دوفصل‌نامه علمی منادی امنیت فضای تولید و تبادل اطلاعات (افتا)
سال ۱۴۰۱، جلد ۲۰، شماره ۱ (۱۸-۲۵)
https://monadi.isc.org.ir

ارائه‌شده در هجدهمین کنفرانس بین‌المللی انجمن رمز ایران در دانشگاه اصفهان، ۱۰ و ۱۱ شهریور ۱۴۰۰

آنالیز آسیب‌پذیری مدارهای دیجیتال در برابر تروجان سخت‌افزار زمانی مبتنی بر عملگر خازنی★

فاطمه خورمیزی و بیژن علیزاده*

دانشکده مهندسی برق و کامپیوتر، پردیس دانشکده‌های فنی، دانشگاه تهران، تهران، ایران

اطلاعات مقاله

کلمات کلیدی:
تروجان سخت‌افزار زمانی
آنالیز آسیب‌پذیری
شناسایی مبتنی بر تأخیر مسیر
شناسایی مبتنی بر آزمون منطقی

dor: 20.1001.1.24763047.1401.11.1.3.0

نوع مقاله: پژوهشی

چکیده

تروجان سخت‌افزاری یک نمونه از تهدیدهای امنیتی برای سخت‌افزار به شمار می‌آید که با اعمال تغییراتی مخفیانه در سخت‌افزار به اهداف مخرب خود می‌رسد. از روش‌های عمده مقابله با آن می‌توان به روش‌های شناسایی و طراحی برای امنیت نام برد. در روش‌های شناسایی، تکنیک‌های شناسایی تروجان سخت‌افزاری معرفی می‌شوند و در روش‌های طراحی برای امنیت تکنیک‌هایی برای شناسایی راحت‌تر و یا ممانعت از درج تروجان سخت‌افزاری ارائه می‌شوند. یک تروجان سخت‌افزاری هوشمند بایستی با در نظر گرفتن این موارد به صورت مخفیانه به سخت‌افزار ورود پیدا کند. با توجه به گستردگی تروجان‌های سخت‌افزاری، انتظار می‌رود زمینه‌های حضور تروجان‌های جدید بررسی شوند. در این مقاله سعی داریم با معرفی نمونه‌ی جدید تروجان سخت‌افزار زمانی که به کمک عملگر خازنی پیاده‌سازی می‌شود، به بررسی نقاط آسیب‌پذیر در برابر آن بپردازیم. تروجانی که با برهم زدن مقررات زمانی موجب خطای زمانی و رفتار نادرست سخت‌افزار می‌شود. از این رو نقاطی که در روش‌های شناسایی تروجان سخت‌افزاری ضعیف‌تر عمل می‌کنند، بررسی شده‌اند و بر اساس آن آنالیزی برای تحلیل آسیب‌پذیری مدارها مطرح شد. نتایج حاصل از آنالیز آسیب‌پذیری حاکی از آن است که تعداد نقاط آسیب‌پذیر مدارها در برابر تروجان سخت‌افزاری معرفی شده اندک هستند و می‌توان برای آن‌ها به دنبال روشی برای ایمن‌سازی در برابر این تروجان گشت.

۱ مقدمه

تحقیقاتی که در دهه‌های اخیر در زمینه امنیت سخت‌افزار انجام شده‌اند، وجود نوعی تهدید با عنوان تروجان سخت‌افزاری را نشان می‌دهند. در پی گسترش توزیع ساخت مدارات الکترونیکی برای کاهش هزینه و زمان تولید، سخت‌افزار ممکن است در طی مراحل تولید تا ساخت، دستخوش تغییرات عمدی و مخرب شود به گونه‌ای که از دید طراحان پنهان باشد و شناسایی آن سخت باشد. در واقع هدف از حمله‌کنندگان تروجان سخت‌افزاری، پنهان نگه داشتن ماهیت خود در حین آسیب رساندن به سخت‌افزار است [۱، ۲]. راهکارهایی که برای مقابله با تروجان سخت‌افزاری ارائه شده‌اند عمدتا در سه دسته تقسیم می‌شوند:

(۱) شناسایی تروجان سخت‌افزاری [۱]
(۲) طراحی بر مبنای امنیت [۲]
(۳) جداسازی فرآیند ساخت به منظور امنیت [۳]

در زمینه تشخیص تروجان، روش‌هایی ارائه شده‌اند که می‌توانند بدون وجود ساختارهای تشخیص اضافی در مدار، تروجان را در صورت وجود

★از کمیته علمی هجدهمین کنفرانس بین‌المللی انجمن رمز ایران برای داوری این مقاله تشکر می‌شود.
*نویسنده مسئول
آدرس‌های رایانامه: fatemeh.khormizi@ut.ac.ir (فاطمه خورمیزی)، b.alizadeh@ut.ac.ir (بیژن علیزاده)

[1] hardware trojan detection [2] design for trust [3] split manufacturing for trust

شناسایی کنند. در زمینه طراحی برای امنیت، روش‌هایی ارائه شده‌اند که با اصلاح طراحی مدارها، سعی بر سهولت شناسایی تروجان یا جلوگیری از درج موثر آن دارند. تحقیقات در زمینه جداسازی فرآیند ساخت به دنبال یافتن موثرترین تقسیم‌بندی فرآیند ساخت در میان کارخانجات سازنده هستند تا هم در هزینه‌های ساخت صرفه‌جویی شود و هم از در اختیار گذاشتن تمام لایه‌های مدار به دست یک کارخانه نامطمئن جلوگیری شود [۳].

