تحلیل الگوریتم رمز جریانی HC-256′ بر اساس حملهٔ تمایز

احمدرضا ویزندان[1]، جواد شیخ زادگان[2] و عبدالرسول میرقدری[1]
[1] دانشگاه امام حسین علیه السلام، [2]پژوهشکدهٔ پردازش هوشمند علائم

چکیده:

رمزهای جریانی یکی از مهم‌ترین انواع الگوریتم‌های رمزنگاری متقارن است که به‌لحاظ قابلیت‌های ویژه و مناسب در برخی از کاربردهایی مانند امنیت شبکه‌ها و ارتباطات مخابراتی، ارزیابی امنیتی آن‌ها در حوزهٔ شبکه‌های ارتباطی از اهمّیت به‌سزایی برخوردارند و پروژهٔ بین‌المللی eSTREAM در راستای افزایش فعالیت در این شاخهٔ رمزنگاری نقش به‌سزایی ایفا نمود. در این مقاله یکی از الگوریتم‌های رمز جریانی پایه‌آرایه‌ای با استفاده از حملهٔ تمایز، مورد ارزیابی تحلیلی قرار می‌گیرد. درواقع ایدهٔ اصلی این مقاله در معرّفی دو دسته تمایزگر، تمایزگر پایه و بهبودیافته برروی الگوریتم رمز جریانی HC-256′ می‌باشد که در حملهٔ پایه، نیاز به $2^{56.3}$ رشته‌کلید خروجی می‌باشد، درحالی‌که در حملهٔ بهبودیافته نیاز به $2^{55.4}$ رشته‌کلید خروجی می‌باشد. بنابراین در حملهٔ بهبودیافته به‌طور تقریبی سی مرتبه کمتر از حملهٔ پایه، نیاز به رشته‌کلید خروجی از الگوریتم رمز داریم.

واژگان کلیدی: حملهٔ تمایز، تحلیل الگوریتم، eSTREAM، HC-256′

۱- مقدمه

امروزه با افزایش پیچیدگی شبکه‌های مخابراتی و رایانه‌ای، کنترل آسیب‌پذیری‌ها و تهدیدها یکی از مسایل جدّی می‌باشد. لذا یکی از راه‌های ایجاد امنیت اطّلاعات در برابر حمله و سایر خرابی‌ها، در سامانه‌های ارتباطی آتی، توسعهٔ جدّی در نظریه و مدل‌سازی ارزیابی‌های امنیتی سیستم‌های رمزکننده می‌باشد. *شانون* بنیان‌گذار نظریهٔ اطّلاعات، سیستم رمز امن کاملی را پیشنهاد کرد که در آن هر حرف از پیام اصلی با حرفی از یک دنبالهٔ کاملاً تصادفی ترکیب می‌شود و پیام رمزی را تشکیل می‌دهد *(Menezes, 1965)*. از آن‌جایی‌که این دنبالهٔ ترکیب‌شده با پیام اصلی، به‌طور کامل تصادفی است، امنیت سیستم را تضمین می‌کند. رمزهای جریانی که یکی از مهم‌ترین شیوه‌های رمزنگاری متقارن است، به‌طور منحصربه‌فرد تک‌تک حروف متن پیام آشکار را رمز می‌کند که این تبدیل، با زمان تغییر می‌کند؛ که در مقایسه با شیوهٔ دوم رمزهای متقارن، رمزهای قالبی، یک قالب از حروف متن پیام آشکار را رمز می‌کند و این تبدیل رمزنگاری، متغیر با زمان نمی‌باشد *(Menezes, 1965)*. به‌طور معمول رمزهای جریانی نسبت به رمزهای قالبی در کاربردهای سخت‌افزاری سریع‌تر بوده و پیچیدگی مدارات سخت‌افزاری کمتری دارند. هم‌چنین آن‌ها قابلیت‌های ویژه و مناسب‌تری در برخی از کاربردها (به‌طور نمونه در کاربردهای مخابراتی) مانند کاربردهایی که با حافظه‌های میانی[1] محدود و انتشار خطای محدود سرو کار دارند، از خود نشان می‌دهند *(Rijmen, 2010)*. شایان ذکر است، بعضی از رمزهای جریانی در کاربردهای نرم‌افزاری در مقایسه با رمزهای قالبی مانند AES، چهار الی پنج برابر سریع‌تر می‌باشند. در حال حاضر دانش نظری زیادی بر روی رمزهای جریانی وجود دارد و اصول طراحی و تحلیل‌های گوناگونی بر روی این‌گونه رمزها ارائه گردیده است. در این اواخر، پیشنهادهای زیادی برای رمزهای جریانی مشخّصی مشاهده گردید *(ECRYPT, 2008 and NESSIE, 1999)*. هم‌چنین بسیاری از رمزهای جریانی

[1] Buffer

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

اختصاصی شده، درعمل استفاده می‌گردد که برخی از آن‌ها در ابتدا به‌صورت محرمانه استفاده می‌شد که بعدها با انتشار آن‌ها ضعف‌هایی در ردّ آن‌ها مشاهده گردید، به‌عنوان مثال الگوریتم A5 مورد استفاده در شبکهٔ تلفن‌همراه GSM و RC4 *(Wikipedia, 2008)*. درکل می‌توان الگوریتم‌های رمز جریانی را به دو گروه رمزجریانی هم‌زمان[1] و غیرهم‌زمان[2] تقسیم کرد. در رمزهای جریانی هم‌زمان، مولد رشته‌کلید شبه‌تصادفی، غیر وابسته به متن‌رمزی و متن‌آشکار می‌باشد؛ درصورتی‌که در رمز جریانی غیرهم‌زمان رشته‌کلید رمز وابسته به متن‌رمزی و متن‌آشکار علاوه‌بر کلید اصلی می‌باشد. امروزه الگوریتم‌های رمزجریانی پیشرفته در چندین ساختار طراحی می‌گردند که یکی از آن‌ها استفاده از آرایه‌های بزرگ در حالت داخلی[3] می‌باشد که رشته‌کلید تولیدشده، تابعی از محتویات این آرایه است *(Paul and Preneel, 2005)*. گفتنی است که پس از اتمام فرآیند انتخاب استاندارد رمزگذاری پیشرفته (AES)، رمزهای قالبی کانون توجّه جامعهٔ رمزنگاری گردید. برخی از افراد بیان داشتند که آیا هنوز هم، برنامهٔ عملی برای رمزهای جریانی می‌تواند وجود داشته باشد؟ و یا دلیلی برای انجام تحقیقات روی این دسته از الگوریتم‌ها وجود دارد؟

