ارزیابی غیر خطی بودن توابع رمزنگاری به کمک شبکه عصبی

حسام محمدحسینی
دانشکده فنی و مهندسی،
دانشگاه تربیت مدرس
hesam.mhosseini@gmail.com

بهروز حاجیان‌نژاد
دانشکده مهندسی برق، دانشگاه
صنعتی خواجه نصیرالدین طوسی
behrouz.hajian@gmail.com

احمدرضا شرافت
دانشکده فنی و مهندسی،
دانشگاه تربیت مدرس
sharafat@isc.iranet.net

چکیده: در این مقاله با استفاده از شبکه عصبی پرسپترون چند لایه، روشی برای بررسی میزان غیرخطی بودن S-boxها ارائه کرده‌ایم. هر S-box تابعی بولی را مشخص می‌کند که m بیت ورودی را به n بیت خروجی، $m \to n$ ، تبدیل می‌کند. با توجه به اینکه در رمزهای قطعه‌ای، تنها بخش غیرخطی الگوریتم، S-boxها هستند، میزان مقاومت رمزهای قطعه‌ای در برابر حملات، عموما تنها به مناسب بودن طراحی S-boxها، یعنی به غیرخطی بودن آنها، بستگی دارد. در روش پیشنهادی، میزان غیرخطی بودن یک S-box با تعداد m بیت ورودی و تعداد n بیت خروجی به صورت حداقل تعداد نورون‌ها در لایه میانی شبکه عصبی، برای شناسایی دیتای ورودی و خروجی S-box تعریف گردیده است. به این ترتیب می‌توان تقریبی از میزان غیرخطی بودن هر تابع بولی را بدست آورد. از این روش می‌توان برای بررسی مناسب بودن S-box های جدید در سیستم‌های رمز قطعه‌ای استفاده کرد، به خصوص در مواردی که بزرگی ابعاد S-boxها، تحلیل ریاضی معیارهای مختلف غیرخطی بودن را بسیار پیچیده می‌کند. به کمک این روش، S-boxهای اصلی سیستم DES، یعنی S-boxهای $4 \to 6$ و نیز S-boxهای $4 \to 4$ موجود در DES مطالعه و شناسایی گردیده‌اند. این نتایج همچنین تاییدی بر وابسته بودن طراحی S-boxهای DES به یکدیگر است.

واژه های کلیدی: توابع باینری، غیرخطی بودن، سیستم رمز قالبی، شبکه‌های عصبی پرسپترون چند لایه، DES، S-box.

۱- مقدمه

مهمترین بخش در هر الگوریتم رمز قطعه‌ای، تابع دور (Round Function) است. تابع دور مجموعه‌ای از چندین عضو به همراه یک جزء اصلی و مهم به نام S-box (Substitution Box) است که تحت یک معماری و ساختار مشخص، سازماندهی می‌شوند. تابع دور ممکن است شامل یک یا چند S-box در مجموعه خود باشد. در الگوریتم های DES [۱]، Twofish [۲]، و Rijndael [۳] توابع دور شامل چندین S-box هستند. به غیر از S-box، سایر اجزاء در مجموعه تابع دور لزوماً عناصری غیر خطی نبوده بلکه عموماً تبدیلاتی خطی بوده که با هدف افزایش شباهت بین S-box ها یا به منظور ایجاد مصونیت در مقابل حمله‌ای خاص در نظر گرفته می‌شوند. در حقیقت S-box، عضو اصلی در مجموعه تابع دور و غیرخطی بودن آن شرط لازم برای کفایت تابع دور است. هر S-box یک جانشینی از 2^m ورودی به 2^n خروجی است، که با توجه به الگوریتم مورد استفاده، می‌تواند

وابسته به کلید الگوریتم (بطور دقیق تر زیر کلید دور مربوطه در الگوریتم) و یا مستقل از کلید باشد. معمولا S-boxها چندین جایگشت دارند که در یک ساختار مشخص، تحقق بسیاری از ویژگی‌های مطلوب رمزنگاری بر عهده آنهاست. بطور مثال S-box های الگوریتم DES، شش بیت ورودی و چهار بیت خروجی دارند که در داخل خود از چهار جایگشت خاص با چهار بیت ورودی و چهار بیت خروجی استفاده می‌کنند. در مجموعه هشت S-box بکار رفته در تابع دور الگوریتم DES، از ۳۲ جایگشت مربعی $4 \to 4$ استفاده شده است. مصداقی دیگر S-box های الگوریتم رمز قطعه‌ای Twofish است که دارای ۸ بیت ورودی و ۸ بیت خروجی بوده و به مانند DES (ولی در یک معماری متفاوت) از جایگشت‌های مربعی $4 \to 4$ در داخل خود استفاده می‌کنند.

