باسمه تعالی

گزارش پروژه درس رمزنگاری پیشرفته

عنوان:

حملات Boomerang

انجام‌دهنده:

علیرضا شیرزاد

99201754

تابستان ۱۴۰۱

تاریخ تحویل: 1401/5/20	حملات Boomerang	درس رمزنگاری پیشرفته
99201754	علیرضا شیرزاد	

چکیده

پس از معرفی حمله‌ی تفاضلی و پدیدآمدن تکنیک‌های مقاومت در برابر این نوع حملات، حملاتی به نام حملات بومرنگ معرفی شدند که نیازی به یک مشخصه‌ی تفاضلی کامل برای یک الگوریتم رمز ندارد، بلکه با استفاده از یک مشخصه‌ی نصفه حمله را اجرا می‌کند. تحلیل پیچیدگی این حملات کار آسانی نیست چرا که از اتصال 4 مشخصه‌ی مختلف بدست می‌آید که می‌توانند کاملا از هم مستقل باشند. برای بهبود دقت در بدست آوردن پیچیدگی حمله‌ی بومرنگ، حمله‌ی ساندویچ مطرح شد که لایه‌ی میانی‌ای را جهت اتصال دو مشخصه‌ی بالایی و پایینی پیشنهاد داد.در واقع احتمال رخداد بومرنگ به حاصل ضرب 4 مشخصه در احتمال رخداد چهارتایی بومرنگ در لایه‌ی میانی تبدیل می‌شود. حال اگر این لایه شامل یک دور از الگوریتمی برمبنای جعبه جانشینی باشد، احتمال رخدادن چهارتایی بومرنگ در لایه میانی با استفاده از جدول BCT قابل محاسبه است که از جدول DDT در حمله‌ی تفاضلی الهام گرفته شده است. جدول BCT خواص جالب دارد که به برخی از آن‌ها می‌پردازیم و رابطه‌ی آن را با جدول DDT بررسی می‌کنیم، سپس نتیجه می‌گیریم که جدول BCT 4-ینکواخت با خاصیت تفاضلی 4-یکنواخت وجود ندارد و بهترین جدول BCT 6-ینکواخت است.در ادامه اگر لایه‌ی میانی شامل بیش از یک لایه جعبه جانشینی باشد، محاسبه‌ی احتمال آن کمی پیچیده‌تر می‌شود، چرا که جدول BCT دیگر کارایی نخواهد داشت. در این سناریو جداول جدیدی به نام LBCT ، UBCT و EBCT به عنوان تعمیمی از BCT پیشنهاد شده است. بر اساس این جداول الگوریتمی برای محاسبه‌ی احتمال و بالتبع پیچیدگی الگوریتم معرفی شده است. در آخر نیز پیشنهاداتی برای شناخت بیشتر جداول BCT، یافتن بهترین مسیر بومرنگ و نحوه‌ی دفاع در مقابل چنین حملاتی ارائه شده است.

فهرست مطالب

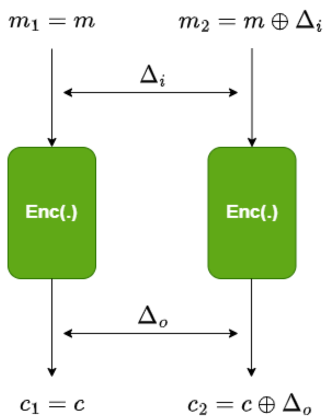
فهرست اشکال

فهرست جداول

فصل 1

۱ مقدمه

پس از گسترش استفاده از رمزهای قالبی و معرفی استاندارد DES [1] ، تکنیک‌های مختلفی برای تحلیل این رمزها ارائه شد. یکی از تکنیک‌های پرکاربرد و محبوب، تحلیل تفاضلی [2]–[4] است که براساس مفهوم تمایزگر[1] عمل می‌کند. اساس کار حملات تمایزگری، پیدا کردن یک ویژگی در الگوریتم رمزنگاری است که بین خروجی الگوریتم رمز و یک خروجی تصادفی، تمایز ایجاد می‌کند. در حملات تفاضلی، تمایزگر ارائه شده، رابطه‌ی بین تفاضل متن رمزشده و تفاضل متن اصلی است، بدین صورت که اگر تفاضل معینی از متن اصلی را در ورودی الگوریتم رمزنگاری قرار دهیم، با احتمال قابل قبولی، تفاضل معینی از متن رمزشده را در خروجی خواهیم دید.



شکل 1: تحلیل تفاضلی

در یک الگوریتم ایده‌آل (جایگشت تصادفی)، تفاضل متن رمزشده در خروجی توزیع یکنواخت دارد و داریم

$$p_D = Pr[\Delta_i \to \Delta_o] = \Pr[c_1 \oplus c_2 = \Delta_o | m_1 \oplus m_2 = \Delta_i] = \frac{1}{2^b}$$

اما در الگوریتم‌های غیر ایده آل مقدار $\mathrm{p_D}$ بیشتر از مقدار فوق‌الذکر است که همین باعث تمایز بین یک الگوریتم رمزنگاری و یک جایگشت تصادفی می‌شود. همچنین این حملات معمولا با یک فاز استخراج کلید همراه هستند که در زمانی کمتر از زمان جستجوی کور[2]، کلید را بدست می‌آورند. در حملات تفاضلی در ابتدا مقدار cp_D^{-1} زوج متن اصلی منتخب با تفاضل ورودی Δ_i انتخاب می‌شوند. منطق این حمله بر این اساس است که چون احتمال $\Delta_i \to \Delta_o$ برابر با p_D است پس، در هر p_D^{-1} زوج متن اصلی، حدودا یک بار این اتفاق می‌افتد. برهمین اساس یک احتمال تفاضلی برای $R-1$ راند اول الگوریتم رمز انتخاب می‌شود، سپس برای هر زوج متن اصلی و متن رمز شده، دور آخر متن رمز شده وارد الگوریتم رمزگشایی جزئی[3] با یک کلید کاندیدا می‌شود تا متن رمز شده در دور $R-1$ بدست بیاید. سپس تفاضل در این لایه محاسبه می‌شود، اگر این تفاضل Δ_o بود، به شمارنده‌ی این کلید یک مقدار اضافه می‌شود [3] . به این حملات $R-1$ می‌گویند.

[1] Distinguisher
[2] Brute Force
[3] Partial Decryption

پس از ارائه‌ی حمله‌ی فوق توسط Biham و Shamir ، طراحان الگوریتم‌های رمزنگاری روش‌هایی را در پیش گرفتند تا الگوریتم‌ها در مقابل حملات تفاضلی امن باقی بمانند. از این روش‌ها می‌توان به بهبود مشخصه‌ی تفاضلی s-boxها، بهبود پراکنش[۴] و اضافه کردن لایه‌ی همبستگی زدایی [5] اشاره کرد. در واقع نتیجه‌ی به کار گیری همه‌ی این روش‌ها، بهبود مشخصه‌ی تفاضلی الگوریتم و کاهش احتمال وقوع تفاضل‌ها در الگوریتم هستند.

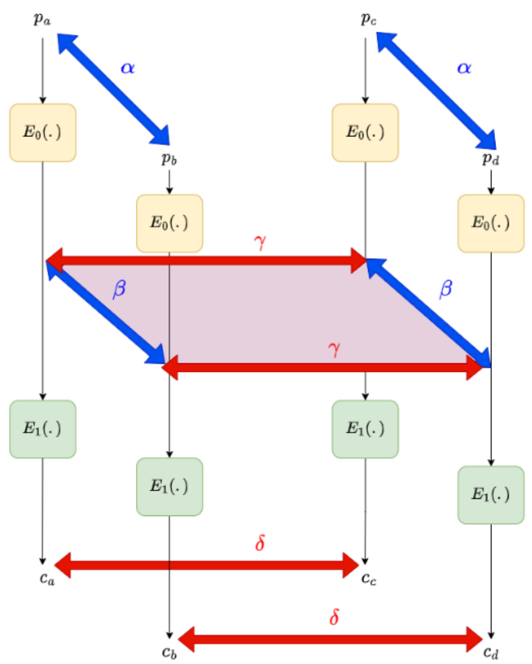
در مقابل این روش‌های دفاعی، استراتژی‌های حمله‌ی جدیدی نیز توسعه پیدا کرد که یکی از مهم‌ترین آن ها، حمله‌ی بومرنگ [6] می‌باشد که در این گزارش به آن می‌پردازیم. حمله‌ی بومرنگ برای حمله به الگوریتم‌های رمزی پیشنهاد شد که مشخصه‌ی تفاضلی خوبی ندارند و احتمال p_D آن‌ها بسیار پایین است. در این حمله الگوریتم رمزنگاری $E(.)$ به دو بخش $E_1(.)$ و $E_0(.)$ تقسیم می‌شود به طوری که $E(.) = E_1(.) \circ E_0(.)$. اساس این حمله بر این حقیقت استوار است که یک الگوریتم رمز با این که مشخصه‌ی تفاضلی خوبی برای تمام N دور خود نداشته باشد، اما ممکن است برای بخشی از دورهای خود که متعلق به $E_0(.)$ و $E_1(.)$ می‌باشد، مشخصه‌هایی با احتمال بالا وجود داشته باشد. بر همین اساس فرض کنید که

$$\Pr\left[\alpha \overset{E_0}{\rightarrow} \beta\right] = p \text{ و } \Pr\left[\gamma \overset{E_1}{\rightarrow} \delta\right] = q$$

آن‌گاه حمله‌ی بومرنگ بر اساس احتمال زیر تعریف می‌شود:

$$\Pr[E^{-1}(E(m) \oplus \delta) \oplus E^{-1}(E(m \oplus \alpha) \oplus \delta) = \delta] \cong p^2q^2 \quad \text{(معادله 1)}$$

در واقع احتمال این واقعه برابر با این است که هر مشخصه‌ی تفاضلی $\alpha \overset{E_0}{\rightarrow} \beta$ و $\gamma \overset{E_1}{\rightarrow} \delta$ دوبار اتفاق بیفتد که با شرط استقلال واقعه‌ها از هم (که در آینده می‌بینیم شرط درستی نیست) برابر می‌شود با p^2q^2.



