تشخیص تروجان سخت‌افزاری برمبنای تحلیل توان مصرفی، با استفاده از الگوریتم PCA و شبکه عصبی مصنوعی MLP

علی فریدونی[۱]، محمدعلی دوستاری[۲]، حامد یوسفی[۳و۴]

[۱] دانشجو کارشناسی ارشد، گروه مهندسی کامپیوتر، دانشکده فنی، دانشگاه شاهد، تهران، a.fereidouni@shahed.ac.ir

[۲] استادیار گروه مهندسی کامپیوتر، دانشکده فنی، دانشگاه شاهد، تهران، doostari@shahed.ac.ir

[۳]دانشجو دکتری، گروه مهندسی برق، دانشکده فنی، دانشگاه شاهد، تهران، h.yusefi@shahed.ac.ir

[۴]پژوهشگاه توسعه فناوری‌های پیشرفته خواجه نصیرالدین طوسی، تهران، h.yusefi@rcdat.ir

چکیده

به علت جهانی‌شدن صنعت نیمه‌هادی و طراحی مراحل مختلف تراشه در نقاط مختلف جهان، تولید تراشه به‌طور فزاینده‌ای از طریق برون‌سپاری انجام می‌شود. این امر یک خطر مهم برای مدارهای مجتمع‌هایی است که در کاربردهای مهم امنیتی استفاده می‌شود. مهاجمان می‌توانند تراشه‌ها را در هنگام ساخت در کارخانه‌های غیرقابل‌اعتماد تغییر دهند و یا ممکن است در فازهای مختلف طراحی به‌نوعی به طراحی دست برده شود و تغییراتی در آن اعمال شود. این تغییرات مخرب و توابع پنهان به‌عنوان «تروجان سخت‌افزاری» نامیده می‌شود. کشف چنین مدارهای تروجان دار با استفاده از راهبردهای آزمون معمولی، تقریبا غیرممکن است. در پژوهش‌های انجام‌شده روش‌های مختلفی برای کشف تروجان ارائه‌شده است که روش کشف با استفاده از پارامترهای کانال جانبی از مهم‌ترین و مؤثرترین آن‌هاست. در این روش‌ها با تحلیل‌های آماری و اعمال الگوریتم‌های مختلف بر روی پارامترهای کانال جانبی می‌توان به وجود تروجان در مدار پی برد. در پژوهش‌های انجام‌شده هر الگوریتم و تحلیل به‌تنهایی قادر به کشف ۱۰۰ درصد تروجان‌های کوچک نیست لذا درروش جدید ارائه‌شده در این مقاله به کمک ترکیب الگوریتم PCA و شبکه‌های عصبی مصنوعی MLP نشان داده می‌شود که توان مصرفی مدار AES128 آلوده به تروجان نشت کلید از نمونه طلایی آن به‌راحتی قابل‌تفکیک بوده و با این روش می‌توان تراشه آلوده به تروجان‌های نشت کلیدی که ردپای بسیار کوچکی بر روی مساحت و توان مصرفی دارند را با دقت بسیار بالایی تشخیص داد.

کلمات کلیدی

تروجان سخت‌افزاری، Hardware Trojan Detection**، پارامترهای کانال جانبی، توان مصرفی تراشه،الگوریتم** PCA**، شبکه عصبی مصنوعی** MLP

۱- مقدمه

استفاده‌های فراوان از مدارهای مجتمع و سخت‌افزارهای مختلف در حوزه‌های بحرانی گوناگونی همچون صنایع نظامی، اقتصادی، ارتباطات، پزشکی و غیره (مواردی که عواقب یک حمله موفقیت‌آمیز می‌تواند بسیار جدی و پرهزینه باشد)، اهمیت مطالعه و پژوهش در حوزه امنیت سخت‌افزار را بیشتر نموده است. سخت‌افزار به‌عنوان پایین‌ترین لایه هر سکوی[۱] محاسباتی، بالاترین سطح دسترسی را داراست و سوءاستفاده از آسیب‌پذیری‌های امنیتی احتمالی در این لایه با سازوکارهای امنیتی موجود در سطح نرم‌افزاری قابل‌شناسایی و پیشگیری نیست و حمله‌کننده می‌تواند در این سطح کنترل کامل بر سامانه هدف را، داشته باشد [۱].

یکی از آسیب‌پذیری‌های مطرح‌شده و موردپژوهش در سطح سخت‌افزار، ورود تروجان سخت‌افزاری به مدار مجتمع مورداستفاده است. تروجان‌های سخت‌افزاری می‌توانند به‌صورت تغییرات سخت‌افزاری در مدارهای مجتمع خاص‌منظوره (ASICها)، بخش‌های تجاری عام‌منظوره و یا به‌صورت دست‌کاری میان‌افزار در جریان بیتی FPGA پیاده‌سازی گردند و عملکرد[۲] و اطمینان‌پذیری[۳] سامانه‌های سخت‌افزاری را تحت تأثیر قرار دهند. تروجان‌های سخت‌افزاری امروزه تهدیدی جدی برای امنیت اطلاعات محسوب می‌شوند، زیرا فناوری‌های نیمه‌هادی و مدارهای مجتمع به‌سرعت درحال‌توسعه و پیشرفت بوده و امکان اطمینان به این مدارهای مجتمع به جهت انجام صحیح کار مشخص‌شده برای آن‌ها (انجام فقط کار تعریف‌شده و مشخص‌شده برای آن‌ها) همواره یکی از نگرانی‌های امنیتی بوده است. با این نگاه، امنیت سخت‌افزار در سال‌های اخیر به یک موضوع مهم تحقیقاتی در دنیا تبدیل‌شده است.

