باسمه تعالی

گزارش پروژه اول درس معماشناسی کاربردی

موضوع سمینار:

گسترش S-box سیستم رمز متقارن قطعه‌ای Twofish

نگارندگان:

سید محمد مهدی احمدپناه ۹۴۱۳۱۰۸۶

سید امیر حسین ناصرالدینی ۹۴۱۳۱۰۱۹

استاد:

دکتر بابک صادقیان

پاییز ۱۳۹۴

فهرست مطالب

معرفی و کلیات

در دانش معماشناسی، توابع رمزنگاری قطعه‌ای متقارن جایگاه ویژه‌ای دارند. سیستم رمز Twofish یکی از همین سیستم‌های رمز است، که جزو ۵ سیستم رمز راه یافته به مرحله نهایی مسابقه AES است. نسخه اولیه این سیستم رمز، بلوک‌های بیتی ورودی متن واضح را به بلوک‌های ۱۲۸بیتی متن رمز شده تبدیل می‌کند. در این پروژه سعی شده است که این سیستم رمز گسترش یافته و توانایی تبدیل بلوک‌های ۲۵۶بیتی متن واضح ورودی به بلوک‌های ۲۵۶بیتی متن رمز شده را داشته باشد. طبق تعریف اولیه پروژه در این پروژه می‌بایست ساختار کلی الگوریتم ثابت مانده و فقط S-Box های آن گسترش یابند. این سیستم رمز باید به گونه‌ای گسترش یابد که ویژگی‌های آماری سیستم رمز ابتدایی از بین نروند و در حد امکان افزایش یابند. در ادامه به بررسی پروژه و نتایج به دست آمده در مراحل مختلف انجام پروژه تشریح شده است.

فصل اول: سیستم رمزنگاری متقارن قطعه‌ای Twofish

معرفی کلی سیستم

سیستم رمز متقارن Twofish یک سیستم رمز قطعه‌ای است که در طراحی آن از ساختار فایستل بهره-گیری شده است. این سیستم رمز ۱۶ دور است. تابع F در این سیستم رمز یک تابع پوشاست. قطعه‌های ورودی ۱۲۸بیت و خروجی نیز یک قطعه‌ی ۱۲۸بیتی است. در نسخه اولیه این سیستم رمز کلید ۱۲۸بیتی است، اما امکان استفاده از کلید های با طول بیشتر تا ۲۵۶ بیت نیز وجود دارد. [1]

معرفی S-box

یک S-box یک جابجایی غیرخطی است که غالبا به صورت یک جدول نمایش داده می‌شود. استفاده از S-box ها در سیستم‌های رمز قطعه‌ای امری شایع است. S-box ها در تعداد بیت‌های ورودی و خروجی، تنوع زیادی دارند. و شیوه ساخت آنها می‌تواند به صورت بی‌قاعده[۱] و یا بر اساس الگوریتم خاصی باشد. S-box اولین بار در سیستم رمز Lucifer مورد استفاده قرار گرفت و سپس در سیستم رمز DES و پس از آن در غالب سیستم‌های رمز نگاری مورد استفاده قرار گرفته است. سیستم رمز Twofish از چهار S-box متفاوت که دارای ویژگی‌های زیر هستند بهره می‌برد:

- ۸-بیت ورودی
- ۸-بیت خروجی
- وابسته به کلید
- پوشا[۲] (احتمال تبدیل ورودی به هر کدام از مقادیر فضای برد[۳] وجود دارد)

