله یکی از الگوریتم‌های رمزجریانی پایه آرایه‌ای را با استفاده از حمله تمایز مورد تحلیل قرار می‌دهد. در واقع، ایده اصلی در معرفی یک دسته تمایزگر بر روی الگوریتم رمز جریانی HC-256′ می‌باشد که در این حمله، نیاز به 2^{556} معادلات خطی می‌باشد.

کلیدواژه‌ها: حمله تمایز، تمایزگر پایه، ارزیابی تحلیلی، HC-256′، eSTREAM

1- Cryptanalysis

۲- دانشگاه جامع امام حسین(ع)، مرکز تحقیقات فتح، Email: A_vizand@yahoo.com
۳- پژوهشکده پردازش هوشمند علائم

۱- مقدمه

امروزه زیرساخت‌های حیاتی کشور در زمینه‌های مختلف از جمله اقتصادی، صنعتی، نظامی و... وابستگی زیادی به توسعه پدیده فناوری اطلاعات و ارتباطات پیدا کرده‌اند و مزیت‌های زیادی نصیب بشر شده است؛ اما همین پدیده، آسیب‌پذیری‌ها، تهدیدها و چالش‌های جدید، پیچیده، مبهم و خطرناکی را در تمامی حوزه‌های مذکور ایجاد نموده است. با توجه به رشد فزاینده پیچیدگی شبکه‌های مخابراتی و رایانه‌ای، کنترل آسیب‌پذیری‌ها و تهدیدها یکی از مسائل جدی بشمار می‌آید. لذا یکی از راه‌های ایجاد امنیت اطلاعات، در برابر حمله و سایر خرابی‌ها در سامانه‌های ارتباطی، توسعه جدی در نظریه و پیاده‌سازی ارزیابی‌های امنیتی سامانه‌های رمزکننده می‌باشد.

سامانه‌های رمزنگاری به دو دسته عمده تقسیم می‌شوند: نوع اول، سامانه رمزنگاری متقارن می‌باشد که در آن، گیرنده و فرستنده پیام بر روی مقدار مشخص و مخفی به‌عنوان کلیدخصوصی توافق می‌نمایند و نباید شخص دیگری به این کلید دست پیدا کند. نوع دوم، سامانه رمزنگاری نامتقارن با کلید عمومی است که علت عمده ابداع آن، رفع مشکلات مربوط به توزیع کلید در این سامانه رمزنگاری می‌باشد. شایان ذکر است که سامانه‌های رمزنگاری متقارن نیز به دو گروه تقسیم‌بندی می‌شوند: رمز قالبی و رمز جریانی.

رمزهای جریانی، یکی از مهم‌ترین شیوه‌های رمزنگاری متقارن می‌باشند که به‌طور یکتا، تک‌تک حروف متن اصلی را رمز می‌نماید که این تبدیل با زمان تغییر می‌کند که در مقایسه با شیوه رمزهای قالبی، یک قالب از حروف متن اصلی را رمز می‌نماید و تبدیل رمزنگاری در این شیوه، متغیر با زمان نمی‌باشد[۱۴]. به‌طور معمول، رمزهای جریانی نسبت به رمزهای قالبی در کاربردهای سخت‌افزاری سریع‌تر بوده و قابلیت‌های ویژه و مناسب‌تری در برخی از کاربردها مانند کاربردهای با حافظه‌های میانی[1] محدود و انتشار خطای محدود، از خود نشان می‌دهند [۱۳].

الگوریتم‌های رمز جریانی به دو گروه تقسیم می‌شوند: رمزجریانی هم‌زمان[2] و رمزجریانی غیرهم‌زمان[3]. در رمزهای جریانی هم‌زمان، مولد رشته کلید اجرایی مستقل از متن اصلی و متن رمزشده است؛ در صورتی‌که در رمز جریانی غیرهمزمان، رشته‌کلید اجرایی، تابعی از کلید محرمانه و تعداد معینی از نمادهای رمز شده قبلی است[۱۴].

به‌منظور کمک به جستجو و پیدایش رمزهای جریانی جدید، مجموعه ECRYPT پروژه eSTREAM را در سال ۲۰۰۴ آغاز کرد[۶]. در واقع این مجموعه به‌عنوان یک مجموعه تحقیقاتی از اول فوریه سال ۲۰۰۴ با حمایت اتحادیه اروپا و با آرمان هدایت و گسترش فعالیت‌های پژوهشی کشورهای عضو در زمینه امنیت اطلاعات و به‌طور خاص در شاخه رمزشناسی و ته‌نقش‌نگاری رقمی[4] فعالیت خود را آغاز کرد. یکی از بهترین راه‌های ارتقاء تحقیقات در رمزهای جریانی در پروژه eSTREAM می‌توانست فراخوان طرح‌های جدید باشد که آخرین مرحله این پروژه می‌توانست یک بسته مفید در طرح‌های رمز جریانی باشد. برای کمک به طراحان، دو دیدگاه معرفی شد[۶] :

دیدگاه ۱- رمزهای جریانی برای کاربردهای نرم‌افزاری با ورودی-خروجی بسیار زیاد.