در این مقاله سعی داریم با بررسی ویژگی‌های تروجان‌های سخت‌افزاری، نمونه‌ای از تروجان سخت‌افزاری را معرفی کنیم که در نقاط حساس و آسیب‌پذیر مدار ورود پیدا می‌کند تا با برهم زدن مقررات زمانی موجب خطای زمانی در مدار شود. تروجان سخت‌افزار زمانی[1] برای فرار از شناسایی و حفظ خاصیت پنهان‌کاری در نقاطی درج می‌شود که فعال‌سازی آن در روش‌های شناسایی به ندرت رخ می‌دهد. شناسایی تروجان سخت‌افزاری در دو مرحله قبل از ساخت تراشه و بعد از ساخت تراشه تقسیم می‌شود. در مرحله قبل از ساخت، از روش‌های متداول درستی‌سنجی مدارهای الکترونیکی برای تشخیص تروجان استفاده می‌شود که ضمانتی برای تشخیص کامل تروجان سخت‌افزاری وجود ندارد. اما در مرحله پس از ساخت، دو روش مهمی که وجود دارد، روش‌های آنالیز سیگنال کانال جانبی[2] و آزمون منطقی[3] می‌باشند. در روش ارزیابی سیگنال کانال جانبی، وجود تروجان با مشاهده اختلاف میان اندازه‌گیری‌های پارامترهای فیزیکی چون توان، تأخیر و دما در مدار تحت آزمون با مدار طلایی شناسایی می‌شود. در روش آزمون‌های منطقی، بردارهای آزمون برای فعال‌سازی تروجان به نحوی تولید می‌شوند که اثر تروجان در خروجی مدار نمایان شود [۱]. تروجان سخت‌افزار زمانی در نقاطی از مدار درج می‌شود که شناسایی آن با این دو روش به راحتی امکان‌پذیر نباشد. بر همین اساس، آنالیزی برای شناسایی نقاط آسیب‌پذیر در برابر این نوع تروجان‌سخت‌افزاری انجام شده است. به کمک این روش می‌توان با داشتن توصیف سطح گیت هر مداری، نقاط آسیب‌پذیر آن را بر اساس میزان سختی در روش‌های شناسایی پیدا کرد. با این وجود، سهم علمی که این مقاله در بردارد به صورت زیر است:

(۱) معرفی تروجان سخت‌افزاری زمانی که با ورود به نقاط حساس مدارهای دیجیتال موجب خطای زمانی در مدار می‌شود
(۲) معرفی آنالیزی برای شناسایی نقاط حساس و آسیب‌پذیر مدارها در برابر تروجان سخت‌افزاری زمانی

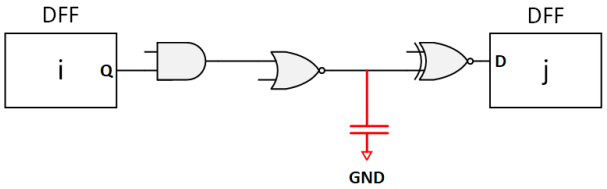
در ادامه ترتیب ارائه مطالب بدین صورت است. در بخش ۲ به معرفی تروجان سخت‌افزار زمانی پرداخته می‌شود. در بخش ۳ روش شناسایی نقاط آسیب‌پذیر در برابر تروجان سخت‌افزار زمانی توضیح داده خواهد شد تا در پی آن آنالیز آسیب‌پذیری پیشنهادی معرفی شود. نتایج حاصل از آنالیز آسیب‌پذیری بر روی مدارهای مختلف در بخش ۴ آمده است. در نهایت نتیجه‌گیری و کارهایی که می‌تواند در راستای این تحقیق انجام شود در بخش ۵ آورده شده است.

۲ معرفی تروجان سخت‌افزار زمانی در شکل عملگر خازنی

تروجان سخت‌افزاری به ساده‌ترین مفهوم، یک مدار جانبی متشکل از دو بخش فعال‌کننده و عملگر در مدار اصلی می‌باشد. واحد فعال‌کننده به طور مداوم سیگنال‌های خاصی از مدار را کنترل می‌کند تا زمانی که یک اتفاق مشخص برای آن‌ها رخ دهد و در این صورت واحد فعال‌کننده تروجان را فعال می‌کند و واحد عملگر وارد کار می‌شود تا اثر نهایی تروجان بر روی مدار واقع شود. بر اساس حالات مختلفی که می‌توان برای پیاده‌سازی فعال‌کننده و عملگر در نظر گرفت، دسته بندی در [۳] ارائه شده است که در آن هم فعال‌کننده و هم عملگر به دو بخش یکسان دیجیتال و آنالوگ و بخش‌های دیگر تقسیم شده‌اند. تروجان سخت‌افزار زمانی به صورت یک عملگر آنالوگی در مدار پیاده‌سازی می‌شود. به عبارت دیگر تروجان سخت‌افزار زمانی با درج خازن در نقاط آسیب‌پذیر مدار در پی تغییر تأخیر مسیر است. مرجع [۴] دسته‌بندی کامل‌تری برای تروجان‌های سخت‌افزاری ارائه داده است. در این دسته‌بندی، تروجان‌های سخت‌افزاری به لحاظ ورود در فازهای توسعه مدار (از توصیف اولیه سخت‌افزار تا ساخت و مونتاژ)، ورود در سطح‌های توصیفی مدار (سطح سیستم تا سطح فیزیکی)، مکانیزم فعال‌سازی، اثرات، محل قرارگیری و مشخصات فیزیکی تقسیم می‌شوند. در این دسته‌بندی نیز شاهد گستردگی امکان بروز تروجان‌های سخت‌افزاری هستیم به طوری که در [۴] صراحتا به طراحی تروجان‌های سخت‌افزاری نوین اشاره شده است. نمونه‌های متعددی از حمله‌های زمانی به سیستم‌های رمزنگاری تاکنون طراحی شده‌اند [۲]. در [۵] یک مدل حمله زمانی به FPGA برای کشف کلید رمز الگوریتم رمزنگاری طراحی شده است که با کمک شبکه حلقه‌های نوسان‌ساز موجب افت ولتاژ منبع توان و در نتیجه افزایش تأخیر گیت‌ها می‌شود. با افزایش تأخیر گیت‌ها، تأخیر انتشار داده در مسیر زیاد می‌شود و با دیر رسیدن داده به انتهای مسیر، خطای زمانی در مسیرهای بحرانی رخ می‌دهد.