ساختار سیستم

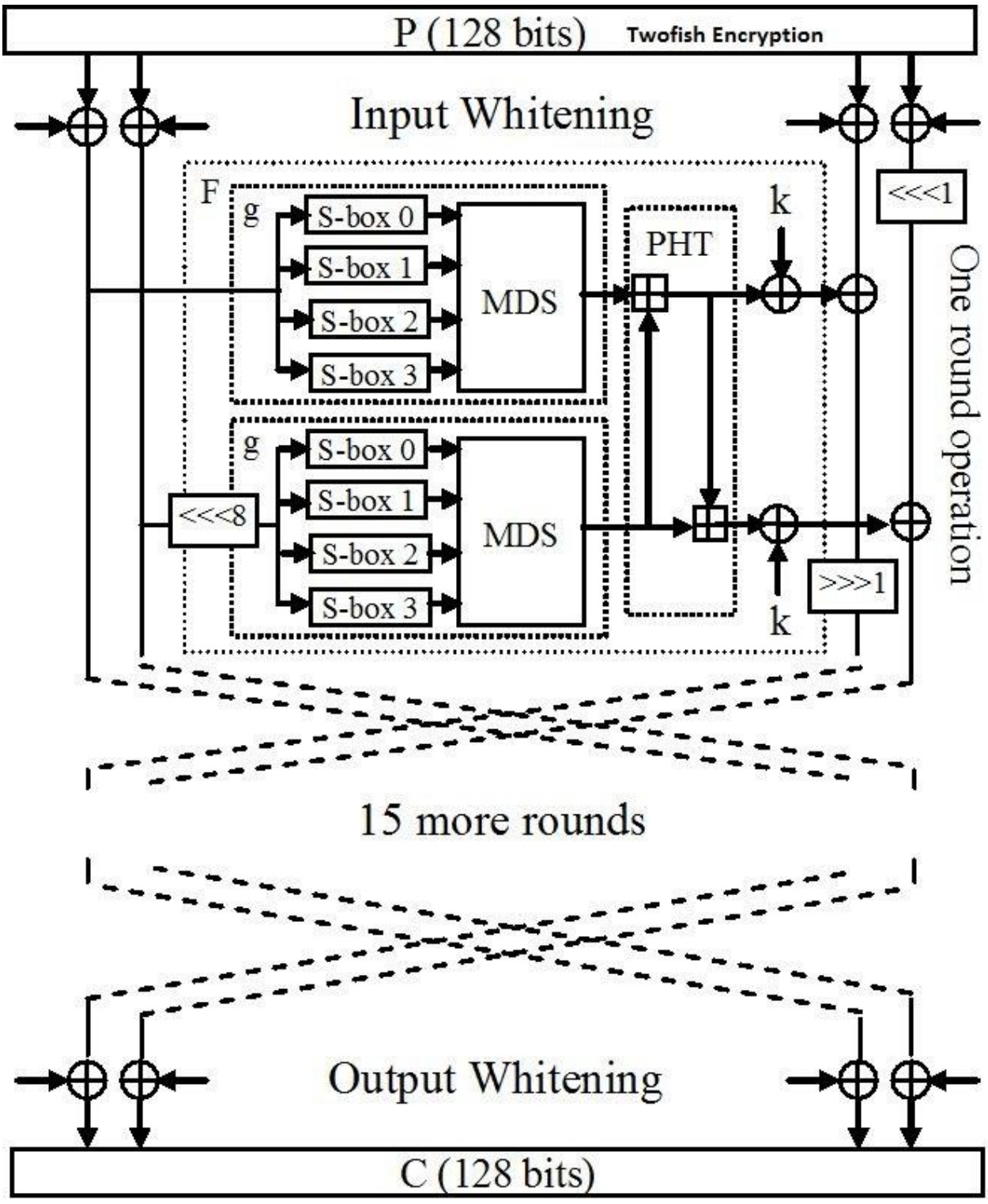
ساختار به صورت فایستل به همانند شکل زیر است:

---

۱ Randomly

۲ Bijective

۳ Range


P (128 bits)
Twofish Encryption
Input Whitening
F
g
S-box 0
S-box 1
S-box 2
S-box 3
MDS
PHT
k
<<<1
One round operation
g
S-box 0
S-box 1
S-box 2
S-box 3
MDS
<<<8
>>>1
k
15 more rounds
Output Whitening
C (128 bits)

فصل دوم: نسخه گسترش یافته سیستم رمز Twofish

طراحی نسخه گسترش یافته

طبق ساختار بررسی شده در فصل گذشته این سیستم رمز از یک تابع f ، یک تابع g ، s-box و ماتریس MDS بهره می‌برد. درون تابع f تابع g قرار دارد و هسته تابع g که هر s-box را تشکیل می‌دهد، تابعی به نام q است. در سیستم رمز Twofish از دو تابع q0 و q1 استفاده شده است که عملکرد آنها بسیار شبیه به هم است. تابع g قلب سیستم است و تابع q قلب تابع g. در بررسی برای گسترش تابع q که یک تابع ۸بیت-۸بیت است به یک تابع ۱۶بیت-۱۶بیت تبدیل شد، به گونه‌ای که ویژگی‌های آماری خود را به خوبی حفظ کند و حتی در برخی موارد ویژگی‌های آماری بهتری نیز فراهم سازد. ساختار تابع q در نسخه اصل سیستم رمز Twofish اینگونه است:

$$a_0 = \left\lfloor \frac{x}{16} \right\rfloor$$
$$b_0 = x \bmod 16$$
$$a_1 = a_0 \oplus b_0$$
$$b_1 = a_0 \oplus ROR_4(b_0, 1) \oplus 8a_0 \bmod 16$$
$$a_2 = t_0[a_1]$$
$$b_2 = t_1[b_1]$$
$$a_3 = a_2 \oplus b_2$$
$$b_3 = a_2 \oplus ROR_4(b_2, 1) \oplus 8a_2 \bmod 16$$
$$a_4 = t_2[a_3]$$
$$b_4 = t_3[b_3]$$
$$y = (16b_4 + a_4) \pmod{2^8}$$

در تابع ذکر شده x ورودی تابع و y خروجی تابع است. اما در نسخه گسترش یافته تابع که در آن اندازه ورودی و خروجی دو برابر شده است عملکرد تابع به شرح زیر است:

$$a_0 = \left\lfloor \frac{x}{4096} \right\rfloor$$
$$b_0 = \left\lfloor \frac{x}{256} \right\rfloor$$
$$c_0 = \left\lfloor \frac{x}{16} \right\rfloor$$
$$d_0 = x \bmod 16$$
$$a_1 = a_0 \oplus b_0$$
$$b_1 = a_0 \oplus ROR_4(b_0, 1) \oplus 8a_0 \bmod 16$$
$$a_2 = t_0[a_1]$$
$$b_2 = t_1[b_1]$$
$$a_3 = a_2 \oplus b_2$$
$$b_3 = a_2 \oplus ROR_4(b_2, 1) \oplus 8a_2 \bmod 16$$
$$a_4 = t_2[a_3]$$
$$b_4 = t_3[b_3]$$
$$c_1 = c_0 \oplus d_0$$

