ELEKTRONIK
TIDNINGEN

Pankaj Rohatgi
säkerhetsexpert
Cryptography Research

Strömvariation avslöjar koden

Kryptokoder kan knäckas med amperemätare om FPGA:n inte är säkrad mot strömförbrukningsanalys.
Pankaj Rohatgi visar hur. Och hur du skyddar dig,

Redaktör
Jan Tångring
jan@etn.se
0734-17 13 09

EMBEDDED EXPERT

Strömvariation avslöjar koden

Kryptokoder kan knäckas med amperemätare om FPGA:n inte är säkrad mot strömförbrukningsanalys

Av Pankaj Rohatgi, Cryptography Research

Pankaj Rohatgi är expert på sidoattacker och andra aspekter på systemsäkerhet. Han kom till CRI efter en trettonårig karriär som säkerhetsforskare och chef på IBM där han bland annat deltog i utvecklingen av kryptoprocessorn IBM 4758 och ledde olika säkerhetsprojekt med kommersiella och statliga kunder.

Energin som förbrukas i en FPGA beror på omkopplingsaktiviteten i dess transistorer, vilken i sin tur beror på vilken typ av operationer som genomförs. Mätningar av förändringar i strömförbrukning eller elektromagnetiska fält under körning ger angriparen information om de data som bearbetas.

Vi ska här visa hur två metoder kallade SPA och DPA (simple respektive differential power analysis) kan användas för att avslöja kryptonycklar.

De implementeringar som har störst sårbarhet för SPA är de där strömförbrukningen varierar starkt med vilka bitar som finns i nyckeln. I många implementeringar av modulär exponentiering för RSA och Diffie-Hellman förekommer nyckelberoende sekvenser av kvadreringar och multipliceringar. Och i elliptiska kurvkryptosystem (ECC) beräknas skalärprodukter ofta i en nyckelberoende sekvens av fördubblingar och additioner. Mönstren i operationerna kan avslöja den hemliga nyckeln i en enda enkel mätserie.

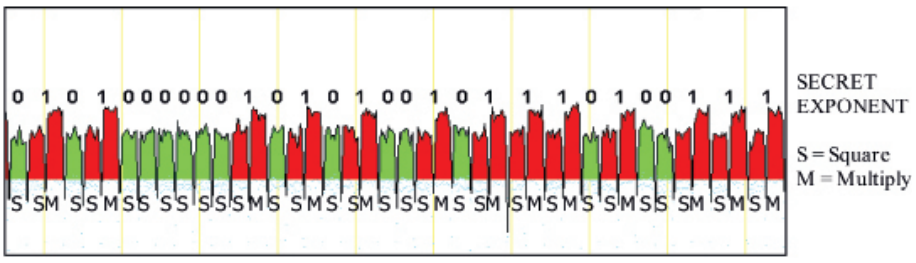
Figur 1 visar strömkurvan under en RSA-operation som använder en typisk sekvens av kvadreringar- och multipliceringar. De två operationerna har olika strömprofiler. Den hemliga exponenten återskapas från den observerade sekvensen av operationer. En etta motsvarar en kvadrering (låg ström) följd av en multiplicering (hög ström), medan en nolla endast inbegriper en kvadrering. I Figur 1 är kvadrerings- och multiplikationssteg gröna respektive röda.

En DPA-attack (differentiell effektanalys) kombinerar separata strömmätningar med statistiska metoder enligt antagandet att den totala strömförbrukningen korrelerar med beräkningarna. Man tar till DPA när läckaget från en enskild kryptotransaktion är litet, brusigt och maskerat av andra aktiviteter. Genom att fokusera på mellanliggande värden som beror på ett fåtal bitar i nyckeln är det möjligt att bestämma just dessa bitar. För varje möjligt värde på nyckelbitarna förutspår man vilka mellanvärden som kommer att uppträda och letar sedan efter korrelationer mellan mätningar och förutspådda bitar i mellanliggande värden.

Som visas i **Figur 2**, sammanfaller korrelationstopparna med att förutspådda mellanliggande värden för givna nyckelbitar bearbetas. För andra (felaktiga) värden på nyckelbitarna, saknas korrelationstoppar, eller så är de mycket lägre. När dessa nyckelbitar är fastställda kan samma söndra-och-härska-princip upprepas för nyckelbitar i andra mellanliggande värden.

