بررسی آسیب‌پذیری‌های TrustZone ARM

سید جواد موسوی حسینی
دانشجو ارشد مهندسی کامپیوتر گرایش شبکه‌های کامپیوتری، دانشگاه جامع امام حسین (ع)
علیرضا فرجی
دانشجو ارشد مهندسی کامپیوتر گرایش شبکه‌های کامپیوتری، دانشگاه جامع امام حسین (ع)

چکیده

آسیب‌پذیری امنیتی پردازنده‌هایی که از معماری کورتکس کمپانی آرم استفاده می‌کنند دستگاه‌های زیادی را تحت‌تأثیر قرار خواهد داد. این معماری اغلب در دستگاه‌های همراه مورداستفاده قرار گرفته است. مشکلات امنیتی مرتبط با حافظه میلیون‌ها سی‌پی‌یو ی تولید کمپانی اینتل را تحت‌تأثیر قرار داده است. کمپانی ARM نیز تأیید کرده که تعداد بسیاری از سی‌پی‌یو هایی که از معماری کورتکس (Cortex) استفاده می‌کنند از آسیب‌پذیری‌های امنیتی رنج می‌برند. معماری کورتکس در طیف وسیعی از دستگاه‌های اندرویدی و دستگاه‌هایی که از سیستم‌عامل IOS Apple استفاده می‌کنند استفاده شده است. ما در این مقاله ابتدا به معرفی TrustZone پرداخته و سپس در ادامه به بررسی آسیب‌پذیری‌های آن می‌پردازیم.

واژگان کلیدی: آسیب‌پذیری، امنیت، آرم، معماری، کورتکس

مقدمه

با ظهور دستگاه‌های تلفن همراه در سال‌های اخیر، چشم‌انداز محاسبات را به طور قابل‌توجهی تغییر داده‌اند. از اوایل سال ۲۰۱۴، استفاده از اینترنت در دستگاه‌های تلفن همراه از رایانه شخصی بیشتر شده است. یک سیستم معمولاً فقط در سطح نرم افزار ایمن می‌شود. با این حال، با ایجاد بررسی‌های امنیتی در سخت افزار سیستم می‌توان به سطح بیشتری از امنیت دست یافت. این ایده توسط مفهوم محیط‌های اجرایی مورد اعتماد (TEE) پیاده سازی شده است(Platform, 2013).

یک استاندارد صنعتی در مورد آنچه TEE به طور گسترده پذیرفته شده بیان می‌کند که TEE یک منطقه امن از پردازنده اصلی یک دستگاه است که باید اجرای ایمن ایزوله نرم‌افزار امنیتی مجاز را ارائه دهد. علاوه بر این، باید بتواند در زمینه‌های احراز هویت کاربر، پردازش و جداسازی مورداعتماد، اعتبارسنجی تراکنش، استفاده از منابع امن و صدور گواهی برنامه‌هایی داشته باشد. همچنین می‌توان از TEE برای محافظت از محتوای دیجیتالی مانند ویدئوهای پخش‌شده در برابر سرقت با نگه‌داشتن همه برنامه‌های مرتبط در ناحیه امن پردازنده استفاده کرد. ازآنجایی‌که ARM به طور گسترده در اکثر دستگاه‌های کنترل‌کننده موبایل و میکروکنترلر مستقر شده است، هدف TrustZone تأمین امنیت برای آن پلتفرم‌ها است.

آشنایی با پردازنده‌های ARM

تا به امروز 11 نسخه از معماری‌های ARMتعریف شده است، یعنی ARMv1 تا ARMv11 که یکی از مهم‌ترین ویژگی‌های آن استفاده از RISC هست.

RISC که مخفف Reduced instruction Set Computing یا مجموعه دستورهای ساده شده است در واقع نوعی از طراحی CPU است که پایه و اساس آن، ساده‌سازی دستورها است که منجر به بازده بالا و سرعت‌بخشیدن به اجرای دستورها می‌شود. پردازنده‌ای که بر اساس این طراحی ساخته می‌شود را RISC می‌نامند. مهم‌ترین و معروف‌ترین معماری که بر اساس RISC طراحی شده، ARM است. چندین نوع مختلف از معماری برای پردازنده‌های ARM وجود دارد که از آن جمله می‌توان به ARMv3،V2، Arm v7 و ... اشاره کرد. کمپانی‌ها برای استفاده از هر کدام از این طراحی‌ها باید گواهی مربوط به آن را از ARM دریافت کنند. کمپانی‌ها از این معماری در ساخت پردازنده‌های موردنظر خود بهره برده و در نهایت با یکپارچه‌سازی آن با واحد پردازش گرافیک، حافظه رم و قسمت کنترلر باند رادیویی سیستم روی - یک - چیپ خود را می‌سازند(soc) . پس لازم است بدانید که کل SoC بر اساس معماری ARM تولید نمی‌شود و تنها بخش CPU آن بر مبنای معماری ARM طراحی و تولید می‌گردد.