بر اساس دسته‌بندی ارائه‌شده در [۲] روش‌های مختلفی برای کشف تروجان‌های سخت‌افزاری وجود دارد که روش تشخیص با استفاده از پارامترهای کانال جانبی از مهم‌ترین روش‌ها است. اساس این روش به این صورت است که هر تغییر ناخواسته در مدار اثراتی بر پارامترهای کانال جانبی از قبیل توان مصرفی پویا یا ایستا و یا تأخیر و یا امواج الکترومغناطیسی دارد [۳]. درعین‌حال، تغییرات فرایند ساخت[۴] در فناوری‌های پیشرفته نانوتکنولوژی و نویز اندازه‌گیری (به‌خصوص برای تروجان‌های کوچک) که می‌توانند اثر مدار تروجان را پوشش دهد، چالشی‌های اصلی مرتبط با این روش‌ها هستند. به همین دلیل روش‌های کانال جانبی موجود برای تنظیم روند به‌سوی روش‌هایی همچون نرمال‌سازی و تنظیم نویز اندازه‌گیری با میانگین‌گیری از چندین اندازه‌گیری برای خلاصی از نویز تصادفی، گرایش دارند [۲].

در [۴] با شبیه‌سازی، از ارتباط ذاتی بین جریان گذرای منبع (I_{DDT})[۵] و جریان خاموش منبع (I_{DDQ})[۶] برای حذف تغییرات فرآیند ساخت کمک گرفته‌شده و توانسته است مصرف توان مدار طلایی را از تروجاندار تفکیک کند. در [۵] یک روش کانال جانبی ارائه‌شده که اثرات مصرف توان گذرا را با استفاده از الگوریتم تحلیل مؤلفه‌های اصلی باهدف تشخیص تروجان سخت‌افزاری مشخص و مقایسه می‌کند. این روش برای نتایج اندازه‌گیری با استفاده از یک FPGA مبتنی بر آزمون برای طراحی‌های بزرگ ازجمله الگوریتم رمزنگاری AES128 بیتی معتبر است. نتایج تجربی نشان می‌دهد که این روش می‌تواند تروجان‌های کوچکی (کمتر از ۱/۱ درصد) را تحت نویز و تغییرات فرآیند ساخت تشخیص دهد. در [۶] یک هسته IP محاسبات چرخشی دیجیتال (CORDIC) ۱۸بیتی به‌عنوان یک‌مدار طلایی (بدون-تروجان) و یک شمارنده ۲ بیتی که مانند یک بمب زمانی از تراشه عمل می‌کند، به‌عنوان مدار تحت آزمون استفاده‌شده است. نسبت مساحت معادل بین مدار تروجان و مدار طلایی حدود ۰/۱ درصد است.

در [۷] با استفاده از شبکه عصبی (ELM) و پیاده‌سازی تروجان سخت‌افزاری حدود ۰/۱۵ درصد بر روی FPGA، توانسته است مدار آلوده به تروجان را از مدار طلایی با موفقیت حدود ۹۰ درصد بر مبنای جریان مصرفی مدار تشخیص دهد. در [۸] به کمک یادگیری ماشین با استخراج ویژگی‌های خاصی از نت لیست مدار در مرحله طراحی توانسته است به تفکیک تروجان از مدار طلایی برسد. اما ویژگی‌های استفاده‌شده وابسته به نوع مدار طلایی و تروجان درج‌شده در آن است.

در [۹] از یک تروجان محدودیت در سرویس استفاده‌شده است که در عملکرد مدار رمزنگاری AES اختلال ایجاد می‌کند. جهت تشخیص این تروجان با پیاده‌سازی آن بر روی FPGA، به کمک توان مصرفی و یادگیری ماشین توانسته است آن را در حالت فعال بودن به‌احتمال ۹۵ درصد و در حالت غیرفعال بودن آن به‌احتمال ۸۱ درصد تشخیص دهد.

در این مقاله ما به دنبال روشی هستیم که بتوانیم تروجان درج‌شده در مدار AES128 را مستقل از نوع تروجان و فعال شدن آن، با موفقیت تقریبا صد درصد تشخیص دهیم. به همین منظور روش نوینی ارائه داده و نشان می‌دهیم چگونه با استخراج ویژگی از توان مصرفی مدار طلایی و تروجاندار به کمک الگوریتم PCA و یادگیری الگوی مصرف توان مدار به کمک شبکه عصبی MLP می‌توان تروجان‌هایی که تأثیر بسیار کمی بر روی مساحت و توان مصرفی مدار اصلی دارند را با نرخ موفقیت تقریبا ۱۰۰ درصد تشخیص داد.

در ادامه قسمت‌های مختلف مقاله به‌صورت زیر بخش‌بندی شده‌اند:

بخش دوم: معرفی الگوریتم PCA و شبکه عصبی مصنوعی استفاده‌شده درروش پیشنهادی. بخش سوم: محک‌های استفاده‌شده برای آزمون روش پیشنهادی و تأثیر آن‌ها بر روی مساحت و توان مصرفی مدار اصلی. بخش چهارم: نحوه پیاده‌سازی مدار تروجاندار بر روی FPGA و اندازه‌گیری توان مصرفی و در بخش پنجم روش پیشنهادی را معرفی و نتایج حاصل از آزمون را ارائه می‌دهیم.