شکل 2: حمله‌ی بومرنگ

طبیعتا با چنین مشخصه‌ای می‌توان حمله‌ی تمایزگری زیر را طراحی کرد:

[4] Diffusion

الگوریتم 1: حمله تمایزگری بومرنگ

ورودی: پاسخگوی رمزنگاری و رمزگشایی، مجموعه $M = \{(p_1, p_2) | p_2 = p_1 \oplus \alpha\}$

1. یک متن اصلی p_a را از مجموعه M انتخاب کن و مقدار $p_b = p_a \oplus \alpha$ را از روی آن بساز.
2. حاصل $c_a = E(p_a)$ و $c_b = E(p_b)$ را محاسبه کن.
3. سپس مقادیر $c_c = c_a \oplus \delta$ و $c_d = c_b \oplus \delta$ را محاسبه کن.
4. در نهایت مقدار $p_c = E^{-1}(c_c)$ و $p_d = E^{-1}(c_d)$ را محاسبه کن.
5. چک کن که آیا $p_c \oplus p_d = \alpha$ برقرار است یا خیر، اگر برقرار بود به شمارنده یکی اضافه کن.
6. اگر در نهایت مقدار شمارنده از حد آستانه بیشتر بود، خروجی 1 بده، در غیر این صورت 0 بده.

برای این که حمله‌ی فوق موفقیت‌آمیز باشد، تقریبا نیاز به $p^{-2}q^{-2}$ جفت متن اصلی نیاز است. حال براساس همین تمایزگر می‌توان حمله‌ی استخراج کلید R-1 را اجرا کرد. نکته‌ی قابل توجه در حمله این است که تنها مقادیر α و δ توسط مهاجم کنترل می‌شوند و مقادیر β و γ هر چیزی می‌توانند باشند، تنها به شرط اینکه هر 4 مشخصه‌ی تفاضلی به صورت همزمان برقرار باشند. با این تحلیل، احتمال اینکه یک جفت متن اصلی در شرط بومرنگ صدق کند به شکل زیر افزایش پیدا می‌کند:

$$\Pr[E^{-1}(E(m) \oplus \delta) \oplus E^{-1}(E(m \oplus \alpha) \oplus \delta) = \delta] \cong \sum_{\beta} \Pr[\alpha \overset{E_0}{\rightarrow} \beta] . \sum_{\gamma} \Pr[\gamma \overset{E_1}{\rightarrow} \delta] \quad \text{(معادله 2)}$$

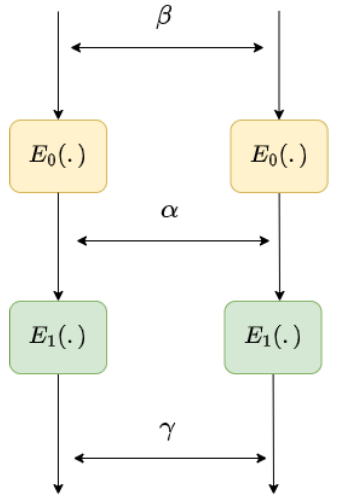
به عنوان مثال در الگوریتم رمزنگاری COCONUT98 [5] حمله‌ی تفاضلی هیچ مزیتی نسبت به حمله‌ی جستجوی کور ندارد و مقدار احتمال برقراری هر مشخصه‌ای تقریبا معادل $\frac{1}{2^{64}-1}$ است، در حالی که حاصل معادله 2 برای $\alpha = \delta = (e_{10}, e_{31})$ در این الگوریتم رمز برابر با $\frac{1}{1900}$ است که نشانگر عملکرد عالی حمله‌ی بومرنگ دارد.

نکته‌ی مهم دیگری که به همراه حمله‌ی بومرنگ به آن اشاره شد، این است که آیا می توان ایده‌های حمله‌ی تفاضلی بریده[5] [7]را در ساختار بومرنگ اجرا کرد یا خیر. حملات تفاضلی بریده حملاتی هستند که در آن تفاضل همه‌ی بیت‌ها مهم نیست، بلکه تفاضل برخی از بیت‌ها کافی است. نکته‌ی قابل توجه این است که تفاضل‌های معکوس که اساس کار حمله‌ی بومرنگ است، لزوما در حملات بریده برقرار نیستند. به همین دلیل احتمال موفقیت در حمله‌ی بومرنگ باید با دقت بیشتری به شکل زیر نوشته شود

$$\text{p} \approx \sum_{w \oplus x \oplus y \oplus z = 0} \Pr\left[\alpha \overset{E_0}{\rightarrow} w\right] . \Pr\left[\delta \xrightarrow{E_1^{-1}} x\right] . \Pr\left[\delta \xrightarrow{E_1^{-1}} y\right] . \Pr[z \xrightarrow{E_0^{-1}} \alpha] \quad \text{(معادله 3)}$$

همچنین حمله‌ی دیگری که در مقاله‌ی معرفی بومرنگ به آن پرداخته می‌شود، حمله‌ی برونگرا[6] است که به نحوی دوگانِ حمله‌ی بومرنگ است. در این حمله از یک درون یک الگوریتم‌های رمزنگاری به سمت بیرون حرکت می‌کنیم، بدین صورت که تفاضل ورودی را در درون و تفاضل خروجی را در بیرون قرار می‌دهیم. به عبارتی این حمله براساس دو مشخصه‌ی $\alpha \xrightarrow{E_0^{-1}} \beta$ و $\alpha \overset{E_1}{\rightarrow} \gamma$ اجرا می‌شود.

[5] Truncated Differential Attack
[6] Inside-Out Attack
در واقع Inside-Out به معنی پشت و رو می‌باشد اما در اینجا به معنی از داخل به بیرون است که ترجمه‌ی مناسب آن برونگرا می‌باشد.



شکل 3: حملهی برونگرا

الگوریتم 2: حمله تمایزگری برونگرا

1. به مقدار کافی زوج متن اصلی (p_1, p_2) و متناظرا زوج متن رمز شده (c_1, c_2) را ذخیره کن به طوری که حدود Rتا از آن‌ها تفاضل α را در میانه داشته باشند.
2. از این بین تعداد زوج‌هایی را که در شرط $p_1 \oplus p_2 = \beta$ و $c_1 \oplus c_2 = \gamma$ صدق می‌کنند را بشمار. اگر این تعداد از تعداد آستانه خیلی بیشتر بود، 1 خروجی بده در غیر این صورت 0 خروجی بده.

در الگوریتم بالا، برای یک زوج تصادفی در N زوج انتخاب شده فرض کنید

$$\Pr[p_1 \oplus p_2 = \beta] = p_0$$

$$\Pr[c_1 \oplus c_2 = \gamma] = p_1$$

لذا ما به صورت تصادفی و نرمال $N_1 = p_1 p_0 N$ زوج با تفاضل‌های درست می‌بینیم، اما در مجموعه زوج‌هایی که انتظار داریم R زوج دارای تفاضل میانی α باشند، مقدار مورد انتظار مشاهده‌ی زوج‌های با تفاضل ابتدا و انتهای درست $N_1 + R$ می‌باشد. به طور سرانگشتی اگر $R \gg \sqrt{N_1}$ باشد، تمایزگر به خوبی اجرا می‌شود.

1-1 مرور ادبیات

پس از معرفی حمله‌ی بومرنگ [6]، انواع مختلفی از این حمله معرفی شد که به اختصار به برخی از آن‌ها می‌پردازیم. در [8] حمله‌ی بومرنگ تقویت‌شده[7] معرفی شده که حمله‌ی بومرنگ را به یک حمله‌ی متن اصلی معلوم تبدیل می‌کند که باعث کاهش احتمال برقراری شرط بومرنگ به $p^2q^2 2^{-n}$ (که p و q به ترتیب احتمال‌های برقراری مشخصه‌های بالا و پایین و n اندازه بلوک است) و بالتبع، افزایش پیچیدگی داده می‌شود.

[7] Amplified Boomerang Attack

یکی از کارهای مهم در زمینه حملات بومرنگ، مقاله Murphy [9] است که در آن در رابطه با احتمال برقراری یک ساختار بومرنگ بحث‌های مهمی شده. در واقع Murphy نشان می‌دهد که مشخصه‌های انتخابی برای E_0 و E_1 از هم مستقل نیستند، لذا احتمال برقراری مشخصه برابر با ضرب توان دوم مشخصه‌های E_0 و E_1 نیست. وی نشان می‌دهد که برخی از مشخصه‌ها با هم سازگار نیستند و احتمال رخداد آن‌ها صفر است و حمله را بی‌معنی می‌کند، از طرفی برخی از مشخصه‌ها نیز احتمال رخداد بسیار بالاتری نسبت به میزان تخمین زده شده دارند که حمله را کاراتر می‌کند. در واقع این عدم توانایی در تخمین درست از احتمال برقراری ساختار بومرنگ، انگیزه‌ی اصلی بحث‌های پیش‌رو در این گزارش است. در بخش‌های بعدی می‌بینیم که این خلاء چگونه پر خواهد شد.

۱-۲ سازمان‌دهی گزارش

در فصل اول به مفهوم نقطه‌ی اتصال در حمله‌ی بومرنگ می‌پردازیم و مشاهده می‌کنیم که انتخاب نقطه‌ی اتصال مناسب چقدر به کاهش پیچیدگی حمله کمک می‌کند. سپس به مطالعه انواع نقاط اتصال و روش‌های سوییچ کردن می‌پردازیم. در فصل دوم به مطالعه‌ی ابزاری به نام BCT می‌پردازیم که مطالعه‌ی سوییچ‌های بومرنگ را بسیار ساده تر کرده، سپس به خواص آن می‌پردازیم و بین BCT و DDT مقایسه‌هایی انجام می‌دهیم. در فصل سوم به تعمیم مفهوم BCT اختصاص یافته است. در این فصل با جداول LBCT ، UBCT و EBCT آشنا می‌شویم که در مطالعه‌ی نقطه‌های اتصال چند دوری بسیار کارامد هستند.