محبوب‌ترین CPUهای موجود در بازار اکنون از معماری ARMv7 (۳۲ بیتی، یعنی Cortex-A8، Cortex-A9، Cortex-M4) یا ARMv8 (۶۴ بیتی، یعنی Cortex-A53، Cortex-A57) استفاده می‌کنند.

معماری ARM دارای ویژگی‌های زیر است(Ngabonziza et al., 2016) :

۱) دارای یک پرونده ثبت یکنواخت بزرگ است. [1]
۲) عملیات پردازش داده فقط بر روی رجیسترها عمل می‌کند، نه مستقیماً روی حافظه.
۳) حالت‌های آدرس دهی ساده.
۴) دستورالعمل‌هایی که یک شیفت را با یک عملیات حسابی و منطقی ترکیب می‌کند.
۵) حالت‌های آدرس دهی افزایش خودکار و کاهش خودکار برای بهینه سازی حلقه‌های برنامه.
۶) بارگیری و ذخیره دستورالعمل‌های متعدد برای به حداکثر رساندن توان داده.

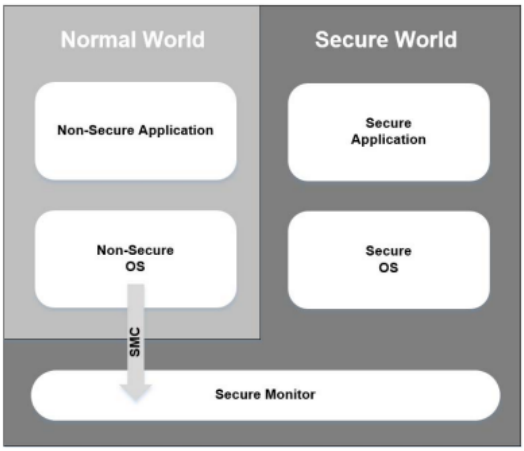
[1] large uniform register file

بررسی TrustZone

TrustZone(Holdings, 2009) یک افزونه امنیتی سخت افزاری اختیاری از معماری پردازنده ARM است که شامل باس یکپارچه و تجهیزات جانبی سیستم است. امنیت TrustZone بر اساس ایده پارتیشن بندی تمام سخت افزار و نرم افزار System on Chip (SoC) به دو محیط است: محیط امن و محیط عادی. محیط امن یعنی همه چیز زمانی که حالت پردازنده امن است اجرا می شود و محیط عادی همه چیزهایی است که وقتی پردازنده در حالت غیر ایمن است اجرا می شود. با توجه به اینکه محیط امن محدود نیست موانع سخت افزاری برای جلوگیری از دسترسی اجزای عادی جهان به منابع امن جهانی ایجاد شده است. به طور خاص، سیستم حافظه از دسترسی محیط عادی به چند مورد جلوگیری می‌کند :

۱) مناطقی از حافظه فیزیکی که به عنوان محیط امن تعیین شده اند
۲) کنترل‌های سیستمی که برای محیط امن اعمال می‌شوند.
۳) سوئیچینگ وضعیت، فرای تعدادی از مکانیسم‌های تایید شده.

این پارتیشن بندی ممکن است فیزیکی یا مجازی باشد. به عنوان مثال، یک هسته پردازشگر فیزیکی توسط محیط معمولی و ایمن به روش زمان‌بندی[2] به اشتراک گذاشته می‌شود، که به هر دو محیط این توهم[3] را می‌دهد که صاحب پردازنده است.
محیط امن امکان ایجاد یک محیط قابل برنامه ریزی جدا شده را به وجود می‌آورد که می‌تواند طیف گسترده ای از برنامه‌های امنیتی را اجرا کند.



شکل ۱ - شمای TrustZone بر ARM Cortex-A

بررسی آسیب پذیری Armv8-M Architecture-Stack sealing (CVE-2020-16273)

گزارشی به Arm ارائه شده است که نشان می‌دهد نرم‌افزار امنی که روی پردازنده‌های Armv8-M اجرا می‌شود ممکن است در برابر حملات ایجاد شده از حالت غیر امن آسیب‌پذیر باشد. اگر نرم‌افزار Secure هنگام ایجاد پشته‌ها، یا هنگام انجام انتقال غیراستاندارد بین حالت‌ها به‌درستی پشته‌های Secure را مدیریت نکند، این آسیب پذیری صرفاً در نرم افزار رخ می‌دهد و به آن Stack Sealing می‌گویند. فقط در پردازنده‌های Armv8-M که از پسوند امنیتی TrustZone استفاده می‌شود، یعنی کد در حال اجرا در هر دو حالت امن و غیرایمن وجود دارد، ضروری است. هیچ تغییری در سخت افزار لازم نیست(Cerdeira et al., 2020).