ساختاری که جایگشت‌های داخلی S-box تحت آن سازماندهی می‌شوند، جزو ویژگی‌های هر الگوریتم بوده و به شکل‌های متفاوتی طراحی می‌شود. به بیان دیگر، در اختیار داشتن جایگشت‌های مناسب رمزنگاری، شرط لازم برای تحقق یک S-box خوب و بکارگیری آنها تحت یک ساختار صحیح در S-box شرط کافی برای این مهم است. جایگشت‌های اشاره شده، که کوچکترین (و مهمترین) عضو در یک الگوریتم رمز قطعه‌ای محسوب می‌شوند، از نظر محتوایی، توابع بولی با خواص رمزنگاری هستند که باید اهداف رمزنگاری مختلفی را برآورده سازند. تاکنون معیارهای متعددی بمنظور ارزیابی توابع بولی جهت بکارگیری در کاربردهای رمزنگاری مطرح شده‌اند که در حالت کلی، آنها را با سه معیار زیر دسته بندی کرده‌اند.

- میزان غیر خطی بودن
- میزان همبستگی بین ورودی و خروجی تابع
- میزان انتشار

خانواده اول، بصورت حداقل فاصله همینگ موجود بین مؤلفه های باینری تابع و تابع آفاین معادل تعریف می‌شود [٤]. خانواده دوم، شامل معیارهای تعیین میزان همبستگی خروجی و ورودی توابع است. مشخصه‌هایی نظیر متوازن بودن[1] [٥]، منظم بودن[2] [٦]، ارتجاعی[3] بودن[٧] و مصونیت همبستگی[4] [٨] در این خانواده قرار دارند. خانواد سوم مربوط به خواص انتشار و نحوه ارزیابی آنهاست. معیارهائی نظیر بهمنی[5] [٩]، کامل بودن[6] [١٠]، بهمنی اکید[7] [١١]، بهمنی اکید مراتب بالا[8] [١٢]، و انتشار از درجه k[9] [١٣]، برخی از اعضای خانواده سوم به شمار می‌روند.

یافتن تابعی که تمامی ویژگی‌ها را در حد کمال دارا باشد امکان پذیر نبوده و افزایش یک ویژگی، کاهش ویژگی‌های دیگر یا از دست رفتن برخی خصوصیات را به همراه خواهد داشت. بر این اساس جستجوی توابعی که معیارهای بیشتری را بصورت همزمان محقق سازند و تعیین نقطه بیشینه و بهینه در بین تمامی معیارها و ویژگی‌های مطلوب رمزنگاری همواره یکی از مباحث مورد توجه در رمزشناسی بوده است [١٤]، [١٥]، و [١٦].

جستجوی تابع مناسب از نقطه نظر تمامی معیارها و تعیین نقطه بهینه در بین همه ویژگی‌ها، مستلزم در اختیار داشتن یک معیار مادر یا تابع کفایت[10] مناسب است. تا کنون توابع کفایت مختلفی بمنظور ارزیابی توابع بولی جهت کاربردهای رمزنگاری معرفی شده و تلاش شده است تا حد امکان در برگیرنده تعداد بیشتری معیارهای رمزنگاری بوده و خروجی تابع، به نقطه بهینه در بین تمامی ویژگی‌ها نزدیک باشد.

در این مقاله روشی نو بمنظور ارزیابی توابع بولی معرفی می‌کنیم. ابزار مورد نظر، شبکه عصبی و سنجه مورد استفاده، پیچیدگی پیاده‌سازی عصبی تابع مورد ارزیابی است. به این منظور چندین تابع بولی $4 \to 4$ (٤ بیت ورودی و ٤ بیت خروجی) قوی و ضعیف از منظر رمزنگاری را انتخاب کرده و به روش اشاره شده مورد ارزیابی قرار گرفته‌اند. توابع بولی قوی مورد آزمون، توابع مورد استفاده در S-box های الگوریتم DES هستند که در فضای توابع بولی $4 \to 4$ جزو بهترین‌ها از حیث معیارهای رمزنگاری پیش گفته هستند. توابع ضعیف مورد استفاده از بین توابع بولی $4 \to 4$ و با ویژگی‌های محدود رمزنگاری ساخته شده‌اند. توابع ضعیف و قوی منتخب با استفاده از شبکه عصبی پیاده‌سازی شده و پیچیدگی پیاده‌سازی آنها (همانطور که به تفصیل در بخش‌های بعدی بیان شده است) نیز تأیید کننده این موضوع هستند. آزمون تجربی ارزیابی کفایت توابع بولی جهت کاربردهای رمزنگاری با