با توجه به مطالب گفته شده، در اینجا نوعی دیگر از حمله زمانی به سخت‌افزار یا به عبارت دیگر تروجان سخت‌افزار زمانی معرفی می‌شود. براساس دسته‌بندی کلی در [۴] این تروجان در فاز طراحی و در سطح توصیفی گیت وارد مدار می‌شود. مکانیزم فعال‌سازی آن وابسته به مقادیر ورودی کاربر می‌باشد، به طوری که اگر مقادیر ورودی باعث شود مسیری که دارای تروجان سخت‌افزار زمانی است فعال شود، آنگاه مدار دچار مشکلات زمانی می‌شود که می‌تواند بر روی رفتار مدار اثر بگذارد. محل قرارگیری این تروجان در مدارهای ترتیبی است. برای فهم بهتر این تروجان، مثالی در شکل ۱ آورده شده است. همان طور که گفته شد، این تروجان زمانی در سطح گیت به مدار ورود می‌کند. بنابراین یک مسیر از مدار نشان داده شده است که تروجان به صورت خازن در آن درج شده است. با افزایش بار خازنی مسیر، تأخیر مسیر افزایش می‌یابد. بدین ترتیب تروجان سخت‌افزار زمانی به صورت یک عملگر آنالوگ پیاده سازی می‌شود.

[1] timing hardware trojan [2] side-channel signal analysis [3] logic testing



شکل ۱. ورود تروجان سخت‌افزار زمانی در توصیف سطح گیت

تروجان سخت‌افزار زمانی با افزایش تأخیر مسیر به دنبال بروز خطای زمانی از طریق خطای زمان نگهداری [1] است. خطای زمان نگهداری زمانی رخ می‌دهد که تأخیر اضافی ناشی از تروجان، محدودیت زمانی تعیین شده برای مسیر را برهم بزند. معادله (۱) محدودیت زمانی مسیر را نشان می‌دهد که برای دریافت درست داده در فلیپ‌فلاپ [2] مقصد بایستی داده حداقل زمانی به اندازه زمان نگهداری فلیپ‌فلاپ مقصد ($t_{s,j}$) پس از زمان کلاک تا خروجی فلیپ‌فلاپ مبدا ($t_{cq,i}$) و زمان انتشار در مسیر ($D_{i,j}$) در ورودی فلیپ‌فلاپ مقصد پایدار بماند. در صورتی تأخیر ناشی از تروجان به اندازه‌ای زیاد باشد که داده نتواند حداقل به اندازه زمان نگهداری فلیپ‌فلاپ مقصد در ورودی آن پایدار باشد، داده به درستی ذخیره نمی‌شود و خطای زمانی رخ می‌دهد [۶].

$$D_{i,j} < T_{clk} - t_{s,j} - t_{cq,i} \qquad (۱)$$

اما در کنار ویژگی‌های ذکر شده، مهم‌ترین ویژگی تروجان سخت‌افزاری محل درج آن در مدارها هست. تروجان سخت‌افزار زمانی باید در نقاطی از مدار وارد شود که علاوه بر حفظ خاصیت پنهان‌کاری، موجب بروز خطا در مدار شود. در ادامه به توضیح نحوه یافتن نقاط آسیب‌پذیر در برابر تروجان سخت‌افزار زمانی پرداخته می‌شود.

۳ شناسایی نقاط آسیب‌پذیر

همان‌طور که گفته شد، مهم‌ترین نکته‌ای که باید برای هر تروجان سخت‌افزاری مدّنظر داشت، محل درج آن در سخت‌افزار است که با حفظ خاصیت پنهان‌کاری، به هدف مخرب خود دست یابد. تروجان سخت‌افزار زمانی هم از این قاعده مستثنا نمی‌باشد و سعی در انتخاب نقاطی از مدارها دارد که با درج در آنجا، شناسایی آن با روش‌های آزمون منطقی و آنالیز سیگنال کانال جانبی به راحتی امکان‌پذیر نباشد. چون این تروجان بر روی تأخیر مسیر اثر می‌گذارد، باید از شناسایی توسط روش‌های مبتنی بر تأخیر مسیرها که از روش‌های آنالیز سیگنال جانبی است پنهان بماند. از این رو دو رویکرد برای شناسایی نقاط آسیب‌پذیر در برابر تروجان سخت‌افزار زمانی داریم:

(۱) شناسایی نقاطی که در روش شناسایی مبتنی بر تأخیر مسیرها ضعیف عمل می‌کنند

(۲) شناسایی نقاطی که در روش آزمون منطقی، آزمون‌پذیری سخت‌تری دارند.

[1] setup time violation [2] flip-flop hold time

در ابتدا توضیحات مرتبط و لازم هر دو رویکرد داده می‌شود تا در ادامه براساس آن‌ها، یک آنالیز کاملی برای شناسایی نقاط آسیب‌پذیر ارائه شود.