شبکهٔ ECRYPT پروژهٔ چندمرحله‌ای eSTREAM را جهت شناسایی و بررسی رمزهای جریانی جدید در سال ۲۰۰۴ شروع کرد و در سال ۲۰۰۸ به اتمام رسانید. این پروژه در دو دستهٔ مجزا اجرا گردید: رمزهای جریانی جهت کاربردهای نرم‌افزاری با سرعت بالا و رمزهای جریانی جهت کاربردهای سخت‌افزاری با محدویت‌های منابع. در اوّلین مرحله از این پروژه در سال ۲۰۰۵، ۳۴ الگوریتم با توجّه به شرایط اعلام شده در مسابقه شرکت کردند *(Babbage, et al, 2005)*. یکی از الگوریتم‌های معرفی شده در مرحلهٔ نهایی پروژهٔ eSTREAM، الگوریتم رمز جریانی HC-128 می‌باشد که توسط *وو* ارائه گردید *(Wu, 2008)*. قابل توجّه است که وو، قبل از ارائه این الگوریتم، الگوریتم HC-256 *(Wu, 2004)* را در کنفرانس FSE در سال ۲۰۰۴ ارائه نمود و هم‌چنین شکل تغییر یافتهٔ الگوریتم HC-256، رمز جریانی HC-256′ را ارائه کرد *(Wu, 2004)* که در این مقاله مورد تحلیل قرار می‌گیرد. رمزهای خانوادهٔ HC از نوع رمزهای با انتقال کلمه‌به‌کلمه و مبتنی بر چرخش آرایه‌ها و جمع‌کننده‌ها بوده و در دسته‌بندی رمزهای جریانی از گروه رمزهای هم‌زمان محسوب می‌شوند. تحلیل‌های اولیهٔ طراح بر روی این رمز، نشان می‌دهد که به‌دست آوردن کلید در این رمز بسیار سخت، و لازمهٔ آن جستجوی کامل فضای کلید است. با توجّه به این‌که اغلب ساختار رمزهای جریانی از یک مولد رشته‌کلید شبه‌تصادفی برای تولید یک رشتهٔ طولانی از علائم دودویی استفاده می‌کنند، لذا امنیت یک رمز جریانی، به‌طور دقیق وابسته به چگونگی شباهت این رشتهٔ تولیدشده به یک رشتهٔ تصادفی واقعی می‌باشد. در حملهٔ تمایز، تحلیل‌گر تلاش می‌نماید که نشان دهد رفتار رشتهٔ تولیدشده، مانند یک رشتهٔ تصادفی واقعی نمی‌باشد *(Hell, et al, 2009)*. در بخش حملهٔ تمایز پایه بر روی این الگوریتم، نشان داده‌شده که این حمله نیاز به 2^{556} خروجی الگوریتم رمز دارد و درنهایت این مقدار در حملهٔ بهبودیافته کاهش می‌یابد.

پردازش الگوریتم رمز HC-256′ به‌صورت کلمه‌ای[4] است و طول هر کلمه ۳۲ بیت می‌باشد و الگوریتم از یک کلید اصلی و بردار حالت اولیه[5] به‌طول ۲۵۶ بیت برای تولید رشته‌کلید خروجی بهره می‌جوید. از آن‌جایی‌که این الگوریتم به شکل ممتازی طراحی گردیده و تاکنون مورد تحلیل قرار نگرفته است، بنابراین دستیابی به نتایج تحلیل، جذاب به‌نظر می‌رسد. درواقع هدف از این مقاله ارزیابی تحلیلی الگوریتم رمز HC-256′ مبتنی برحملهٔ تمایز می‌باشد که در ادامه به‌دقت یک گروه از تمایزگرها مورد بررسی قرار خواهد گرفت که نیاز به آزمایش تقریباً $2^{551.6}$ معادلهٔ خطی دارد. هر کدام از اجزای این معادلات شامل خروجی رشته کلید شبه تصادفی الگوریتم می‌باشد که با توجّه به این موضوع، تعداد رشته‌کلیدهای مورد نیاز از خروجی الگوریتم در این حمله برابر $2^{555.4}$ خواهد شد و نسبت به حملهٔ پایه به‌طور تقریبی سی برابر بهبودی مشاهده می‌گردد.

این مقاله از بخش‌هایی به شرح ذیل تشکیل شده است. در بخش بعدی نمادهایمان را تعریف می‌کنیم. در بخش سه به‌طور مختصر الگوریتم HC-256′ معرفی می‌گردد. حملهٔ تمایز پایه در بخش چهار ارائه می‌گردد. در بخش پنج حملهٔ تمایز بهبودیافته بیان شده و آخرین بخش به نتیجه‌گیری اختصاص پیدا خواهد کرد.

[1] Synchronous
[2] Asynchronous
[3] Internal state
[4] word-oriented
[5] Initial Value

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

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]

$x \gg y$: 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 $0 \le x < 2^{32}$

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

s_i: رشته‌کلید تولیدشده در مرحلهٔ i (یعنی $i + 1$اُمین تکرار از مولد رشته‌کلید شبه‌تصادفی[2])

- عبارت‌های s_i^j، $(h_1(x))^j$، $(h_2(x))^j$ و $(Q[x])^j$ به‌ترتیب بیان‌گر j اُمین بیت از عبارت‌های s_i، $h_1(x)$، $h_2(x)$، $Q[x]$ می‌باشند.
- اگر x یک کلمه باشد، سپس $x^{(i)}$ عبارت است از i اُمین بایت از x که $x^{(0)}$ بایت پایین رتبه و $x^{(3)}$ بایت بالا رتبه خواهد بود.
- P و Q به‌عنوان جعبه‌های جانشینی[3] در الگوریتم HC-256′ به‌کار می‌رود که هر کدام از جدول‌ها، ۱۰۲۴ عنصر ۳۲ بیتی دارند.
- K بیان‌گر یک کلید به اندازهٔ ۲۵۶ بیت می‌باشد.
- IV بیان‌گر یک مقدار اولیه به اندازهٔ ۲۵۶ بیت می‌باشد.

۳-تشریح الگوریتم HC-256′

در این بخش الگوریتم HC-256′ به‌طور خلاصه معرفی می‌گردد. این الگوریتم از یک کلید، K و یک مقدار اوّلیه، IV به اندازهٔ ۲۵۶ بیت استفاده می‌نماید.

می‌توان $IV = IV_0 \| \cdots \| IV_7, K = K_0 \| \cdots \| K_7$ تعریف نمود که هر کدام از K_i و IV_i $(i = 0, \ldots, 7)$ به اندازهٔ ۳۲ بیت می‌باشند. حالت داخلی[4] الگوریتم HC-256′ از دو جدول P و Q تشکیل شده که هر کدام دارای ۱۰۲۴ عنصر ۳۲ بیتی می‌باشند *(Wu, 2004)*.