استفاده از شبکه عصبی بر روی چندین تابع بولی کوچک ضعیف و قوی، و کسب نتایج مورد انتظار، بکارگیری این روش را برای توابع بزرگتر (که ارزیابی آنها از طریق سایر روشها مشکل است) ممکن می‌سازد.

در ادامه، و نظر به اینکه از توابع $4 \to 4$ الگوریتم DES به عنوان توابع قوی استفاده شده است، نحوه بکارگیری توابع مذکور در مجموعه S-box ها و در ساختار تابع دور الگوریتم اشاره شده، به اختصار معرفی می‌شود. در بخش سوم، جزییات شبکه‌های عصبی مورد استفاده برای شناسایی S-box های DES $6 \to 4$ را بیان نموده و نتایج هر شبکه را ارائه کرده‌ایم. در بخش ٤ به مطالعه S-boxهای $4 \to 4$موجود در DES پرداخته‌ایم. در بخش ٥ تعدادی S-box $4 \to 4$ ضعیف، از نظر میزان غیرخطی بودن، در مقایسه با چهار S-box $4 \to 4$ موجود در DES S-box1 طراحی شده‌اند و سرانجام در بخش ٦، نتیجه گیری و پیشنهاد برای استفاده از نتایج بدست آمده به منظور ارزیابی توابع، ارائه شده است.

۲- نحوه عملکرد سیستم رمز قالبی DES

الگوریتم DES به عنوان اولین رمزقطعه‌ای استاندارد از زمان معرفی توسط NSA در سال ۱۹۷۷ مورد بررسی‌های فراوانی قرار گرفته است. این الگوریتم رمز قطعه‌ای، هر قطعه ٦٤ بیتی از ورودی (متن اصلی) را بوسیله کلید ٥٦ بیتی به متن رمز شده ٦٤ بیتی تبدیل می‌کند[۱۷]. قسمت‌های مختلف الگوریتم، که در امنیت الگوریتم تأثیرگذار هستند، تاکنون در مقالات مختلف مورد مطالعه فراوان قرار گرفته اند.

در الگوریتم DES و پس از انجام یک جایگشت ابتدایی، تعداد ٦٤ بیت ورودی به دو بخش ۳۲ بیتی R_0 و L_0 تقسیم می‌شوند. الگوریتم در ١٦ دور به دیتای ورودی اعمال گردیده و در نهایت متن رمزشده بدست می‌آید.

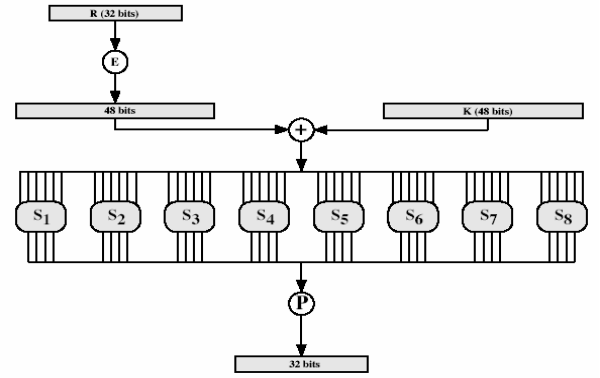
در دور i ام، ورودی‌های این دور، یعنی R_{i-1}و L_{i-1}، به کمک زیر کلید این دور، یعنی K_i، خروجی‌های این دور را تولید می‌نمایند. برای دور i ام داریم

$$\begin{cases} L_i = R_{i-1} \\ R_i = L_{i-1} \oplus f\ (R_{i-1}, K_i) \end{cases}$$

در مورد خروجی حاصل از دور ١٦ام، جایگشت صورت نمی‌گیرد یعنی

$$\begin{cases} L_{16} = L_{15} \oplus f\ (R_{15}, K_{16}) \\ R_{16} = R_{15} \end{cases}$$

پس از خروج از دور شانزدهم، تعداد ٦٤ بیت بوسیله عکس جایگشت اولیه، جایگشت می‌یابند و متن رمزشده بدست می‌آید. شکل ۱ جایگاه S-box ها را در تابع دور نشان می‌دهد.



