
Foto: Bet_Noire / iStock

RISC-V med fokus på sikkerhed

Den særdeles fleksible og licensfrie RISC-V arkitektur bliver stadig mere udbredt. Men hvordan realiserer man i praksis en sikker implementering?

Af John Hallman, Product Manager, Trust and Security, OneSpin Solutions

Der er mange grunde til den stadig større interesse for RISC-V processorarkitekturen. Blandt de vigtigste er den høje fleksibilitet og muligheden for at optimere RISC-V hardwaren til specifikke arbejdsbelastninger (workloads).

Den procesteknologiske udvikling, som beskrevet i Moore's lov, udvikler sig ikke helt på samme niveau som tidligere, og derfor er det vigtigt at kunne lave tilpasninger og optimeringer for at understøtte de stadig højere processeringskrav, der ikke længere alle kan realiseres via procesteknologiske fremskridt.

RISC-V arkitekturen udmærker sig endvidere ved at være fri for licens- og royalty-omkostninger, hvilket gør det muligt for en større gruppe af virksomheder at udvikle innovative og kostoptimerede slutprodukter. Det gælder ikke mindst inden for områder som IoT og wearable enheder, hvor der f.eks. er integreret AI-kapabilitet.

Udviklere af systemchip (SoC) løsninger benytter ofte tredjeparts RISC-V IP-processorkerner, og disse designs og deres tilhørende toolchains kan udvides med specialtilpassede (custom) instruktioner, der er dedikeret til brug i den aktuelle SoC-implementering.

Gennem tilgang til et højkvalitets verifikationsmiljø omkring den valgte RISC-V og faciliteter til test på systemniveau kan udviklerne få en vis sikkerhed for, at den valgte IP-byggeblok ikke har nogle kritiske bugs.

Desværre er dette i mange applikationer ikke tilstrækkeligt, da der vil være andre seriøse risici, der skal tages højde for, som det belyses i det følgende.

Sårbarheder og trojanske heste

Traditionelt har sikkerhedsmæssige sårbarheder i elektroniksystemer været relateret til problemer på systemniveau eller i software.

Men på det seneste er hardware IP-byggeblokke – herunder ikke mindst processor IP – udviklet sig til at blive et centralt problem (se figur 1). Processor-implementeringerne benytter pipeline-baserede mikroarkitekturer og inkluderer ofte features til ydelses- og power-optimering.

Den større kompleksitet forøger risikoen for at overse ikke alene nogle funktionelle bugs, men også nogle sikkerhedsmæssige sårbarheder.

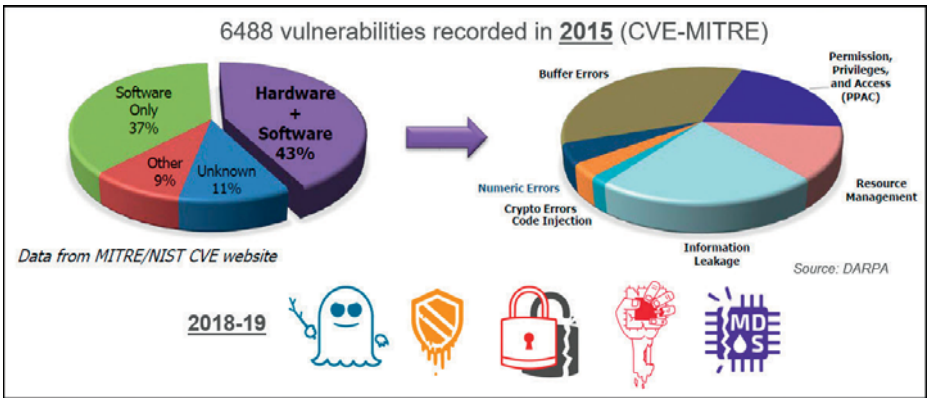
De forskere, der opdagede Meltdown- og Spectre-angrebene i begyndelsen af 2018, har demonstreret, at ydeevne-optimeringsfunktioner i processorer kan bruges på utilsigtede måder, så der kan påføres store skader. Siden da er der opdaget mange flere sårbarheder i både high-end og low-end processorer.