۲ مسئله‌ی اتصال در بومرنگ

در سال 2011، این حقیقت توسط Murphy کشف شد که امکان انتخاب مشخصه‌های E_0 و E_1 به صورت کاملا مستقل وجود ندارد[9].در واقع برقراری معادله 1 زیر سئوال رفت، چرا که وی نشان داد، دو مشخصه‌ی انتخابی برای E_0 و E_1 می توانند کاملا نامنطبق باشند و احتمال بوجود آمدن توامانِ چنین مشخصه‌هایی صفر باشد. همچنین در سناریوهایی احتمال بازگشت بومرنگ از مقدار محاسبه‌شده یعنی p^2q^2 بسیار بیشتر خواهد بود. چنین دستاورهایی نشان داد که مشخصه‌های بالایی و پایینی، از یکدیگر مستقل نیستند بلکه در نقطه‌ی اتصال به یکدیگر وابسته هستند.

نقطه‌ی اتصال در واقع مساحت هاشورزده‌ای است که در شکل 2 نشان‌داده شده است. در واقع در تحلیل بومرنگ دو مشخصه‌ی بالا و پایین می‌توانند مستقل باشند تا جایی که در نقطه‌ی اتصال با هم متنافر نباشند. نقطه‌ی اتصال باعث کاهش احتمال p^2q^2 می‌شود و اگر اتصال متنافر برقرار شود، این احتمال به صفر می‌رسد.

۲-۱ انواع اتصالات

در ادامه به معرفی و بحث در رابطه با چند اتصال معروف در حمله‌ی بومرنگ می‌پردازیم که نمایانگر عدم استقلال مشخصه‌ها و اهمیت نقطه‌ی اتصال دو مشخصه می‌باشد:

۲-۱-۱ اتصال غیرممکن

گاهی اوقات مشخصه‌های بالایی و پایینی با هم تطابق ندارند، به صورتی که احتمال برگشت بومرنگ صفر است. به عنوان مثال در [9]ذکر شده که در یک DES چهار دوری که به دو زیرالگوریتم دو دوری E_0 و E_1 تقسیم می‌شود، پیدایش بومرنگی با مشخصات زیر صفر است:

$$\Delta = (\Upsilon_9, 0) \xrightarrow{E_0} \Delta^* = \Delta \; s.t. \Upsilon_9 = 0x19600000$$

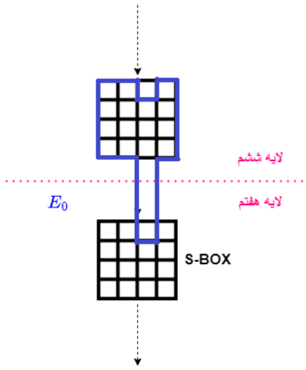
$$\nabla = (\Upsilon_B, 0) \xrightarrow{E_0} \nabla^* = \nabla \; s.t. \Upsilon_B = 0x1B600000$$

در واقع طبق آنالیز بومرنگ، احتمال وقوع یک چهارتایی با این مشخصات $\left(\frac{1}{234}\right)^4$ است، منتها اثبات می‌شود که وقوع این چهارتایی امکان‌پذیر نیست.

۲-۱-۲ اتصال نردبانی[8]

اتصال نردبانی [10] پیشنهاد می‌کند که نقطه‌ی اتصال را لزوما بعد از یک لایه‌ی کامل و بر روی یک حالت کامل از الگوریتم رمزنگاری خود نگیریم، بلکه می‌توانیم نقطه‌ی اتصال را تلفیقی از دو حالت در نظر بگیریم، یعنی بخشی از نقطه‌ی اتصال در حالت i ام و بخشی از آن در حالت $i+1$ام باشد. برای درک اتصال نردبانی به عنوان مثال در شکل 4 تصور کنید که خانه‌ی $b_{0,2}$ در لایه‌ی هفتم در بخش پایینی فعال و در بخش بالایی غیر فعال باشد.

[8] Ladder Switch

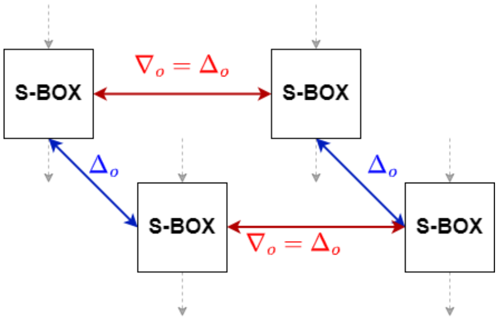


شکل 4: نمونه‌ای از اتصال نردبانی در الگوریتم AES192

اگر در حالت سنتی بومرنگ بخواهیم بین لایه‌ها گذار را انجام دهیم، این گذار باعث کاهش احتمال کل پیدایش بومرنگ می‌شود،چرا که تفاضل لایه‌ی بالایی باید با لایه‌ی پایینی سازگار باشد. حال فرض کنید که Sbox خانه‌ی (0,2) را به الگوریتم E_0 اضافه کنیم، بدین ترتیب چون این Sbox در الگوریتم بالایی تفاضلی ندارد، در نقطه‌ی اتصال هیچ هزینه‌ای برای ما نخواهد داشت و احتمال کل نمی‌کاهد. با توجه به شکل 2، دلیل این امر این است که تفاضل صفر در سمت اولیه‌ی E_0 با انتقال γ مجددا تفاضل صفر خواهند داشت، لذا بخشی از تفاضل β با احتمال 1 ساخته می‌شود و دیگر نیازی به نگرانی در رابطه با عدم تطابق یا کاهش احتمال p^2q^2 نخواهیم بود. البته کلمات دیگر نقطه‌ی اتصال ممکن است مشکل ساز باشند اما در این کلمه‌ی بخصوص هیچ هزینه‌ای پرداخت نشده است.

۳-۱-۲ اتصال جانشینی[9]

در صورتی که در لایه‌ی آخر E_0، جعبه‌ی جانشینی‌ای وجود داشته باشد که تفاضل خروجی آن، با تفاضل ابتدای الگوریتم پایینی برابر باشد، احتمال برقراری همین تفاضل در سمت دیگر، 1 می‌شود و هیچ هزینه‌ی احتمالی‌ای برای ما نخواهد داشت. [10]



شکل 5 : اتصال جانشینی

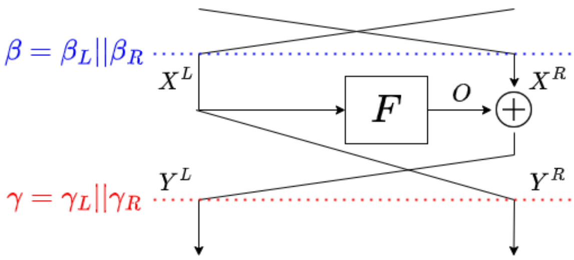
در واقع با تفاضل $\nabla_0 = \Delta_o$ صرفا جای تفاضل برعکس می‌شود و تفاضل تغییری نمی‌کند.

۴-۱-۲ اتصال فیستلی[10]

در این اتصال، فرض می‌شود که الگوریتم رمز به شکل فیستلی می‌باشد. حال نکته‌ی جالب این است که ما می‌توانیم یک راند از الگوریتم را به صورت رایگان و بدون هزینه در اتصال بومرنگ داشته باشیم [10]. در ابتدا به شکل زیر توجه بفرمایید:

---

[9] S-Box Switch
[10] Feistel Switch



شکل 6: اتصال فیستلی

در ابتدا هزینه‌ی بال چپ ساختار فیستل را محاسبه می‌کنیم:

$$\begin{aligned} X_3^L \oplus X_4^L &= Y_3^R \oplus Y_4^R = (Y_1^R \oplus \gamma_R) \oplus (Y_2^R \oplus \gamma_R) \\ &= Y_1^R \oplus Y_2^R = X_1^L \oplus X_2^L = \beta \end{aligned} \quad \text{(معادله4)}$$

که مشاهده می‌کنیم بدون هیچ هزینه‌ای تفاضل برقرار می‌شود، حال برای اینکه بال راست ساختار نیز هزینه‌ای نداشته باشد باید داشته باشیم:

$$\begin{aligned} X_3^R \oplus X_4^R &= (O_3 \oplus Y_3^L) \oplus (O_4 \oplus Y_4^L) = (F(X_3^L) \oplus Y_3^L) \oplus (F(X_4^L) \oplus Y_4^L) \\ &= (F(Y_3^R) \oplus Y_3^L) \oplus (F(Y_4^R) \oplus Y_4^L) \\ &= (F(Y_1^R \oplus \gamma_R) \oplus Y_3^L) \oplus (F(Y_2^R \oplus \gamma_R) \oplus Y_4^L)) \\ &= (F(X_1^L \oplus \gamma_R) \oplus Y_3^L) \oplus (F(X_2^L \oplus \gamma_R) \oplus Y_4^L) \\ &= (F(X_1^L \oplus \gamma_R) \oplus Y_3^L) \oplus (F(X_1^L \oplus \beta_L \oplus \gamma_R) \oplus Y_4^L) \\ &= (F(X_1^L \oplus \gamma_R) \oplus (F(X_1^L) \oplus X_1^R \oplus \gamma_L)) \oplus (F(X_1^L \oplus \beta_L \oplus \gamma_R) \\ &\oplus (F(X_1^L \oplus \beta_L) \oplus (X_1^R \oplus \beta_R) \oplus \gamma_L)) \\ &= F(X_1^L \oplus \gamma_R) \oplus F(X_1^L) \oplus F(X_1^L \oplus \beta_L \oplus \gamma_R) \oplus F(X_1^L \oplus \beta_L) \\ &\oplus \beta_R = \beta_R \end{aligned}$$

(معادله 5)

که این بدین معنی است که باید داشته باشیم:

$$F(X_1^L \oplus \gamma_R) \oplus F(X_1^L) \oplus F(X_1^L \oplus \beta_L \oplus \gamma_R) \oplus F(X_1^L \oplus \beta_L) = 0 \quad \text{(معادله 6)}$$

عبارت 6 در شرایط مختلفی صفر می‌شود که یکی از تکنیک‌های بررسی آن در [11] جدول اتصالات بومرنگ فیستلی[11] است. البته در [12] پیشنهاد شده که بگیریم $\beta_l = \gamma_R$ تا معادله 6 برقرار باشد. در هر صورت در ساختار فیستلی می‌توان یک دور را به رایگان داشت و این مسئله‌ی مهمی در نقطه‌ی اتصال است.