شکل ۱- جایگاه S-box ها در تابع دور.

بوسیله هشت S-box در هر دور، تعداد ٤٨ بیت ورودی به ۳۲ بیت خروجی نگاشته می‌شود[۱۷] و [۱۸]. در DES، ساختار S-boxها در تمام دورها ثابت است. در کلی‌ترین حالت، یک S-box یک نگاشت از $GF(2^m)$، برای m بیت ورودی، به $GF(2^n)$، برای n بیت خروجی است. به عبارت دیگر می‌توان هر S-box را به صورت تابع $S-box: GF(2^m) \to GF(2^n)$ نمایش داد. با توجه اینکه ورودی‌ها و خروجی‌ها، بردارهای باینری هستند، از عبارت $m \to n$ برای نمایش تابع مذکور استفاده می‌کنیم. همانطور که اشاره شد هر S-box $6 \to 4$ DES، در داخل خود، از چهار تابع S-box مربعی $4 \to 4$ استفاده می‌کند.

۳- ارزیابی توابع بولی با استفاده از شبکه عصبی: بررسی S-boxهای DES به عنوان تابع بولی $6 \to 4$

به منظور ارزیابی توابع بولی، از شبکه پرسپترون چندلایه استفاده کرده‌ایم. به کمک نرم افزار MATLAB شبکه‌های متعددی را بررسی کرده که در ادامه معرفی می‌کنیم. با توجه به

تعداد ورودی‌ها و خروجی‌ها در هر تابع (٦ بیت ورودی و ٤ بیت خروجی)، تمام شبکه‌های این بخش دارای ٦ نورون در لایه ورودی و ٤ نورون در لایه خروجی هستند. معیار برای پاسخ مطلوب شبکه (یادگیری مناسب) به صورت زیر در نظر گرفته شد. به ازای اعمال ورودی‌های تست، خروجی حاصل از شبکه ابتدا با دستور round به نزدیک‌ترین عدد صحیح مجاور تبدیل و این مقدار با خروجی مطلوب متناظر مقایسه می‌شود. در صورت انطباق کامل برای تمام دیتای تست، شبکه مناسب بدست آمده است.

اکنون تعداد نورون‌ها در لایه میانی و نیز اثر توابع فعالیت مختلف در یادگیری شبکه را بررسی می‌کنیم. با توجه به اینکه برای هر S-box DES $6\to4$، تعداد $2^6=64$ ورودی متفاوت وجود خواهد داشت، استفاده از بیشتر از 64 نورون در لایه میانی موجه نیست[١٩] . البته با توجه به انتخاب تصادفی مقادیر اولیه پارامترها در شبکه عصبی، بیان مناسب‌تر این است که با انتخاب مناسب توابع فعالیت و نیز انتخاب تصادفی مقادیر اولیه ضرایب در شبکه، کافی بودن تعداد دیتای آموزشی و اعمال دیتای آموزشی تا حصول یادگیری توسط شبکه، حداکثر به تعداد 2^n نورون در لایه میانی، یعنی به تعداد خروجی‌ها، برای شناسایی مطلوب نیاز داریم. ورودی‌ها ابتدا به صورت باینری $\{0,1\}$ اعمال شده‌اند ولی نظر به اینکه نتایج حاصل در صورت اعمال به شکل $\{-1,1\}$ مناسب‌تر می‌شوند، ورودی‌ها را به این شکل به شبکه اعمال کرده‌ایم. با یک لایه پنهان، یادگیری مطلوب حاصل گردید. در تمامی شبیه‌سازی‌ها نسبت $\frac{1}{2^m}$ از بردارهای ورودی را برای تست در نظر گرفته و بقیه بردارهای ورودی را به تعداد e مرتبه برای آموزش و یادگیری شبکه عصبی به شبکه اعمال می‌کنیم، که e تعداد epoch های دیتای آموزشی اعمال شده به شبکه است. هر بار اعمال تمامی دیتای آموزشی به شبکه عصبی، یک epoch نامیده می‌شود [٢٠] .

شبکه‌های مورد مطالعه به ترتیب عبارتند از:

١- شبکه پرسپترون سه لایه، تابع فعالیت لایه میانی خطی(purelin)، تابع فعالیت لایه خروجی خطی(purelin)، تعداد نورون لایه میانی n، تعداد دیتای آموزشی ٦٣، تعداد دیتای تست ١. با این ساختار شبکه، برای هیچکدام از توابع، یعنی هیچ یک از هشت S-box، نتیجه مطلوب حاصل نشد.