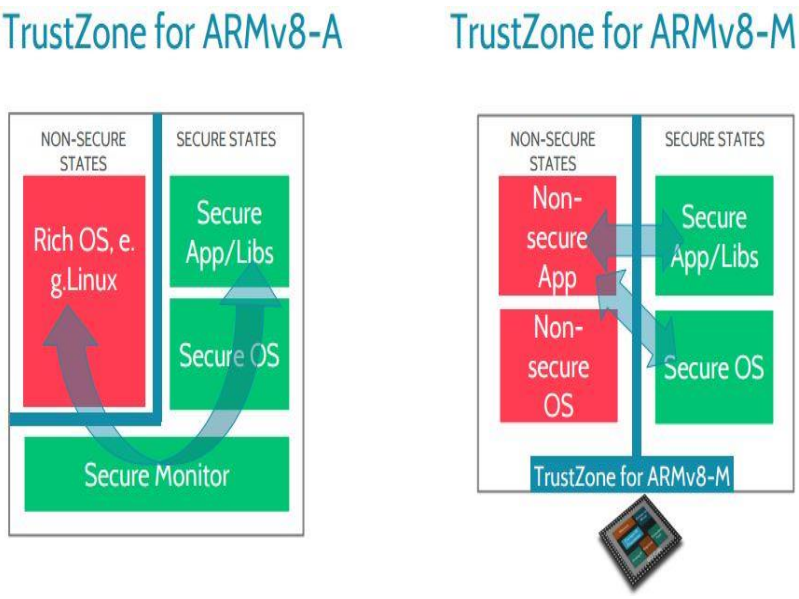
[2] time-sliced fashion
[3] illusion

این آسیب‌پذیری می‌تواند به یک عامل مخرب اجازه دهد تا یک پشته در نرم‌افزار Secure world راه‌اندازی کند بدون اینکه فوراً یک استثناء خطا در دنیای امن ایجاد کند. این تغییر می‌تواند منجر به عملکرد نادرست اجرای کد امن شود که می‌تواند باعث انکار سرویس از کد امن یا عملکرد نادرست پلت فرم شود.

اگر عملیات مهر و موم پشته در نرم افزار Secure انجام نشود، می‌تواند به مهاجمی که کد را در حالت غیر ایمن اجرا می‌کند، اجازه می‌دهد تا بدون ایجاد خطا، یک حمله تحت جریان پشته را راه اندازی کند. این فقط می‌تواند بر روی نرم‌افزارهای پردازنده‌های مبتنی بر Armv8-M با پسوند TrustZone تأثیر بگذارد که نرم‌افزار را در هر دو حالت امن و غیرایمن اجرا می‌کند.

در یک پردازنده مبتنی بر Armv8-M با پسوند امنیتی، اگر نرم افزار Secure فقط از نشانگر پشته اصلی (MSP_S) استفاده کند و فقط از دستورالعمل BLXNS برای جابجایی از حالت امن به حالت غیر ایمن یا از طریق دنباله‌های وقفه استفاده کند، سیستم تحت تأثیر این آسیب پذیری قرار نگرفته است. با این حال، برای اطمینان از محافظت از سیستم در برابر حملات زیر جریان پشته، Arm توصیه می‌کند که پشته‌های امنی که در سیستم استفاده می‌شوند، مهر و موم شوند.

نرم‌افزاری که بر روی پردازنده‌های مبتنی بر Armv8-M با پسوندهای امنیتی برای جداسازی بین حالت‌های امن و غیرایمن اجرا می‌شود، و در جایی که یک عامل مخرب ممکن است کد خود را در دنیای غیر امن اجرا کند، نیاز دارد که همه پشته‌های امن مهر و موم شوند. نرم‌افزار حالت امن که روی پردازنده‌های Arm Cortex®-M23، Cortex-M33، Cortex-M35P یا Cortex-M55 اجرا می‌شود، و هر نرم‌افزار حالت امنی که بر روی یک پردازنده مبتنی بر Armv8-M اجرا می‌شود که برنامه‌های افزودنی امنیتی توسعه‌یافته تحت مجوز را پیاده‌سازی کرده است.



شکل ۲: TrustZone for ARM v8-M vs TrustZone for ARM v8-A

بررسی آسیب پذیری Spectre Branch History Buffer (CVE-2022-23960)

اسپکتر یک آسیب است که یک برنامه را جهت دسترسی به محل‌های دلخواه در فضای حافظه برنامه ترغیب می‌نماید. یک مهاجم

ممکن است محتوای حافظه در دسترس را بخواند و سپس به‌طور بالقوه توانایی دسترسی به داده‌های حساس را داشته باشد(Fitzek et al., 2015)

ARMگزارش داده‌است که اکثر پردازنده‌های آن‌ها آسیب‌پذیر نیستند و لیستی از پردازنده‌های خاص را که تحت آسیب‌پذیری اسپکتر قرار می‌گیرند، منتشر کرد:

Cortex-R7، Cortex-R8، Cortex-A8، Cortex-A9، Cortex-A15، Cortex -A17، Cortex-A57، Cortex-A72، Cortex-A73 و ARM Cortex-A75سایر هسته‌های CPU سفارشی تولیدکنندگان که مجموعه‌های دستورالعمل ARM را اجرا می‌کنند، مانند نمونه‌هایی که در اعضای جدید پردازنده‌های سری A اپل وجود دارد، نیز آسیب‌پذیر هستند(Deb, 2018; Kocher et al., 2019)

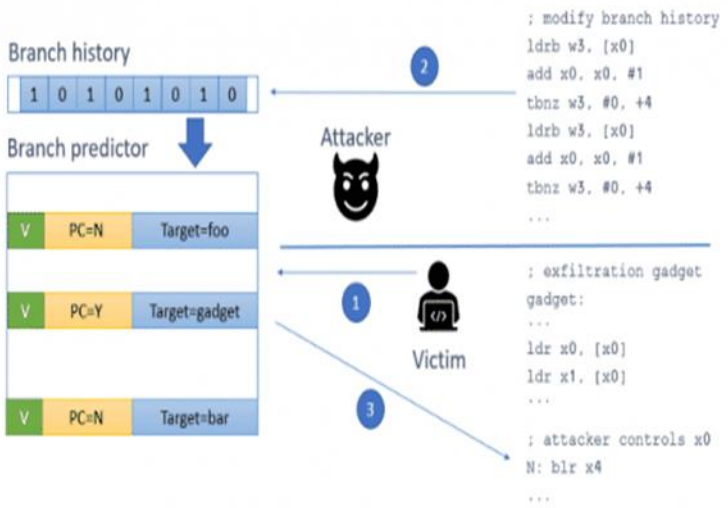


Figure 3. (1) Victim runs normally, executing an indirect branch with different safe targets ("foo" and "bar"), and some other branch to a potential (given the right register values) exfiltration "gadget". This makes several branch predictions in the victim's context. (2) The attacker tampers the global branch history, forcing the combination of PC=N and history to alias with the "gadget" entry. (3) The victim executes an allegedly safe branch that is mis-predicted , redirecting the control flow to a gadget that, with attacker controlled registers, causes speculative leakage.

شکل ۳: Spectre Branch History Buffer

بررسی آسیب پذیری Attacking TrustZone on devices lacking memory protection (DMA Attack)

یک محیط اجرای قابل اعتماد (TEE) تعبیه شده در هسته‌های پردازنده که توسط برخی از فروشندگان ماژول‌های ARM ارائه می‌شود که به طور کامل با مشخصات TrustZone مطابقت ندارند، و ممکن است منجر به آسیب پذیری در سیستم شود. حمله DMA از دنیای ناامن به دنیای امن و طراحی و اجرای این حمله در یک سخت افزار ناامن واقعی را انجام می‌دهد(Pinto & Santos, 2019).

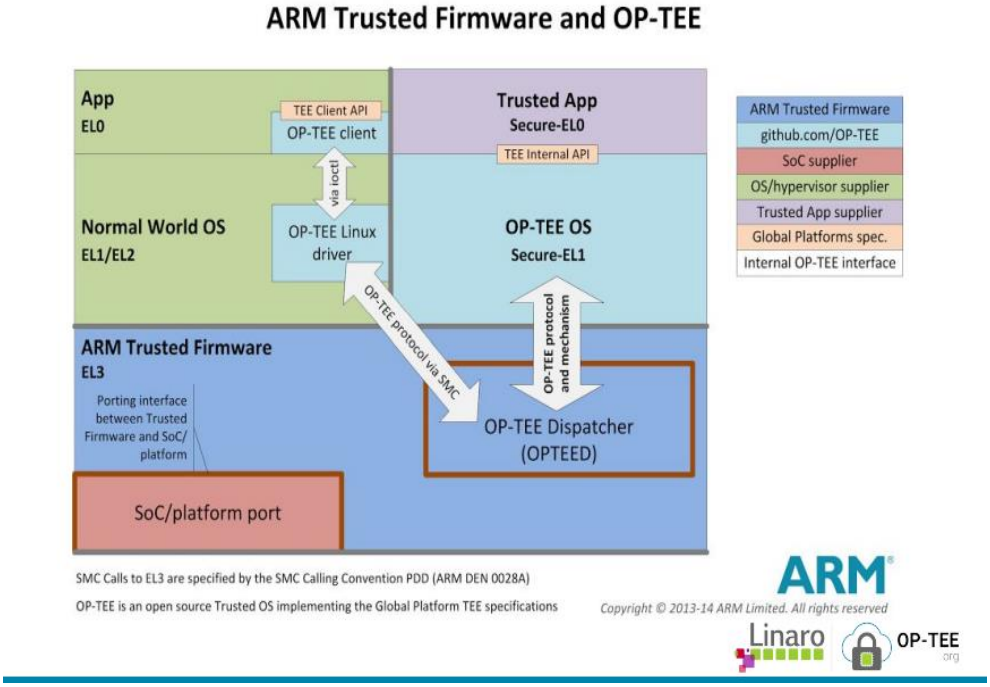
اگرچه ARM TrustZoneیک راه عالی برای پیاده سازی مکانیسم‌های امنیتی در دستگاه‌های تعبیه شده در اینترنت اشیا است، اما همچنان مستعد اجرای ناکافی سخت افزار و نرم افزار است. بنابراین، سخت‌افزار شرکت‌های مختلف مانند گوگل، سامسونگ، هواوی و غیره ممکن است همچنان تحت تأثیر آسیب‌پذیری‌های شدیدی باشد که کل مجموعه امنیتی را به خطر می‌اندازد یکی از ویژگی‌های کلیدی AMBA (معماری پیشرفته گذرگاه میکروکنترلر) AXI (رابط توسعه پذیر پیشرفته) جداسازی فضای آدرس است. بنابراین،