دیدگاه ۲- رمزهای جریانی برای کاربردهای سخت‌افزاری با منابع محاسباتی محدود.

در طول دوره برگزاری پروژهٔ eSTREAM، مهم‌ترین ملاک‌های ارزیابی عبارت بودند از [۱]:

۱. امنیت؛
۲. کارآیی در مقایسه با الگوریتم رمز قالبیAES؛
۳. کارآیی در مقایسه با سایر طرح‌ها؛
۴. استدلال و توجیه‌های پشتیبان؛
۵. سادگی و انعطاف‌پذیری؛
۶. تکامل و وضوح طرح.

به‌طور مسلم، امنیت مقدم بر همه چیز است.

در نهایت، پروژه چند مرحله‌ای eSTREAM در سال ۲۰۰۸ به اتمام رسید. یکی از الگوریتم‌های موفق، الگوریتم رمز جریانی HC می‌باشد که توسط وو تشریح گردید[۹]. طراح، قبل از ارائه آن، الگوریتم HC-256 [۷] را در کنفرانس FSE[5] در سال ۲۰۰۴ ارائه نمود که شکل تغییریافته این الگوریتم، یعنی HC-256′ را هم معرفی نمود[۷]. پردازش الگوریتم رمز HC-256′ به‌صورت کلمه‌ای[6] می‌باشد که طول هر کلمه ۳۲ بیت است و الگوریتم از یک کلید اصلی و بردار حالت اولیه[7] به‌طول ۲۵۶ بیت برای تولید رشته‌کلید اجرایی بهره می‌جوید[۷]. از آنجایی که این الگوریتم به شکل ممتازی طراحی گردیده و تاکنون مورد تحلیل قرار نگرفته است، بنابراین دست‌یابی به نتایج تحلیل، جذاب به‌نظر می‌رسد.

جدول ۱- نتایج دیدگاه‌ها برای طراحان بر اساس اندازه K و IV

دیدگاه	اندازه کلید (بیت)	اندازهIV (بیت)	اندازه برچسب (بیت)
1	128	64,128	----------
1A	128	64,128	128،32, 64, 96
2	80	32 , 64	-------
2A	80	32 , 64	32یا 64

1- Buffer
2- Synchronous
3- Asynchronous

4- Digital watermarking
5- Fast Software Encryption
6- word-oriented
7- Initial Value

در واقع، هدف از ارائه مقاله، ارزیابی تحلیلی الگوریتم رمز HC-256′ مبتنی بر حمله تمایز می‌باشد. در این حمله تحلیل‌گر تلاش می‌نماید که تعیین کند آیا یک رشته بیت مشخص مانند $z = z_1, z_2, \dots, z_N$ شبیه به رشته تولیدشده از الگوریتم رمز جریانی در نظر گرفته‌شده می‌باشد و یا آن رشته فقط یک رشته تصادفی می‌باشد؟ در واقع تمایزگر، الگوریتمی است که حداقل یک رشته مشخصی به‌عنوان ورودی می‌گیرد و به‌طور صحیح پاسخ سؤال بالا را می‌دهد.

در ادامه، به‌دقت یک گروه از تمایزگرها مورد بررسی قرار خواهد گرفت و نشان داده خواهد شد که حمله تمایز بر روی الگوریتم HC-256′ نیاز به 2^{556} معادلات خطی دارد که هر معادله شامل رشته خروجی الگوریتم رمز می‌باشد.

این تحقیق از نوع بنیادی است؛ زیرا با استفاده از حمله تمایز که به‌طورکامل بنیادی است، الگوریتم رمز مورد تحلیل قرار خواهد گرفت و از لحاظ روش، توصیفی- اکتشافی است زیرا با توصیف ویژگی‌های الگوریتم، نقطه ضعف را کشف می‌نماییم. روش تحلیل، از نوع تحلیل ریاضی می‌باشد.

این مقاله از بخش‌هایی به‌شرح ذیل تشکیل شده است. در بخش بعدی نمادهایمان را تعریف می‌کنیم. در بخش ۳ به‌طور مختصر الگوریتم HC-256′ معرفی می‌گردد. حمله تمایز در بخش ۴ ارائه می‌گردد و در نهایت، آخرین بخش به نتیجه‌گیری اختصاص پیدا خواهد کرد.

۲- نمادها و عملگرها

نمادها و عملگرهای مورد استفاده در این نوشتار به‌شرح ذیل می‌باشد.