۱.۳ آسیب‌پذیری در شناسایی مبتنی بر آزمون منطقی

در این قسمت نقاطی از مدار برای درج تروجان سخت‌افزار زمانی مناسب‌تر هستند که در روش شناسایی با آزمون منطقی ضعیف‌تر عمل کنند و به عبارتی آسیب‌پذیرتر باشند. در روش آزمون منطقی با اعمال بردارها آزمون به ورودی‌های مدار، سعی بر فعال کردن تروجان و انتقال اثر آن تا خروجی مدار دارند. از این رو در این روش، نقاطی از مدار شناسایی سختی دارند که قابلیت آزمون‌پذیری کمتری داشته باشد. در آزمون مدارهای الکترونیکی دو مولفه کنترل‌پذیری [3] و رویت‌پذیری [4] برای هر گره مدار تعریف می‌شود تا قابلیت آزمون‌پذیری مدار مشخص شود. منظور از کنترل‌پذیری گره، میزان سختی تنظیم کردن مقدار منطقی گره به یک مقدار مشخص از طریق ورودی‌های مدار است به طوری که گره‌های ورودی مدار بیشترین کنترل‌پذیری را دارند. همچنین منظور از رویت‌پذیری گره، میزان سختی مشاهده کردن تغییر مقدار گره در خروجی مدار است به طوری که گره‌های خروجی بیشترین مقدار رویت‌پذیری را دارند. در اینجا برای تعیین مقدار کنترل‌پذیری و رویت‌پذیری از الگوریتم SCOAP [5] معرفی شده در [۷] استفاده شده است. در این الگوریتم سه پارامتر کنترل‌پذیری به مقدار یک CC1، کنترل‌پذیری به مقدار صفر CC0 و رویت‌پذیری CB بر اساس تعداد گره‌های لازم تعریف شده است. CC1 به صورت تعداد گره‌هایی که باید از ورودی به مقدار مشخصی تنظیم شوند تا این گره به مقدار یک تنظیم شود، تعریف می‌شود. مقدار CC0 برابر است با تعداد گره‌هایی از ورودی مدار که باید به مقدار مشخص تنظیم شوند تا گره مدّنظر به مقدار صفر برسد. مقدار CB نیز به همین صورت تعریف می‌شود و برابر است با تعداد گره‌هایی که باید به مقدار مشخصی تنظیم شوند تا مقدار گره مدّنظر در خروجی نمایان شود. در [۷] مثالی از نحوه محاسبه این پارامترها برای مدارهای ترتیبی آورده شده است. در این اندازه‌گیری، هر چه مقدار کنترل‌پذیری و رویت‌پذیری محاسبه شده برای یک گره بیشتر باشد، نشان می‌دهد که کنترل‌پذیری و رویت‌پذیری آن گره بیشتر است.

در روش‌های آزمون، آزمون‌پذیری یک گره که دچار خطای چسبندگی به یک [6] شده است به صورت مجموع کنترل‌پذیری گره به مقدار صفر و رویت‌پذیری آن گره تعریف می‌شود. به عبارتی با کنترل کردن مقدار گره به مقدار مخالف با مقدار خطا، خطای گره فعال شده و در خروجی قابل رؤیت هست. با توجه به همین مفهوم برای تروجان سخت‌افزار زمانی نیز پارامتر آزمون‌پذیری تعریف می‌شود. با این تفاوت که برای آزمودن تروجان سخت‌افزار زمانی باید تروجان فعال شود یعنی گذاری در گره‌ای که دارای تروجان است رخ دهد. این به معنی کنترل کردن گره به مقدار یک و صفر به صورت متوالی هست. همچنین تروجان در اثر فعال شدن،

[3] controllability [4] observability [5] Sandia Controllability/Observability Analysis Program [6] stuck-at-1 fault

اثر خود به خروجی مدار انتقال دهد. بر این اساس معادله (۲) مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی را برای گره x نشان می‌دهد.

$$T_{THT}(x) = CC1 + CC0 + CB \qquad (۲)$$

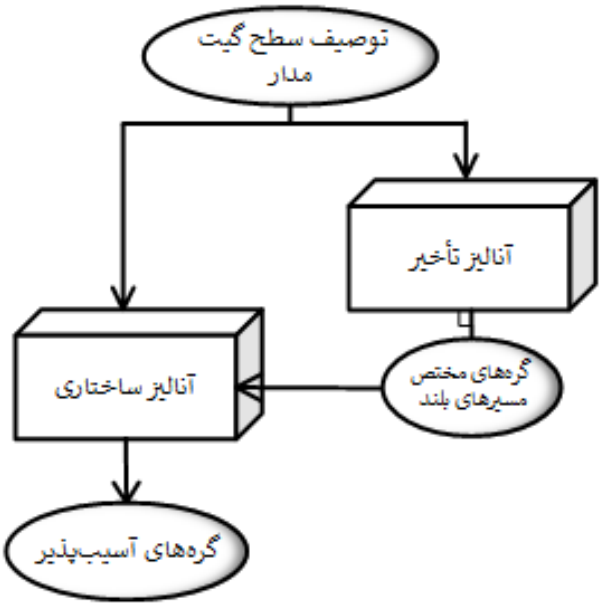
پس از تعیین مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی برای تمام گره‌های مدار، بیشترین مقدار آزمون‌پذیری زمانی ($T_{THT,\max}$) در میان گره‌های مدار به عنوان مرجع سخت‌ترین گره آزمون در مدار مشخص می‌شود تا گره‌های آسیب‌پذیر در روش شناسایی آزمون منطقی براساس این مرجع معرفی شوند.