فقدان آنها باعث ایجاد جدایی ناامن حافظه بین دنیای عادی و دنیای امن می‌شود.

OP-TEEیکی از سیستم عامل‌های رایجی است که روی TrustZone اجرا می‌شود

(Open Portable Trusted Execution Environment) OP-TEEیک سیستم عامل محبوب، منبع باز و TrustZone است. ما از OP-TEE به عنوان یک سیستم عامل TrustZone مرجع برای نشان دادن حمله ارائه شده استفاده کردیم، اگرچه این حمله به دلیل اشکال OP-TEE نیست بلکه یک ویژگی سخت افزاری از دست رفته است. حمله ما به مهاجم اجازه می‌دهد تا کد دلخواه را در دنیای امن اجرا کند یا داده‌های دلخواه را از دنیای امن در سیستم عامل غنی بخواند. حمله ما یک حمله کنترل جریان بر روی هسته OP-TEE است. همچنین، یک آسیب‌پذیری سخت‌افزاری را در SoC نشان می‌دهیم که ARM TrustZone را به خطر می‌اندازد. با استفاده از حمله DMA، ما توانایی جایگزینی برنامه‌های قابل اعتماد را با برنامه‌های مخرب به دست می‌آوریم. علاوه بر این، ما یک حمله به رایانه Raspberry Pi را نشان می‌دهیم و توضیح می‌دهیم که چگونه این روش بر روی دیگر پلتفرم‌ها تأثیر می‌گذارد. همچنین اقداماتی را برای کاهش این آسیب‌پذیری ارائه می‌کند. حمله زمانی که AMBA AXI حضور داشت امکان پذیر نبود. متأسفانه AMBA AXI در چند دستگاه سخت افزاری مدرن از جمله Raspberry Pi 3،4 و Jetson Nano وجود ندارد.

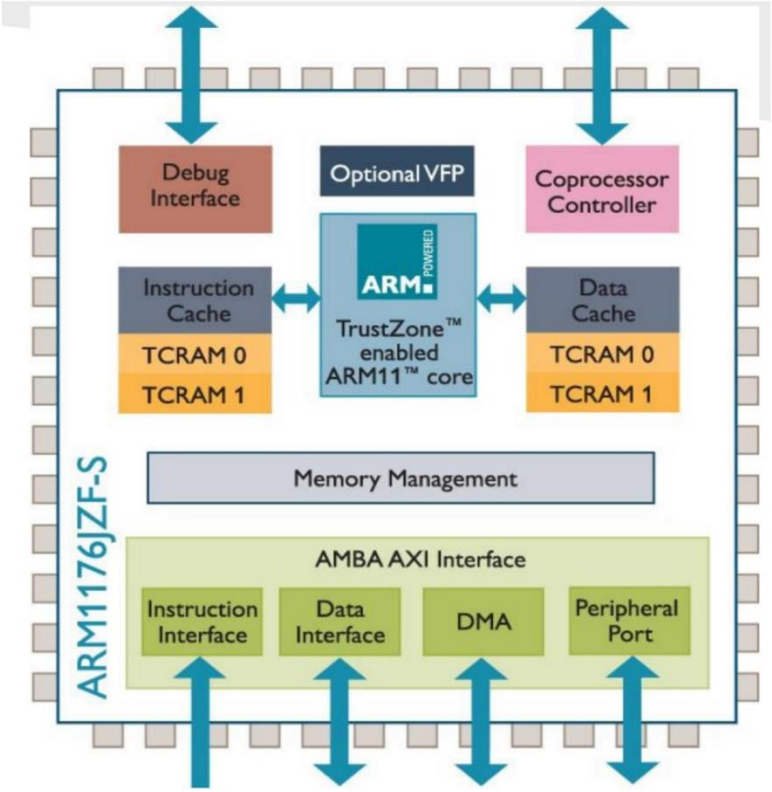
استفاده از حملات DMA در TrustZone طیف وسیعی از احتمالات حمله را در اختیار شما قرار می‌دهد. ، ما استفاده از حملات DMA را برای انجام ACE (اجرای کد دلخواه) نشان می‌دهیم. با این حال، می‌توان از این روش برای خواندن کد دلخواه از حافظه فیزیکی استفاده کرد که به موجب آن یک کاربر مخرب می‌تواند به داده‌های حساس دسترسی پیدا کند.