۲- مبانی استخراج ویژگی و تشخیص الگوی رفتاری

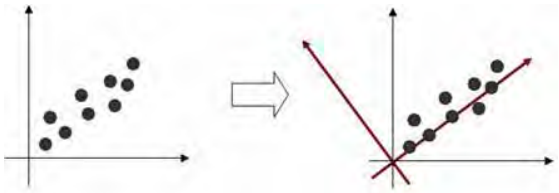
استخراج ویژگی به معنی تبدیل داده‌های اولیه (با همه متغیرها) به یک مجموعه داده با تعداد کمتری متغیر است که نشان‌دهنده ویژگی‌های اصلی هستند. الگوریتم‌های مختلفی برای استخراج ویژگی و کاهش ابعاد

سیگنال‌ها وجود دارد که الگوریتم PCA[۷] یکی از بهترین آن‌ها است. شبکه عصبی مصنوعی نیز از روش‌های قدرتمند برای استخراج ویژگی و تشخیص الگوی رفتاری یک سیگنال است. در این مقاله از الگوریتم PCA و شبکه عصبی MLP جهت استخراج ویژگی و تشخیص الگوی رفتاری سیگنال‌های توان مصرفی استفاده‌شده است.

۲-۱- الگوریتم PCA

الگوریتم PCA یکی از بهترین روش‌ها برای کاهش ابعاد داده به‌صورت خطی است. یعنی با حذف ضرایب کم‌اهمیت به‌دست‌آمده از این تبدیل، اطلاعات ازدست‌رفته نسبت به روش‌های دیگر کمتر است. در این روش محورهای مختصات جدیدی برای داده‌ها تعریف‌شده و داده‌ها بر اساس این محورهای مختصات جدید بیان می‌شوند. اولین محور باید در جهتی قرار گیرد که واریانس داده‌ها بیشینه شود (یعنی در جهتی که پراکندگی داده‌ها بیشتر است). دومین محور باید عمود بر محور اول به‌گونه‌ای قرار گیرد که واریانس داده‌ها بیشینه شود. به همین ترتیب محورهای بعدی عمود بر تمامی محورهای قبلی به‌گونه‌ای قرار می‌گیرند که داده‌ها در آن جهت دارای بیشترین پراکندگی باشند. در شکل ۱ این مطلب برای داده‌های دوبعدی نشان داده‌شده است.

در الگوریتم PCA هدف پیدا کردن ماتریس انتقالی[۸] است که ورودی را به محور مختصاتی انتقال دهد که داده‌ها در آن بیشترین واریانس را داشته باشند.



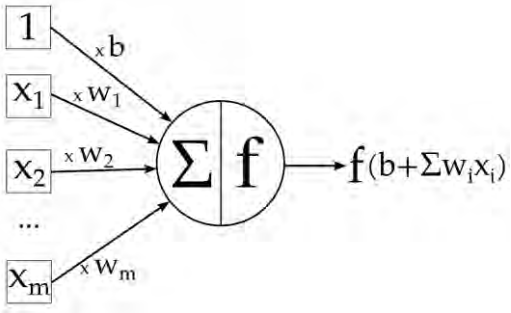
شکل ۱: انتخاب محورهای جدید برای داده‌های دوبعدی درروش PCA

۲-۲- شبکه عصبی مصنوعی

شبکه‌های عصبی مصنوعی روشی برای محاسبه هستند که بر پایه اتصال به‌هم‌پیوسته چندین واحد پردازشی ساخته می‌شود. این شبکه‌ها از تعداد دلخواهی سلول یا گره یا واحد یا نورون تشکیل می‌شود که مجموعه ورودی را به خروجی ربط می‌دهند. شبکه عصبی مصنوعی روشی عملی برای یادگیری توابع گوناگون نظیر توابع با مقادیر حقیقی، توابع با مقادیر گسسته و توابع با مقادیر برداری است. یادگیری شبکه عصبی در برابر خطاهای داده‌های آموزشی مصون بوده و این‌گونه شبکه‌ها با موفقیت به مسائلی نظیر شناسایی گفتار، شناسایی و تعبیر تصاویر، و یادگیری روبات اعمال‌شده است. در این پژوهش از شبکه عصبی MLP به‌منظور یادگیری و تشخیص الگوی رفتاری سیگنال‌های توان مصرفی تراشه‌های طلایی و تروجاندار استفاده‌شده است.

این نوع شبکه عصبی برمبنای یک واحد محاسباتی به نام پرسپترون ساخته می‌شود. در شکل ۲ یک پرسپترون، برداری از ورودی‌های با مقادیر حقیقی را گرفته و یک ترکیب خطی از این ورودی‌ها را محاسبه می‌کند. اگر حاصل، از یک مقدار آستانه بیشتر بود خروجی پرسپترون برابر با ۱ و در غیر این صورت معادل ۱- خواهد بود. به‌بیان‌دیگر خروجی هر نورون را می‌توان تابعی از ورودی‌های شبکه، وزن یال‌ها[۹]، یک حد آستانه[۱۰] و تابع محرک[۱۱] آن نوشت. در فرمول (۱) متغیر X بردار ورودی، W وزن یال‌ها، n اندازه ورودی و b مقدار بایاس شده نورون یا حد آستانه است.

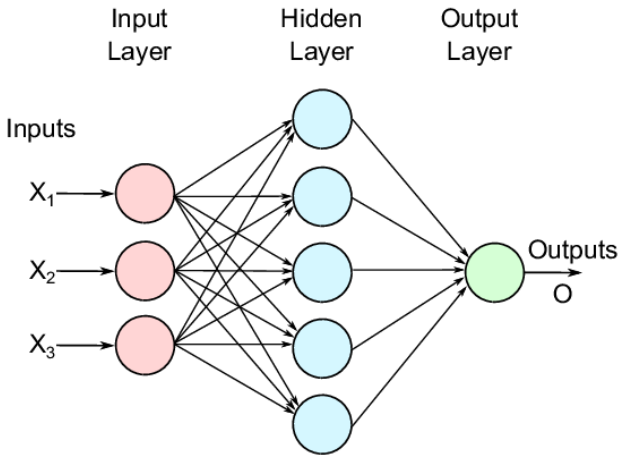
$$y = f(\sum_{1}^{n} W_i X_i + b) \qquad (۱)$$



شکل ۲: مدلی از یک نورون شبکه عصبی مصنوعی

نورون‌ها می‌توانند از توابع محرک متفاوتی جهت تولید خروجی استفاده کنند. که از رایج‌ترین آن‌ها می‌توان به توابع لگاریتم سیگموئیدی تانژانت سیگموئیدی و تابع محرک خطی اشاره کرد.