۲.۳ آسیب‌پذیری در شناسایی مبتنی بر تأخیر مسیر

اساس روش شناسایی مبتنی بر تأخیر مسیرها، اندازه‌گیری تأخیر مسیر در مدار مشکوک به تروجان سخت‌افزاری و مقایسه آن با تأخیر همان مسیر در مدار مدل طلایی است چرا که تأخیر مسیر در حضور تروجان سخت‌افزاری زیاد می‌شود. مطالعات در زمینه طراحی به منظور امنیت برای سهولت شناسایی با این روش نشان می‌دهد که مسیرهای طولانی [1] نسبت به دیگر مسیرها آسیب‌پذیرترند [۸]. علت آن حضور بیشتر نوسانات فرآیند در مسیرهای طولانی است. نوسانات فرآیند از دو مؤلفه‌ی die-to-die و with-in-die تشکیل می‌شوند. در مؤلفه die-to-die سیگنال‌های کانال جانبی در هر نمونه از مدار مجتمع تغییر می‌کنند ولی در مؤلفه with-in-die سیگنال‌ها در هر نقطه از یک مدار مجتمع تغییر می‌کنند. با تغییرات with-in-die، تأخیر دو گیت مشخص در دو نقطه متفاوت از یک مدار متفاوت است. اگر این دو گیت از یک مسیر باشند و به هم نزدیک‌تر باشند، تفاوت تأخیر آن‌ها کمتر است و بالعکس هر چه دو گیت در مسیر از هم دورتر باشند، تفاوت تأخیر آن‌ها بیشتر است و تغییرات تأخیر مسیر بیشتر است. از این رو مسیرهای طولانی تغییرات زمانی ناشی از نوسانات فرآیند بیشتری نسبت به مسیرهای کوتاه دارند و همین سبب می‌شود شناسایی تروجان‌های سخت‌افزاری در این مسیرها با مشکل روبرو شود. لازم به ذکر است که در مسیرهای کوتاه علاوه بر این که تغییرات زمانی ناشی از نوسانات فرآیند کمتر است، تأخیر ناشی از تروجان سخت‌افزار زمانی باید به قدری باشد که این مسیر دچار خطای زمانی شود، به همین دلیل تأخیر تروجان زمانی با تأخیر مسیر قابل قیاس می‌شود و این امر شناسایی آن را با روش‌های مبتنی بر تأخیر مسیرها آسان‌تر می‌کند.

اما در مسیرهای طولانی گره‌هایی برای درج تروجان سخت‌افزاری مناسب هستند که تنها مختص مسیرهای طولانی باشند تا بدین ترتیب اثر تروجان روی مسیرهای کوتاه و بحرانی صفر باشد. برای همین هست که در [۸] و [۹] سعی شده است هر کدام به روشی گره‌های مختص مسیرهای طولانی را به مسیرهای کوتاه تقلبی تجهیز کنند تا درج تروجان سخت‌افزاری را سخت کنند.

[1] long paths



شکل ۲. روند پیشنهادی آنالیز آسیب‌پذیری مدارهای سطح گیت

۳.۳ روش پیشنهادی برای آنالیز آسیب‌پذیری

بنابر توضیحات داده شده در دو زیربخش قبل، روند پیشنهادی آنالیز آسیب‌پذیری مدارهای سطح گیت مطابق با شکل ۲ خواهد بود که آنالیز آسیب‌پذیری را به دو بخش آنالیز تأخیر و آنالیز ساختاری تقسیم کرده است. در آنالیز تأخیر نقاطی از مدار مشخص می‌شوند که در روش شناسایی مبتنی بر تأخیر مسیر ضعیف عمل می‌کنند و در آنالیز ساختاری نقاطی مشخص می‌شوند که در آزمون منطقی ضعیف هستند که از تجمیع آن‌ها نقاط آسیب‌پذیر مدار مشخص می‌شود. بنابراین ابتدا در آنالیز تأخیر از ظرفیت خازنی گیت‌های موجود در مسیر برای محاسبه تأخیر مسیر استفاده شده است و با توجه به ظرفیت خازنی هر مسیر، مسیرهای طولانی تفکیک شده‌اند. بدین صورت که بیشترین مقدار ظرفیت خازنی مسیر به عنوان مسیر بحرانی انتخاب شده است تا ۷۰٪ این مقدار به عنوان حدی برای تأخیر مسیرهای بحرانی و شبه بحرانی در نظر گرفته شود. همچنین ۵۰٪ مقدار بیشترین ظرفیت خازنی مسیر نیز برای تأخیر حدی مسیرهای کوتاه به حساب می‌آید. با داشتن این دو مقدار، مسیرهایی که ظرفیت خازنی بین این دو مقدار دارند، مسیرهای طولانی هستند. پس از تعیین مسیرهای طولانی، گره‌هایی که تنها مختص این مسیرها باشند و در مسیرها کوتاه و بحرانی نباشند، به عنوان گره‌های آسیب‌پذیر معرفی شده در آنالیز تأخیر جدا می‌شوند. پس از آن در آنالیز ساختار با داشتن توصیف سطح گیت مدار، به محاسبه پارامترهای کنترل‌پذیری و رویت‌پذیری براساس الگوریتم SCOAP پرداخته می‌شود. با داشتن مقدار کنترل‌پذیری و رویت‌پذیری تمام گره‌های مدار، مقدار آزمون‌پذیری تروجان سخت‌افزاری برای گره‌ها محاسبه می‌شود تا از میان گره‌های مختص مسیرهای طولانی گره‌هایی که مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی بیشتری دارند انتخاب شوند. بدین ترتیب خروجی این آنالیز گره‌هایی است که در صورت درج تروجان سخت‌افزار زمانی، هم در روش شناسایی مبتنی بر تأخیر مسیرها

الگوریتم ۱ آنالیز آسیب‌پذیری

Input: Gate-level Netlist, Paths Information

Output: Vulnerable Nets ($VNets$)

1: $Paths = \text{Extrat\_Path}(Paths\ Information)$;
2: $PC_{\max} = \max(\forall\ capacitance\ of\ p_i \in Paths)$;
3: $Lpaths = \{p_i \in Paths \mid 50\%PC_{\max} < Cap_{p_i} < 70\%PC_{\max}\}$
4: $Paths = Paths - LPaths$
5: $UNets = \{n_i \in LPaths \mid n_i \notin Paths\}$
6: $Nets = \text{THT\_Testability}(Netlist)$
7: $T_{THT,\max} = \max(\forall\ THT\ testability\ of\ n_i \in Nets)$;
8: $VNets = \{n_i \in UNets \mid T_{THT,n_i} > 30\%T_{THT,\max}\}$;
9: **Return** $VNets$

و هم روش شناسایی آزمون منطقی به سختی قابل شناسایی هستند.