٢- شبکه پرسپترون سه لایه، تابع فعالیت لایه میانی غیرخطی(tansig)، تابع فعالیت لایه خروجی خطی(purelin)، تعداد نورون لایه میانی n، تعداد دیتای آموزشی ٦٣ ، تعداد دیتای تست ١. این شبکه نیز با تعداد قابل قبول نورون در لایه میانی و نیز تعداد epoch زیاد قادر به شناسایی نبود.

کم بودن دیتای آموزشی با توجه به تعداد پارامترهای مستقل شبکه برای یادگیری، دلیل اصلی عدم شناسایی است؛ به همین دلیل با افزودن مقداری نویز به دیتای ورودی، تعداد آن را افزایش داده تا شبکه توانایی یادگیری آن را پیدا کند. این روشی است که عموما در اینگونه موارد به کار گرفته می‌شود[٢٠] . برای این منظور به ازای هر ورودی، تعداد ١٠ ورودی با میانگین ورودی اصلی و واریانس ٠,٠٠٥ با استفاده از توزیع گاوسی تولید شد. در ادامه مقاله، نسبت افزایش اطلاعات ورودی را با k، تعداد epochهای اعمال دیتا برای آموزش شبکه را با e و انحراف از معیار استفاده شده برای افزایش دیتای ورودی را با σ نمایش می‌دهیم.

٣- شبکه پرسپترون سه لایه، تابع فعالیت لایه میانی خطی(purelin)، تابع فعالیت لایه خروجی خطی(purelin)، تعداد نورون لایه میانی n، تعداد دیتای آموزشی ٦٣٠ ، تعداد دیتای تست ١٠. این شبکه با تعداد مناسبی نورون در لایه میانی، موفق به یادگیری نشد.

٤- شبکه پرسپترون سه لایه، تابع فعالیت لایه میانی غیرخطی(tansig)، تابع فعالیت لایه خروجی خطی (purelin)، تعداد نورون لایه میانی n، تعداد دیتای آموزشی ٦٣٠ ، تعداد دیتای تست ١٠. این شبکه توانست توابع را شناسایی نماید. تعداد نورون‌های مورد نیاز برای شناسایی کامل هر تابع در جدول ١ ذکر شده است. این جدول برای دو اجرای متفاوت و با تعداد epoch ٣٠٠ بدست آمده است.

جدول ۱- تعداد نورون لایه میانی برای شناسایی.

	S1	S2	S3	S4	S5	S6	S7	S8	متوسط
آزمایش اول	22	22	23	21	22	21	20	20	21.375≈21
آزمایش دوم	22	19	21	19	22	22	20	21	20.75≈21

جدول ۲- تعداد (e) epoch لازم برای شناسایی (تعداد نورون لایه میانی $n = 25$).

	S1	S2	S3	S4	S5	S6	S7	S8	متوسط
epoch (e)	20	15	18	18	21	15	19	15	17.625≈18

جدول ۳- بررسی واریانس نویز در تعداد نورون‌های لایه میانی لازم برای یادگیری.

	S1	S2	S3	S4	S5	S6	S7	S8	متوسط
$\sigma^2 = 0.05$	28	26	31	28	27	27	28	27	27.875≈28
$\sigma^2 = 0.005$	25	22	23	24	23	23	24	21	23.125≈23
$\sigma^2 = 0.0005$	21	23	23	19	21	23	22	23	21.875≈22

جدول ٤ – بررسی میزان افزایش دیتای ورودی در تعداد نورون لازم برای یادگیری.

میزان افزایش دیتا k	S1	S2	S3	S4	S5	S6	S7	S8	متوسط
5	25	22	23	22	21	19	21	22	21.875≈22
10	25	22	23	24	23	23	24	21	23.125≈23
15	24	20	22	22	25	23	24	23	22.75≈23

در مورد این شبکه، پس از بدست آمدن نتایج بالا، تعداد epoch لازم برای یادگیری نیز بررسی شد. در جدول ۲ حداقل تعداد epoch لازم برای شناسایی توابع در یک اجرا بیان شده‌اند. در این حالت ۲۵ نورون در لایه میانی در نظر گرفته شده است. جدول ۳ اثر واریانس نویز اضافه شده به دیتا را در تعداد نورون لایه میانی مورد نیاز برای شناسایی کامل به ازای ٤٠ epoch، یعنی $e = 40$ بار اعمال دیتای آموزشی نشان می‌دهد. می‌بینیم که واریانس نویزی در حدود پنج هزارم مناسب بوده و کاهش بیشتر واریانس موجب کاهش چشمگیری در تعداد نورون‌های مورد نیاز نمی‌شود. در ادامه، تاثیر نسبت افزایش دیتا نیز مورد بررسی قرار گرفت. جدول ٤ نتایج را به ازای افزایش‌های متفاوت در میزان دیتا، با واریانس نویز ۰۰۵. و تعداد ٤٠ epoch اعمال ورودی، یعنی $e = 40$ و $\sigma = 0.005$، نشان می‌دهد.