شکل ۴: Arm Trusted Firmware and OP-TEE

شکل (۴) تراشه BCM2837 را نشان می‌دهد. این Broadcom SOC از تراکنش‌های TrustZone و DMA از طریق AMBA Advanced Microcontroller Bus Architecture AXI (Advanced Extensible Interface) پشتیبانی می‌کند. همانطور که قبلاً ذکر شد، پیاده سازی همه فروشندگان با کل مشخصات سخت افزاری مطابقت ندارد. برای مثال، شکل نشان می‌دهد که BCM2837

دارای گذرگاه AXI صحیح است، اما فاقد TZASC و TZPC است که آن را در برابر حملات DMA آسیب‌پذیر می‌کند(Stajnrod, 2021).



شکل (۴): تراشه BCM2837

TZASCیک ابزار جانبی با معماری گذرگاه میکروکنترلر پیشرفته (AMBA) سازگار با سیستم روی تراشه (SoC) است. این یک کنترل‌کننده فضای آدرس با کارایی بالا و بهینه‌سازی منطقه با رابط‌های گذرگاه AMBA روی تراشه است که با پروتکل AMBA Advanced Extensible Interface (AXI) و پروتکل AMBA Advanced Peripheral Bus (APB) مطابقت دارد.

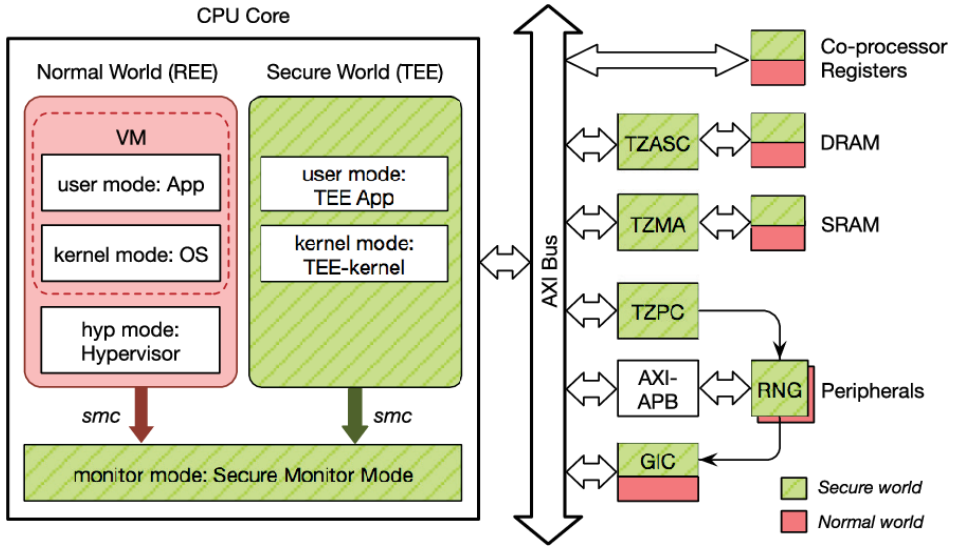
TrustZone Protection Controller (TZPC)، TZProtCtrl، یک دستگاه جانبی SoC سازگار با AMBA است که توسط ARM Limited توسعه، آزمایش و دارای مجوز است. TZPCیک رابط نرم افزاری برای بیت‌های حفاظتی در یک سیستم امن در طراحی TrustZone فراهم می‌کند. انعطاف پذیری سیستم را برای شما فراهم می‌کند تا بتوانید مناطق مختلف حافظه را به عنوان امن یا غیر ایمن پیکربندی کنید.

TZPC دارای ویژگی‌های زیر است:

دارای بیت‌های حفاظتی است که به شما امکان می‌دهد تا ۲۴ ناحیه حافظه را به صورت ایمن یا غیر ایمن برنامه ریزی کنید.