بر اساس مراحل نشان داده شده آنالیز آسیب‌پذیری در شکل ۲، شبه‌کدی در الگوریتم ۱ آورده شده است که نحوه انجام دقیق آنالیز آسیب‌پذیری مدارها را نشان می‌دهد. ورودی الگوریتم توصیف سطح گیت مدار و اطلاعات مسیرهای آن می‌باشد. اطلاعات مسیرهای مدار با هر ابزار سنتزی قابل وصول می‌باشد. خروجی الگوریتم نیز گره‌های آسیب‌پذیر در برابر تروجان سخت‌افزار زمانی است. در ابتدا با کمک تابع تعریف شده Extract_Path مسیرهای موجود در مدار به همراه گره‌های مسیر و ظرفیت خازنی گیت‌های موجود در آن استخراج شده و در مجموعه Paths ذخیره می‌شوند (خط ۱). پس از آن برای انجام آنالیز تأخیر بایستی بیشترین مقدار ظرفیت خازنی مسیر به عنوان مسیر بحرانی مدار پیدا شود (خط ۲). با معلوم شدن بیشترین مقدار ظرفیت خازنی مسیر مدار، مسیرهای طولانی با ظرفیت خازنی بین ۵۰ تا ۷۰ درصد بیشترین ظرفیت خازنی مسیر مدار، از بقیه مسیر های کوتاه و بحرانی جدا می‌شوند (خط ۳-۴). پس از آن گره‌های مختص مسیرهای طولانی به عنوان گره‌های آسیب‌پذیر در روش شناسایی مبتنی بر تأخیر مسیر شناسایی می‌شوند (خط ۵). پس از اتمام آنالیز تأخیر، آنالیز ساختاری با تابع THT_Testability شروع می‌شود که پس از محاسبه کنترل‌پذیری و رویت‌پذیری گره‌ها از روی توصیف سطح گیت مدار، مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی را برای تمام گره‌ها حساب می‌کند و در مجموعه Nets ذخیره می‌کند (خط ۶). پس از آن بیشترین مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی مدار به عنوان سخت‌ترین گره در شناسایی آزمون منطقی پیدا می‌شود تا معیاری برای اندازه‌گیری سختی شناسایی تروجان در آزمون منطقی باشد (خط ۷). برای تعیین حد سختی آزمون گره‌های مختص مسیرهای طولانی در آنالیز ساختاری، به بررسی مدارهای مورد مطالعه پرداخته شد. در این بررسی مشاهده شد گره‌های مختص مسیرهای طولانی دارای مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی کمتر از ۳۰ درصد بیشترین مقدار آزمون‌پذیری هستند. از این رو ۳۰ درصد بیشترین مقدار آزمون‌پذیری برای کمترین حد میزان آزمون‌پذیری تروجان سخت‌افزار زمانی در نظر گرفته می‌شود. با توجه به این مقدار، از میان گره‌های مختص مسیرهای طولانی، گره‌هایی که آزمون‌پذیری تروجان سخت‌افزار زمانی بیشتر از این مقدار دارند، به عنوان گره‌های آسیب‌پذیر در این آنالیز باقی می‌مانند (خط ۸).

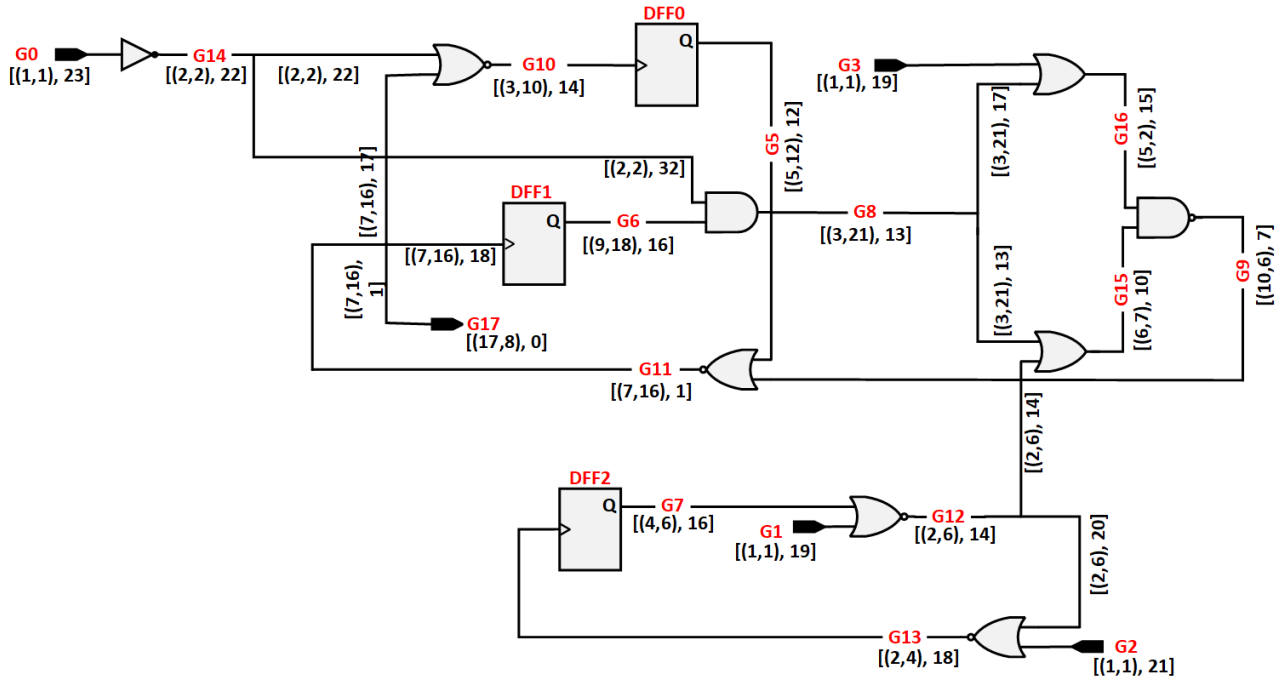
برای فهم بهتر این الگوریتم، آنالیز پیشنهادی بر روی مدار محک s27 اعمال می‌شود. شکل ۳ مدار s27 را به همراه مقادیر پارامترهای کنترل‌پذیری و رویت‌پذیری به صورت [(CC0, CC1), CB] نشان می‌دهد. همان‌طور که در شکل پیداست ۲۰ مسیر در این مدار موجود می‌باشد. با توجه به بیشترین ظرفیت خازنی مسیر در این مدار، چهار مسیر