Sidekanaler og transiente eksekverings-angreb kan bryde ind i ellers sikre enklaver og tillade, at ondsindede applikationer lækker fortrolige data eller endda overtager kontrollen med systemet. Og i modsætning til software kan hardwareproblemer ikke let 'repareres' med over-the-air opdateringer. Hvis man skal løse et hardwareproblem ved hjælp af software, medfører det ofte alvorlige ydelsesmæssige forringelser.

En mindre sandsynlig risiko, hvis alvorlighed dog er endnu større, er tilstedeværelsen af ondsindede logik- eller hardware trojanske heste i RISC-V-kernen. En trojansk hest er en logikfunktion, der bevidst er designet til at komme 'snigende', og som primært aktiveres under meget sjældne omstændigheder, der kun kendes af angriberen.

En specifik sekvens af data og kontrolhændelser, der ikke vil indtræde, mens

...FORTSÆTTES NÆSTE SIDE



Figur 1. CVE-MITRE databasen registrerede 6488 sårbarheder i 2015, hvoraf 43 procent kan klassificeres som software-assisterede hardware-baserede sårbarheder. I 2018 og 2019 har forskere opdaget og rapporteret adskillige sårbarheder i processorer – herunder Meltdown og Spectre, Foreshadow, ZombieLoad, og RIDL og Fallout. Kilde: DARPA and OneSpin.

FORTSAT FRA SIDE 17:

systemet kører inden for dets tiltænkte anvendelsesområder, udløser trojan-logikken, som efterfølgende leverer f.eks. en skadelig payload, lækker en hemmelighed eller ødelægger systemfunktionen.

SoC-løsninger, der benytter open source eller tredjeparts RISC-V-kerner, kan ikke længere ignorere disse risici.

Det kan være svært at sikre, at en processor gør, hvad den skal gøre. Men at sikre, at den *ikke* gør noget, som den *ikke* skal gøre, er en endnu mere udfordrende opgave, der stadig stort set er uadresseret.

I sikkerhedskritiske systemer, hvor databeskyttelse er en helt altafgørende parameter, er der brug for effektive løsninger af høj kvalitet, der adresserer mulige sikkerhedssårbarheder og truslen fra trojanske heste.

Smart hardware-sikring

For at sikre den påkrævede troværdighed (trust) og sikkerhed i RISC-V IP'er kræves der innovative og effektive tekniske løsninger, der er komplementære i forhold til de tilgange, som normalt bruges i forbindelse med sikring af funktionel korrekt opførsel i de tiltænkte anvendelsesscenarier (se figur 2).

IP-leverandørerne er ansvarlige for leveringen af 'state-of-the-art' trust og sikkerheds-verifikationsløsninger, mens IP-integratorer skal have tilgang til uafhængige sikringsløsninger, som kan implementeres hurtigt og uden dyb kendskab til IP implementeringsdetaljerne.

Formelle metoder kan på udtømmende vis analysere hardwarefunktioner og levere bevis for, at IP'en eller SoC præcis matcher den forventede opførsel, der ofte er opsamlet i SystemVerilog assertions (påstande).

Hardware formel verifikations-teknikker, der bruger kommercielle model-checkere, er blevet adopteret i stor stil gennem det seneste årti. Mens nogle veldefinerede formelle verifikationsopgaver kan automatiseres ved brug af Apps, så kræves der generelt en signifikant ingeniørmæssig indsats for at opsamle IP'ens forventede opførsel med afsæt i assertions. Der er desuden ikke nogen garanti for, at der er blevet skrevet tilstrækkeligt mange assertions.

Udokumenterede funktioner eller utilsigtede gab i de beskrevne sæt af assertions kan give anledning til ikke-verificeret IP-funktionalitet.