+: $x + y$ یعنی $x + y \bmod 2^{32}$ که $0 \le y < 2^{32}$ و $0 \le x < 2^{32}$

$\boxminus$: $x \boxminus y$ یعنی $x-y \bmod 1024$

$\oplus$: یای انحصاری

$\|$: الحاق[1]

$x \ll y$: انتقال به راست x به اندازه y بیت.

$x \ll y$: انتقال به چپ x به اندازه y بیت.

$x \ggg n$: یعنی $((x \gg n) \oplus (x \ll (32 - n))$ که $0 \le n < 32$ و $0 \le x < 2^{32}$ (چرخش به راست)

$x \lll n$: یعنی $((x \ll n) \oplus (x \gg (32 \quad n))$ که $0 \le n < 32$ و $0 \le x < 2^{32}$،(چرخش به چپ)

s_i: رشته کلید تولیدشده در مرحله i (یعنی $i+1$ امین تکرار از مولد رشته کلید شبه تصادفی[2])

- عبارت‌های s_i^j، $(h_1(x))^j$، $(h_2(x))^j$ و $(Q[x])^j$ به‌ترتیب بیانگر j امین بیت از عبارت‌های s_i، $h_1(x)$، $h_2(x)$ و $Q[x]$ می‌باشند.
- اگر x یک کلمه باشد، سپس $x^{(i)}$ عبارت است از i امین بایت از x که $x^{(0)}$ بایت پایین‌رتبه و $x^{(3)}$ بایت بالارتبه خواهد بود.
- P و Q به‌عنوان جعبه‌های جانشینی[3] در الگوریتم $HC-256$ به‌کار می‌رود که هر کدام از جدول‌ها، دارای ۱۰۲۴ عنصر ۳۲ بیتی می‌باشند.

۳- تشریح الگوریتم رمز جریانی HC-256′

در این بخش، الگوریتم $HC-256$ به‌طور خلاصه معرفی می‌گردد[۷]. این الگوریتم از یک کلید، K و یک مقدار اولیه، IV به اندازه ۲۵۶ بیت استفاده می‌نماید. می‌توان $K = K[0] \| \cdots \| K[7]$، $IV = IV[0] \| \cdots \| IV[7]$ تعریف نمود که هر کدام از $K[i]$ و $IV[i]$ $(i = 0, \dots, 7)$ به اندازه ۳۲ بیت می‌باشند. حالت‌داخلی[4] الگوریتم $HC-256$ از دو جدول P و Q تشکیل شده که هر کدام دارای ۱۰۲۴ عنصر ۳۲ بیتی می‌باشند.

۳-۱- فرآیند پیش‌محاسبات: الگوریتم برپایی و اجرای کلید و مقداردهی اولیه

- آرایه $R[0, \dots, 2559]$ توسط توسیع K و IV به‌صورت ذیل به‌دست می‌آید.

$$R_i = \begin{cases} k_i & 0 \le i \le 7 \\ IV_{i-8} & 8 \le i \le 15 \\ f_2(R_{i-2}) + R_{i-7} + f_1(R_{i-15}) + R_{i-16} + i & 16 \le i \le 2559 \end{cases}$$

که توابع f_1 و f_2 به‌صورت زیر تعریف می‌گردند.

$$f_1(x) = (x \ggg 7) \oplus (x \ggg 18) \oplus (x \gg 3),$$
$$f_2(x) = (x \ggg 17) \oplus (x \ggg 19) \oplus (x \gg 10).$$

- جدول‌های P و Q با استفاده از آرایه‌های R به‌صورت زیر به‌روز رسانی می‌گردند.

$$P[i] \leftarrow P[i] \boxplus g_1(P[i \boxminus 3], P[i \boxminus 10]) \oplus P[i \boxminus 1] \quad \text{for } 0 \le i \le 1023$$
$$Q[i] \leftarrow Q[i] \boxplus g_2(Q[i \boxminus 3], Q[i \boxminus 10]) \oplus Q[i \boxminus 1] \quad \text{for } 0 \le i \le 1023$$

- الگوریتم، ۴۰۹۶ مرتبه اجرا می‌شود و مقادیر خروجی با عناصر جدول‌ها، به‌صورت زیر تعویض می‌گردد.