$$(G۶ > G۸ > G۱۶ > G۹ > G۱۱),$$
$$(G۶ > G۸ > G۱۵ > G۹ > G۱۱ > G۱۰),$$
$$(G۶ > G۸ > G۱۶ > G۹ > G۱۱ > G۱۰),$$
$$(G۶ > G۸ > G۱۵ > G۹ > G۱۱ > G۱۰)$$

از مسیرهای طولانی مدار محسوب می‌شوند. در این چهار مسیر گره $G۶$ تنها گره مختص به این چهار مسیر طولانی است و بقیه گره‌ها در مسیرهای بحرانی و کوتاه حضور دارند. مقدار آزمون‌پذیری تروجان سخت‌افزاری $G۶$ برابر با ۴۳ می‌باشد که با توجه به بیشترین مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی مدار که همین گره می‌باشد، این گره می‌تواند جزء گره‌های آسیب‌پذیر مدار در برابر تروجان سخت‌افزار زمانی باشد.

۴ نتایج

برای بررسی آنالیز آسیب‌پذیری پیشنهادی در برابر تروجان سخت‌افزار زمانی معرفی شده، الگوریتم معرفی شده برای آنالیز آسیب‌پذیری به کمک زبان برنامه‌نویسی C++ پیاده‌سازی شده است. همچنین از ابزار سنتز Design Compiler برای گرفتن اطلاعات مسیرهای مدار و از شبیه‌ساز Modelsim برای شبیه‌سازی زمانی استفاده شده است. در ادامه هشت مدار محک با توصیف سطح گیت سری ISCAS89 [۱۰] به همراه اطلاعات مسیرهای آنها به الگوریتم داده شده است. جدول ۱ نتایج حاصل از آنالیز آسیب‌پذیری بر روی این مدارها را نشان می‌دهد. در این جدول منظور از DNets گره‌های قابل شناسایی هستند که مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی محدودی دارند. در مقابل گره‌هایی در مدار هستند که مقدار آزمون‌پذیری نامحدودی دارند. به این معنا که توانایی فعال شدن در آزمون منطقی را ندارند. در نهایت از گره‌های آسیب‌پذیر شناسایی شده در این آنالیز برای درج تروجان‌سخت‌افزار زمانی به صورت خازن، استفاده شده است. برای بررسی عملکرد تروجان سخت‌افزار زمانی، همان طور که در بخش معرفی آن آمده است، برای فعال کردن تروجان‌سخت‌افزار زمانی، ورودی‌های مدار باید به گونه‌ای تنظیم شوند که یک گذار از ورودی به سمت گره شامل تروجان سخت‌افزار زمانی تولید شود و این گذار در نهایت به خروجی رود تا با افزایش تأخیر، خطای زمانی ایجاد شود. با تنظیم ورودی‌های مدار در شبیه‌سازی زمانی، نتایج به دست آمده



شکل ۳. آنالیز آسیب‌پذیری مدار s27 به همراه مقادیر کنترل‌پذیری و رویت‌پذیری

نشان می‌دهد که در مسیرهای شامل این گره‌ها، با فعال شدن مسیر و تروجان‌سخت‌افزار زمانی مسیر دچار خطای زمانی می‌شود. اما در تحلیل نتایج به دست آمده از آنالیز آسیب‌پذیری مدارهای محک، همان‌طور که در جدول ۱ معلوم است مدارهای مختلف نتایج مختلفی از خود نشان داده اند که تحلیل برخی از آن‌ها آورده شده است:

s27: این مدار همان مدار نمونه‌ای است که روند آنالیز آسیب‌پذیری بر روی آن توضیح داده شد. پس از تعیین مسیرهای طولانی در آن، از میان ۱۷ گره مدار، تنها یک گره مختص مسیرهای طولانی بود که مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی آن در حدی بود که جزو گره آسیب‌پذیر مدار باشد.

s1488: در این مدار با توجه به تعداد کم مسیرهای طولانی، تعداد گره‌های مختص آن‌ها کم شده است و تنها یک گره مختص به مسیرهای طولانی پیدا شده است که این گره مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی کمی داشته و در نتیجه در روش آزمون منطقی به راحتی قابل شناسایی هست و آسیب‌پذیر نمی‌باشد.

s13270: همان‌طور که از مقدار $T_{THT,\max}$ پیداست، این مدار برای آزمون منطقی پیچیده است به طوری که از ۹۲۹۰ گره تنها ۱۷۶۷ گره مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی محدودی دارند. در این مدار نیز تعداد مسیرهای طولانی نسبت به مابقی مسیرها کم می‌باشد که این، اشتراک گره‌های مسیرهای طولانی با آن‌ها را بالا برده، به طوری که گره مختص به مسیرهای طولانی وجود ندارد.

s35932: نسبت تعداد مسیرهای طولانی به مابقی مسیرها از مدارها دیگر بیشتر است و همین امر موجب افزایش تعداد گره‌های مختص مسیرهای طولانی و در نتیجه گره‌های آسیب‌پذیر شده است.