$d_1 = c_0 \oplus ROR_4(d_0, 1) \oplus 8c_0 mod\ 16$

$c_2 = t_0[c_1]$

$d_2 = t_1[d_1]$

$c_3 = c_2 \oplus d_2$

$d_3 = c_2 \oplus ROR_4(d_2, 1) \oplus 8c_2 mod\ 16$

$c_4 = t_2[c_3]$

$d_4 = t_3[d_3]$

$a_1 = a_0 \oplus c_0$

$c_1 = a_0 \oplus ROR_4(c_0, 1) \oplus 8a_0 mod\ 16$

$a_2 = t_0[a_1]$

$c_2 = t_1[c_1]$

$a_3 = a_2 \oplus c_2$

$c_3 = a_2 \oplus ROR_4(c_2, 1) \oplus 8a_2 mod\ 16$

$a_4 = t_2[a_3]$

$c_4 = t_3[c_3]$

$b_1 = b_0 \oplus d_0$

$d_1 = b_0 \oplus ROR_4(d_0, 1) \oplus 8b_0 mod\ 16$

$b_2 = t_0[b_1]$

$d_2 = t_1[d_1]$

$b_3 = b_2 \oplus d_2$

$d_3 = b_2 \oplus ROR_4(d_2, 1) \oplus 8b_2 mod\ 16$

$b_4 = t_2[b_3]$

$d_4 = t_3[d_3]$

$a_1 = a_0 \oplus d_0$

$d_1 = a_0 \oplus ROR_4(d_0, 1) \oplus 8a_0 mod\ 16$

$a_2 = t_0[a_1]$

$d_2 = t_1[d_1]$

$a_3 = a_2 \oplus d_2$

$d_3 = a_2 \oplus ROR_4(d_2, 1) \oplus 8a_2 mod\ 16$

$a_4 = t_2[a_3]$

$d_4 = t_3[d_3]$

$c_1 = c_0 \oplus b_0$

$b_1 = c_0 \oplus ROR_4(b_0, 1) \oplus 8c_0 mod\ 16$

$c_2 = t_0[c_1]$

$b_2 = t_1[b_1]$

$c_3 = c_2 \oplus b_2$

$b_3 = c_2 \oplus ROR_4(b_2, 1) \oplus 8c_2 mod\ 16$

$c_4 = t_2[c_3]$

$b_4 = t_3[b_3]$

$y = (4096d_4 + 256c_4 + 16b_4 + a_4)\ (mod\ 2^{16})$

فلسفه طراحی

در طراحی نسخه گسترش یافته سیستم رمز از ایده‌ی موجود در طراحی نسخه اصل سیستم استفاده شده است؛ به این صورت که عملیات انجام شده در تابع q از عملیات بر روی ۸بیت ورودی به عملیات بر روی یی ورودی ۱۶بیتی تغییر یافت همچنین خروجی ۸بیتی تابع نیز به ۱۶ بیت تغییر یافت. در نسخه گسترش یافته به دلیل انجام عملیات بر روی قطعه‌ی ۱۶ بیتی تعداد عملیات بیشتر شده، اما وجود عملیات بیشتر باعث تامین ویژگی آماری در خروجی تابع شده است.

نتایج به دست آمده

در سیستم رمز گستر یافته، برای هر دور از سیستم رمز ویژگی بهمنی در نظر گرفته شده است که با ۲۵۶۰ تست نتایج زیر به دست آمده اند:

1,	1,	2,	1,	4,	3,	6,	5,	5,	5,
1,	4,	6,	1,	5,	3,	5,	6,	5,	6,
4,	5,	4,	5,	5,	3,	5,	4,	4,	3,
4,	5,	8,	5,	3,	5,	5,	6,	5,	3,
5,	6,	5,	3,	5,	4,	4,	6,	5,	5,

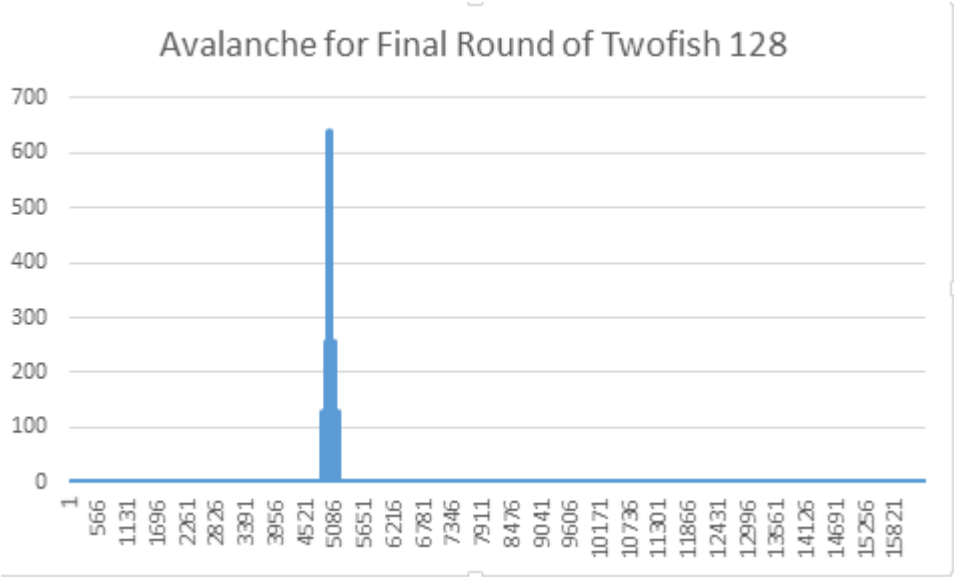
نتایج ذکر شده در بالا برای نسخه گسترش یافته سیستم رمز است. در زیر بخشی از نتایج برای نسخه اصل سیستم رمز ذکر شده است.