1- Concatenation

2- PRGB: pseudo random bit generation
3- S-BOX
4- Internal State

$$P[i \bmod ۱۰۲۴] = P[i \bmod ۱۰۲۴] + P[i \boxminus ۱۰] + g_۱(P[i \boxminus ۳], P[i \boxminus ۱۰۲۳]) \oplus h_۱(P[i \boxminus ۱۲])$$
$$\text{for } ۰ \le i \le ۱۰۲۳$$

$$Q[i \bmod ۱۰۲۴] = Q[i \bmod ۱۰۲۴] + Q[i \boxminus ۱۰] + g_۲(Q[i \boxminus ۳], Q[i \boxminus ۱۰۲۳]) \oplus h_۲(P[i \boxminus ۱۲])$$
$$\text{for } ۰ \le i \le ۱۰۲۳$$

در اینجا توابع $g_۱$ و $g_۲$ به شکل زیر تعریف می‌گردند.

$$g_۱(x,y) = ((x \ggg ۱۰) \oplus (y \ggg ۲۳)) + Q[(x \oplus y) \bmod ۱۰۲۴],$$
$$g_۲(x,y) = ((x \ggg ۱۰) \oplus (y \ggg ۲۳)) + P[(x \oplus y) \bmod ۱۰۲۴].$$

بعد از اتمام مراحل بالا، الگوریتم رمز جهت تولید رشته‌کلید اجرایی آماده می‌باشد.

۳-۲- الگوریتم مولد رشته کلید اجرایی

مولد رشته‌کلید شبه‌تصادفی، عناصر هر کدام از جدول‌ها را در هر مرحله به‌روزرسانی می‌نماید و یک رشته‌کلید خروجی ۳۲ بیتی (کلمه) تولید می‌کند.

$i = ۰$
Repeat until (enough keystream bits are generated)
{
$j = i \bmod ۱۰۲۴$;
$P[j] = P[j] + P[j \boxminus ۱۰] + g_۱(P[j \boxminus ۳], P[j \boxminus ۱۰۲۳])$;
$s_{۲i} = h_۱(P[j \boxminus ۱۲]) \oplus P[j]$;
$Q[j] = Q[j] + Q[j \boxminus ۱۰] + g_۲(Q[j \boxminus ۳], Q[j \boxminus ۱۰۲۳])$;
$s_{۲i+۱} = h_۲(Q[j \boxminus ۱۲]) \oplus Q[j]$;
$i = i + ۱$; each increment of i corresponds to 2 steps.
}

در روابط بالا $h_۲$ و $h_۱$ به‌صورت زیر بیان می‌گردند.

$$h_۱(x) = Q[x^{(۰)}] + Q[۲۵۶ + x^{(۱)}] + Q[۵۱۲ + x^{(۲)}] + Q[۷۶۸ + x^{(۳)}],$$
$$h_۲(x) = P[x^{(۰)}] + P[۲۵۶ + x^{(۱)}] + x^{(۲)}] + P[۷۶۸ + x^{(۳)}]$$

۴- حمله تمایز

در این بخش، حمله تمایز به الگوریتم رمز جریانی $HC-۲۵۶$ مورد بررسی قرار می‌گیرد؛ با این فرض که فرآیند پیش‌پردازش و مقداردهی اولیه کامل[1] باشد. در امین مرحله، جدول‌های P و Q به‌صورت زیر به‌روزرسانی می‌گردند:

$$P[i] = P[i] + P[i \boxminus ۱۰] + g_۱(P[i \boxminus ۳], P[i \boxminus ۱۰۲۳]),$$
$$Q[i] = Q[i] + Q[i \boxminus ۱۰] + g_۲(Q[i \boxminus ۳], Q[i \boxminus ۱۰۲۳])$$

و همچنین

$$s_{۲i} = h_۱(P[i \boxminus ۱۲]) \oplus P[i \bmod ۱۰۲۴],$$
$$s_{۲i+۱} = h_۲(Q[i \boxminus ۱۲]) \oplus Q[i \bmod ۱۰۲۴]$$

اگر $۱۰ \le i < ۱۰۲۴$ و با استفاده از این حقیقت $P[i] = s_{۲i} \oplus h_۱(Z_i)$ و $Q[i] = s_{۲i+۱} \oplus h_۲(Y_i)$ ، تابع بازخورد را می‌توان به‌صورت زیر نوشت. در اینجا Z_i و Y_i به‌ترتیب $P[i \boxminus ۱۲]$ و $Q[i \boxminus ۱۲]$ در امین مرحله می‌باشند.

$$s_{۲i} \oplus h_۱(z_i) = \left(s_{۲(i-۲,۱۱)} \oplus h'_۱(z_{i-۲,۱۱})\right) + \left(s_{۲(i-۱۰)} \oplus h_۱(z_{i-۱۰})\right) + g_۱\left(s_{۲(i-۳)} \oplus h_۱(z_{i-۳}), s_{۲(i-۲,۱۷)} \oplus h'_۱(z_{i-۲,۱۷})\right) \quad (۱)$$