۳-۱- فرآیند پیش محاسبات: الگوریتم برپایی و اجرای کلید و مقداردهی اوّلیه

- آرایهٔ $R[0, \ldots, 2559]$ توسط توسیع K و IV به‌صورت ذیل به‌دست می آید:

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

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 آرایه R به‌صورت زیر به‌روز رسانی می‌گردند:

$$P[i] = R_{i+512} \quad \text{for } 0 \le i \le 1023$$
$$Q[i] = R_{i+1536} \quad \text{for } 0 \le i \le 1023$$

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

$$P[i] = P[i] + P[i \boxminus 10] + g_1(P[i \boxminus 3], P[i \boxminus 1023]) \oplus h_1(P[i \boxminus 12]) \text{ for } 0 \le i \le 1023$$

$$Q[i] = Q[i] + Q[i \boxminus 10] + g_2(Q[i \boxminus 3], Q[i \boxminus 1023]) \oplus h_2(P[i \boxminus 12]) \text{ for } 0 \le i \le 1023$$

در اینجا توابع g_1 و g_2 به‌شکل زیر تعریف می‌گردند:

$$g_1(x, y) = \big((x \ggg 10) \oplus (y \ggg 23)\big) + Q[(x \oplus y) \bmod 1024]$$
$$g_2(x, y) = \big((x \ggg 10) \oplus (y \ggg 23)\big) + P[(x \oplus y) \bmod 1024]$$

بعد از مراحل بالا، فرآیند پیش‌پردازش و مقداردهی اوّلیه به اتمام رسیده و الگوریتم رمز جهت تولید رشته‌کلید رمز آماده می‌باشد.

۳-۲- الگوریتم مولد رشته کلید

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

```
i = 0
Repeat until (enough keystream bits are generated)
{
```

[1] Concatenation
[2] PRGB: pseudo random bit generation
[3] S-BOX
[4] Internal State

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

$j = i \bmod 1024;$

$P[j] = P[j] + P[j \boxminus 10] + g_1(P[j \boxminus 3], P[j \boxminus 1023])$;

$s_{2i} = h_1(P[j \boxminus 12]) \oplus P[j];$

$Q[j] = Q[j] + Q[j \boxminus 10] + g_2(Q[j \boxminus 3], Q[j \boxminus 1023]);$

$s_{2i+1} = h_2(Q[j \boxminus 12]) \oplus Q[j];$

$i = i + 1$;each increment of i corresponds to 2 steps.

}

در روابط بالا h_1 و h_2 به‌صورت زیر بیان می‌گردند.

$$h_1(x) = Q[x^{(0)}] + Q[256 + x^{(1)}] + Q[512 + x^{(2)}] + Q[768 + x^{(3)}]$$

$$h_2(x) = P[x^{(0)}] + P[256 + x^{(1)}] + P[512 + x^{(2)}] + P[768 + x^{(3)}]$$

۴- حملهٔ تمایز پایه

در این بخش حملهٔ تمایز پایه به الگوریتم رمز جریانی HC-256′ مورد بررسی قرار می‌گیرد با این فرض که فرآیند پیش پردازش و مقداردهی اوّلیه کامل[1] باشد. درواقع در این مرحله شیوهٔ کار نزدیک به کار وو می‌باشد *(Wu, 2004, 2008 and Hell, et al, 2009)*. در i اُمین مرحله، جدول‌های P وQ به‌صورت زیر به‌روزرسانی می‌شوند:

$$P[i] = P[i] + P[i \boxminus 10] + g_1(P[i \boxminus 3], P[i \boxminus 1023])$$

$$Q[i] = Q[i] + Q[i \boxminus 10] + g_2(Q[i \boxminus 3], Q[i \boxminus 1023])$$

و هم‌چنین

$$s_{2i} = h_1(P[i \boxminus 12]) \oplus P[i \bmod 1024]$$
$$s_{2i+1} = h_2(Q[i \boxminus 12]) \oplus Q[i \bmod 1024]$$

اگر $i < 1024 \leq 10$ و با استفاده از این حقیقت $P[i \bmod 1024] = s_{2i} \oplus h_1(z_i)$ و $Q[i \bmod 1024] = s_{2i+1} \oplus h_2(Y_i)$ ، تابع بازخور را می‌توان به‌صورت زیر نوشت:

$$s_{2i} \oplus h_1(z_i) = \left(s_{2(i-2048)} \oplus h_1'(z_{i-2048})\right) + \left(s_{2(i-10)} \oplus h_1(z_{i-10})\right) + g_1\left(s_{2(i-3)} \oplus h_1(z_{i-3}), s_{2(i-2047)} \oplus h_1'(z_{i-2047})\right) \quad (۱)$$

و

$$s_{2i+1} \oplus h_2(Y_i) = \left(s_{2(i-2048)+1} \oplus h_2'(Y_{i-2048})\right) + \left(s_{2(i-10)+1} \oplus h_2(Y_{i-10})\right) + g_2\left(s_{2(i-3)+1} \oplus h_2(z_{i-3}), s_{2(i-2047)+1} \oplus h_2'(Y_{i-2047})\right) \quad (۲)$$

شایان ذکر است که توابع $h_1(x)$، $h_1'(x)$ و $h_2(x)$ و $h_2'(x)$ به‌جهت این‌که از جعبه‌های جانشنی متفاوتی هستند، باهم فرق دارند. در این‌جا z_i و Y_i به‌ترتیب $P[i \boxminus 12]$ و $Q[i \boxminus 12]$ در i اُمین مرحله می‌باشند. همان‌طوری‌که مشاهده می‌گردد، عملگرهای + و $\oplus$ در بیت پایین‌رتبه[2] مانند هم عمل می‌نمایند؛ بنابراین معادله‌های (۱) و (۲) را می‌توان به شکل زیر مجدد نوشت.

$$s_{2i}^0 \oplus s_{2(i-2048)}^0 \oplus s_{2(i-10)}^0 \oplus s_{2(i-3)}^{10} \oplus s_{2(i-2047)}^{23} = (h_1(z_i))^0 \oplus (h_1'(z_{i-2048}))^0 \oplus (h_1(z_{i-10}))^0 \oplus (h_1(z_{i-3}))^{10} \oplus (h_1'(z_{i-2047}))^{23} \oplus (Q[r_i])^0 \quad (۳)$$

و

$$s_{2i+1}^0 \oplus s_{2(i-2048)+1}^0 \oplus s_{2(i-10)+1}^0 \oplus s_{2(i-3)+1}^{10} \oplus s_{2(i-2047)+1}^{23} = (h_2(Y_i))^0 \oplus (h_2'(Y_{i-2048}))^0 \oplus (h_2(Y_{i-10}))^0 \oplus (h_2(Y_{i-3}))^{10} \oplus (h_2'(Y_{i-2047}))^{23} \oplus (P[s_i])^0 \quad (۴)$$

که

$$r_i = s_{2(i-3)} \oplus h_1(z_{i-3}) \oplus s_{2(i-2047)} \oplus h_1'(z_{i-2047})$$
$$s_i = s_{2(i-3)+1} \oplus h_2(z_{i-3}) \oplus s_{2(i-2047)+1} \oplus h_2'(Y_{i-2047})$$

از طرفی هنگامی‌که $10 \leq i \leq 1024$ باشد الگوریتم مولد رشته‌کلید، به‌طور دائم جدول‌هایP و Q را به‌روزرسانی می‌نماید؛ لذا می‌توان تابع بازخور را به‌صورت زیر نمایش داد.