Antalet mätningar som krävs för en framgångsrik DPA-attack mot en viss nyckel beror på signal-brus-egenskaperna i det angripna systemet. Processen kan automatiseras och även attacker som omfattar miljoner operationer är relativt enkla att utföra med hjälp av ett vanligt digitalt minnesoscilloskop och en PC.

Det har publicerats många vetenskapliga rapporter om FPGA:ers sårbarhet för effektanalys. En av de allra första handlade om SPA-attacker på elliptisk kurvkryptografi i en FPGA. Andra har avhandlat attacker på AES och DES på FPGA:er. Detaljer skiljer sig, men de grundläggande principerna för SPA och DPA är desamma för mjukvara, FPGA:er och asicar. Flexibiliteten och den låga kostnaden gör FPGA:er till lämpliga plattformar för forskning kring effekt-



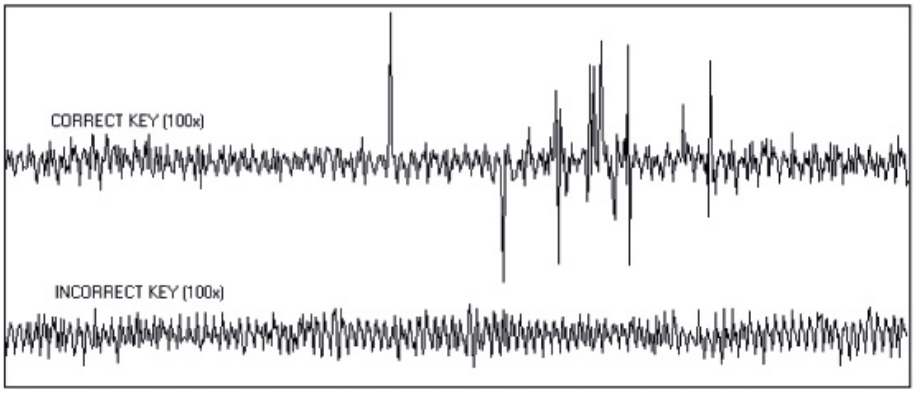
Figur 1: Strömkurva av ett segment av en RSA-exponentieringsoperation. Sådana är sårbara för enkel strömförbrukningsanalys (SFA). Sekvenserna visas tillsammans med bitar från den återskapade hemliga exponenten.

analys, både för att analysera sårbarhet och motåtgärder. Många forskningsgrupper använder det FPGA-baserade utvecklingskortet Sasebo (Side-channel Attack Standard Evaluation Board) som utvecklats av AIST, Japans nationella institut för avancerad industriell vetenskap och teknik.

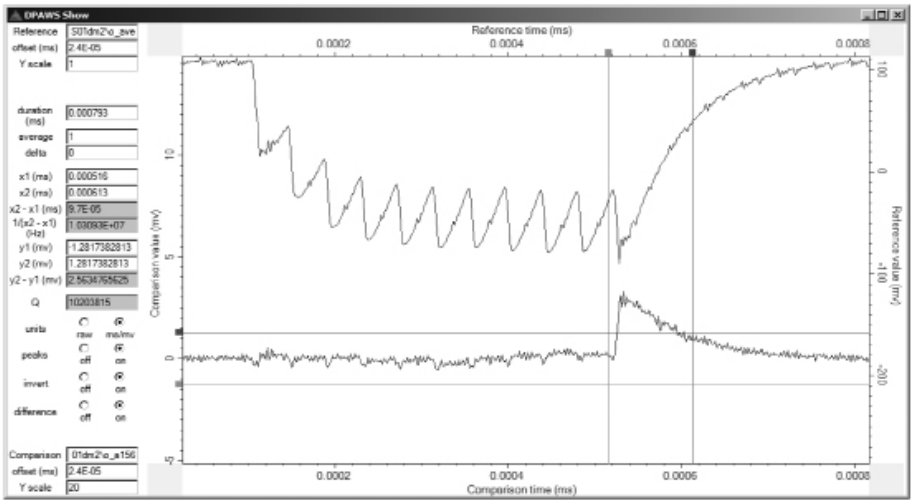
Sårbarheten för SPA och DPA i FPGA:er finns både på plattformsnivå och i den programmerade logiken.