و

$$s_{۲i+۱} \oplus h_۲(Y_i) = \left(s_{۲(i-۲,۳۱)+۱} \oplus h'_۲(Y_{i-۲,۳۱})\right) + \left(s_{۲(i-۱۰)+۱} \oplus h_۲(Y_{i-۱۰})\right) + g_۲\left(s_{۲(i-۳)+۱} \oplus h_۲(z_{i-۳}), s_{۲(i-۲,۳۷)+۱} \oplus h'_۲(Y_{i-۲,۳۷})\right). \quad (۲)$$

شایان ذکر است که توابع $h_۲(x)$ ،$h'_۱(x)$ ،$h_۱(x)$ و $h'_۲(x)$ به‌جهت اینکه از جعبه‌های جانشینی متفاوتی هستند، باهم فرق دارند. همان‌طور که مشاهده می‌گردد عملگرهای $+$ و $\oplus$ در بیت پایین‌رتبه[2] مانند هم عمل می‌نمایند[۱۰] بنابراین، معادله‌های (۱) و (۲) را می‌توان مجدداً به شکل زیر نوشت.

$$s'_{۲i} \oplus s'_{۲(i-۲,۳۱)} \oplus s'_{۲(i-۱۰)} \oplus s'_{۲(i-۳)} \oplus s'^{۳۲}_{۲(i-۲,۳۱)} = (h_۱(z_i))^{۰} \oplus (h'_۱(z_{i-۲,۳۱}))^{۰} \oplus (h_۱(z_{i-۱۰}))^{۰} \oplus (h_۱(z_{i-۳}))^{۱۰} \oplus (h'_۱(z_{i-۲,۳۱}))^{۳۲} \oplus (Q[n])^{۰} \quad (۳)$$

و

1- Perfect
2- Least Significant Bit (LSB)

$$s_{۲i+۱}^{\cdot} \oplus s_{۲(i-۲۰۴۸)+۱}^{\cdot} \oplus s_{۲(i-۱۰)+۱}^{\cdot} \oplus s_{۲(i-۳)+۱}^{۱} \oplus s_{۲(i-۲۰۴۷)+۱}^{۲} = (h_{۲}(Y_i))^{\cdot} \oplus (h_{۲}'(Y_{i-۲۰۴۸}))^{\cdot} \oplus (h_{۲}(Y_{i-۱۰}))^{\cdot} \oplus (h_{۲}(Y_{i-۳}))^{۱۰} \oplus (h_{۲}'(Y_{i-۲۰۴۷}))^{۳} \oplus (P[s_i])^{\cdot} \quad (۴)$$

که

$$\begin{aligned} r_i &= s_{۲(i-۳)} \oplus h_{۱}(z_{i-۳}) \oplus s_{۲(i-۲۰۴۷)} \oplus h_{۱}'(z_{i-۲۰۴۷}) \\ s_i &= s_{۲(i-۳)+۱} \oplus h_{۲}(z_{i-۳}) \oplus s_{۲(i-۲۰۴۷)+۱} \oplus h_{۲}'(Y_{i-۲۰۴۷}). \end{aligned}$$

از طرفی هنگامی‌که $۱۰ \le i \le ۱۰۲۴$ باشد الگوریتم مولد رشته‌کلید تصادفی، به‌طور دائم جدول‌های P و Q را به‌روزرسانی می‌نماید. لذا می‌توان تابع بازخورد را به‌صورت زیر نمایش داد:

$$s_{۲i}^{\cdot} \oplus s_{۲(i-۲۰۴۸)}^{\cdot} \oplus s_{۲(i-۱۰)}^{\cdot} \oplus s_{۲(i-۳)}^{۱} \oplus s_{۲(i-۲۰۴۷)}^{۲} \oplus s_{۲(i-۲۰۴۷)+۱}^{۲} \oplus s_{۲(i-۲۰۴۸)+۱}^{\cdot} \oplus s_{۲(i-۱۰)+۱}^{\cdot} \oplus s_{۲(i-۳)+۱}^{۱} = (h_{۱}(z_i))^{\cdot} \oplus (h_{۱}'(z_{i-۲۰۴۸}))^{\cdot} \oplus (h_{۱}(z_{i-۱۰}))^{\cdot} \oplus (h_{۱}(z_{i-۳}))^{۱۰} \oplus (h_{۱}'(z_{i-۲۰۴۷}))^{۳} \oplus (Q[r_i])^{\cdot} \oplus (h_{۲}(Y_i))^{\cdot} \oplus (h_{۲}'(Y_{i-۲۰۴۸}))^{\cdot} \oplus (h_{۲}(Y_{i-۱۰}))^{\cdot} \oplus (h_{۲}(Y_{i-۳}))^{۱۰} \oplus (h_{۲}'(Y_{i-۲۰۴۷}))^{۳} \oplus (P[s_i])^{\cdot} \quad (۵)$$