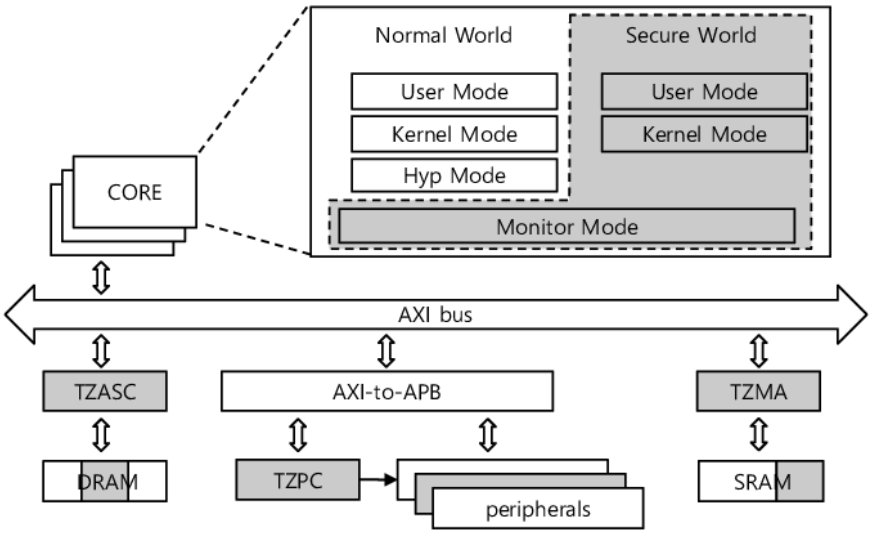
دارای بیت‌های منطقه ایمن است که به شما امکان می‌دهد یک منطقه از RAM داخلی را به دو قسمت امن و غیر ایمن تقسیم کنید.

در مناطقی که دارای یک رابط سیستم AMBA APB است، هیچ حالت انتظار APBیا پاسخ خطای Slave ایجاد نمی‌کند و بنابراین با پروتکل AMBA 2 APB سازگار است.



شکل (۵):TrustZone Protection Controller

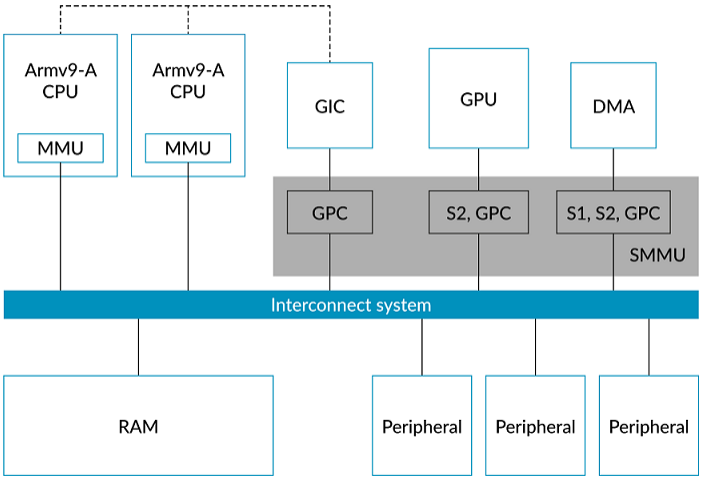
هنگام انتخاب SoC، باید الزامات دستگاه را با ویژگی‌های SoC مقایسه کنید. در هنگام انتخاب SoC، می‌خواهیم مطمئن شویم که معماری SoC دارای تمام تراشه‌های مورد نیاز برای ARM TrustZone برای درست کار کردن است (TZASC، TZPC، گذرگاه پشتیبانی‌شده و غیره). متأسفانه، بررسی معماری SoC همیشه آسان و خودکار نیست زیرا همه فروشندگان معماری SoC خود را منتشر نمی‌کنند. ما پیشنهاد می‌کنیم که فروشندگان SoC در مورد معماری خود در مورد ویژگی‌های امنیتی شفاف‌تر باشند. ما همچنین توصیه می‌کنیم که سازندگان مطمئن شوند که سخت افزار SoC آنها از مشخصات TrustZone ARM Core و TrustZone پشتیبانی می‌کند. در مواردی که TrustZone کاملاً سازگار در دسترس نیست (فقدان سخت‌افزار در SoC که TrustZone را ایمن می‌کند) استفاده شود. پشتیبانی از واحد مدیریت حافظه سیستم (SMMU) (که با نام IOMMU نیز شناخته می‌شود) به سیستم‌ها اجازه می‌دهد تا جداول صفحه نمایه را با تجهیزات جانبی به اشتراک بگذارند و سازگاری با پشتیبانی از دستگاه مجازی را در سطح سیستم با مدل حافظه معماری Arm فراهم می‌کند.



شکل (۶):TrustZone Address Space Controller

استفاده از SMMU (شبیه به IOMMU در Intel x86) برای پیکربندی آدرس‌های خاص برای کنترلرهای DMA. SMMU به عنوان MMU برای دسترسی BUS کار می‌کند، بنابراین هرگونه دسترسی به حافظه از طریق BUS با مجوز پیکربندی شده برای آدرس دسترسی مطابقت دارد. با SMMU و پیکربندی صحیح، حمله DMA از طریق تجهیزات جانبی امکان پذیر نخواهد بود. همچنین مهم است که توجه داشته باشید که یک مهاجم هسته می‌تواند این پیکربندی را تغییر دهد. در مورد Raspberry Pi، با غیرفعال کردن کنترل‌کننده DMA، یک کاربر یا ابزار جانبی غیرمجاز نمی‌تواند از تراکنش‌های DMA استفاده کند. دنیای امن را روی یک رم متفاوت بدون نگاشت کنترلر DMA تنظیم کنید تا هیچ رابط فیزیکی بین دنیای عادی و امن وجود نداشته باشد.