مجموعه‌ای از این پرسپترون ها با آرایش لایه‌ای خاص یک شبکه عصبی MLP را تشکیل می‌دهند (شکل ۳). می‌توان وزن‌ها و حد آستانه‌های هر نورون آن را برای ورودی‌ها و خروجی‌های مشخصی آموزش داد. جهت آموزش شبکه عصبی از الگوریتم‌های مختلفی مانند الگوریتم ژنتیک، مومنتوم، لونبرگ-مارکوات (LM) و گرادیان نزولی (CG) استفاده می‌شود. از بین روش‌های مختلف به روش پس انتشار خطا، الگوریتم لونبرگ-مارکوات (LM)، به دلیل همگرایی سریع در آموزش شبکه‌های با اندازه متوسط، برای استفاده در تحقیق حاضر انتخاب‌شده است.



شکل ۳: مدلی از یک شبکه عصبی با یک‌لایه مخفی

۳- تروجان‌های محک و وب‌سایت Trust-hub[۱۲]

در این مقاله جهت پیاده‌سازی و ارزیابی روش پیشنهادی از دو نوع تروجان استفاده‌شده است. این تروجان‌ها در سطح RTL به مدار طلایی AES128 اضافه می‌شوند و از نوع نشت اطلاعات رمزنگاری است که از سایت trust-hub گرفته‌شده‌اند. این سایت محیط مناسبی جهت تبادل ایده-ها، محک‌ها، ابزارها و منابع آموزشی است که منابع آن توسط بنیاد ملی علوم[۱۳] حمایت می‌شود. در این سایت انواع تروجان در سطوح مختلف مداری وجود داشته که پژوهشگران می‌توانند به‌راحتی کد HDL یا HSPICE آن را دریافت کرده و به‌عنوان محک استفاده کنند. در ادامه عملکرد تروجان‌های مورداستفاده در این مقاله و تأثیر آن‌ها بر روی مدار اصلی توضیح داده‌شده است.

۳-۱- تروجان T300

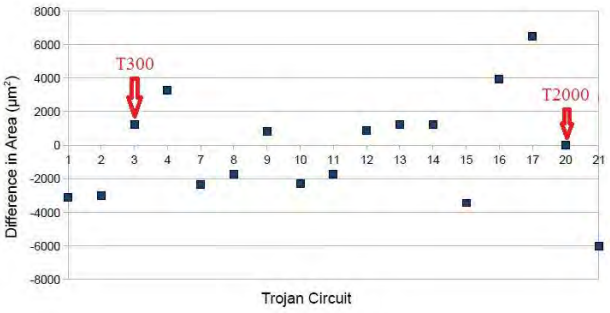
این تروجان یک حمله به بلوک رمز و زمان‌بندی کلید متناظر با آن در الگوریتم AES128 را نشان می‌دهد. ایده اصلی به این صورت است که به‌صورت مصنوعی یک حالت میانی در زمان‌بندی کلید و ورودی مدار اضافه می‌شود، اما ممکن است به‌طور طبیعی در عملیات رمز اتفاق نیفتد. ساختار این تروجان بدین ترتیب است که ابتدا بیت‌های ورودی و بیت‌های کلید باهم AND شده، سپس با بیت‌های میانی XOR شده و خروجی نهایی به‌عنوان فعال‌کننده مدار نشت جریان استفاده می‌شود. تروجان در هر دوره زمان‌بندی کلید، یک بایت از دوره کلید را نشت می‌دهد. مدار نشتی یک ثبات انتقال‌دهنده است که با مقدار اولیه تصادفی از صفرها و یک‌ها مقداردهی شده است. این ثبات انتقال‌دهنده فقط درزمانی فعال می‌شود که بیت فعال‌کننده از مدار نشتی یک باشد. درنتیجه توان مصرفی کل مدار به‌صورت توان مصرفی پویا افزایش می‌یابد. حمله‌کننده به کمک این تروجان می‌تواند به کلید تراشه رمزنگاری دست یابد.

۳-۲- تروجان T2000

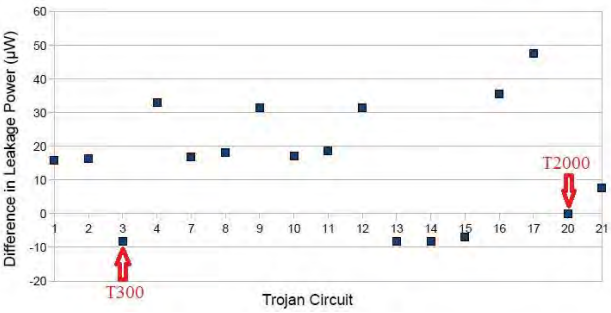
این تروجان پس از تشخیص یک توالی خاص از ورودی‌های مشخص، کلید مخفی از مدار AES128 را در جریان نشتی نشت می‌دهد. مدار نشت جریان یک ثبات انتقال‌دهنده شامل کلید مخفی رمزنگاری و دو عدد گیت NOT است. بیت باارزش این ثبات انتقال‌دهنده به‌عنوان ورودی NOT اول و خروجی این NOT به ورودی NOT دوم متصل می‌شود. هرگاه بیت باارزش ثبات انتقال‌دهنده مقدار صفر را در خود داشته باشد، یک مسیر مستقیم بین منبع و زمین از PMOS مربوط به NOT اول و NMOS مربوط به NOT دوم در زمان محدودی برقرار می‌شود. درنتیجه با اندازه‌گیری مقدار جریان نشتی می‌توان به کلید مخفی پی برد.