۵ نتیجه‌گیری و کارهای آینده

در این مقاله سعی شد نوعی تروجان سخت‌افزار زمانی معرفی شود که به صورت یک عملگر خازنی به نقاط آسیب‌پذیر ورود می‌کند و در صورت فعال شدن موجب خطای زمانی در مدار می‌شود. پس از آن آنالیزی برای شناسایی نقاط آسیب‌پذیر بر اساس عملکرد ضعیف در روش‌های شناسایی مطرح شد. در این آنالیز نقاطی که مختص مسیرهای طولانی باشند و دارای مقدار آزمون‌پذیری تروجان سخت‌افزار زمانی بالایی باشند، نقاط آسیب‌پذیرند. با بررسی نتایج حاصل از آنالیز آسیب‌پذیری مدارها، مشخص شد که عمدتا تعداد این نقاط در مدار اندک هستند. از این رو از کارهای پیشنهادی در راستای این مقاله بررسی امکان امنیت در برابر این تروجان سخت‌افزار زمانی است. در واقع می‌توان با بررسی روش‌های جلوگیری از خطاهای زمانی، امکان رفع اثر مخرب تروجان سخت‌افزار زمانی را لحاظ کرد.

مراجع

[1] Swarup Bhunia and M Tehranipoor. The hardware trojan war. *Cham,, Switzerland: Springer*, 2018.

[2] Wei Hu, Chip-Hong Chang, Anirban Sengupta, Swarup Bhunia, Ryan Kastner, and Hai Li. An overview of hardware security and trust: Threats, countermeasures, and design tools. *IEEE Transactions on Computer-Aided*

جدول ۱. نتایج حاصل از آنالیز آسیب‌پذیری بر روی مدارهای محک

Circuits	s38584	s35932	s9234	s13207	s5378	s1488	s344	s27
PathCap(uF)	7.68	7	3.71	6.07	1.76	5.09	1.32	0.43
#Paths	349701	183597	230816	271186	7943	450	338	20
#LPaths	60834	52800	32992	38562	2139	69	127	2
#Nets	22144	19557	6056	9290	3173	674	184	17
#UNets	973	630	33	0	73	1	5	1
$T_{THT,\max}$	28432	304	846887	23954246	73617	437	608	43
#DNets	20374	17828	3296	1767	2993	667	184	17
#VNets	0	304	13	0	19	0	1	1

2015.

[10] Maksim jenihhin. http://www.pld.ttu.ee/~maksim/benchmarks. Accessed: December 2020.

Design of Integrated Circuits and Systems, 40(6):1010–1038, 2020.

[3] Kan Xiao, Domenic Forte, Yier Jin, Ramesh Karri, Swarup Bhunia, and Mohammad Tehranipoor. Hardware trojans: Lessons learned after one decade of research. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 22(1):1–23, 2016.

[4] Bicky Shakya, Tony He, Hassan Salmani, Domenic Forte, Swarup Bhunia, and Mark Tehranipoor. Benchmarking of hardware trojans and maliciously affected circuits. *Journal of Hardware and Systems Security*, 1(1):85–102, 2017.

[5] Jonas Krautter, Dennis RE Gnad, and Mehdi B Tahoori. Fpgahammer: Remote voltage fault attacks on shared fpgas, suitable for dfa on aes. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 44–68, 2018.

[6] Jayaram Bhasker and Rakesh Chadha. *Static timing analysis for nanometer designs: A practical approach*. Springer Science & Business Media, 2009.

[7] Michael Bushnell and Vishwani Agrawal. *Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits*, volume 17. Springer Science & Business Media, 2004.

[8] Arash Nejat, David Hely, and Vincent Beroulle. Escalation: Leveraging logic masking to facilitate path-delay-based hardware trojan detection methods. *Journal of Hardware and Systems Security*, 2(1):83–96, 2018.

[9] Seyed Mohammad Hossein Shekarian and Morteza Saheb Zamani. Improving hardware trojan detection by retiming. *Microprocessors and Microsystems*, 39(3):145–156,

MonadiAfta-Bianual Scientific Journal for Cyberspace Security
2022, Volume 20, Number 1 (pp. 18–25)
https://monadi.isc.org.ir

Presented at the ISCISC 2021 in University of Isfahan, Isfahan, Iran

Vulnerability Analysis of Digital Circuits Against Capacitor-based Timing Hardware Trojan★

Fatemeh Khormizi and Bijan Alizadeh *

School of Electrical and Computer Engineering, College of Engineering, University of Tehran, 14395-515, Tehran, Iran

ARTICLE INFO.

Keywords:
timing hardware trojan
vulnerability analysis
path-delay based detection
logic testing based detection

dor: 20.1001.1.24763047.1401.11.1.3.0

Type: research paper

ABSTRACT

Hardware Trojan is a hardware security threat that attempts to insert in the circuit and modifies the hardware stealthy. Trojan detection and design-for-trust are the main defensive strategies against hardware Trojan. The target of Trojan detection is to verify hardware Trojan and in design-for-security, the security techniques are presented for facilitating detection or preventing hardware Trojan insertion. In this work, we introduce a capacitor-based timing hardware Trojan (THT) model and then discuss how to analyze the vulnerability of gate-level circuits against such THT model. For THT that violates timing constraints in the circuit, the susceptible nets are recognized. Susceptible nets to THT are vulnerable nets in path-delay analysis and logic testing detection approaches and they are not detectable easily. The experimental results show that the number of vulnerable nets to the capacitor-based THT model is small enough so that a design-for-trust approach can be proposed.

★ The ISCISC 2021 Program Committee effort is highly acknowledged for reviewing this paper.
* Corresponding author
Email addresses: fatemeh.khormizi@ut.ac.ir (Fatemeh Khormizi), b.alizadeh@ut.ac.ir (Bijan Alizadeh)