$$s_{2i}^0 \oplus s_{2(i-2048)}^0 \oplus s_{2(i-10)}^0 \oplus s_{2(i-3)}^{10} \oplus s_{2(i-2047)}^{23} \oplus s_{2i+1}^0 \oplus s_{2(i-2048)+1}^0 \oplus s_{2(i-10)+1}^0 \oplus s_{2(i-3)+1}^{10} \oplus s_{2(i-2047)+1}^{23} = (h_1(z_i))^0 \oplus (h_1'(z_{i-2048}))^0 \oplus (h_1(z_{i-10}))^0 \oplus (h_1(z_{i-3}))^{10} \oplus (h_1'(z_{i-2047}))^{23} \oplus (Q[r_i])^0 \oplus (h_2(Y_i))^0 \oplus (h_2'(Y_{i-2048}))^0 \oplus (h_2(Y_{i-10}))^0 \oplus (h_2(Y_{i-3}))^{10} \oplus (h_2'(Y_{i-2047}))^{23} \oplus (P[s_i])^0 \quad (۵)$$

[1] Perfect

[2] Least Significant Bit (LSB)

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

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

برای $2048 \times \alpha + 10 \le i,j < 2048 \times \alpha + 1023$ که α عضو مجموعهٔ اعداد طبیعی می‌باشد و $j \neq i$ می‌توان معادلهٔ (۵) را به‌صورت زیر نوشت:

$$\begin{aligned} & s^0_{2j} \oplus s^0_{2(j-2048)} \oplus s^0_{2(j-10)} \oplus s^{10}_{2(j-3)} \oplus \\ & s^{23}_{2(j-2047)} s^0_{2i+1} \oplus s^0_{2(j-2048)+1} \oplus s^0_{2(j-10)+1} \oplus \\ & s^{10}_{2(j-3)+1} \oplus s^{23}_{2(j-2047)+1} \\ & = (h_1(z_j))^0 \oplus (h_1'(z_{j-2048}))^0 \oplus (h_1(z_{j-10}))^0 \oplus \\ & (h_1(z_{j-3}))^{10} \oplus (h_1'(z_{j-2047}))^{23} \oplus (Q[r_j])^0 \oplus \\ & (h_2(Y_j))^0 \oplus (h_2'(Y_{j-2048}))^0 \oplus (h_2(Y_{j-10}))^0 \oplus \\ & (h_2(Y_{j-3}))^{10} \oplus (h_2'(Y_{j-2047}))^{23} \oplus (P[s_j])^0 \end{aligned} \quad (۶)$$

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

$$\begin{aligned} & s^0_{2i} \oplus s^0_{2(i-2048)} \oplus s^0_{2(i-10)} \oplus s^{10}_{2(i-3)} \oplus \\ & s^{23}_{2(i-2047)} s^0_{2i+1} \oplus s^0_{2(i-2048)+1} \oplus s^0_{2(i-10)+1} \oplus \\ & s^{10}_{2(i-3)+1} \oplus s^{23}_{2(i-2047)+1} = s^0_{2j} \oplus s^0_{2(j-2048)} \oplus \\ & s^0_{2(j-10)} \oplus s^{10}_{2(j-3)} \oplus s^{23}_{2(j-2047)} s^0_{2i+1} \oplus \\ & s^0_{2(j-2048)+1} \oplus s^0_{2(j-10)+1} \oplus s^{10}_{2(j-3)+1} \oplus \\ & s^{23}_{2(j-2047)+1} \end{aligned} \quad (۷)$$

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

$$\begin{aligned} & (h_1(z_i))^0 \oplus (h_1'(z_{i-2048}))^0 \oplus (h_1(z_{i-10}))^0 \oplus \\ & (h_1(z_{i-3}))^{10} \oplus (h_1'(z_{i-2047}))^{23} \oplus (Q[r_i])^0 \oplus \\ & (h_2(Y_i))^0 \oplus (h_2'(Y_{i-2048}))^0 \oplus (h_2(Y_{i-10}))^0 \oplus \\ & (h_2(Y_{i-3}))^{10} \oplus (h_2'(Y_{i-2047}))^{23} \oplus (P[s_i])^0 \\ & = (h_1(z_j))^0 \oplus (h_1'(z_{j-2048}))^0 \oplus (h_1(z_{j-10}))^0 \oplus \\ & (h_1(z_{j-3}))^{10} \oplus (h_1'(z_{j-2047}))^{23} \oplus (Q[r_j])^0 \oplus \\ & (h_2(Y_j))^0 \oplus (h_2'(Y_{j-2048}))^0 \oplus (h_2(Y_{j-10}))^0 \oplus \\ & (h_2(Y_{j-3}))^{10} \oplus (h_2'(Y_{j-2047}))^{23} \oplus (P[s_j])^0. \end{aligned} \quad (۸)$$

یادآوری می‌گردد:

$$\begin{aligned} z_i &= z_{i-2048} + z_{i-10} + g_1(z_{i-3}, z_{i-2047}) \\ z_j &= z_{j-2048} + z_{j-10} + g_1(z_{j-3}, z_{j-2047}) \\ Y_j &= Y_{j-2048} + Y_{j-10} + g_2(Y_{j-3}, Y_{j-2047}) \\ Y_i &= Y_{i-2048} + Y_{i-10} + g_2(Y_{i-3}, Y_{i-2047}) \end{aligned} \quad (۹)$$

می‌توان معادلهٔ (۸) را به‌صورت زیر تقریب زد:

$$H(x_1) = H(x_2) \quad (1)$$

H یک جعبهٔ جانشنی ۲۷۶ بیت به یک بیت می‌باشد. x_1 و x_2 دو ورودی تصادفی ۲۷۶ بیتی می‌باشد.

$$x_1 = z_{i-3} \| z_{i-10} \| z_{i-2047} \| z_{i-2048} \| Y_{i-3} \| Y_{i-10} \| Y_{i-2047} \| Y_{i-2048} \| r_i \| s_i$$

$$x_2 = z_{j-3} \| z_{j-10} \| z_{j-2047} \| z_{j-2048} \| Y_{j-3} \| Y_{j-10} \| Y_{j-2047} \| Y_{j-2048} \| r_j \| s_j$$

برای به‌دست آوردن نرخ تصادم[1] می‌توان از قضیهٔ زیر استفاده نمود:

قضیه: اگر s یک جعبهٔ 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 $2^{-m} + 2^{-n} - 2^{-m-n}$ می‌باشد *(Wu, 2008)*.

اثبات. اگر $x_1 = x_2$، دراین‌صورت $s(x_1) = s(x_2)$ می‌گردد. اگر $x_1 \neq x_2$ دراین‌صورت $s(x_1) = s(x_2)$ با احتمال 2^{-n} . چون $x_1 = x_2$ با احتمال 2^{-m} می‌باشد و $x_1 \neq x_2$ با احتمال $1 - 2^{-m}$ صورت می‌پذیرد احتمال $s(x_1) = s(x_2)$ برابر است با $2^{-m} + (1 - 2^{-m})2^{-n}$. □

با استفاده از این قضیه، احتمال معادلهٔ (۱۰) برابر $\frac{1}{2} + 2^{-277}$ می‌باشد. بنابراین احتمال معادلهٔ (۷) هم با همین احتمال یعنی $\frac{1}{2} + 2^{-277}$ می‌باشد.