Dagens FPGA-plattformar kan innehålla hårda IP-block för räkneoperationer, bitströmsdekryptering, lösenordsskyddad konfigurering, skyddade segment och nyckelskåp, cpu-kärnor, med mera. Dessa komponenter implementeras vanligen i standardiserade cellblock snarare än i den programmerbara logiken. Utan effektiva motåtgärder kan dessa komponenter kompromitteras via SPA och DPA.

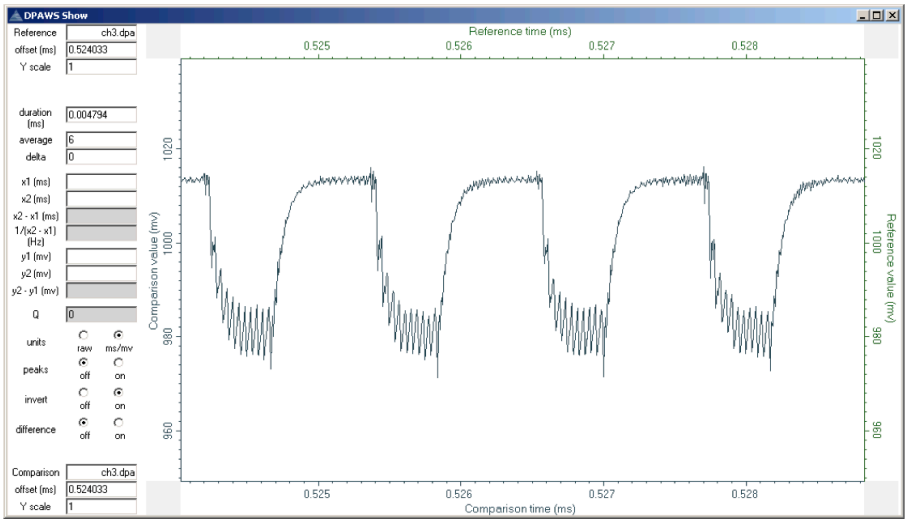
Mängder av publicerade rapporter demonstrerar effektanalysattacker mot krypteringsalgoritmer på FPGA:er. Rapporterna visar att implementationer i logiknätet — när motåtgärder saknas — är mycket sårbara för DPA-attacker och i vissa fall även för SPA-attacker.



Figur 2: DPA: Korrelationen mellan strömkurvor för ett förutspått mellanvärde och en korrekt (upptill) respektive felaktig (nedtill) gissning.



Figur 3: En DPA-attack på Sasebos AES-implementering. Den övre kurvan är den genomsnittliga AES-kurvan, och den nedre kurvan visar korrelationen mellan strömkurvor med ett förutspått mellanliggande värde i omgång 10 för en korrekt gissning av en byte i nyckeln.



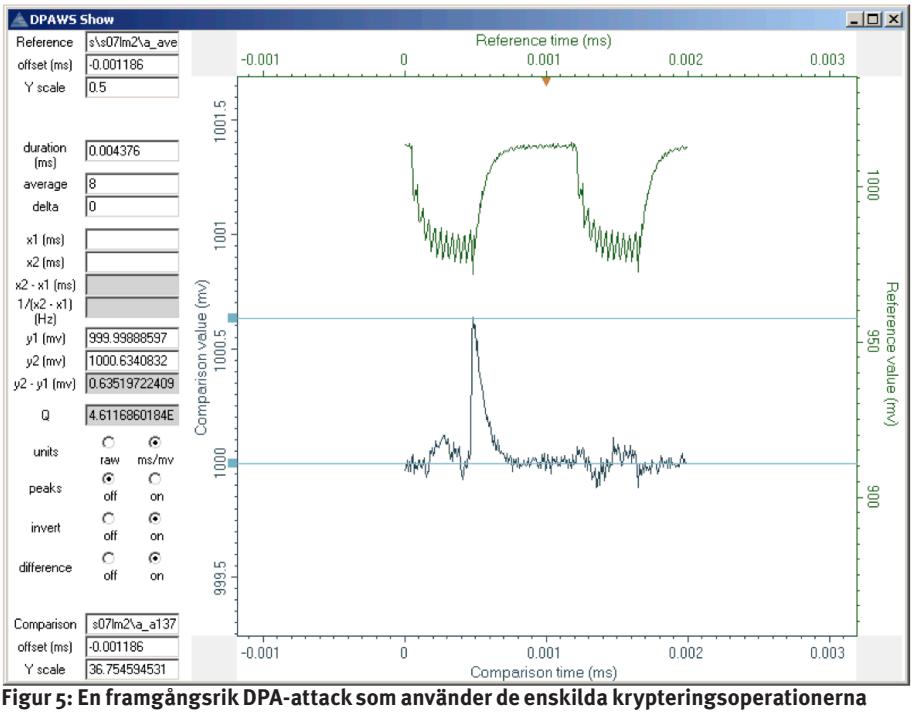
Figur 4: Sasebo-GII AES-implementering som körs i CBC-läge samtidigt som ett 256 kb stort meddelande krypteras. Fyra block är avbildade och den fullständiga kurvan innehåller mätningar från 16384 stycken AES-blockkryptooperationer.