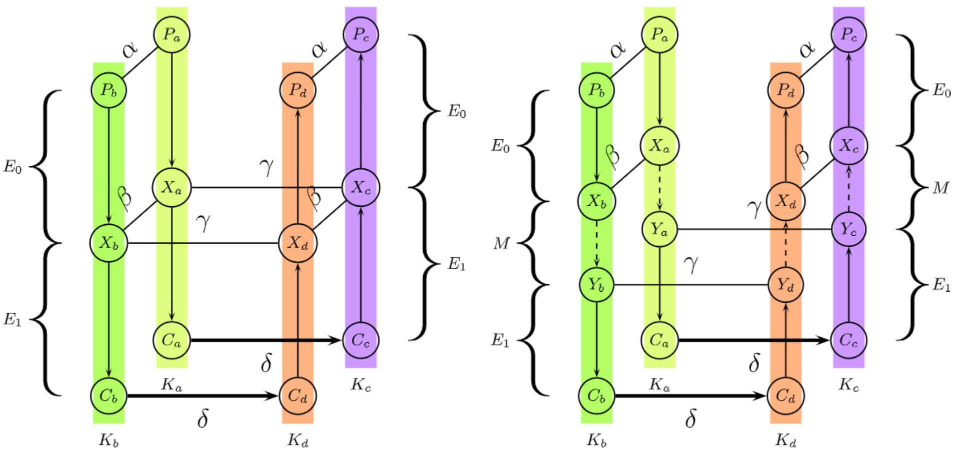
[11] Feistel Boomerant Connectivity Table (FBCT)

۲-۲ حمله‌ی ساندویچ[۱۲]

همانطور که در بخش قبل دیدیم، اتصالات و گذار از E_0 به E_1 می‌تواند به طرق مختلفی انجام شود که هر کدام هزینه‌های خاص خودش را دارد. برای مدل سازی این هزینه‌ها در نقطه‌ی اتصال، ساختاری به شکل ساندویچ پیشنهاد شد [12]. در این حمله، الگوریتم رمز ما به 3 زیر الگوریتم تقسیم می‌شود.

$$E(.) = E_0(.) \circ E_m(.) \circ E_1(.) \qquad \text{(معادله 7)}$$

در واقع الگوریتم $E_m(.)$ نقطه‌ی اتصال و احتمال گذار دو مشخصه را مدل‌سازی می‌کند. توجه داشته باشید که حمله‌ی ساندویچ هیچ تفاوتی با حمله‌ی بومرنگ در الگوریتم حمله ندارد، بلکه تفاوت در آنالیز احتمالاتی الگوریتم و مدل‌سازی ریاضی آن است.



ب) حمله‌ی بومرنگ

الف) حمله‌ی ساندویچ

شکل 7: تفاوت حملات بومرنگ و ساندویچ

در حمله‌ی اصلی بومرنگ، اگر دوتایی (p_a, p_b) برای مشخصه‌ی اول زوج درستی باشند و دوتایی‌های (C_a, C_c) و (C_b, C_d) نیز نسبت به مشخصه‌ی دوم و سوم زوج‌های درستی باشند، خواهیم داشت

$$(X_a \oplus X_b = \beta) \wedge (X_a \oplus X_c = \gamma) \wedge (X_b \oplus X_d = \gamma) \qquad \text{(معادله 8)}$$

که نتیجه می‌دهد

$$X_c \oplus X_d = \beta$$

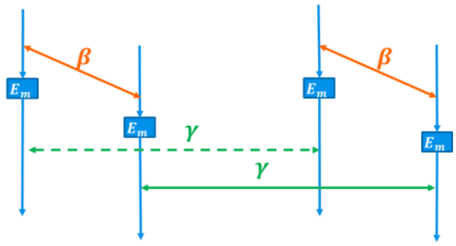
حال رابطه‌ی $p_c \oplus p_d = \alpha$ با احتمال p برقرار است. اما در حمله‌ی ساندویچ به جای رابطه‌ی 8، داریم:

$$(X_a \oplus X_b = \beta) \wedge (Y_a \oplus Y_c = \gamma) \wedge (Y_b \oplus Y_d = \gamma) \qquad \text{(معادله 9)}$$

که X_iها حاصل رمزنگاری توسط E_0 و Y_iها حاصل رمزگشایی توسط E_1 هستند. لذا احتمال تشکیل ساختار بومرنگ برابر می‌شود با p^2q^2r به طوری که

$$r = \Pr[X_c \oplus X_d = \beta | (X_a \oplus X_b = \beta) \wedge (Y_a \oplus Y_c = \gamma) \wedge (Y_b \oplus Y_d = \gamma)] \qquad \text{(معادله 10)}$$

[12] Sandwich Attack

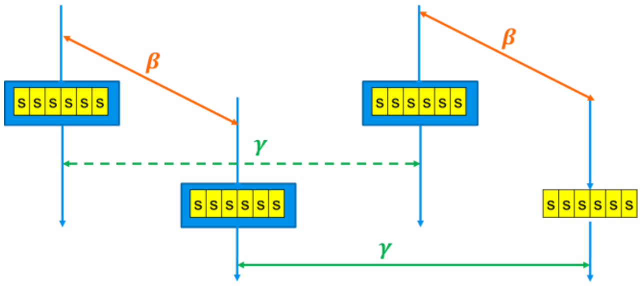


شکل 8: مشخصه‌ی اتصال در حمله‌ی ساندویچ

بدون هیچ شرطی روی ساختار E_m و تفاضل‌های بالایی و پایینی، مقدار r بسیار پایین و حوالی 2^{-n} خواهد بود، اما همانطور که در بخش‌های قبل دیدیم، به استفاده از تکنیک‌هایی می‌توان نقطه‌ی اتصال یا E_m را به شکلی تنظیم کرد که مقدار r بسیار بالا و یا حتی 1 باشد، به طوری که بخش E_m به صورت رایگان بدست بیاید.

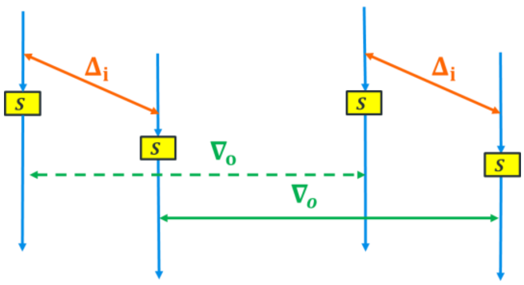
۳ جدول اتصالات بومرنگ

همانطور که در بخش قبل دیدیم، تاثیر مشخصه‌ی بومرنگ از اتصال دو مشخصه را می‌توان توسط یک لایه‌ی میانی به نام E_m مدل‌سازی کرد. این لایه معمولا شامل یک یا دو راند از تابع رمزگذاری است. از آن جایی که هر تابع رمزگذاری معمولا از یک لایه جعبه جانشینی استفاده می‌کند، لذا درست مانند حمله‌ی تفاضلی می‌توان مشخصه‌ی بومرنگ در کل الگوریتم رمزگذاری را به مشخصه‌های جعبه‌های جانشینی هر لایه کاهش داد.



شکل 9: لایه میانی در حمله‌ی ساندویچ که به جعبه‌های جانشینی مختلفی تقسیم می‌شود

با استفاده از روابط خطی پیش و پس از جعبه‌های جانشینی می‌توان دریافت که اگر β و γ تفاضل‌های لایه‌ی اتصال باشند، پس Δ_i و ∇_o تفاضل‌های یک جعبه جانشینی به خصوص هستند.



شکل 10: مشخصه‌ی اتصال هر جعبه جانشینی در لایه میانی

۳-۱ مفهوم

تعریف1. در تحلیل تفاضلی، جدول توزیع تفاضل ها[13] به صورت زیر تعریف می‌شود:

$$\pi(a,b) = \#\{x \in \{0,1\}^n | S(x) \oplus S(x \oplus a) = b\}$$

در واقع تعداد جواب‌های معادله‌ی $S(x) \oplus S(x \oplus a) = b$ در خانه‌ی (a,b) جدول قرار می‌گیرد.

در حمله‌ی بومرنگ، مفهومی بسیار مشابه به جدول توزیع تفاضل‌ها داریم.

[13] Differential Distribution Table (DDT)

تعریف2. فرض کنید که تفاضل بوجود آمده‌ی ورودی توسط الگوریتم رمز E_0 بر روی S-box برابر با Δ_i و تفاضل خروجی که در الگوریتم رمز E_1 تعریف شده است برابر با ∇_o باشد. حال می‌توان جدول BCT [13] را تشکیل داد به طوری که در خانه‌ی (Δ_i, ∇_o) دفعات بوجود آمدن 4تایی صحیح قرار داشته باشد.

$$\tau(\Delta_i, \nabla_o) = \#\{x \in \{0,1\}^n | S^{-1}(S(x) \oplus \nabla_o) \oplus S^{-1}(S(x \oplus \Delta_i) \oplus \nabla_o) = \Delta_i\}$$

در واقع تعداد جواب‌های معادله $S^{-1}(S(x) \oplus \nabla_o) \oplus S^{-1}(S(x \oplus \Delta_i) \oplus \nabla_o) = \Delta_i$ در خانه (a, b) قرار می‌گیرد. بدین ترتیب احتمال بوجود آمدن 4تایی صحیح به صورت زیر محاسبه می‌شود:

$$p_{\{\Delta_i, \nabla_o\}} = \frac{\tau(\Delta_i, \nabla_o)}{2^n}$$

۲-۳ مثال

به عنوان مثال جداول DDT و BCT الگوریتم رمز PRESENT به شرح زیر آمده است:

جدول1 : جدول توزیع تفاضل‌های (DDT) در جعبه جانشینی الگوریتم رمز PRESENT

Δ_o

Δ_i	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	4	0	0	0	4	0	4	0	0	0	4	0	0
2	0	0	0	2	0	4	2	0	0	0	2	0	2	2	2	0
3	0	2	0	2	2	0	4	2	0	0	2	2	0	0	0	0
4	0	0	0	0	0	4	2	2	0	2	2	0	2	0	2	0
5	0	2	0	0	2	0	0	0	0	2	2	2	4	2	0	0
6	0	0	2	0	0	0	2	0	2	0	0	4	2	0	0	4
7	0	4	2	0	0	0	2	0	2	0	0	0	2	0	0	4
8	0	0	0	2	0	0	0	2	0	2	0	4	0	2	0	4
9	0	0	2	0	4	0	2	0	2	0	0	0	2	0	4	0
a	0	0	2	2	0	4	0	0	2	0	2	0	0	2	2	0
b	0	2	0	0	2	0	0	0	4	2	2	2	0	2	0	0
c	0	0	2	0	0	4	0	2	2	2	2	0	0	0	2	0
d	0	2	4	2	2	0	0	2	0	0	2	2	0	0	0	0
e	0	0	2	2	0	0	2	2	2	2	0	0	2	2	0	0
f	0	4	0	0	4	0	0	0	0	0	0	0	0	0	4	4

جدول2 : جدول اتصالات بومرنگ در جعبه جانشینی الگوریتم رمز PRESENT

∇_o

Δ_i	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
1	16	0	4	4	0	16	4	4	4	4	0	0	4	4	0	0
2	16	0	0	6	0	4	6	0	0	0	2	0	2	2	2	0
3	16	2	0	6	2	4	4	2	0	0	2	2	0	0	0	0
4	16	0	0	0	0	4	2	2	0	6	2	0	6	0	2	0
5	16	2	0	0	2	4	0	0	0	6	2	2	4	2	0	0
6	16	4	2	0	4	0	2	0	2	0	0	4	2	0	4	8
7	16	4	2	0	4	0	2	0	2	0	0	4	2	0	4	8
8	16	4	0	2	4	0	0	2	0	2	0	4	0	2	4	8
9	16	4	2	0	4	0	2	0	2	0	0	4	2	0	4	8
a	16	0	2	2	0	4	0	0	6	0	2	0	0	6	2	0
b	16	2	0	0	2	4	0	0	4	2	2	2	0	6	0	0
c	16	0	6	0	0	4	0	6	2	2	2	0	0	0	2	0
d	16	2	4	2	2	4	0	6	0	0	2	2	0	0	0	0
e	16	0	2	2	0	0	2	2	2	2	0	0	2	2	0	0
f	16	8	0	0	8	0	0	0	0	0	0	8	0	0	8	16

۳-۳ توجیح اتصالات خاص

همانطور که گفته شد، اتصالات خاصی در [9], [10] معرفی شد که عبارتند از اتصال غیرممکن، اتصال نردبانی، اتصالی فیستلی و اتصال Sbox. در ادامه به این موضوع می‌پردازیم که چگونه می‌توان این اتصالات را با استفاده از BCT تشخیص داد.

۳-۳-۱ اتصال غیرممکن

زمانه که اتصال غیرممکن اتفاق بیفتد ، بومرنگ برنمیگردد، بدین معنی که احتمال پیدایش چهارتایی درست، صفر خواهد شد. از آن‌جایی که $p, q > 0$ پس به ناچار $r = 0$ که نتیجه می‌دهد

$$\tau(\Delta_i^*, \nabla_o^*) = 0$$

پس نقاطی از جدول BCT که صفر شده است، متعلق به اتصال‌های غیر ممکن است.

۳-۳-۲ اتصال نردبانی

اتصال نردبانی مروبط به زمانی است که در ورودی یا خروجی یک جعبه جانشینی، تفاضل صفر قرار داشته باشد. همانطور که در بخش قبل توضیح دادیم، در اینجا هیچ هزینه‌ای بابت این لایه داده نمی‌شود ، چرا که احتمال این اتصال 1 است، که نتیجه می‌دهد

$$\tau(\Delta_i^*, \nabla_o^*) = 2^n$$

۳-۳-۳ اتصال جانشینی

در این اتصال داشتیم که اگر تفاضل Δ_i در ورودی یک جعبه جانشینی و تفاضل Δ_o در خروجی آن باشد، آن‌گاه داریم

$$S^{-1}(S(x) \oplus \Delta_o) \oplus S^{-1}(S(x \oplus \Delta_i) \oplus \Delta_o) = \Delta_i$$

این بدین معنی است که اگر در خانه‌ی (Δ_i, Δ_o) از جدول DDT عددی غیر صفر مانند m باشد، در خانه‌ی (Δ_i, Δ_o) از جدول BCT نیز حداقل m وجود دارد.

با نتایج بدست آمده از مطالعه اتصال جانشینی، لم زیر به صورت مستقیم نتیجه می‌شود:

لم1.

$$\tau(a,b) \geq \pi(a,b)$$

حال سئوال اینجاست که آیا فرم بسته‌ای از $\tau(a,b)$ بر حسب $\pi(a,b)$ وجود دارد یا خیر. پاسخ مثبت است. در مقاله‌ی [14] برای اولین بار چنین فرمی ارائه شده است.

قضیه1:

تعریف می‌کنیم

$$\mathcal{U}_{a,b}^{S} = \{x \in \mathbb{F}_{2^n}: S(x) \oplus S(x \oplus a) = b\}$$

$$\mathcal{V}_{a,b}^{S} = \{S(x) \in \mathbb{F}_{2^n}: S(x) \oplus S(x \oplus a) = b\}$$

حال داریم

$$\tau(a,b) = \pi(a,b) + \sum_{\gamma \neq 0,b} \#(\mathcal{V}_{a,\gamma}^{S} \cap (\mathcal{V}_{a,\gamma}^{S} \oplus b))$$

۴-۳-۳ یکنواختی

همانطورکه به یاد داریم، یکی از راه‌های مقابله با حملات تفاضلی، کاهش بزرگترین خانه‌ی درج‌شده در جدول DDT بود. بدین منظور تعریف می‌کنیم:

$$\tau_s = \max_{\mathrm{a,b} \in \mathbb{F}_2^n, a \neq 0} \tau_s(a,b)$$

$$\pi_s = \max_{\mathrm{a,b} \in \mathbb{F}_2^n, a \neq 0} \pi_s(a,b)$$

حال به یک الگوریتم رمزی که $\pi_s = \pi$ داشت، می‌گفتیم الگوریتم π-یکنواخت، همچنین طیف تفاضلی یک الگوریتم به شکل زیر تعریف می‌شد

$$\{\pi_s(a,b), \in \mathbb{F}_2^n\{0\}, b \in \mathbb{F}_2^n\}$$

می‌دانیم که در تحلیل تفاضلی، کمترین میزان یکنواختی تفاضلی 2 است که به چنین توابع یا جعبه‌های جانشینی، APN [14] گفته می‌شود. برای جعبه‌های جانشینی $\mathbb{F}_2^n \to \mathbb{F}_2^n$ و n های فرد، پیدا کردن APNها کار سختی نیست[15] اما برای n های زوج تنها برای $n = 6$ یک

[14] Almost Perfect Nonlinear

APN پیدا شده و برای nهای دیگر به بهترین جواب دست‌یافته یعنی 4-یکنواخت بسنده شده است. به همین دلیل تمرکز اصلی این گزارش بر روی جعبه‌های 4-ینکواخت است.

حال سئوال طبیعی‌ای که پیش می‌آید این است که آیا می‌توان جعبه‌های جانشینی‌ای تعریف کرد که یکنواختی BCT آن‌ها کمینه باشد؟ در واقع همانطور که از روابط BCT و DDT پیداست، کمترین مقدار یکنواختی BCT یک الگوریتم برابر خواهد بود با مقدار یکنواختی DDT آن الگوریتم. پس بهتر است اینگونه بپرسیم که آیا جعبه‌های جانشینی با BCT 4-یکنواخت وجود دارد یا خیر؟ در پاسخ به این سئوال [13] نشان داد که این کار بسیار سخت است، اما پاسخ قطعی به این پرسش در [14] مطرح شد که در ادامه به تحلیل آن می‌پردازیم. در ابتدا لازم است یک لم بیان شود و از آن‌ها در تحلیل استفاده شود.

لم2: فرض کنید F و G دو جایگشت معادل همگر[15] از $\mathbb{F}_2^n$ باشند، بدین معنی که دو جایگشت همگر A_1 و A_2 وجود داشته باشد به طوری که $G = A_2 \circ F \circ A_1$ حال داریم:

$$\forall a, b \in \mathbb{F}_2^n: \quad \tau_G(a, b) = \tau_F(L_1(a), L_2^{-1}(b))$$

که در آن L_1 و L_2 بخش‌های خطی A_1 و A_2 هستند.

طبق لم بالا، یکنواختی BCT در جایگشت‌های همگر برابر هستند، بدین معنی که برای هر کلاس همگر، کافی است نماینده‌ی آن کلاس را بررسی کنیم و نیازی به بررسی کل کلاس نیست. در واقع این خاصیت به ما کمک کرد که فضای جستجوی خود را بسیار محدود کنیم. با گذاشتن شرط 4-یکنواخت بودن DDT و غیرخطی بودن بهینه $\mathcal{L}(S) = 8$ جدول زیر بدست می‌آید.