حال تمام اجزا، به‌منظور بناکردن حملهٔ 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[2] ۲۰ خروجی از معادلهٔ (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′ به‌دست می‌آید.

یادآوری می‌گردد که از نتایج بالا داریم $p = \frac{1}{2} + 2^{-277}$ و $p' = \frac{1}{2}$ و هنگامی‌که N بزرگ باشد هر دو توزیع دوجمله‌ای[3] را با توزیع نرمال می‌توان تقریب زد. حال اگر $|\mu - \mu'| > 2(\sigma + \sigma')$ یعنی برای $N > 2^{556}$،

[1] Collision Rate
[2] XOR
[3] Binomial

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

خروجی الگوریتم رمز را با احتمال ۰/۹۷۷۲ می‌توان از یک رشتهٔ تمام تصادفی تمایز داده شود.

۵- حملهٔ تمایز بهبود یافته

در این بخش حملهٔ تمایز بهبودیافته به الگوریتم رمز جریانی HC-256′ ارائه خواهد شد. نشان خواهیم داد که معادلهٔ (۷) وقتی‌که $i = j + 10$ می‌تواند با احتمال بیشتری نگه داشته شود.

۵-۱- فرآیند بهبود حمله

در این بخش ارزیابی تحلیلی الگوریتم، شبیه به حملهٔ پایه می‌باشد که در بخش قبل مورد بررسی قرار گرفت.

با شرط $2048 \times \alpha + 10 \le i,j < 2048 \times \alpha +$ 1023 و $i = j + 10$ معادله‌های (۷) و (۸) را می‌توان به شکل زیر کاهش داد:

$$\begin{aligned} & s^{0}_{2(j+10)} \oplus s^{0}_{2(j-2038)} \oplus s^{10}_{2(j+7)} \oplus \\ & s^{23}_{2(j-2037)} \oplus s^{0}_{2(j+10)+1} \oplus s^{0}_{2(j-2038)+1} \oplus \\ & s^{10}_{2(j+7)+1} \oplus s^{23}_{2(j-2037)+1} = s^{0}_{2(j-2048)} \oplus \\ & s^{0}_{2(j-10)} \oplus s^{10}_{2(j-3)} \oplus s^{23}_{2(j-2047)} \oplus \\ & s^{0}_{2(j-2048)+1} \oplus s^{0}_{2(j-10)+1} \oplus s^{10}_{2(j-3)+1} \oplus \\ & s^{23}_{2(j-2047)+1} \end{aligned} \qquad (۱۱)$$

بنابراین

$$\begin{aligned} & (h_1(z_{j+10}))^0 \oplus (h_1'(z_{j-2038}))^0 \oplus \\ & (h_1(z_{j+7}))^{10} \oplus (h_1'(z_{j-2037}))^{23} \oplus \\ & (Q[r_{j+10}])^0 \oplus (h_2(Y_{j+10}))^0 \oplus \\ & (h_2'(Y_{j-2038}))^0 \oplus (h_2(Y_{j+7}))^{10} \oplus \\ & (h_2'(Y_{j-2037}))^{23} \oplus (P[s_{j+10}])^0 \\ = & (h_1'(z_{j-2048}))^0 \oplus (h_1(z_{j-10}))^0 \oplus (h_1(z_{j-3}))^{10} \oplus \\ & (h_1'(z_{j-2047}))^{23} \oplus (Q[r_j])^0 \oplus (h_2'(Y_{j-2048}))^0 \oplus \\ & (h_2(Y_{j-10}))^0 \oplus (h_2(Y_{j-3}))^{10} \oplus (h_2'(Y_{j-2047}))^{23} \oplus \\ & (P[s_j])^0 \end{aligned} \qquad (۱۲)$$

فرض کنید هنگامی‌که معادلهٔ (۱۲) برقرار باشد، بیان می‌داریم که رویداد L اتفاق افتاده است. با فرض این‌که الگوریتم پیش‌پردازش و مقداردهی اوّلیه کامل باشد می‌توانیم وضعیت‌های متفاوتی را به‌صورت زیر در نظر بگیریم:

۵-۲- وضعیت یک

فرض کنید E رویداد

$$z_{j-2038} \parallel z_{j+7} \parallel z_{j-2037} = z_{j-2048} \parallel z_{j-3} \parallel z_{j-2047}$$

$$Y_{j-2038} \parallel Y_{j+7} \parallel Y_{j-2037} = Y_{j-2048} \parallel Y_{j-3} \parallel Y_{j-2047}$$

باشد. در این‌جا z و Y دو متغیر تصادفی با توزیع یکنواخت می‌باشند. حال وقتی رویداد E با احتمال $\Pr[E] = 2^{-96}.2^{-96} = 2^{-192}$ اتفاق بی‌افتد، معادلهٔ (۱۲) به معادلهٔ (۱۳) کاهش پیدا می‌نماید.

$$\begin{aligned} & (h_1(z_{j+10}))^0 \oplus (Q[r_{j+10}])^0 \\ & \oplus (h_2(Y_{j+10}))^0 \oplus (P[s_{j+10}])^0 \\ & = (h_1(z_{j-10}))^0 \oplus (Q[r_j])^0 \oplus (h_2(Y_{j-10}))^0 \\ & \oplus (P[s_j])^0 \end{aligned} \qquad (۱۳)$$

یادآوری می‌گردد:

$$(h_1(z_{j+10}))^0 = (Q[z^{(0)}_{(j+10)}])^0 \oplus (Q[256 + z^{(1)}_{(j+10)}])^0 \oplus (Q[512 + z^{(2)}_{(j+10)}])^0 \oplus (Q[768 + z^{(3)}_{(j+10)}])^0$$

$$(h_1(z_{j-10}))^0 = (Q[z^{(0)}_{(j-10)}])^0 \oplus (Q[256 + z^{(1)}_{(j-10)}])^0 \oplus (Q[512 + z^{(2)}_{(j-10)}])^0 \oplus (Q[768 + z^{(3)}_{(j-10)}])^0$$

$$(h_2(Y_{j+10}))^0 = (P[Y^{(0)}_{(j+10)}])^0 \oplus (P[256 + Y^{(1)}_{(j+10)}])^0 \oplus (P[512 + Y^{(2)}_{(j+10)}])^0 \oplus (P[768 + Y^{(3)}_{(j+10)}])^0$$

$$(h_2(Y_{j-10}))^0 = (P[Y^{(0)}_{(j-10)}])^0 \oplus (P[256 + Y^{(1)}_{(j-10)}])^0 \oplus (P[512 + Y^{(2)}_{(j-10)}])^0 \oplus (P[768 + Y^{(3)}_{(j-10)}])^0$$