با درنظر گرفتن سمت چپ معادله (۵) می‌توان حمله پایه را به‌صورت زیر اعمال نمود:

بـــرای $۲۰۴۸ \times \alpha + ۱۰ \le i, j < ۲۰۴۸ \times \alpha + ۱۰۲۴$ و $j \neq i$ می‌توان معادله (۵) را به‌صورت زیر نوشت:

$$\begin{aligned} &s_{۲j}^{\cdot} \oplus s_{۲(j-۲۰۴۸)}^{\cdot} \oplus s_{۲(j-۱۰)}^{\cdot} \oplus s_{۲(j-۳)}^{۱} \oplus s_{۲(j-۲۰۴۷)}^{۲} \oplus s_{۲(j-۲۰۴۷)+۱}^{۲} \oplus s_{۲(j-۲۰۴۸)+۱}^{\cdot} \ominus s_{۲(j-۱۰)+۱}^{\cdot} \oplus s_{۲(j-۳)+۱}^{۱} \oplus s_{۲(j-۲۰۴۷)+۱}^{۲} \\ &= (h_{۱}(z_j))^{\cdot} \oplus (h_{۱}'(z_{j-۲۰۴۸}))^{\cdot} \oplus (h_{۱}(z_{j-۱۰}))^{\cdot} \oplus (h_{۱}(z_{j-۳}))^{۱۰} \oplus (h_{۱}'(z_{j-۲۰۴۷}))^{۳} \oplus (Q[r_j])^{\cdot} \oplus (h_{۲}(Y_j))^{\cdot} \oplus (h_{۲}'(Y_{j-۲۰۴۸}))^{\cdot} \oplus (h_{۲}(Y_{j-۱۰}))^{\cdot} \oplus (h_{۲}(Y_{j-۳}))^{۱۰} \oplus (h_{۲}'(Y_{j-۲۰۴۷}))^{۳} \oplus (P[s_j])^{\cdot} \end{aligned} \quad (۶)$$

به‌عبارتی، اگر سمت چپ معادله‌های (۵) و (۶) برابر باشد، داریم:

$$s_{۲i}^{\cdot} \oplus s_{۲(i-۲۰۴۸)}^{\cdot} \oplus s_{۲(i-۱۰)}^{\cdot} \oplus s_{۲(i-۳)}^{۱} \oplus s_{۲(i-۲۰۴۷)}^{۲} \oplus s_{۲(i-۲۰۴۷)+۱}^{۲} \ominus s_{۲(i-۲۰۴۸)+۱}^{\cdot} \oplus s_{۲(i-۱۰)+۱}^{\cdot} \oplus s_{۲(i-۳)+۱}^{۱} \oplus s_{۲(i-۲۰۴۷)+۱}^{۲}$$

$$= s_{۲j}^{\cdot} \oplus s_{۲(j-۲۰۴۸)}^{\cdot} \oplus s_{۲(j-۱۰)}^{\cdot} \oplus s_{۲(j-۳)}^{۱} \oplus s_{۲(j-۲۰۴۷)}^{۲} \oplus s_{۲(j-۲۰۴۷)+۱}^{۲} \oplus s_{۲(j-۲۰۴۸)+۱}^{\cdot} \oplus s_{۲(j-۱۰)+۱}^{\cdot} \oplus s_{۲(j-۳)+۱}^{۱} \oplus s_{۲(j-۲۰۴۷)+۱}^{۲}$$

سپس طرف راست معادله‌های (۵) و (۶) نیز برابر خواهند بود:

$$\begin{aligned} &(h_{۱}(z_i))^{\cdot} \oplus (h_{۱}'(z_{i-۲۰۴۸}))^{\cdot} \oplus (h_{۱}(z_{i-۱۰}))^{\cdot} \oplus (h_{۱}(z_{i-۳}))^{۱۰} \oplus (h_{۱}'(z_{i-۲۰۴۷}))^{۳} \oplus (Q[r_i])^{\cdot} \oplus (h_{۲}(Y_i))^{\cdot} \oplus (h_{۲}'(Y_{i-۲۰۴۸}))^{\cdot} \oplus (h_{۲}(Y_{i-۱۰}))^{\cdot} \oplus (h_{۲}(Y_{i-۳}))^{۱۰} \oplus (h_{۲}'(Y_{i-۲۰۴۷}))^{۳} \oplus (P[s_i])^{\cdot} = \\ &(h_{۱}(z_j))^{\cdot} \oplus (h_{۱}'(z_{j-۲۰۴۸}))^{\cdot} \oplus (h_{۱}(z_{j-۱۰}))^{\cdot} \oplus (h_{۱}(z_{j-۳}))^{۱۰} \oplus (h_{۱}'(z_{j-۲۰۴۷}))^{۳} \oplus (Q[r_j])^{\cdot} \oplus (h_{۲}(Y_j))^{\cdot} \oplus (h_{۲}'(Y_{j-۲۰۴۸}))^{\cdot} \oplus (h_{۲}(Y_{j-۱۰}))^{\cdot} \oplus (h_{۲}(Y_{j-۳}))^{۱۰} \oplus (h_{۲}'(Y_{j-۲۰۴۷}))^{۳} \oplus (P[s_j])^{\cdot} \end{aligned} \quad (۸)$$