جدول3 : جدول آنالیز BCT برای نمایندگان کلاس‌های همگر جایگشت‌های 4-یکنواخت و 8-غیرخطی

	Representative	$\mathcal{L}(S)$	[DeC07]	[LP07]	n_0	n_2	n_4	n_6	n_8	n_{10}	n_{16}	tau
1	[8, 0, 1, 12, 15, 5, 6, 7, 4, 3, 10, 11, 9, 13, 14, 2]	8	3	G_3	120	60	15	30	0	0	0	**6**
2	[2, 0, 1, 8, 3, 11, 6, 7, 4, 9, 10, 15, 12, 13, 14, 5]	8	6	G_5	108	72	27	18	0	0	0	**6**
3	[8, 0, 1, 12, 2, 5, 6, 9, 4, 3, 10, 11, 7, 13, 14, 15]	8	2	G_6	104	80	27	10	4	0	0	**8**
4	[8, 0, 1, 9, 2, 5, 13, 7, 4, 6, 10, 11, 12, 3, 14, 15]	8	8	G_{11}	100	85	30	5	5	0	0	**8**
5	[4, 0, 1, 15, 2, 11, 6, 7, 3, 9, 10, 5, 12, 13, 14, 8]	8	1	G_{13}	105	78	28	11	2	1	0	**10**
6	[2, 0, 1, 8, 3, 13, 6, 7, 4, 9, 10, 5, 12, 11, 14, 15]	8	4	G_4	112	72	23	14	0	4	0	**10**
7	[2, 0, 1, 8, 3, 15, 6, 7, 4, 9, 5, 11, 12, 13, 14, 10]	8	5	G_7	105	80	30	5	0	5	0	**10**
8	[4, 8, 1, 2, 3, 11, 6, 7, 0, 9, 10, 14, 12, 13, 5, 15]	8	7	G_{12}	110	75	25	10	0	5	0	**10**
9	[8, 14, 1, 2, 3, 5, 6, 7, 4, 12, 10, 11, 9, 13, 0, 15]	8	9	G_9	108	69	28	14	5	1	0	**10**
10	[8, 14, 1, 2, 3, 5, 6, 7, 4, 9, 15, 11, 12, 13, 0, 10]	8	10	G_{14}	108	70	27	13	6	1	0	**10**
11	[8, 15, 1, 2, 3, 5, 12, 7, 4, 9, 10, 11, 6, 13, 14, 0]	8	11	G_{15}	108	70	27	13	6	1	0	**10**
12	[8, 15, 1, 2, 3, 5, 6, 13, 4, 9, 10, 11, 12, 7, 14, 0]	8	12	G_{10}	108	69	30	12	3	3	0	**10**
13	[12, 0, 1, 9, 3, 5, 4, 7, 6, 2, 10, 11, 8, 13, 14, 15]	8	13	G_2	107	64	32	8	12	0	2	**16**
14	[12, 11, 1, 2, 3, 5, 4, 7, 6, 9, 10, 0, 8, 13, 14, 15]	8	14	G_1	107	60	36	12	8	0	2	**16**
15	[12, 9, 1, 2, 3, 5, 4, 7, 6, 0, 10, 11, 8, 13, 14, 15]	8	15	G_8	103	72	32	0	16	0	2	**16**
16	[8, 14, 1, 2, 3, 5, 4, 7, 6, 9, 10, 0, 12, 13, 11, 15]	8	16	G_0	107	64	32	8	12	0	2	**16**

در جدول بالا مشاهده می‌کنیم که کمترین مقدار یکنواختی BCT 6 است. پس هیچ جعبه جانشینی‌ای وجود ندارد که به طور همزمان در دو جدول BCT و DDT 4-یکنواخت باشد.

[15] Affine Equivalent

فصل 4

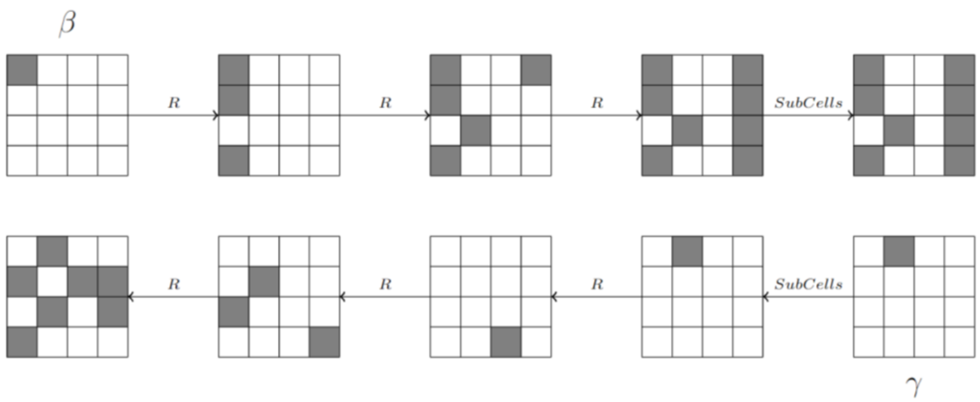
۴ گسترش لایه‌ی میانی

همانطور که در بخش 2.2 گفته شد، لایه‌ی میانی وظیفه‌ی مدل سازی وابستگی مشخصه‌ی بالایی و پایینی را برعهده داشت، بدین صورت که با انتخاب مناسب مشخصه‌ها در نقطه‌ی اتصال می‌توانستیم دو مشخصه را از یکدیگر مستقل کنیم و احتمال گذار از یک مشخصه به مشخصه‌ی دیگر را 1 کنیم. همچنین در فصل قبل دیدیم که با در نظر گرفتن یک لایه جعبه جانشینی در E_m می‌توانیم از ابزار قدرتمندی به نام BCT استفاده کنیم.حال سئوال مهمی که مطرح می‌شود این است که E_m می‌تواند تا چه اندازه بزرگ باشد؟ آیا می‌تواند چند لایه جعبه جانشینی داشته باشد؟ اگر چندلایه باشد آیا می‌توان از BCT استفاده کرد؟

برای پاسخ به سئوالات بالا می‌توان از لم زیر شروع کرد که بسیار شهودی است و می‌توان گفت، تعمیم سوییچ نردبانی است که در بخش دوم مطرح کردیم.

لم3. [15]در E_m اگر پخش پیشروی تفاضل β هیچ تداخلی با پخش پَسروی γ نداشته باشد، احتمال تولید یک چهارتایی برای E_m، 1 خواهد بود.

برای توجیح لم بالا به مثال زیر توجه کنید.

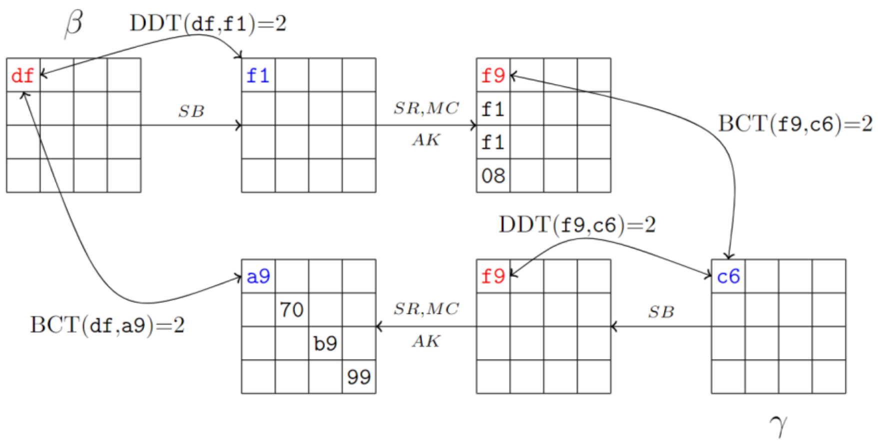


شکل 11: 4 دور از الگوریتم SKINNY برای E_m

همانطور که در شکل بالا مشاهده می‌شود در هر لایه، تفاضل‌های پخش‌شده از سمت β و تفاضل‌های پخش‌شده از سمت γ هیچ تداخلی ندارند، بدین معنی هیچ کدام از خانه‌های فعال دو تفاضل روی هم نیفتاد. این نتیجه می‌دهد که در هر جعبه جانشینی یا Δ_i و یا ∇_o صفر هستند. با استفاده از لم بالا می‌توان دریافت که E_m تا جایی می‌تواند ادامه پیدا کند که شرط لم صادق بماند.

۴-۱-۱ جدول اتصالات بومرنگ برای لایه‌ی میانی چند دوری

پس از در نظر گرفتن لم بخش قبل، حال سئوال بعدی این است که اگر شرایط لم اتفاق نیفتد، چگونه می‌توان در حالت کلی احتمال گذار را تعیین کرد؟ آیا کماکان BCT ابزار مناسبی است؟ پاسخ منفی است. به عنوان مثال به سناریوی شکل زیر توجه کنید.



شکل 12: ناکارآمدی BCT برای محاسبه احتمال گذار در لایه میانی چند دوری (2 دور AES)

در شکل بالا دو لایه جعبه جانشینی وجود دارد. خانه $(0,0)$ را در نظر بگیرید. در این خانه دو گذار در هر لایه از جعبه‌های جانشینی انجام میگیرد که هر کدام از آن‌ها با جدول BCT مشخص می‌شود. برای هر دو گذار 2 پاسخ وجود دارد بدین معنی که با یک احتمال غیر صفر چهارتایی صحیح برای این ساختار وجود دارد، لکن با بررسی خانه‌ی $(0,0)$ در دور اول E_m می‌توان دریافت که جواب درست باید در دو معادله زیر صدق کند:

$$S^{-1}(S(x) \oplus a9) \oplus S^{-1}(S(x \oplus df) \oplus a9) = df$$

$$S(x) \oplus S(x \oplus df) = f1$$

حال واقعیت این است که این دستگاه معادلات هیچ جوابی ندارد، لذا نتیجه می‌گیریم که تحلیل با BCT اشتباه بوده است.

۴-۲ جدول اتصالات بالایی و پایینی بومرنگ

برای حل این مشکل، جداول جدیدی پیشنهاد شد که برای لایه‌های میانی بیشتر از یک راند قابل استفاده می‌باشد. این جداول در [15] به نام‌های BDT و BDT' معرفی شدند، منتها در [16] نام‌های دیگری برای آن‌ها پیشنهاد شد که ما از آن‌ها استفاده می‌کنیم.