۳-۳- تأثیر تروجان‌ها بر روی مدار طلایی

بر اساس مطالعه و شبیه‌سازی‌هایی که در [۱۰] انجام‌شده است با سنتز ۲۰ تروجان نشت کلید در فنّاوری ۴۵ نانومتر و ۹۰ نانومتر و بررسی مصرف توان ایستا و پویا در مدار تروجاندار با مدار طلایی، این تروجان‌ها ردپاهای بسیار ریزی در مساحت و توان مصرفی داشته‌اند. بررسی اثرات توان مصرفی در ابزار synopsys design compiler انجام‌شده است. مطابق با شکل ۴ و شکل ۵، تروجان T2000 در فنّاوری ۹۰ نانومتر کوچک‌ترین ردپا را بر روی مساحت و جریان نشتی دارد که مساحت مدار اصلی را حدود ۲ میکرومتر مربع کاهش می‌دهد. همچنین هر دو تروجان معرفی‌شده کمترین تأثیر را نسبت به تروجان‌های دیگر بر روی توان مصرفی داشته‌اند. درنتیجه انتخاب این دو تروجان گزینه‌های محکمی برای اثبات روش پیشنهادی در این مقاله می‌باشند.



شکل ۴: تأثیر تروجان‌های نشت کلید بر روی مساحت مدار AES128 در [۱۰]



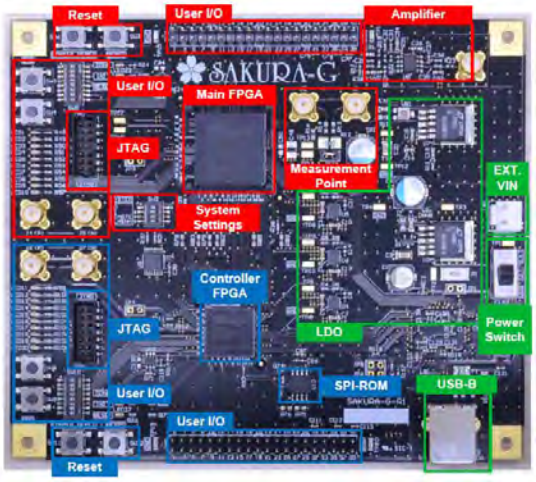
شکل ۵: تأثیر تروجان‌های نشت کلید بر روی جریان نشتی مدار AES128 در [۱۰]

۴- پیاده‌سازی و اندازه‌گیری توان‌مصرفی مدار طلایی و تروجان‌دار

برای دریافت سیگنال توان مصرفی باید از تجهیزات خاصی استفاده شود. در حالت واقعی لازم است جریان کشیده از منبع ولتاژ محاسبه شود، به این منظور باید مداری طراحی شود تا نقش تقویت‌کنندگی را برای سیگنال جریان داشته باشد، چون جریان مصرفی در تراشه‌های امروزی بسیار کم است. در این پژوهش از مدار سکورا[۱۴] استفاده‌شده است (شکل ۶). مدار سکورا به‌منظور تحقیق و پژوهش در حوزه‌ی امنیت سخت‌افزار ازجمله SCA[۱۵] ،

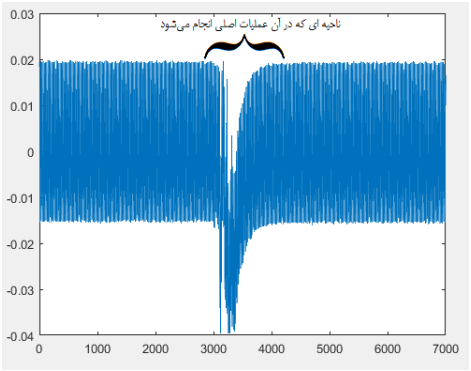
PUF[۱۷]،FIA[۱۶] ساخته‌شده است. این مدار شامل دو FPGA از خانواده‌ی Spartan-6 است. یکی به‌عنوان کنترل‌کننده و دیگری به‌عنوان تراشه‌ی اصلی است.

جهت پیاده‌سازی مدارهای طلایی و تروجاندار، کد سخت‌افزاری مدار AES128 را توسط نرم‌افزار ISE یک‌بار در حالت طلایی و بار دیگر با افزودن کد تروجان سخت‌افزاری آن، سنتز کرده و بر روی مدار سکورا برنامه‌ریزی می‌شود. ماژول top کد سخت‌افزاری، شامل ماشین حالتی است که ورودی را از طریق واسط سریال دریافت و حالت‌های دریافت کلید و دریافت ورودی و شروع عملیات رمزنگاری در آن پیاده‌سازی شده است. علاوه بر آن کد سخت‌افزاری شامل تریگری است که به پایه مشخصی از مدار متصل است. از این پایه جهت تریگر شدن و شروع اندازه‌گیری توان مصرفی اسیلوسکوپ استفاده‌شده است.