Figur 3 visar en DPA-attack på en testimplementering av AES-128 som inkluderas med plattformen Sasebo-GII. Implementeringen är rättfram och enkel med en iteration per klockcykel och en klockfrekvens på 24 MHz. Den övre kurvan visar genomsnittlig strömförbrukning över 10 000 krypteringar, mätt över ett 1 Ω-motstånd på VCC-sidan. De elva dipparna motsvarar de elva klockcykler det tar att göra AES-operationen (implementeringen inleds med en XOR-operation mellan nyckel och data, följd av 10 iterationer). Den nedre kurvan visar korrelationen av strömkurvor med ett förutspått mellanliggande värde i början av omgång 10, för en korrekt gissning av en byte i nyckeln. Den skarpt stigande kanten i korrelationskurvan i början av omgång 10 bekräftar att gissningen är korrekt.

DPA-attacken ovan utförs genom att man externt anropar AES-blockkrypteringsoperationen cirka 10 000 gånger med en okänd nyckel och slumpmässiga data. Därmed registreras mindre än 5 ms beräkningstid. En minuts bearbetningstid på en PC är därefter allt som krävs för att avslöja en nyckel på 16 byte. Verktyget som används är analysmjukvaran Cryptography Research DPA Workstation.

Angriparen kan samla in betydligt större datamängder för sin attack från FPGA:er än från smartkort och annan bandbreddsbegränsad hårdvara.

Figur 4 visar en del av en obearbetad strömkurva från en AES-implementering på Sasebo-GII som utför krypteringar i bulk i CBC-läge. En ensam strömkurva under bulkkrypteringen representerar åtminstone tiotusentals individuella blockoperationer och kan överföras till en PC på bara några sekunder. **Figur 5** visar resultatet av en framgångsrik DPA-attack som bara utnyttjade en enda strömkurva. Först bröts kurvan ner i



Figur 5: En framgångsrik DPA-attack som använder de enskilda krypteringsoperationerna från kurvan i Figur 4. Den genomsnittliga kurvan visas upptill och korrelationskurvan för den korrekta gissningen (137) för byte 7 av nyckeln i sista omgången visas nedtill.

enskilda blockoperationer och sedan användes DPA för att analysera operationerna.

Det var Cryptography Research som upptäckte SPA och DPA i mitten av 1990-talet. Vi har också de grundläggande patenten på motåtgärder, bland annat:

Minskat läckage: Gör sekvensen av operationer mindre beroende av nyckel och mellanvärden. Balansering kan reducera variationen i strömförbrukning. Metoden kräver extra försiktighet på grund av asymmetrier i routinginfrastrukturen i FPGA:er. Det övergripande målet är att reducera läckagets signal-brusförhållande, för att på sätt öka antalet strömmätningar som en angripare behöver göra.

Brusgenerering: Ett annat sätt att öka antalet strömkurvor angriparen behöver mäta, är att addera brus där angriparen mäter effekten. Brus kan genereras i amplituddomänen genom att man exempelvis förbrukar slumpmässigt vald strömstyrka, eller i tidsdomänen genom att man randomiserar timingen för operationer.