تعریف. جدول اتصالات پایینی بومرنگ[۱۶]

$$LBCT(\Delta_0, \Delta_1, \nabla_0) = \#\{x \in \{0,1\}^n | S^{-1}(S(x) \oplus \nabla_0) \oplus S^{-1}(S(x \oplus \Delta_0) \oplus \nabla_0) = \Delta_0 , S(x) \oplus S(x \oplus \Delta_0) = \Delta_1\}$$

تعریف. جدول اتصالات بالایی بومرنگ[۱۷]

$$UBCT(\nabla_0, \nabla_1, \Delta_0) = \#\{x \in \{0,1\}^n | S(S^{-1}(x) \oplus \Delta_0) \oplus S(S^{-1}(x \oplus \nabla_0) \oplus \Delta_0) = \nabla_0, S^{-1}(x) \oplus S^{-1}(x \oplus \nabla_0) = \nabla_1\}$$

همچنین جدول دیگری برای لایه‌های میانی با بیشتر از 2 دور در [2] پیشنهاد شد منتها تعریف نشد، این جدول در [3] به صورت زیر تعریف شد:

تعریف. جدول اتصالات تعمیم‌یافته بومرنگ[۱۸]

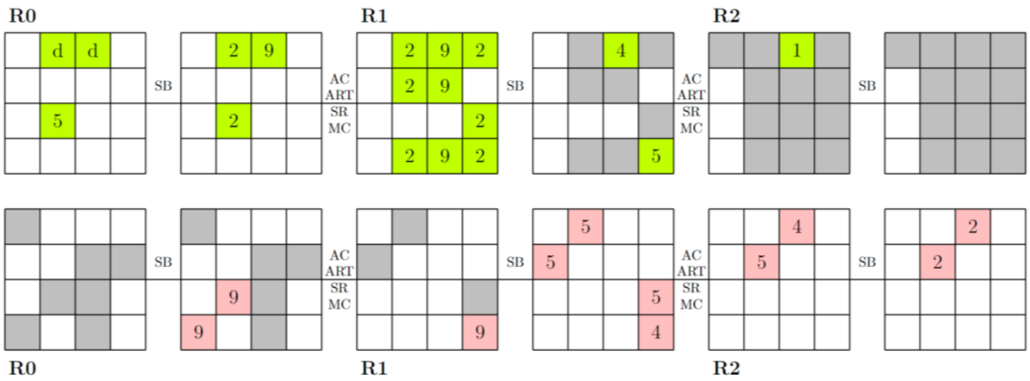
[16] Lower Boomerang Connectivity Table (LBCT)
[17] Upper Boomerang Connectivity Table (UBCT)
[18] Extended Boomerang Connectivity Table (EBCT)

$$EBCT(\Delta_0, \Delta_1, \nabla_0, \nabla_1) = \#\{x \in \{0,1\}^n | S^{-1}(S(x) \oplus \nabla_0) \oplus S^{-1}(S(x \oplus \Delta_0) \oplus \nabla_0) \\ = \Delta_0 , S(x) \oplus S(x \oplus \Delta_0) = \Delta_1, S(x) \oplus S(x \oplus \nabla_1) = \nabla_0\}$$

حال با استفاده از این جداول می‌توان احتمال گذار یک لایه‌ی میانی با چند دور را محاسبه کرد. به عنوان مثال فرض کنید که دور میانی شامل 3 دور الگوریتم SKINNY-64 باشد که در آن تفاضل‌های بالایی و پایینی به شکل زیر باشند:

$$\Delta_0 = [0, d, d, 0,0,0,0,0,0,5,0,0,0,0,0,0]$$

$$\nabla_0 = [0,0,2,0,0,2,0,0,0,0,0,0,0,0,0,0]$$



شکل 13: مشخصه‌ی بومرنگ در لایه‌ی میانی الگوریتم SKINNY-64

در مثال بالا خانه‌های رنگی تفاضل معلوم، خانه‌های سفید تفاضل صفر و خانه‌های خاکستری تفاضل نامعلوم دارند. در ردیف اول، پیشروی تفاضل بالایی و در ردیف دوم، پسروی تفاضل پایینی را می‌توان مشاهده کرد.حال برای محاسبه احتمال این گذار باید به شکل زیر و به صورت لایه به لایه عمل کرد:

- **لایه 0:** در این لایه تنهار 3 جعبه جانشینی فعال هستند. مشاهده می‌شود که جعبه‌های شماره 1 و 2 دارای تفاضل پایینی نیستند، لذا کافی است احتمال رخداد تفاضل را توسط جدول DDT محاسبه کنیم. همچنین جعبه شماره 9 با تفاضل مشخص در تفاضل بالایی و تشکیل یک حلقه‌ی بومرنگ احتمال خود را از جدول UBCT دریافت می‌کند.
- **لایه‌های بعدی:** به همان شکلی که در لایه 0 عمل کردیم.

حال مقدار این احتمال برابر می‌شود با

$$P\left(\Delta_0 \overset{E}{\rightleftarrows} \nabla_0\right) \\ = [\Pr_{DDT}(d,2).\ \Pr_{DDT}(d,9).\ \Pr_{UBCT}(5,2,9)].[\Pr_{BCT}(2,5).\ \Pr_{DDT}(9,4).\ \Pr_{BCT}(2,5).\ \Pr_{EBCT}(2,5,9,4)].[\Pr_{LBCT}(1,4,2).\ \Pr_{DDT}{}^2(5,2)]$$

۴-۳ یافتن احتمال تعمیم یافته

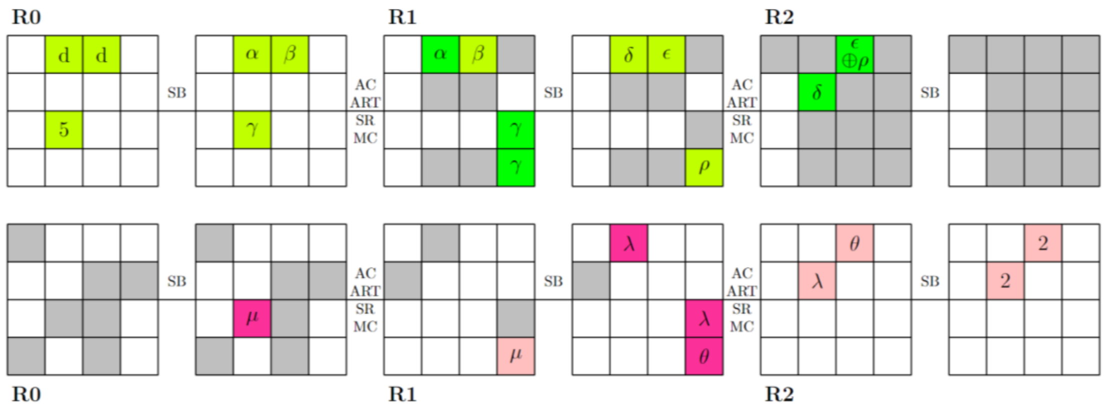
همانطور که در بخش قبل دیدیم، احتمال‌های لایه میانی براساس جداول اتصالات بالایی و پایینی و تعمیم یافته محاسبه می‌شد که در آن تفاضل هر دور از لایه‌ی میانی ثابت فرض شده بود. اما در حمله‌ی بومرنگ تنها چیزی که برای حمله‌کننده اهمیت دارد، تفاضل ورودی بالایی و پایینی است. محدود کردن تفاضل‌های لایه‌های میانی باعث کاهش احتمال حمله خواهد شد. به همین دلیل می‌بایست روی حالات میانی مختلف، احتمال را جمع بست به شکلی که هر کدام باعث بوجود آمدن چهارتایی بومرنگ شوند. برای این کار الگوریتم زیر پیشنهاد داده می‌شود:

الگوریتم 3: حمله تمایزگری بومرنگ [16]

ورودی: یک تعداد دور مشخص، تفاضل ورودی مشخصه بالایی، تفاضل خروجی مشخصه پایینی

1. صفرهای تفاضل ورودی بالایی را به سمت خروجی و صفرهای تفاضل خروجی پایینی را به سمت ورودی گسترش بده.
2. برای هر ورودی جعبه جانشینی در مشخصه بالایی با تفاضل ناصفر و تفاضل پایینی ناصفر، خانه را علامت گذاری کن. این کار را برای مشخصه‌ی پایینی نیز به صورت معکوس انجام بده. توجه داشته باشید که خانه‌های لایه اول و آخر علامت‌گذاری نمی‌شوند.
3. هر خانه علامت گذاری شده‌ای که به یک خانه علامت گذاری شده در لایه‌ی بعدی می‌رود را به صورت بازگشتی علامت گذاری کن.
4. برای هر خانه‌ی علامت گذاری شده در لایه بالایی و پایینی مشخصه یک متغیر تولید کن، به شکلی که ورودی جعبه‌های جانشینی براساس رابطه‌ی خطی‌ای از بقیه متغیرها بدست بیایند. همین کار را به صورت برعکس برای مشخصه‌ی پایینی انجام می‌دهیم.
5. با استفاده از فرمول جمع احتمالات، احتمال کل را بر روی حالات مختلف متغیرهای میانی حساب کن و خروجی بده.

برای توضیح بیشتر، مراحل بالا را برای همان مثال بخش قبل اجرا می‌کنیم.



شکل 14: مشخصه‌ی بومرنگ در لایه‌ی میانی الگوریتم SKINNY-64 پس از اجرای الگوریتم 3

مرحله اول: همانطور که می‌بینید صفرهای لایه R_0 به صورت مستقیم در مشخصه‌ی بالایی و صفرهای لایه R_2 به صورت معکوس در لایه‌ی پایینی منتشر می‌شوند و خانه‌های خالی و خاکستری را پدید می‌آورند. در واقع در مرحله‌ی اول خانه‌های رنگی، خاکستری هستند.

مرحله دوم: خانه‌های ورودی جعبه‌های جانشینی (به غیر از لایه اول) مشخصه‌ی بالایی که در مشخصه بالا و پایین تفاضل غیرصفر دارند را با زنگ سبز علامت میزنیم. همچنین خانه‌های ورودی جعبه‌های جانشینی (به غیر از لایه دوم) مشخصه‌ی پایینی که در مشخصه بالا و پایین تفاضل غیرصفر دارند را با زنگ بنفش علامت میزنیم.

مرحله سوم: حال باید به صورت بازگشتی خانه‌ها را رنگ‌آمیزی کرد. برای مثال خانه 2 و 5 در ورودی جعبه جانشینی دور دوم را در مشخصه‌ی بالایی در نظر بگیرید. برای دانستن این خانه‌ها می‌بایست خانه‌های 1 ، 2 و 15 رادر لایه قبل بدانیم، به همین دلیل این خانه‌ها به رنگ لیمویی در می‌آیند. همچنین همین کار را در مشخصه‌ی پایین با رنگ صورتی انجام می‌دهیم.

مرحله چهارم: برای خانه‌های علامت‌گذاری شده لایه‌های خروجی هر جعبه جانشینی در مشخصه‌ی بالایی یک متغیر در نظر می‌گیریم که عبارتند از $\alpha, \beta, \gamma, \rho, \delta, \epsilon$. سپس لایه‌های بعدی آن‌ها که ورودی جعبه‌های جانشینی می‌باشد را بر اساس این متغیرها محاسبه می‌کنیم.