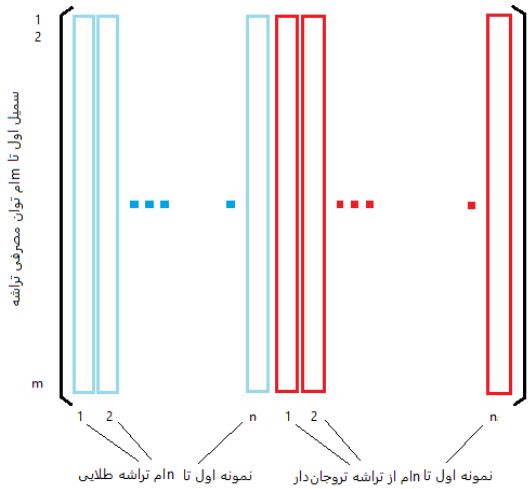
شکل ۶: برد سکورا استفاده‌شده برای اندازه‌گیری توان مصرفی

درنهایت به کمک نرم‌افزاری که توسعه داده‌شده است، plain و کلید را به مدار ارسال و خروجی را دریافت و در صورت صحت خروجی، الگوی توان مصرفی را در فایل CSV ذخیره می‌کنیم. در شکل ۷ نمونه‌ای از سیگنال توان مصرفی اندازه‌گیری شده از FPGA مربوط به مدار AES128 را می‌بینیم.



شکل ۷: نمونه‌ای از سیگنال توان مصرفی پیاده‌سازی مدار AES128

جهت آماده‌سازی جامعه آماری با اعمال ورودی و کلید به‌صورت تصادفی به FPGA، توان مصرفی را طی دفعات متوالی اندازه‌گیری و ذخیره می‌شود. درنهایت میانگین هر ده سیگنال توان مصرفی، به‌عنوان نمونه‌ای از توان مصرفی یک تراشه در نظر گرفته‌شده است. بدین ترتیب یک جامعه آماری ۵۰ تایی از تراشه‌های طلایی و ۵۰ تایی از تراشه‌های تروجاندار در یک ماتریس ایجاد کرده که هر ستون این ماتریس نماینده توان مصرفی یک تراشه در نظر گرفته‌شده است (شکل ۸).



شکل ۸: مدل ماتریس داده توان مصرفی

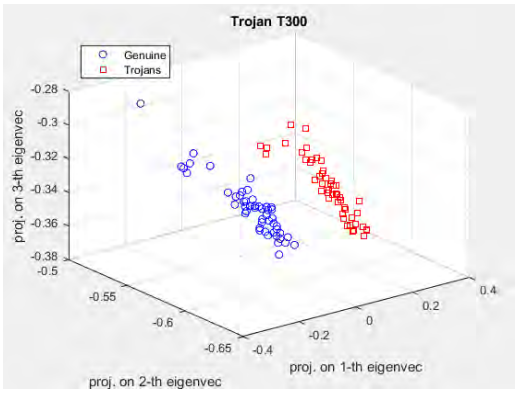
۵- روش پیشنهادی

در این بخش جهت مقایسه و بررسی کارایی روش پیشنهادی، در ابتدا دو روش الگوریتم PCA و شبکه عصبی مصنوعی به‌صورت مجزا بررسی‌شده و درنهایت با استفاده از هر دو روش، روش پیشنهادی معرفی می‌شود.

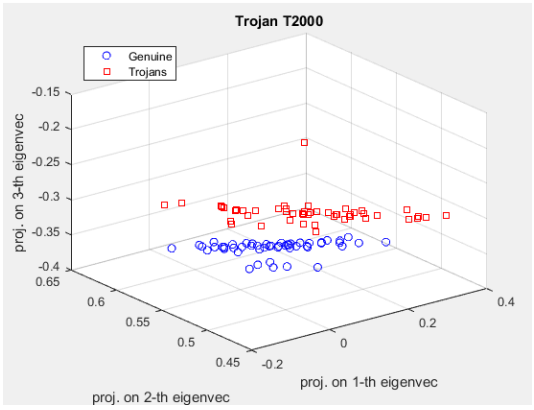
۵-۱- الگوریتم PCA

جهت تفکیک و تشخیص تروجان به کمک الگوریتم PCA، با اعمال الگوریتم بر روی ماتریس داده شکل ۸ ماتریس انتقال (T) به دست می‌آید، به کمک این ماتریس داده‌های ورودی به فضایی که در آن داده‌ها بیشترین واریانس را دارند انتقال می‌یابند. این ماتریس جهت استفاده در مرحله ارزیابی ذخیره می‌شود.

در مرحله ارزیابی جامعه آماری جدیدی آماده کرده و در ماتریس انتقال ضرب می‌شود تا داده‌ها به محور مختصات جدید انتقال یابند. حال با رسم سه بُعد اول از داده‌های خروجی به شکل ۹ و شکل ۱۰ می‌رسیم. همان‌طور که مشاهده می‌شود نواحی تراشه‌های تروجاندار و طلایی از یکدیگر جداشده ولی مرز مشخصی برای تفکیک نوع تراشه‌ها وجود ندارد. همچنین با توجه به تأثیر کمتر تروجان T2000 بر روی توان مصرفی، مشاهده می‌شود که تفکیک این تروجان سخت‌تر از تروجان T300 است.



شکل ۹: تفکیک تروجان T300 از مدار اصلی با الگوریتم PCA

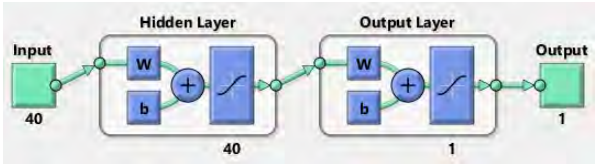


شکل ۱۰: تفکیک تروجان T2000 از مدار اصلی با الگوریتم PCA

۵-۲- شبکه عصبی MLP

تشخیص به کمک شبکه عصبی مصنوعی شامل دو فاز آموزش شبکه و طبقه‌بندی است. در فاز آموزش به شبکه عصبی آموزش داده می‌شود که به ازای ورودی‌های مشخص چه نوع خروجی مشخصی باید از خود داشته باشد، سپس در فاز طبقه‌بندی، ورودی‌های آزمون به شبکه عصبی اعمال و خروجی‌ها بررسی می‌شود.