RISC-V's open-source konceptet tillader udvikling af 'prepackaged' og uafhængige sikkerhedsløsninger. OneSpin's 'RISC-V Integrity Verification Solution' kan f.eks. benyttes sammen med mange forskellige mikroarkitekturer.

Den inkluderer modeller for RISC-V ISA og privileged ISA, der kan udvides til at inkludere tilpassede instruktioner.

Et centralt aspekt i denne løsning er, at den er baseret på OneSpin's GapFreeVerification proces, som leverer et strengt bevis på, at det sæt af assertions, der modellerer RISC-V ISA'en, er komplet og helt fri for gabs.

Dette er et aspekt af allerstørste betydning, når detektering af trojanske heste eller udokumenteret logik er en vigtig målsætning.

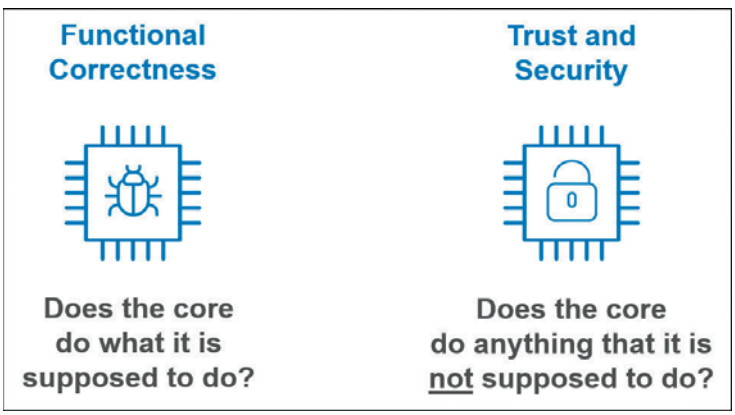
Denne løsning tillader, at SoC-integratorer med begrænset ekspertise i RISC-V eller den RTL-implementering, der er under nærmere granskning, kan overbevises om kvaliteten og troværdigheden af den IP-byggeblok, de ønsker at integrere.

IP-udviklerne kan benytte OneSpin's verifikationsløsning til at detektere sikkerhedssvagheder og funktionelle bugs, før IP'en frigives.

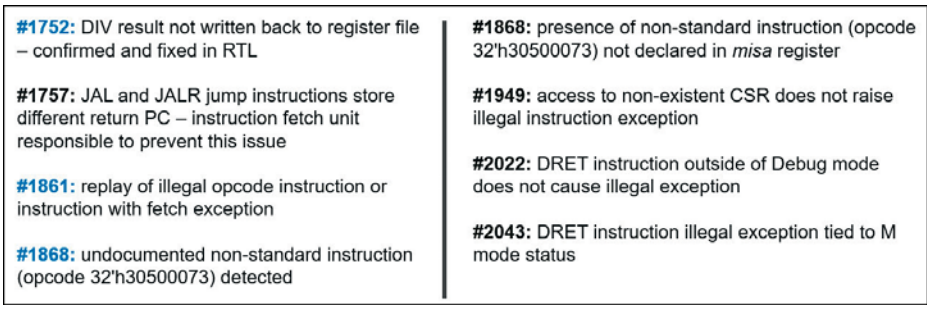
Fungerer det?

RISC-V integritetsforsikrings-processen er blevet succesfuldt implementeret i multiple RTL-designs – herunder RocketCore, der er en open-source, silicium-afprøvet 64-bit RISC-V kerne med et 39-bit virtuelt hukommelsessystem. RocketCore er forsynet med en femtrins, single-issue, 'in-order' pipeline med 'out-of-the-order' færdiggørelse af instruktioner med lange ventetider såsom divisioner. Den understøtter også avancerede features som branch-forudsigelse samt instruktions-gentagelse.

OneSpin's RISC-V Integrity Verification Solution har været anvendt til verifikation af designet inklusiv alle instruktioner,



Figur 2. Verifikation for funktionel korrekt operation giver sikkerhed for, at en processor-implementering opfører sig som specificeret og opfylder kravene fra slutbrugeren. Trust og sikkerhedsverifikation leverer på den anden side troværdighed med hensyn til, at processoren ikke har nogen udokumenterede funktioner, uforudsete sidekanaler, hardwarebaserede trojanske heste eller andre sårbarheder, som kan udnyttes af ondsindede aktører.