با توجه به اینکه هر S-box $6 \to 4$، متشکل از ٤ تابع بولی $4 \to 4$ است و در الگوریتم های رمز قطعه‌ای مدرن نظیر AES و Twofish نیز همچنان از توابع مربعی $4 \to 4$ در داخل S-box ها استفاده شده است، در بخش بعدی به مطالعه ۳۲ تابع چهار در چهار مورد استفاده در مجموعه هشت S-box الگوریتم DES به عنوان توابع قوی می‌پردازیم.

٤- ارزیابی توابع بولی با استفاده از شبکه عصبی: بررسی S-boxهای $4 \to 4$ موجود در سیستم DES

در مورد توابع بولی $4 \to 4$ S-boxهای DES، از روشی مشابه بند چهارم بخش قبل استفاده می‌کنیم، به عبارت دیگر برای شناسایی هر یک از ۳۲ تابع مورد آزمون، از شبکه عصبی پرسپترون سه لایه‌ای با پارامترهای $(k = 10, \sigma = 0.05, e = 40)$ استفاده کرده‌ایم. در هر مورد، تعداد نورون لایه میانی لازم به عنوان تخمینی از میزان کفایت تابع، محاسبه شده است.

نتایج حاصل از ۵۲ بار محاسبه تعداد نورون‌های مورد نیاز در لایه میانی، برای $(k = 10, \sigma = 0.05, e = 40)$، در

جدول های ۵ و ٦ آمده است. ستون‌های جدول‌های ۵ تا ۸ نشان‌دهنده تعداد (یا متوسط تعداد) نورون‌های لایه میانی برای شناسایی ٤ S-box $4\to 4$ موجود در S-box $6\to 4$ DES هستند. جدول ۵ متوسط تعداد نورون‌های لایه میانی برای شناسایی کامل و جدول ٦ حداقل تعداد نورون‌های لازم در اجراهای متفاوت را نشان می‌دهد.

همان‌طور که از جدول های مذکور مشخص است، تعداد ۳۲ تابع بکار گرفته شده در DES، با معیار تعریف شده در این مقاله، دارای خواص مشابه و بسیار نزدیک به هم هستند. به عبارت دیگر، تعداد نورون لازم در لایه میانی برای شناسایی آنها مشابه یکدیگر است.

جدول ۵- متوسط تعداد نورون‌های لایه میانی برای شناسایی.

S-box1	8.6538	8.7692	9.1154	8.8846
S-box2	8.9231	9.1923	9.1346	9.1346
S-box3	9.3654	9.1731	9.1538	8.9615
S-box4	8.5962	9.0577	9.2115	9.0385
S-box5	9.0962	9.0962	8.8654	8.8269
S-box6	9.2308	9.1923	9.2692	9.5
S-box7	9.4423	9.1346	9.3846	9.1731
S-box8	8.8846	8.7308	9.0769	9.1923

جدول ٦- حداقل تعداد نورون‌های لایه میانی برای شناسایی.

S-box1	7	7	7	7
S-box2	7	7	7	7
S-box3	7	7	7	7
S-box4	7	7	7	7
S-box5	7	8	7	7
S-box6	8	8	7	8
S-box7	7	7	7	8
S-box8	7	7	7	7

۵- طراحی و ارزیابی توابع بولی $4\to 6$ ضعیف

اکنون نتایج حاصل در مورد ٤ تابع بولی ضعیف طراحی شده را ارائه می‌کنیم. روابط (۱) تا (٤)، توابع بولی سازنده این چهار تابع را نشان می‌دهند، که در آنها x_i، $i=1,2,...,6$، نشان‌دهنده شش بیت ورودی و y_i، $i=1,2,3,4$، نشان‌دهنده چهار بیت خروجی تابع هستند. همچنین $\overline{x}_i$ بیانگر بیت مکمل x_i است. چهار تابع ضعیف طراحی شده با چهار تابع بکار رفته در اولین S-box الگوریتم DES از نظر تعداد نورون‌های لایه میانی مقایسه شده‌اند. در این مقایسه نیز پارامترهای $(k=10,\sigma=0.05,e=40)$ به کار رفته است. نتایج حاصل از ۳۰ اجرا در جدول های ۷ و ۸ ارائه شده است.