نوع شبکه عصبی استفاده‌شده در این روش، شبکه عصبی MLP با یک‌لایه مخفی و چهل عدد نورون در لایه مخفی، با تابع فعال‌ساز tansig در هر دولایه مخفی و خروجی است (شکل ۱۱).



شکل ۱۱: شبکه عصبی MLP استفاده‌شده

فاز آموزش:

ازآنجایی‌که اندازه ورودی این شبکه چهل است ولی تعداد سمپل‌های سیگنال توان مصرفی ۷۰۰۰ است، بدین منظور چهل سمپل متوالی از ناحیه‌ای که عملیات اصلی در آن انجام می‌شود (شکل ۷)، به‌عنوان ورودی شبکه انتخاب‌شده‌اند. خروجی شبکه نیز به ازای نمونه‌های طلایی(۵۰ نمونه اول)، عدد صفر و نمونه‌های تروجاندار (۵۰ نمونه دوم)، عدد ده در نظر گرفته‌شده است. درنهایت با فرضیات بالا، شبکه عصبی را آموزش داده و برای فاز طبقه-بندی ذخیره می‌کنیم.

فاز طبقه‌بندی:

جهت ارزیابی این روش، جامعه آماری جدیدی آماده کرده و ماتریس ورودی آن را به شبکه ذخیره‌شده اعمال می‌کنیم. اگر خروجی بزرگ‌تر از ۵ باشد تراشه تروجاندار، در غیر این صورت تراشه بدون تروجان است.

نتیجه این ارزیابی در سه بار تکرار آموزش شبکه عصبی در جدول ۱ نشان داده‌شده است. مشاهده می‌شود که با تغییر تروجان از T300 به T2000 (کاهش تأثیر تروجان بر روی توان مصرفی)، درصد موفقیت تشخیص نیز کاهش‌یافته است.

جدول ۱: نرخ موفقیت تشخیص تروجان

نوع تروجان	تکرار اول	تکرار دوم	تکرار سوم	میانگین نرخ موفقیت(درصد)
T300	۹۶	۹۸	۹۴	۹۶
T2000	۹۷	۹۳	۸۸	۹۲/۶

۵-۳- استفاده همزمان از الگوریتم PCA و شبکه عصبی MLP

فاز آموزش:

در این روش ابتدا بر روی ماتریس داده، الگوریتم PCA را اعمال کرده و ماتریس انتقال را ذخیره می‌کنیم، سپس چهل بُعد(سمپل) اول از خروجی PCA را به‌عنوان ورودی شبکه عصبی MLP در نظر گرفته و خروجی شبکه عصبی نیز به ازای نمونه‌های طلایی (۵۰ نمونه اول)، عدد صفر و به ازای نمونه‌های تروجاندار(۵۰ نمونه دوم)، عدد ده در نظر گرفته‌شده است، در پایان شبکه عصبی را آموزش داده و آن را برای ارزیابی داده‌های آزمون در فاز طبقه‌بندی ذخیره می‌کنیم.

فاز طبقه‌بندی:

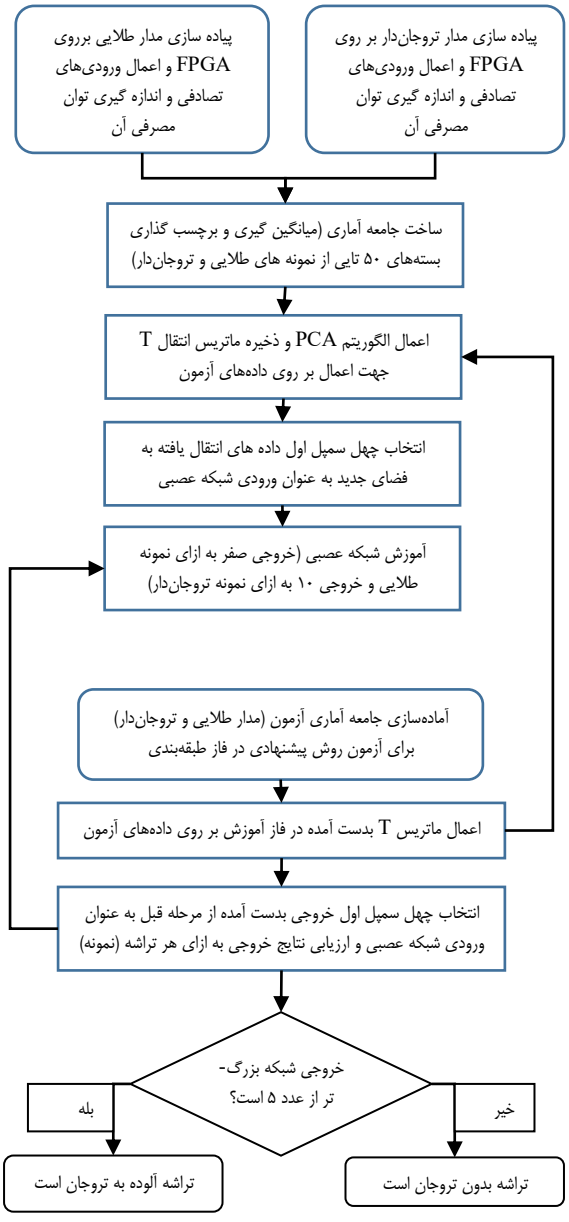
جهت ارزیابی این روش بعد از آماده‌سازی جامعه آماری جدید، آن را در ماتریس انتقال T به‌دست‌آمده در فاز آموزش ضرب کرده تا خروجی PCA حاصل شود، سپس چهل سمپل اول آن برای ارزیابی به شبکه عصبی وارد می‌شود. درنهایت به ازای هر تراشه (نمونه) نتیجه خروجی شبکه عصبی را بررسی می‌کنیم. اگر خروجی بزرگ‌تر از ۵ باشد تراشه تروجاندار، در غیر این صورت تراشه بدون تروجان است. فلوچارت روش پیشنهادی در شکل ۱۲ آورده شده است.