Figur 3. Listen af hændelser, der er detekteret af OneSpin's 'RISC-V Integrity Verification Solutions' og rapporteret til GitHub RocketCore projektet.

privilegie-niveauer, interrupts og udtagelses-mekanismer, og i den forbindelse er otte hændelser (issues) blevet detekteret (figur 3).

Her følger yderligere informationer om tre af disse hændelser:

- *Division 'corner-case':* En deep corner-case, der er forbundet med 'out-of-order' færdiggørelse af divisionsinstruktionen. Denne hændelse kan foranledige, at et softwareprogram, der benytter divisions-operationen, beregner et resultat, hvilket i sagens natur kan medføre, at systemet ikke fungerer som tilsigtet. En sådan hændelse vil dog kun forekomme ved en kombination eller sjældne omstændigheder, hvilket er grunden til, at den ikke er blevet opdaget i forbindelse med tidligere verifikationer.

- *Gentagelse af illegal instruktion:* Dette er ikke en 'corner-case' bug. Gentagelse af en illegal instruktion kan give anledning til tabte processeringscycles, men hvis det kun sker i sjældne tilfælde, får det ingen betydning i praksis. Men der er også andre aspekter at tage højde for. Instruktions-gentagelser kan medføre unødvendige hukommelses-forespørgsler. Disse forespørgsler kan have sideeffekter, som kan udnyttes i forbindelse med sidekanals-angreb. Derfor skal en sådan opførsel enten helt elimineres, eller også skal den være klart forstået og fuldt dokumenteret.

- *Udokumenteret instruktion:* En udokumenteret, ikke-standard instruktion kaldet CEASE, der stopper processorkernen, blev detekteret, hvilket ultimativt betyder, at RISC-V RocketCore kan gøre noget, som den ikke burde. Udokumenterede, gemte funktioner er ikke acceptable, når trust og sikkerhed er et tema. Det gælder også selv om, de relaterer til brugercases, der betragtes som ikke-relevante for slutapplikationen.

RocketCore casestudiet er præsenteret i detaljer i et GOMACTech 2019 paper med titlen 'Complete Formal Verification of RISC-V Processor IPs for Trojan-Free Trusted Ics', som kan læses på onespin.com/resources/white-papers.

Hvad bliver det næste?

Den RISC-V sikringsproces, der er præsenteret her i artiklen, detekterer scenarier, der har betydning for sikkerheden, og på systematisk vis afslører udokumenterede funktioner og hardwaremæssige trojanske heste, der har betydning for processorens opførsel, uanset hvor ofte de forekommer og hvor maskerede, de måtte være.

Men sidekanalerne er ikke systematisk detekterede. Fuld udtømmende detektering af alle sidekanaler kræver en dedikeret løsning med brug af passende teknologi.

Der findes allerede prototyper, der adresserer denne udfordring. Flere informationer findes på onespin.com/resources/technical-articles, hvor der findes en artikel om sidekanals angreb på embedded processorer.

Det er også vigtigt at huske på, at en typisk SoC jo også inkluderer mange andre IP-byggeblokke, som også kan huse hardware-baserede trojanske heste. I modsætning til RISC-V processorkerner er det ikke en selvfølge, at der findes uafhængige sikkerheds-løsninger.

Her ville det være værdifuldt, hvis der fandtes en forholdsvis automatiseret trust-vurderingsproces, som kan benyttes til en vilkårlig IP-byggeblok. En proces, der ikke inkluderer en trusted model af IP-byggeblokken, kan ikke garantere, at der ikke findes en trojansk hest.

Men det er dog muligt at identificere unormale og mistænkelige kodemønstre og kendte signaturer for tilstedeværelsen af trojanske heste samt andre svagheder, som ville kunne give problemer på et senere tidspunkt.