$weak-box\,1-1$

$$\begin{cases} y_1=\overline{x}_3\overline{x}_4\overline{x}_5\oplus\overline{x}_3x_4\oplus x_3\overline{x}_5\oplus x_2 \\ y_2=\overline{x}_4x_5 \\ y_3=x_2x_4x_5\oplus x_3 \\ y_4=x_3\overline{x}_5\oplus x_3 \end{cases} \quad (۱)$$

$weak-box\,1-2$

$$\begin{cases} y_1=x_3\overline{x}_5\oplus x_2 \\ y_2=\overline{x}_2x_3\overline{x}_4 \\ y_3=\overline{x}_4\oplus x_2 \\ y_4=\overline{x}_2\overline{x}_3\overline{x}_4x_3\overline{x}_5\oplus x_3 \end{cases} \quad (۲)$$

$weak-box\,1-3$

$$\begin{cases} y_1=\overline{x}_3\overline{x}_4\overline{x}_5\oplus x_3\overline{x}_5\oplus x_2 \\ y_2=\overline{x}_2\overline{x}_4\overline{x}_5\oplus\overline{x}_4x_5\oplus\overline{x}_2x_5 \\ y_3=x_4\overline{x}_5\oplus x_2 \\ y_4=\overline{x}_2x_4x_5\oplus\overline{x}_2x_5\oplus x_3 \end{cases} \quad (۳)$$

$weak-box\,1-4$

$$\begin{cases} y_1=x_2\overline{x}_3\overline{x}_5\oplus x_3\overline{x}_5\oplus x_2 \\ y_2=\overline{x}_2\overline{x}_5\oplus x_4 \\ y_3=\overline{x}_2\overline{x}_3x_5\oplus x_4\overline{x}_5\oplus x_2 \\ y_4=x_3\overline{x}_5\oplus\overline{x}_2x_5\oplus x_3 \end{cases} \quad (٤)$$

جدول ۷ متوسط تعداد نورون‌های لایه میانی برای شناسایی کامل، و جدول ۸ حداقل تعداد نورون‌های لایه میانی برای شناسایی بین اجراهای متفاوت را نشان می‌دهد. با توجه به نتایج ارائه شده در جدول های مذکور اینگونه مستفاد می‌شود که چهار تابع طراحی شده در این مقاله، با در نظر گرفتن پیچیدگی پیاده‌سازی عصبی به عنوان یک معیار مرجع، در مقایسه با توابع DES ضعیف‌تر هستند.

جدول ۷- متوسط تعداد نورون‌های لایه میانی لازم برای شناسایی.

S-box 1 DES	9.3333	9.3	8.8666	8.9666
Weak S-box	6.7	7.4	8.5	7.8333

جدول ۸- حداقل تعداد نورون‌های لایه میانی لازم برای شناسایی.

S-box 1 DES	8	8	8	7
Weak S-box	5	5	6	7

۶- نتیجه‌گیری

در این مقاله روشی برای تعیین (تقریبی) میزان غیرخطی بودن S-box ها به کمک شبکه‌های عصبی بیان شده است. به کمک این روش، نگاشت غیرخطی از فضای ورودی به فضای خروجی برای S-box های $4 \to 6$ و $4 \to 4$ DES مورد مطالعه قرار گرفت. نتایج نشان می‌دهند تعداد یکسانی نورون در لایه میانی برای شناسایی هر یک از هشت S-box $6 \to 4$ و سی و دو S-box $4 \to 4$ در DES مورد نیاز است. این را می‌توان به عنوان معیاری برای بررسی مناسب بودن S-box طرح شده در هر سیستم رمز قالبی مورد استفاده قرار داد.

به بیان دیگر، باید تعداد نورون‌های لایه میانی لازم برای شناسایی S-boxهای به کار گرفته شده در یک الگوریتم رمز قطعه‌ای، از یک سطح آستانه مشخصی بیشتر باشند. این مقدار را می‌توان تقریبا معادل میزان غیر خطی بودن هر S-box یا درجه غیر خطی بودن آن در نظر گرفت.