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

هرکدام از اعداد صحیح ۳۲ بیتی z و Y را می‌توان به‌صورت چهار بایتی نمایش داد یعنی $z = z^{(3)} \| z^{(2)} \| z^{(1)} \| z^{(0)}$ و $Y = Y^{(3)} \| Y^{(2)} \| Y^{(1)} \| Y^{(0)}$. در این‌جا $z^{(0)}$ ، $Y^{(0)}$ و $z^{(3)}$، $Y^{(3)}$ به‌ترتیب بایت‌های پایین رتبه و بالارتبه z و Y می‌باشند. هم‌چنین رویداد F را می‌توان به‌صورت زیر تعریف نمود:

$$z^{(2)}_{(j+10)} \| z^{(1)}_{(j+10)} \| z^{(0)}_{(j+10)} = z^{(2)}_{(j-10)} \| z^{(1)}_{(j-10)} \| z^{(0)}_{(j-10)}$$
$$Y^{(2)}_{(j+10)} \| Y^{(1)}_{(j+10)} \| Y^{(0)}_{(j+10)} = Y^{(2)}_{(j-10)} \| Y^{(1)}_{(j-10)} \| Y^{(0)}_{(j-10)}$$

حال با داشتن معادلهٔ (۹) روابط زیر را خواهیم داشت:

$$\begin{aligned} Y_{j+10} &= Y_{j-2038} + Y_j + g_2\left(Y_{j+7}, Y_{j-2037}\right) \\ z_{j+10} &= z_{j-2038} + z_j + g_1\left(z_{j+7}, z_{j-2037}\right) \end{aligned} \qquad (۱۴)$$

ملاحظات:

به‌فرض رویداد E رخ دهد، با توجّه به معادله‌های (۹) و (۱۴)، شکل عبارات z_{j+10}، z_{j-10}، Y_{j+10} و Y_{j-10} به‌صورت زیر خواهد بود:

$$z_{j-10} = -A + B - ,z_{j+10} = A + B + C \bmod 2^{32}$$

$$Y_{j+10} = D + E + F \bmod 2^{32} ,C \bmod 2^{32}$$

و $Y_{j-10} = -D + E - F \bmod 2^{32}$. بیت‌های پایین رتبه از عبارات z_{j+10} و Y_{j+10} به‌ترتیب با بیت‌های پایین‌رتبه از عبارات z_{j-10} و Y_{j-10} برابر می‌باشند (*Sekar and Preneel, 2009*). درنتیجه داریم: $\Pr[F|E] = 2^{-23}.2^{-23} = 2^{-46}$.

به‌علاوه بیت‌های بالا رتبه از عبارات z_{j+10} و Y_{j+10} به‌ترتیب با بیت‌های بالارتبه از عبارات z_{j-10} و Y_{j-10} برابر می‌باشند، اگر $Y_{j-10} = Y_{j+10}$ که این مورد با احتمال $2^{-31}.2^{-31} = 2^{-62}$ رخ خواهد داد. به بیان دیگر:

$$\Pr\left[z_{j+10} = z_{j-10}, Y_{j+10} = Y_{j-10} \middle| E\right] = 2^{-62}$$

بنابراین:

$$\Pr\left[z_{j+10} = z_{j-10} \middle| E\right] = 2^{-31} = \Pr\left[(z_{j+10})^{31} = (z_{j-10})^{31}\right],$$

$$\Pr\left[Y_{j+10} = Y_{j-10} \middle| E\right] = 2^{-31} = \Pr\left[(Y_{j+10})^{31} = (Y_{j-10})^{31}\right].$$

وقتی F رخ دهد، معادله (۱۳) به معادله زیر کاهش پیدا می‌کند.

$$\begin{aligned} &\left(Q\left[768 + z^{(3)}_{j+10}\right]\right)^0 \oplus \left(Q[r_{j+10}]\right)^0 \oplus \\ &\left(P\left[768 + Y^{(3)}_{j+10}\right]\right)^0 \oplus \left(P[s_{j+10}]\right)^0 = \\ &\left(Q\left[768 + z^{(3)}_{j-10}\right]\right)^0 \oplus \left(Q[r_j]\right)^0 \oplus \left(P\left[768 + Y^{(3)}_{j-10}\right]\right)^0 \oplus \left(P[s_j]\right)^0 \end{aligned} \qquad (۱۵)$$

حال اگر معادلهٔ (۱۶)

$$r_{j+10} \| r_j \| s_{j+10} \| s_j = 768 + z^{(3)}_{j+10} \| 768 + z^{(3)}_{j-10} \| 768 + Y^{(3)}_{j+10} \| 768 + Y^{(3)}_{j-10} \qquad (۱۶)$$

یا معادلهٔ (۱۷)

$$r_{j+10} \| r_j \| s_{j+10} \| s_j = 768 + z^{(3)}_{j-10} \| 768 + z^{(3)}_{j+10} \| 768 + Y^{(3)}_{j-10} \| 768 + Y^{(3)}_{j+10} \qquad (۱۷)$$

را داشته باشیم سپس معادلهٔ (۱۵) برقرار می‌باشد. از قبل اشاره کردیم که r_j و s_j دو متغیر ده‌بیتی می‌باشند؛ بنابراین احتمال عبارت $r_{j+10} \| r_j \| s_{j+10} \| s_j$ برابر است با 2^{-40}. از طرفی معادله‌های (۱۶) و (۱۷) به‌طور هم‌زمان رخ نمی‌دهند با فرض این‌که: $(z^{(3)}_{j+10})^7 \| (Y^{(3)}_{j+10})^7 \neq (z^{(3)}_{j-10})^7 \| (Y^{(3)}_{j-10})^7$، یعنی $z^{31}_{j+10} \| Y^{31}_{j+10} \neq z^{31}_{j-10} \| Y^{31}_{j-10}$ درنتیجه عبارت $768 + z^{31}_{j+10} \| 768 + Y^{31}_{j+10} \neq 768 + z^{31}_{j-10} \| 768 + Y^{31}_{j-10}$ را خواهیم داشت.

• رویداد L تحت شرایط زیر که از آن به‌عنوان S_1 یاد می‌گردد، رخ می‌دهد.