یادآوری می‌گردد:

$$\begin{aligned} z_i &= z_{i-۲۰۴۸} + z_{i-۱۰} + g_1(z_{i-۳}, z_{i-۲۰۴۷}), \\ z_j &= z_{j-۲۰۴۸} + z_{j-۱۰} + g_1(z_{j-۳}, z_{j-۲۰۴۷}), \\ Y_j &= Y_{j-۲۰۴۸} + Y_{j-۱۰} + g_2(Y_{j-۳}, Y_{j-۲۰۴۷}), \\ Y_i &= Y_{i-۲۰۴۸} + Y_{i-۱۰} + g_2(Y_{i-۳}, Y_{i-۲۰۴۷}). \end{aligned} \quad (۹)$$

می‌توان معادله (۸) را به‌صورت زیر تقریب زد:

$$H(x_1) = H(x_2), \quad (۱۰)$$

H یک جعبه جانشینی ۲۷۶ بیت به ۱ بیت می‌باشد. x_1 و x_2 دو ورودی تصادفی ۲۷۶ بیتی می‌باشند.

$$\begin{aligned} x_1 &= z_{i-۳} \| z_{i-۱۰} \| z_{i-۲۰۴۷} \| z_{i-۲۰۴۸} \| Y_{i-۳} \| Y_{i-۱۰} \| Y_{i-۲۰۴۷} \| Y_{i-۲۰۴۸} \\ x_2 &= z_{j-۳} \| z_{j-۱۰} \| z_{j-۲۰۴۷} \| z_{j-۲۰۴۸} \| Y_{j-۳} \| Y_{j-۱۰} \| Y_{j-۲۰۴۷} \| Y_{j-۲۰۴۸} \end{aligned}.$$

برای به‌دست آوردن نرخ تصادم[1] می‌توان از قضیه زیر استفاده نمود:
قضیه: اگر S یک جعبه جانشینی m بیت به n بیت باشد و تمام عناصر n بیتی به‌طور تصادفی تولید شده و $m \ge n$ باشد. حال x_1 و x_2 دو ورودی تصادفی به S باشند سپس $s(x_1) = s(x_2)$ با احتمال $۲^{-m} + ۲^{-n} - ۲^{-m-n}$ می‌باشد[۹].

1- Collision Rate

اثبات. اگر $x_1 = x_2$، در این صورت $s(x_1) = s(x_2)$. اگر $x_1 \neq x_2$ در این صورت $s(x_1) = s(x_2)$ با احتمال 2^{-n}. $x_1 = x_2$ با احتمال 2^{-m} و $x_1 \neq x_2$ با احتمال $1 - 2^{-m}$. حال احتمال $s(x_1) = s(x_2)$ برابر است با $2^{-n}(1 - 2^{-m}) + 2^{-m}$. با استفاده از این قضیه، احتمال معادله (۱۰) برابر $\frac{1}{2} + 2^{-277}$ می‌باشد. بنابراین احتمال معادله (۷) هم با همین احتمال یعنی $\frac{1}{2} + 2^{-277}$ می‌باشد. حال تمام اجزا، برای بنا کردن حمله تمایز به $HC-256$ مهیا می‌باشد. برای انجام این کار فرض کنید N تعداد تمام معادله‌های(۷) می‌باشد. همچنین فرض کنید که به‌ترتیب D و D' توزیع جمع انحصاری[1] ۲۰ خروجی از معادله (۷) برای الگوریتم رمز $HC-256$ و یک الگوریتم رمز ایده‌آل باشد. میانگین و انحراف استاندارد این توزیع با $\mu = Np$ و $\sigma = \sqrt{Np(1-p)}$ برای D و $\mu' = Np$ و $\sigma' = \sqrt{Np'(1-p')}$ برای D' بدست می‌آید. یادآوری می‌گردد از نتایج بالا داریم $p = \frac{1}{2} + 2^{-277}$ و $p' = \frac{1}{2}$ و هنگامی که N بزرگ باشد هر دو توزیع دو جمله‌ای[2] را با توزیع نرمال می‌توان تقریب زد. حال اگر $|\mu - \mu'| \geq 2(\sigma + \sigma')$ یعنی برای $N \geq 2^{556}$، خروجی الگوریتم رمز را با احتمال ۰/۹۷۷۲ می‌توان از یک رشته تمام تصادفی تمایز داد. حال می‌توان رشته کلیدهای مورد نیاز حمله تمایز را به‌صورت زیر به‌دست آورد. شایان ذکر است که معادله(۷) دارای ۲۰ رشته کلید خروجی می‌باشد لذا در تمایزگر پایه نیاز به $20.2^{556} = 2^{560.3}$ رشته‌کلید خروجی است.