نتیجه ارزیابی تشخیص تروجان‌های T300 و T2000 را با سه بار تکرار آموزش شبکه عصبی، در جدول ۲ می‌بینیم. مشاهده می‌شود که

موفقیت تشخیص در تروجان T300 ۱۰۰ درصد و در تروجان T2000 نزدیک به ۱۰۰ درصد است. به‌طوری‌که قدرت تشخیص در این روش نسبت به حالت قبل (بخش ۵-۲-)، به‌صورت میانگین ۵ درصد افزایش‌یافته و تشخیص مطمئن‌تری داریم.

جدول ۲: نرخ موفقیت تشخیص تروجان درروش پیشنهادی

نوع تروجان	تکرار اول	تکرار دوم	تکرار سوم	میانگین نرخ موفقیت(درصد)
T300	۱۰۰	۱۰۰	۱۰۰	۱۰۰
T2000	۱۰۰	۱۰۰	۹۹	۹۹/۶



شکل ۱۲: فلوچارت روش پیشنهادی

۶- نتیجه‌گیری

در این مقاله ما یک روش نوین تشخیص تروجان سخت‌افزاری مبتنی بر توان مصرفی مدار اصلی، با استفاده همزمان از الگوریتم PCA و شبکه عصبی MLP پیشنهاد کرده‌ایم و در ادامه توانستیم به کمک این روش دو نوع تروجان T300 و T2000 که از محک‌های استاندارد Trust-hub هستند و تأثیر بسیار ناچیزی بر روی مدار اصلی داشتند را به‌طور کامل از مدار طلایی تفکیک داده و احتمال تشخیص صحیح را تقریبا به صد درصد برسانیم. نتایج آزمایش‌ها نشان می‌دهند که دیگر تنها به روش‌های سنتی و الگوریتم‌های آنالیز خطی نمی‌توان بسنده کرد، شبکه‌های عصبی مصنوعی می‌توانند با تشخیص الگوی سیگنال توان مصرفی مدار طلایی در تشخیص تروجان‌های سخت‌افزاری بسیار بهتر عمل کنند.

به‌عنوان پیشنهادی برای کارهای آینده می‌توان این روش را برای انواع دیگر تروجان‌های نشت کلید محک زد. همچنین می‌توان با استفاده از یادگیری عمیق و توسعه شبکه عصبی استفاده‌شده ، دقت تشخیص تروجان-هایی که تأثیر بسیار ناچیزی بر روی مدار دارند را بررسی کرد.

مراجع

[1] M. Bilzor, T. Huffmire, C. Irvine, and T. Levin, "Security checkers: Detecting processor malicious inclusions at runtime," in *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*, 2011, pp. 34-39: IEEE.

[2] M. Tehranipoor and F. Koushanfar, "A survey of hardware trojan taxonomy and detection," *IEEE design & test of computers*, vol. 27, no. 1, pp. 10-25, 2010.

[3] K. Xiao, D. Forte, and M. M. Tehranipoor, "Efficient and secure split manufacturing via obfuscated built-in self-authentication," in *2015 IEEE International symposium on hardware oriented security and trust (HOST)*, 2015, pp. 14-19: IEEE.

[4] B. Hou, C. He, L. Wang, Y. En, and S. Xie, "Hardware Trojan detection via current measurement: A method immune to process variation effects," in *2014 10th International Conference on Reliability, Maintainability and Safety (ICRMS)*, 2014, pp. 1039-1042: IEEE.

[5] L. Wang, H. Xie, and H. Luo, "Malicious circuitry detection using transient power analysis for IC security," in *2013 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (QR2MSE)*, 2013, pp. 1164-1167: IEEE.

[6] C. He, B. Hou, L. Wang, Y. En, and S. Xie, "A novel hardware Trojan detection method based on side-channel analysis and PCA algorithm," in *2014 10th International Conference on Reliability, Maintainability and Safety (ICRMS)*, 2014, pp. 1043-1046: IEEE.

[7] S. Wang, X. Dong, K. Sun, Q. Cui, D. Li, and C. He, "Hardware Trojan detection based on ELM neural network," in *2016 First IEEE International Conference on Computer Communication and the Internet (ICCCI)*, 2016, pp. 400-403: IEEE.

[8] K. Hasegawa, Y. Shi, and N. Togawa, "Hardware Trojan Detection Utilizing Machine Learning

Approaches," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, 2018, pp. 1891-1896: IEEE.

[9] S. Zantout, "Hardware Trojan Detection in FPGA through Side-Channel Power Analysis and Machine Learning," UC Irvine, 2018.

[10] T. Reece and W. H. Robinson, "Analysis of data-leak hardware Trojans in AES cryptographic circuits," in *2013 IEEE International Conference on Technologies for Homeland Security (HST)*, 2013, pp. 467 :274-IEEE.

زیرنویس‌ها

[1] Platform
[2] Functionality
[3] Reliability
[4] Process Variation
[5] Transient current
[6] Quiescent current
[7] Principle component Analysis
[8] Transpose matrix
[9] Weight
[10] Bias
[11] Activation function
[12] Trust-hub.org
[13] National Science Foundation (NSF)
[14] Side-channel Attack User Reference Architecture
[15] Side- Channel Analysis
[16] Fault Injection Attack