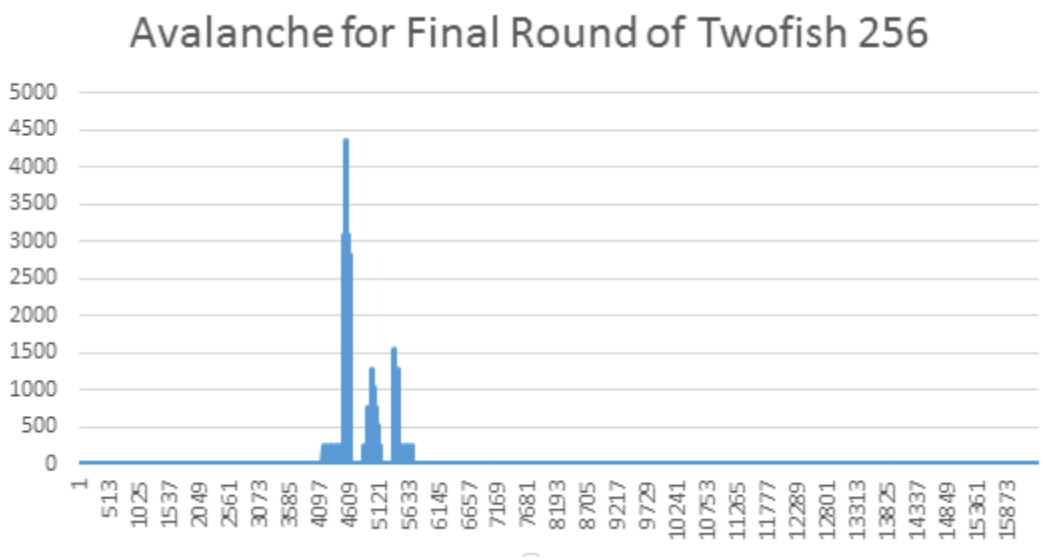
6,	5,	4,	4,	5,	2,	2,	2,	3,	2,
4,	7,	7,	5,	3,	3,	5,	4,	8,	4,
4,	2,	7,	5,	5,	4,	7,	4,	4,	5,
4,	4,	7,	4,	6,	4,	6,	3,	5,	3,
4,	6,	5,	6,	4,	5,	6,	6,	5,	4,

نتایج ذکر شده تنها بخشی از نتایج به دست آمده است؛ نتایج به صورت مفصل و کامل به عنوان پیوست ارائه شده است.

مقایسه نتایج با نتایج به دست آمده از الگوریتم اصلی

همانگونه که در نمودار ذیل دیده می‌شود، نمودار ویژگی‌های آماری نسخه اصل سیستم رمز و نسخه گسترش یافته سیستم رمز را نمایش می‌دهد. نتایج به دست آمده ویژگی‌های آماری دو سیستم رمز را برای ۱۰هزار نمونه ورودی که برای هر ورودی با تغییر هر بیت آن متن رمز شده آن دوباره بررسی شده است را بیان می‌کند. به عبارت دیگر نتایج به دست آمده برای نسخه اصل سیستم رمز که دارای ۱۲۸ بیت ورودی است، حاصل شده از $128 \times 10000 = 1{,}280{,}000$ تست انجام شده و برای نسخه گسترش یافته سیستم که دارای ۲۵۶ بیت ورودی است، حاصل از $256 \times 10000 = 2{,}560{,}000$ تست انجام گرفته شده است.




Avalanche for Final Round of Twofish 256
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
1
513
1025
1537
2049
2561
3073
3585
4097
4609
5121
5633
6145
6657
7169
7681
8193
8705
9217
9729
10241
10753
11265
11777
12289
12801
13313
13825
14337
14849
15361
15873

فصل سوم: نتیجه گیری و جمع بندی

می‌توان گفت که سیستم رمز طراحی شده با توجه به نتایج به دست آمده از نسخه اصل سیستم رمز Twofish ، طراحی قابل قبولی دارد. در بررسی خاصیت بهمنی و کامل بودن سیستم رمز (که فایل‌های مربوطه آن در پیوست ذکر شده اند) در قسمت‌های زیادی بهبودهایی در ویژگی‌های آماری سیستم رمز دیده می‌شود.

فهرست منابع

[۱] B. Schneier, "Twofish: A 128-Bit Block Cipher," [Online]. Available: https://www.schneier.com/cryptography/paperfiles/paper-twofish-paper.pdf.

پیوست

کد برنامه‌های نوشته شده برای انجام کار به زبان‌های Java, C, C++, C# به عنوان پیوست به پروژه ضمیمه شده است. لازم به ذکر است که تمام کدهای پیوست شده (حتی کد اصلی الگوریتم رمز) توسط نگارندگان پروژه پیاده سازی شده است.