TRESOR-HUNT بر این بینش تکیه دارد که دشمنان دارای قابلیت DMA محدود به خواندن حافظه فیزیکی نیستند، بلکه می‌توانند مقادیر دلخواه را نیز در حافظه بنویسند. کلیدهای رمزگذاری هارد دیسک اگر در RAM ذخیره نمی‌شدند، ایمن در نظر گرفته می‌شدند. با این حال، TRESOR-HUNT کدهای مخرب را با استفاده از یک حمله DMA به هسته تزریق می‌کند و سپس کلیدهای رمزگذاری دیسک را از CPU به حافظه سیستم هدف استخراج می‌کند تا با استفاده از یک انتقال معمولی DMA بتوان آنها را از آن بازیابی کرد. نشان می‌دهد که یک دشمن با دسترسی فیزیکی به یک دستگاه می‌تواند با اتصال یک کنترل کننده حافظه مخرب به پین‌های در معرض هر سوکت DIMM RAM، نقش کنترل کننده حافظه دستگاه را جعل کند. با انجام این کار، یک مهاجم دسترسی کامل (READ/WRITE) به حافظه هدف خواهد داشت



شکل (۷): System Memory Management Unit

بحث و نتیجه‌گیری

به طور کلی، می‌توان دید که TrustZone ARM یک پیاده سازی قابل اجرا از محیط اجرای مورد اعتماد است. ARM مکانیزم‌های سخت‌افزاری و نرم‌افزاری را برای جداسازی داده‌های ایمن و ناامن ارائه کرده است. کنترل‌کننده‌های سخت‌افزار سطح قابل قبولی از سازگاری با عقب را فراهم می‌کنند و سعی می‌کنند از کاهش بیش از حد تاخیر جلوگیری کنند. معماری نرم افزار امکان انعطاف پذیری را فراهم می‌کند: محیط امن می‌تواند یک فرآیند پیرو، یک سیستم عامل کامل یا جایی در بین باشد. در این مقاله ما به بررسی برخی از آسیب پذیری‌های موجود آن پرداختیم.

منابع:

vulnerabilities in trustzone-assisted tee systems. 2020 IEEE Symposium on Security and Privacy (SP),

Deb, P. S. (2018). An analysis on effects after mitigating meltdown and spectre vulnerabilities.

Fitzek, A., Achleitner, F., Winter, J., & Hein, D. (2015). The ANDIX research OS—ARM TrustZone meets industrial control systems security. 2015 IEEE 13th International Conference on Industrial Informatics (INDIN),

Holdings, A. (2009). ARM security technology: Building a secure system using trustzone technology. *Retrieved on June*, *10*, 2021.

Kocher, P., Horn, J., Fogh, A., Genkin, D., Gruss, D., Haas, W., Hamburg, M., Lipp, M., Mangard, S., & Prescher, T. (2019). Spectre attacks: Exploiting speculative execution. 2019 IEEE Symposium on Security and Privacy (SP),

Ngabonziza, B., Martin, D., Bailey, A., Cho, H., & Martin, S. (2016). Trustzone explained: Architectural features and use cases. 2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC),

Pinto, S., & Santos, N. (2019). Demystifying arm trustzone: A comprehensive survey. *ACM Computing Surveys (CSUR)*, *51*(6), 1-36.

Platform, G. (2013). Global platform made simple guide: Trusted execution environment (tee) guide. *Derniere visite*, *12*(04).

Stajnrod, R. (2021). *Attacking ARM TrustZone using Hardware vulnerability* The Interdisciplinary Center, Herzliya].

22 th International Conference on
Information Technology,
Computer and Telecommunication
Event Place: Tbilisi,Georgia
www.itctconf.ir
بیست و دومین کنفرانس بین المللی
فناوری اطلاعات، کامپیوتر و مخابرات | گرجستان
22th International Conference on Information Technology,Computer and Telecommunication
مجلات معتبر بین المللی ۲۶ اسفندماه ۱۴۰۲

Investigating TrustZone ARM vulnerabilities

Alireza Faraji **Seyed Javad Mousavi**

Abstract

The security vulnerability of processors using Arm's Cortex architecture will affect many devices. This architecture is often used in mobile devices. Security problems related to memory have affected millions of CPUs manufactured by Intel. ARM has also confirmed that many CPUs that use the Cortex architecture suffer from security vulnerabilities. The Cortex architecture is used in a wide range of Android devices and devices running Apple's iOS operating system. In this article, we first introduce TrustZone and then continue to examine its vulnerabilities.

Keywords: Vulnerability, Security, Logo, Architecture, Cortex