به نظر می‌رسد برای سیستم DES داشتن درجه غیرخطی بودن مشابه هم برای همه S-boxها، نیز جزو معیارهای طراحان سیستم بوده است. نتایج می‌تواند پاسخی به سوال " آیا S-Boxهای DES طبق آنچه طراحان DES ادعا کرده‌اند، مستقل از هم انتخاب شده‌اند؟[۲۱] " نیز بدهد.

۷- مراجع

[1] National Bureau of Standards, NBS FIPS PUB 46, *Data Encryption Standard*, NBS, US Department of Commerce, Jan. 1977.

[2] B. Schneier, et al., *The Twofish Encryption Algorithm*, John Wiley, 1999.

[3] J. Daemen and V. Rijmen, *The Design of Rijndael*, Springer-Verlag, 2001.

[4] K. Nyberg, "On the construction of highly nonlinear permutation," *Advances in Cryptology, Proc. Eurocrypt'92, LNCS*, Springer-Verlag, 1993.

[5] M. H. Dawson and S. E. Tavares, "An expanded set of S-box design criteria based on information theory and its relation to differential-like attacks," *Advances in Cryptology, Proc. Eurocrypt'91, LNCS 547*, pp. 352-367, Springer-Verlag, 1993.

[6] X. M. Zhang and Y. Zheng, "Difference distribution table of a regular substitution box," *Proceedings of the Third Annual Workshop on Selected Areas in Cryptography (SAC'96),* pp. 57-60, August 1996, Kingston, Ontario, Canada.

[7] B. Chor, et al., "The bit extraction problem or t-resilient functions," *Proceedings of the 26th Annual Symposium of Foundations of Computer Science*, pp. 396-407, 1985.

[8] T. Siegenthaler, "Correlation immunity of nonlinear combining functions for cryptographic applications," *IEEE Transactions on Information Theory*, vol. 30, no. 5, pp.776-780, Sep. 1984.

[9] H. Feistel, "Cryptography and computer privacy," *Scientific American,* vol. 228, No. 5, pp. 15-23, 1973.

[10] J. B. Kam and G. I. Davida, "Structured design of substitution permutation networks," *IEEE Transaction on Computer,* vol. C-28, no. 10, pp. 747-753, Oct. 1979.

[11] A. F. Webster and S. E. Tavares, "On the design of S-boxes," *Advances in Cryptology, Proc. CRYPTO'85, LNCS*, Springer-Verlag, 1986.

[12] R. Forre, "The strict avalanche criterion: spectral properties of Boolean functions and an extended definition," *Advances in Cryptology, Proc. Crypto'88*, LNCS, Springer-Verlag, 1990.

[13] B. Preneel, W. V. Leekwijck, L. V. Linden, R. Govaerts and J. Vandewalle, "Propagation characteristics of Boolean functions," *Advances in Cryptology, Proc. Eurocrypt'90*, LNCS, Springer-Verlag, 1990.

[14] S. Maitra and P. Sarkar, "Highly nonlinear resilient functions optimizing Siegenthaler's inequality," *Advances in Cryptology, Proc. Crypto'99*, LNCS, vol. 1666, pp. 198-215, Springer-Verlag, 1999.

[15] J. Sebbery, X. Zhang, and Y. Zheng, "On construction and nonlinearity of correlation immune functions," *Advances in Cryptology, Proc. Eurocrypt'93*, LNCS, vol. 765, pp. 181-199, Springer-Verlag, 1993.

[16] Y. V. Tarannikov, "On resilient Boolean functions with maximal possible nonlinearity," *Proceedings of the First International Conference on Progress in Cryptology*, LNCS, vol. 1977, pp. 19-30, 2000.

[17] W. Stallings, *Cryptography and Network Security*, Prentice-Hall, 2nd ed., 1999.

[18] B. Schneier, *Applied Cryptography*, New York: Wiley, 2nd ed., 1996.

[19] J. E. Dayhoff, *Neural Network Architectures*, Van Nostrand, 1990.

[20] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice-Hall, 2nd ed., 1999.

[21] N. T. Courtois, G. Castagnos, and L. Goubin, "What do DES S-Boxes say to each other?" Available:http://eprint.iacr.org/2003/184/.

[1] Balancedness
[2] Regularity
[3] Resiliency
[4] Correlation Immunity
[5] Avalanche Criteria
[6] Completeness
[7] Strict Avalanche Criteria
[8] Higher Order Strict Avalanche
[9] Propagation of Degree *k*
[10] Fitness Function