۵- نتیجه‌گیری

در این مقاله حمله تمایز به الگوریتم رمز جریانی $HC-256$ ارائه گردید. در ادامه این حمله، احتمال اریبی و داده‌های مورد نیاز برای حمله تمایز بر روی الگوریتم مورد اشاره نشان داده شد. در این ارزیابی تحلیلی احتمال اریبی را 2^{-277} به‌دست آوردیم و همچنین با این احتمال تعداد معادلات مورد نیاز 2^{556} و خروجی‌های مورد نیاز برای حمله یعنی $2^{560.3}$ محاسبه گردید.

1- XOR
2- Binomial

مراجع

1. ECRYPT. eSTREAM: ECRYPT Stream Cipher Project, IST-2002-507932. http://www.ecrypt.eu.org/stream/ (**2008**).
2. NESSIE. New European Schemes for Signatures, Integrity, and Encryption http://www.cryptonessie.org (**1999**).
3. Wikipedia. A5/1-wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/A5/1 (**2008**).
4. Wikipedia. RC4-wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/RC4 (**2008**).
5. Steve Babbage, Christophe De Canni_ere, Anne Canteaut, Carlos Cid, Henri Gilbert, Thomas Johansson, Matthew Parker,Bart Preneel,Vincent Rijmen, Matthew Robshaw, The stream Portfolio .http://www.ecrypt.eu.org/stream/ portfolio.pdf.
6. The eSTREAM Project, available at http://www.ecrypt. eu.org/stream/.
7. H. Wu, "A New Stream Cipher HC-256," In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 226244. Springer, Heidelberg (**2004**), http://eprint.iacr.org/2004/ 092.pdf.
8. G. Sekar, B. Preneel, "Improved Distinguishing Attacks on HC-256," International Workshop on Security (**2009**).
9. H. Wu, "The Stream Cipher HC-128," New Stream Cipher Designs (M.Robshaw and O Billet, eds. vol. 4986 of LNCS, pp.39-47, Springer-Verlag, (**2008**).
10. P. Sarkar, "On Approximating Addition by Exclusive OR," available at .http://eprint.iacr.org/2009/047.pdf.
11. Souradyuti Paul, Bart Preneel, "On the (In)security of Stream Ciphers Based on Arrays and Modular Addition", Cryptology ePrint Archive: Report 2005/448, IACR, (**2005**), Available online at .http://eprint.iacr.org/2005/448
12. M. Hell, T. Johansson, L. Brynielsson, "An overview of distinguishing attacks on stream ciphers", cryptography and communications, Vol.1, No.1, pp.71-94, Springer, (**2009**).
13. V. Rijmen, "Stream Ciphers and the eSTREAM Project", Isecure, Vol. 2, No. 1, (**2010**).
14. .A. Menezes, .P.Oorschot, .S. Vanstone, "Handbook of Applied Cryptography", (**1965**), CRC press.

Passive Defense In Infrastructure Networks Based on Cryptanalysis Stream Cipher Algorithms

A. R. Vizandan[1], A. R Mir Ghadri[1], J. Sheikh Zadegan[2]

Abstract

Communication networks as one of the country's critical infrastructures are considered as testbeds for various communication applications and its use is increasing rapidly.

Due to lack of security in communication networks along with the undeniable benefits, serious and irreparable challenges would be created in the country. Comprehensive understanding of network security problems and measures to overcome the major problems in this area, a significant role in the convention to promote security, safety and sustainability of infrastructure and communications will play a major step toward securing the objectives of passive defense in this regard. One of the most important types of stream ciphers are symmetric encryption algorithms that are appropriate to the specific features and in some applications such as network security and telecommunications infrastructure, the security assessment is an important consideration in the areas of passive defense and international project eSTREAM in line with increased activity in this branch of cryptography can play an important role. This article, cryptanalyzes one of the stream ciphers based on array rolling with distinguishing attack. The main idea of introducing a distinguisher on HC-256' . In this attack we need about 2^{556} linear equations involving binary keystream variables.

Keys Words: *Distinguishing Attack, Cryptanalysis, eSTREAM, Keystream, HC-256´*

1- Fath Research CenterImam Hossein Comprehensive University (Email: A_vizand@yahoo.com)
2- Signature Smart processing Research Center