۱- شرط زیر با احتمال 2^{-192} برقرار است:

$$z_{j-2038} \| z_{j+7} \| z_{j-2037} \| Y_{j-2038} \| Y_{j+7} \| Y_{j-2037} = z_{j-2048} \| z_{j-3} \| z_{j-2047} \| Y_{j-2048} \| Y_{j-3} \| Y_{j-2047}$$

۲- دو عبارت زیر هر دو با احتمال $2^{-23}.2^{-23} = 2^{-46}$ و با در نظر گرفتن شرط یک برقرار است (از قسمت ملاحظات).

$$z^{(2)}_{(j+10)} \| z^{(1)}_{(j+10)} \| z^{(0)}_{(j+10)} = z^{(2)}_{(j-10)} \| z^{(1)}_{(j-10)} \| z^{(0)}_{(j-10)},$$
$$Y^{(2)}_{(j+10)} \| Y^{(1)}_{(j+10)} \| Y^{(0)}_{(j+10)} = Y^{(2)}_{(j-10)} \| Y^{(1)}_{(j-10)} \| Y^{(0)}_{(j-10)}.$$

۳- عبارت زیر با در نظر گرفتن شروط ۱ و ۲ دارای احتمال $1 - 2^{-16}$ می باشد (از قسمت ملاحظات).

$$(z^{(3)}_{j+10})^7 \| (Y^{(3)}_{j+10})^7 \neq (z^{(3)}_{j-10})^7 \| (Y^{(3)}_{j-10})^7$$ یعنی

$$z^{31}_{j+10} \| Y^{31}_{j+10} \neq z^{31}_{j-10} \| Y^{31}_{j-10}$$

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

۴- عبارت

$r_{j+10} \| r_j = 768 + z_{j+10}^{(3)} \| 768 + z_{j-10}^{(3)}$

(با احتمال 2^{-20}) یا

$r_{j+10} \| r_j = 768 + z_{j-10}^{(3)} \| 768 + z_{j+10}^{(3)}$

(با احتمال 2^{-20}) تحت شرط سه احتمال آن $2^{-20} + 2^{-20} = 2^{-19}$

هم‌چنین $s_{j+10} \| s_j = 768 + Y_{j-10}^{(3)} \| 768 + Y_{j+10}^{(3)}$

(با احتمال 2^{-20}) یا $s_{j+10} \| s_j = 768 + Y_{j+10}^{(3)} \| 768 +$

$Y_{j-10}^{(3)}$ (با احتمال 2^{-20}) تحت شرط ۳ احتمال آن $2^{-20} + 2^{-20} = 2^{-19}$

بنابراین احتمال کل $2^{-38} = 2^{-19}.2^{-19}$ خواهد بود.

در نتیجه داریم:

$$\Pr[S_1] = 2^{-192}.2^{-46}.(1-2^{-16}).2^{-38} \approx 2^{-276}.$$

۵-۳- وضعیت دو

شرایط زیر را با عنوان S_2 تعریف می‌نماییم:

۱- شرط زیر با احتمال 2^{-192} برقرار است:

$z_{j-2038} \| z_{j+7} \| z_{j-2037} \| Y_{j-2038} \| Y_{j+7} \| Y_{j-2037} =$

$z_{j-2048} \| z_{j-3} \| z_{j-2047} \| Y_{j-2048} \| Y_{j-3} \| Y_{j-2047}$

۲- دو عبارت زیر ($z_{j+10} = z_{j-10}$ و $Y_{j+10} = Y_{j-10}$) با احتمال $2^{-62} = 2^{-31}.2^{-31}$ و با در نظر گرفتن شرط یک برقرار است (از قسمت ملاحظات).

$z_{(j+10)}^{(3)} \| z_{(j+10)}^{(2)} \| z_{(j+10)}^{(1)} \| z_{(j+10)}^{(0)} = z_{(j-10)}^{(3)} \|$

$z_{(j-10)}^{(2)} \| z_{(j-10)}^{(1)} \| z_{(j-10)}^{(0)}$,

$Y_{(j+10)}^{(3)} \| Y_{(j+10)}^{(2)} \| Y_{(j+10)}^{(1)} \| Y_{(j+10)}^{(0)} = Y_{(j-10)}^{(3)} \|$

$Y_{(j-10)}^{(2)} \| Y_{(j-10)}^{(1)} \| Y_{(j-10)}^{(0)}$.

۳- عبارت $r_{j+10} \| s_{j+10} = r_j \| s_j$ با احتمال 2^{-20} با شروط بالا برقرار است. در نتیجه:

$$\Pr[S_2] = 2^{-192}.2^{-62}.2^{-20} = 2^{-274}$$

از شرط سوم S_1 و شرط دوم S_2 این‌چنین استنباط می‌گردد که S_1 و S_2 دو به دو ناسازگارند[1]. بنابراین

$\Pr[S_1 \cup S_2] = 2^{-276} + 2^{-274} \approx 2^{-273/7}.$

یادآوری می‌گردد بعضی از رویدادهای دیگری ممکن است منجر به وقوع L گردد؛ امّا احتمال وقوع آن‌ها کم‌تر از S_1 و S_2 می‌باشند؛ بنابراین این رویدادها مورد بررسی قرار نگرفته‌اند. وقتی هیچ‌کدام از رویدادهای ذکر شده رخ ندهد، یعنی $(S_1 \cup S_2)^c$ ، معادلهٔ (۸) و به دنبال آن معادلهٔ (۷) با احتمال یکنواخت و به مقدار $\frac{1}{2}$ با شرط این‌که الگوریتم پیش‌پردازش و مقداردهی اوّلیه کامل باشد، منعقد می‌باشد. در نتیجه با به‌کار بردن قانون بیز نتیجهٔ زیر را خواهیم داشت: یادآوری می‌گردد:

$$\Pr[L] = \Pr[L|(S_1 \cup S_2)].\Pr[S_1 \cup S_2]$$

$$+\Pr[L|(S_1 \cup S_2)^c].\Pr[(S_1 \cup S_2)^c] =$$

$$1.2^{-273.7} + 0.5.(1-2^{-273.7}) = \frac{1}{2} + 2^{-274/7}$$

که اگر الگوریتم رمز′HC-256 یک رمز ایده‌آل باشد، احتمال وقوع رشته‌کلیدهای خروجی رمز $\frac{1}{2}$ می‌باشد درحالی‌که در حملهٔ تمایز پایه این مقدار $\frac{1}{2} + 2^{-277}$ نشان داده شد. امّا در این بخش اریبی با ضریب $2^{2.3}$ بهبود داده‌شد. حال با توجه به نتایج بالا می‌توانیم تمایزگر را بر روی الگوریتم ′HC-256 بنا نماییم. درواقع این فرآیند مانند بخش قبلی می‌باشد، به‌طوری‌که در این حالت $p = \frac{1}{2} + 2^{-274/7}$ و $p' = \frac{1}{2}$ می‌باشد (ر.ج، آخرین پاراگراف بخش قبل). حال می‌توان رشته‌کلیدهای مورد نیاز حملهٔ تمایز پایه با حملهٔ تمایز بهبود داده‌شده به‌صورت زیر مورد مقایسه قرار گیرد. شایان ذکر است که معادلهٔ (۷) و (۱۱) به‌ترتیب دارای ۲۰ و ۱۶ رشته کلید خروجی می‌باشند. بنابراین در تمایزگر بهبود داده‌شده نیاز به $16.2^{551/4} = 2^{555/4}$ رشته‌کلید خروجی داریم؛ درحالی‌که در تمایزگر پایه نیاز به $20.2^{556} = 2^{560/3}$ رشته‌کلید خروجی است. بنابراین حملهٔ تمایز بهبودیافته به‌طور تقریبی $2^{4/9} = 30$ برابر کم‌تر، نیاز به رشته‌کلید خروجی دارد.