مرحله پنجم: حال رابطه بومرنگ را برای هر لایه می‌نویسیم و روی متغیرهای میانی جمع می‌بندیم

$$P\left(\Delta_0 \overset{E}{\rightleftarrows} \nabla_0\right) = \sum_{\alpha,\dots,\mu\in\mathbb{F}_2^n} [\Pr_{DDT}(d,\alpha). \Pr_{DDT}(d,\beta). \Pr_{UBCT}(5,\gamma,\mu)]. [\Pr_{UBCT}(\alpha,\delta,\lambda). \Pr_{DDT}(\beta,\epsilon). \Pr_{BCT}(\gamma,\lambda). \Pr_{EBCT}(\gamma,\rho,\mu,\theta)]. [\Pr_{LBCT}(\epsilon \oplus \rho,\theta,2). \Pr_{LBCT}(\delta,\lambda,2)]$$

که این مقدار بسیار بیشتر از یکی از عبارات حاضر در سیگما است.

۵ جمع‌بندی و مراجع

در این گزارش به یکی از تکنیک‌های مهم حمله به رمزهای قالبی به نام حمله‌ی بومرنگ پرداخته شد و فهمیدیم که چگونه می‌توان با استفاده از خاصیت بومرنگ در رمزهای قالبی، یک تمایزگر ساخت. یکی از مهم‌ترین نتایجی که می‌توان از حمله‌ی بومرنگ گرفت این است که مقاومت یک رمز قالبی در مشخصه‌های کامل، هیچ ارتباطی به مقاومت آن در حمله‌ی بومرنگ نخواهد داشت و می‌بایست از روش‌های جدیدی برای مقاومت در برابر این نوع حملات استفاده کرد. البته لازم به ذکر است که کاهش احتمال رخداد تفاضل‌ها در هر دور، کماکان روش موثری در مقابله با حملات بومرنگ به حساب می‌آید چرا که بالاخره این حمله از چند دور از تفاضل‌ها استفاده می‌کند و کاهش احتمال این تکه تفاضل‌ها در احتمال کل و پیچیدگی حمله تاثیر خواهد داشت.

پس از شناخت BCT و حمله‌ی ساندویچ به عنوان یک ابزار تحلیل پیچیدگی مناسب، تقریبا یک روش استاندارد دفاع در برابر حملات بومرنگ معرفی شد و آن‌هم یکنواخت کردن و کاهش اعداد جدول BCT است. در واقع این تکنیک همان تکنیکی است که در دفاع در مقابل حملا تفاضلی استفاده شد و آن هم کاهش عدد یکنواختی[19] جدول BCT است. خوبی این روش این است که با کاهش عدد یکنواختی این جدول، اعداد یکنواختی جدول LBCT ، UBCT و EBCT نیز کاهش می‌یابد که این اتفاق بدین معناست الگوریتم رمز ما در مقابل حملاتی با لایه‌ی میانی چنددوری نیز امن خواهد ماند. اما دیدیم که در جعبه‌های جانشینی 4یکنواخت تفاضلی، BCT 4-یکنواخت وجود ندارد و کمترین مقدار یکنواختی آن 6 است.

در بخش بعدی به این سئوال طبیعی پاسخ دادیم که اگر لایه‌ی میانی از چند دور تشکیل شده باشد چه اتفاقی می‌افتد؟ مشاهده کردیم که دیگر BCT برای تحلیل کافی نیست و پاسخ غلطی در اختیار ما می‌گذارد. بدین ترتیب جداول جدیدی به نام‌های EBCT ، UBCT و LBCT معرفی شدند. سپس الگوریتمی برای یافتن احتمال گذار در لایه میانی پیشنهاد شد که با استفاده از آن می‌توان احتمال تمام گذارهای ممکن در لایه میانی که ابتدا و انتهای ثابتی دارند را محاسبه کرد.

۵-۱ کارهای آتی

در پژوهش‌های انجام شده لایه میانی شامل جعبه‌های جانشینی و تبدیلات خطی است، اما در تکنیک همبستگی زدایی در میانه‌ی الگوریتم ماژول همبستگی زدایی قرار داده می‌شود. در این حالت این ماژول حتما می‌بایست در لایه‌ی میانی ما قرار بگیرد چرا که در صورت قرارگیری در مشخصه‌ی بالایی و پایینی، احتمال مشخصه‌ها را به شدت کاهش می‌دهد. حال سئوالی که پیش می‌آید این است که آیا می‌توان با وجود این ماژول احتمال اتصال را در لایه میانی حساب کرد؟

یکی از پرسش‌های دیگری که پیش می‌آید این است که رابطه بده بستان یکنواختی DDT، ینکواختی BCT و میزان غیرخطی‌بودن یک جعبه جانشینی چگونه است؟ در فصل دوم دیدیم که با فرض 4-یکنواخت بودن DDT و بهینه بودن غیرخطی بودن جعبه‌های جانشینی، در بهترین حال BCTهای 6-یکنواخت داریم. حال می‌توان پرسید در چه حالت‌هایی می‌توانیم جعبه‌هایی بسازیم که BCT 6-یکنواخت داشته باشد.

همچنین یک مسیر پژوهشی دیگر جستجوی مسیر بومرنگ بهینه است. مسیر بومرنگ بهینه مسیری است که کمترین پیچیدگی داده و بالاترین احتمال رخداد را داشته باشد. برای این کار باید لایه‌ی میانی را تا جای ممکن بزرگ در نظر بگیریم و هیچ هزینه‌ای در اتصال ندهیم. طبق بحثی که در بخش 4 داشتیم، اگر پیشروی یک خانه‌ی فعال در مشخصه‌ی بالایی با هیچ خانه‌ی فعالی از مشخصه‌ی پایینی و

[19] Uniformity

پیشروی آن قرار نگیرد، سوییچ نردبانی فعال می‌شود و لایه‌ی میانی هیچ هزینه‌ای برای ما نخواهد داشت. حال آیا می‌توان مشخصه‌های بالا و پایینی طراحی کرد که اندازه لایه پایینی بیشینه شود با این شرط که سوییچ ما نردبانی باشد؟ در مقابل این پژوهش می‌توان به روش‌های دفاعی نیز فکر کرد. به عنوان مثال می‌توان لایه‌های پخشی[۲۰] را به شکلی طراحی کرد که خانه‌های فعال حتما در مقابل همدیگر قرار بگیرد تا هزینه‌ی پرداختی در لایه‌ی میانی افزایش یابد.

۵-۲ مراجع

[1] National Institute of Standards and Technology, "FIPS-46: Data Encryption Standard (DES)," 1979.

[2] H. M. Heys, "A TUTORIAL ON LINEAR AND DIFFERENTIAL CRYPTANALYSIS," *http://dx.doi.org/10.1080/0161-110291890885*, vol. 26, no. 3, pp. 189–221, Jul. 2010, doi: 10.1080/0161-110291890885.

[3] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," *Journal of Cryptology 1991 4:1*, vol. 4, no. 1, pp. 3–72, Jan. 1991, doi: 10.1007/BF00630563.

[4] E. Biham and A. Shamir, "Differential Cryptanalysis of the Data Encryption Standard," *Differential Cryptanalysis of the Data Encryption Standard*, 1993, doi: 10.1007/978-1-4613-9314-6.

[5] S. Vaudenay, "Provable security for block ciphers by decorrelation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1373 LNCS, pp. 249–275, 1998, doi: 10.1007/BFB0028566/COVER/.

[6] D. Wagner, "The boomerang attack," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1636, pp. 156–170, 1999, doi: 10.1007/3-540-48519-8_12/COVER/.

[7] L. R. Knudsen, "Truncated and higher order differentials," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1008, pp. 196–211, 1995, doi: 10.1007/3-540-60590-8_16/COVER.

[8] J. Kelsey, T. Kohno, and B. Schneier, "Amplified boomerang attacks against reduced-round MARS and serpent," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1978, pp. 75–93, 2001, doi: 10.1007/3-540-44706-7_6/COVER/.

[9] S. Murphy, "The return of the cryptographic boomerang," *IEEE Transactions on Information Theory*, vol. 57, no. 4, pp. 2517–2521, Apr. 2011, doi: 10.1109/TIT.2011.2111091.

[10] A. Biryukov and D. Khovratovich, "Related-key cryptanalysis of the full AES-192 and AES-256," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5912 LNCS, pp. 1–18, 2009, doi: 10.1007/978-3-642-10366-7_1/COVER.

[11] H. Boukerrou, P. Huynh, V. Lallemand, B. Mandal, and M. Minier, "On the Feistel Counterpart of the Boomerang Connectivity Table," *IACR Transactions on Symmetric Cryptology*, vol. 2020, no. 1, pp. 331–362, May 2020, doi: 10.13154/TOSC.V2020.I1.331-362.

[12] O. Dunkelman, N. Keller, and A. Shamir, "A practical-time related-key attack on the KASUMI cryptosystem used in GSM and 3G telephony," *Journal of Cryptology*, vol. 27, no. 4, pp. 824–849, Jul. 2014, doi: 10.1007/S00145-013-9154-9/TABLES/5.

[20] Diffusion Layer

[13] C. Cid, T. Huang, T. Peyrin, Y. Sasaki, and L. Song, "Boomerang connectivity table: A new cryptanalysis tool," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10821 LNCS, pp. 683–714, 2018, doi: 10.1007/978-3-319-78375-8_22/TABLES/14.

[14] C. Boura and A. Canteaut, "On the Boomerang Uniformity of Cryptographic Sboxes," *IACR Transactions on Symmetric Cryptology*, vol. 2018, no. 3, pp. 290–310, Sep. 2018, doi: 10.13154/TOSC.V2018.I3.290-310.

[15] H. Wang and T. Peyrin, "Boomerang Switch in Multiple Rounds. Application to AES Variants and Deoxys," *IACR Transactions on Symmetric Cryptology*, vol. 2019, no. 1, pp. 142–169, Mar. 2019, doi: 10.13154/TOSC.V2019.I1.142-169.

[16] S. Delaune, P. Derbez, and M. Vavrille, "Catching the Fastest Boomerangs," *IACR Transactions on Symmetric Cryptology*, vol. 2020, no. 4, pp. 104–129, Dec. 2020, doi: 10.46586/TOSC.V2020.I4.104-129.