Hemlighetsmakeri: Om algoritmerna hålls hemliga måste angriparen göra både reverse engineering och effektanalys. Metoden ger ingen säkerhet när angriparen väl avslöjat den dolda funktionen, men det höjer initialkostnaden. Att kostnaden för påföljande attacker inte ökar bör hållas i åtanke, men metoden är bättre än inget skydd alls.

Slump: Randomisera data på sätt som inte förändrar utdata. Vad gäller publika nyckelsystem kan det fungera bra att maskera eller gömma data och nycklar. För symmetriska algoritmer som AES är det främst maskering av mellanvärden och tabeller som gäller. Tekniken tvingar angriparen använda mer komplexa attacker, som DPA:er av högre ordning, som kräver fler strömmätningar.

Motåtgärder på protokollnivå: Använd protokoll som upprätthåller säkerheten även vid visst informationsläckage. Uppdatera kontinuerligt hemliga värden så angriparen aldrig får tillräckligt med information för att räkna ut värden. Metoden fungerar för både online-tilllämpningar som serverautentisering via utmaning-respons och offline-tillämpningar som laddning av firmware. Den kan dessutom hantera både interaktion med betrodda servrar och P2P-protokoll. Metoden kan inte användas med äldre protokoll som saknar integrerade skydd på protokollnivå, men utvecklare med flexibilitet i protokollen kan med denna metod nå den allra högsta säkerhetsnivån sidoattacker mot strömförbrukningen.

Eftersom DPA-attacker använder signalbehandling för att förstärka läckt information är det i allmänhet en fördel att använda flera samtidiga motåtgärder. Utvecklarna måste överväga vilka metoder som bör användas, givet säkerhetskrav och tekniska begränsningar. FPGA:ernas flexibilitet gör det möjligt att iterativt förfina och testa implementeringar tills önskad nivå på DPA-resistens uppnås. ■

Källor

[1] Paul Kocher, Joshua Jaffe, Benjamin Jun
”Differential Power Analysis”
Advances in Cryptology – Crypto 99 Proceedings, Lecture Notes In Computer Science Vol. 1666
M. Wiener (red.), Springer-Verlag, 1999, s. 388–397.
En rapport finns på http://www.cryptography.com/resources/whitepapers/DPATechInfo.pdf

[2] Sıddıka Berna Ors, Elisabeth Oswald och Bart Preneel
”Power-Analysis Attacks on an FPGA – First Experimental Results”
Cryptographic Hardware and Embedded Systems – CHES 2003, Proceedings, Lecture Notes in Computer Science, Vol 2779
Colin D. Walter, Çetin Kaya Koç, Christof Paar (red.), Springer 2003, s. 35–50.

[3] Francois-Xavier Standaert, Sıddıka Berna Ors, Bart Preneel
”Power Analysis of an FPGA -Implementation of Rijndael: Is Pipelining a DPA Countermeasure”
Cryptographic Hardware and Embedded Systems – CHES 2004, Proceedings, Lecture Notes in Computer Science, Vol 3156
Marc Joye, Jean-Jacques Quisquaters, Springer 2004, s. 30–44.

[4] François-Xavier Standaert, Siddika Berna Örs, Jean-Jacques Quisquater, Bart Preneel
”Power Analysis Attacks Against FPGA Implementations of the DES”
Field Programmable Logic and Application, Proceedings – FPL 2004, Proceedings, Lecture Notes in Computer Science, Vol 3203
Jürgen Becker, Marco Platzner, Serge Vernalde (red.): Springer 2004, s. 84–94

[5] Side-channel Attack Standard Evaluation Board (SASEBO)
http://www.rcis.aist.go.jp/special/SASEBO/index-en.html

[6] Dylan McGrath
”Gartner: ASIC design starts to fall by 22% in ’09”, EE Times, http://www.eetimes.com/news/semi/showArticle.jhtml?articleID=216401584

[7] Jessica Davis
”FPGAs storm military spending”, EDN (Electronic Design, Strategy, News), http://www.edn.com/article/459326-FPGAs_storm_military_spending.php

[8] Steve Trimberger
Trusted design in FPGAs
ur *Proceedings of the 44th Annual Design Automation Conference* (DAC ’07)
ACM, New York, NY, s. 5–8.