۶- نتیجه گیری

در این مقاله حملهٔ تمایز پایه به الگوریتم رمزجریانی ′HC-256 ارائه گردید. در ادامهٔ این حمله، احتمال اریبی و داده‌های مورد نیاز برای حملهٔ تمایز بر روی الگوریتم مورد اشاره بهبود داده شد. در این ارزیابی تحلیلی، احتمال اریبی را 2^{-277} به‌دست آوردیم و هم‌چنین با این احتمال تعداد خروجی‌های مورد نیاز برای حمله، یعنی $2^{560/3}$ محاسبه گردید. در حملهٔ بهبودیافته احتمال اریبی بر روی خروجی‌های این الگوریتم $2^{-274/7}$ به‌دست آمد و برای اجرای حمله نیاز به $2^{555/4}$ رشته کلید خروجی می‌باشد. بنابراین قادر به بهبود داده‌های مورد نیاز حملهٔ تمایز به اندازهٔ سی برابر بوده‌ایم.

[1] Mutually exclusive

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]

۷- مراجع

ECRYPT. eSTREAM: ECRYPT Stream Cipher Project, IST-2002-507932. http://www.ecrypt.eu.org/stream/ (2008)

NESSIE. New European Schemes for Signatures, Integrity, and Encryption. http://www.cryptonessie.org (1999)

Wikipedia. A5/1-wikipedia, the free encyclopedia, 2008. http://en.wikipedia.org/wiki/A5/1

Wikipedia. RC4-wikipedia, the free encyclopedia, 2008.http://en.wikipedia.org/wiki/RC4

Babbage S., Christophe De Canni_ere, Anne Canteaut, Carlos Cid, Henri Gilbert, Thomas Johansson, Matthew Parker,Bart Preneel,Vincent Rijmen, Matthew Robshaw, 2005, The stream Portfoliohttp://www.ecrypt.eu.org/stream.

The eSTREAM Project, available at http://www.ecrypt.eu.org/stream/.

Wu, H., 2004. A New Stream Cipher HC-256, In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 226-244.

Sekar, G., Preneel, B., 2009. Improved Distinguishing Attacks on HC-256, International Workshop on Security.

Wu, H., 2008. The Stream Cipher HC-128, New Stream Cipher Designs (M. Robshaw and O Billet, eds. vol. 4986 of LNCS, pp.39-47.

Sarkar, P., 2004. On Approximating Addition by Exclusive OR, available at http://eprint.iacr.org/2009

Paul, S., Preneel, B., 2005. On the (In)security of Stream Ciphers Based on Arrays and Modular Addition, Cryptology ePrint Archive: Report 2005/448, Available online at http://eprint.iacr.org

Hell, M., Johansson, T., Brynielsson, L., 2009. An overview of distinguishing attacks on stream ciphers, cryptography and communications, vol.1, No.1, pp.71-94, Springer.

Rijmen, V., 2010. Stream Ciphers and the eSTREAM Project, Isecure, Vol.2, No.1,

Menezes, A ., Oorschot, P., Vanstone, S., 1965. Handbook of Applied Cryptography, CRC press.



احمدرضا ویزندان مدرک کارشناسی خود را در رشتهٔ مهندسی مخابرات در سال ۱۳۷۹ از دانشگاه امام‌حسین علیه‌السلام و مدرک کارشناسی ارشد در رشتهٔ مهندسی مخابرات را در سال۱۳۸۴ از دانشگاه آزاد اسلامی واحد تهران جنوب اخذ نموده است و در حال حاضر دانشجوی دکتری رشتهٔ رمز می‌باشد. زمینهٔ مورد علاقه وی رمزنگاری، پنهان‌نگاری و تئوری کدینگ می‌باشد.

نشانی پست الکترونیک ایشان عبارت است از:

a_vizand@yahoo.com



جواد شیخ‌زادگان مدرک کارشناسی مهندسی مخابرات را در خردادماه سال ۱۳۶۳ از دانشکدهٔ مهندسی برق دانشگاه شریف اخذ کرد و بلافاصله در اولین دورهٔ کارشناسی ارشد مخابرات در دانشگاه خواجه نصیرالدین طوسی پذیرفته شد و در تیرماه ۱۳۶۷ این دوره را به پایان رساند. ایشان پس از سپری کردن دورهٔ طرح سربازی خود در مهرماه سال ۱۳۶۹ در اولین دورهٔ دکتری مخابرات در دانشگاه تربیت مدرس پذیرفته شد و در سال ۱۳۷۴ از این رشته فارغ‌التحصیل شد. نامبرده علاوه بر تدریس در دانشگاه‌های مختلف تهران از اواسط سال ۱۳۶۶ تا حال حاضر، از اوایل سال ۱۳۷۰ فعّالیت‌های تحقیقاتی خود را در زمینهٔ پردازش سیگنال‌های صوتی شروع کرد و در سال ۱۳۷۳ موفق به تأسیس یک مؤسسهٔ تحقیقاتی گردید که از سال ۱۳۷۷ با مجوز وزارت علوم، تحقیقات و فناوری به نام پژوهشکدهٔ پردازش هوشمند علائم فعّالیت خود را ادامه می‌دهد. ایشان در حال حاضر عضو هیئت علمی، هیئت امنا و هیئت مدیرهٔ پژوهشکدهٔ مذکور، و نیز عضو هیئت مؤسس و شورای اجرایی انجمن رمز ایران از بدو تأسیس (سال ۱۳۷۹) تاکنون هستند. علاوه بر زمینه‌های تحقیقاتی اصلی وی که عبارتند از: بازشناسی گوینده، بازشناسی زبان و دادگان‌های گفتاری زبان‌فارسی، موضوعاتی از قبیل: رمزنگاری، پنهان‌نگاری و امنیّت اطّلاعات و ارتباطات نیز از زمینه‌های مورد علاقهٔ ایشان می‌باشند.

آدرس پست الکترونیکی نامبرده عبارت است از:

sheikhzadegan@rcisp.ac.ir

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]



عبدالرسول میرقدری مدرک کارشناسی، کارشناسی‌ارشد و دکتری خود را در رشتهٔ آمار به‌ترتیب درسال‌های ۱۳۶۵، ۱۳۶۸و۱۳۸۰ از دانشکدهٔ علوم دانشگاه شیراز اخذ نموده‌اند و در حال حاضر استادیار دانشکده و پژوهشکدهٔ فناوری اطّلاعات و ارتباطات دانشگاه جامع امام‌حسین (ع) می‌باشد. زمینهٔ مورد علاقهٔ وی رمزنگاری، آمار و فرآیندهای تصادفی می‌باشد.

نشانی پست الکترونیک ایشان عبارت است از:

Amrghdri@ ihu.ac.ir

[Downloaded from jsdp.rcisp.ac.ir on 2026-11-09]