DEFINIÇÃO DE UM FLUXO DE PRÉ-PROCESSAMENTO PARA ATACAR POR DPA ARQUITETURAS CRIPTOGRÁFICAS GALS PIPELINE

Luciano Ludwig Loder[1]; Adão Antônio de Souza Junior[2]; Rafael Iankowski Soares[3]

[1]*Instituto Federal Sul-Riograndense - IFSUL – lucianoloder@gmail.com*
[2]*Instituto Federal Sul-Riograndense - IFSUL – adaojr@gmail.com*
[3]*Universidade Federal de Pelotas – UFPEL – rafael.soares@inf.ufpel.edu.br*

1. INTRODUÇÃO

Ataques por canais laterais (do inglês, *side-channel* attacks – SCA) consistem em uma poderosa ferramenta de criptoanálise, apresentada à comunidade acadêmica por Kocher et al. em (KOCHER, 1999). Essa classe de ataques visa comprometer a implementação dos circuitos criptográficos, ao invés do algoritmo criptográfico em si, correlacionando grandezas físicas inerentes ao processamento do circuito criptográfico, como tempo de execução, radiação eletromagnética ou consumo de potência, com as informações secretas processadas. As análises que avaliam o consumo de potência do circuito, são conhecidas como análises por consumo diferencial de potência (do inglês, *Differential Power Analysis* - DPA).

DPA avalia o traço resultante do consumo de potência causado pela execução de uma operação intermediária de um algoritmo de criptografia para um determinado dado de entrada a ser encriptado e compara os traços de consumo adquiridos durante a execução de um conjunto de distintos dados de entrada. Como o hardware é projetado segundo o paradigma síncrono, todas as operações são executadas sequencialmente e na mesma ordem, e ainda, com o mesmo tempo para execução. Deste modo, todos os traços de consumo de potência são alinhados no tempo permitindo que o atacante compare o consumo de uma determinado instante de tempo em relação aos demais.

Na literatura encontram-se vários trabalhos para evitar a fuga de informações por consumo de potência, abordagens conhecidas como contramedidas. Uma contramedida popular é randomizar o fluxo de execução do circuito criptográfico. Como o ataque DPA visa comparar a mesma operação em vários traços de consumo de potência distintos, analisando o mesmo instante de tempo entre vários traços, a randomização do fluxo de execução causará um deslocamento da operação no tempo, dificultando a comparação realizada pelos atacantes.

Em (CLAVIER, 2000), essa contramedida é utilizada em nível de software, enquanto que em (LU, 2008) esta contramedida é aplicada em hardware. Em (AVIRNENI, 2013) é avaliada a variação da frequência do sinal de relógio do circuito como forma de aleatorizar o fluxo de execução do algoritmo. Em (SOARES, 2011) é proposta a utilização de arquiteturas pipeline projetadas segundo o estilo globalmente assíncrono localmente síncrono (do inglês, *globally asynchronous, locally synchronous - GALS*) para evitar ataques DPA através da randomização do tempo de execução do algoritmo e aumento do ruído provocado pelo processamento paralelo do pipeline. Tal arquitetura criptográfica é composta de ilhas síncronas comunicando-se assincronamente, visando inserir atrasos aleatórios e variação da frequência do sinal do relógio enquanto aumenta a vazão de dados. Tal arquitetura mostra-se robusta a ataques DPA convencionais.

Porém na literatura encontram-se trabalhos visando contornar tais contramedidas, visando comprometer circuitos criptográficos dotados de contramedidas. Duas ferramentas para realinhar os traços de consumo de potência são a correlação de fase (do inglês, Phase-Only Correlation - POC) proposta por Nagashima et al. (NAGASHIMA, 2007) e deformação temporal dinâmica (do inglês, Dynamic Time Warping - DTW) (WOUDENBERG, 2009). O POC visa realinhar os traços utilizando a Correlação de fase, porém não é efetivo para alinhar traços obtidos pelo processamento em diferentes frequências de relógio, visto que a correlação entre traços de frequências distintas é extremamente baixa.

O presente trabalho visa propor os primeiros passos para definição de um fluxo de etapas de pré-processamento que permitem a um atacante encontrar vulnerabilidades no desalinhamento dos traços provocado pelas arquiteturas GALS pipeline. Além disso, é discutido o compromisso entre algumas técnicas de processamento revisadas e o esforço computacional necessário para sua aplicação.

2. MATERIAL E MÉTODOS

O presente trabalho foi executado utilizando scripts em MATLAB sobre traços de consumo de potência previamente adquiridos. Os traços foram obtidos pela medição do consumo de potência das arquiteturas GALS pipeline compostas por duas ilhas síncronas implementando o algoritmo criptográfico DES e prototipadas no dispositivo FPGA Xilinx Spartan3. O sinal de relógio de cada ilha é selecionada randomicamente dentre quatro sinais com frequências distintas. Foram realizados experimentos utilizando a arquitetura GALS com pipeline vazio, e foram adquiridos 100 mil traços de consumo de potência por cada configuração, a uma taxa de amostragem de 20G amostras/segundo.

O fluxo de execução de etapas de pré-processamento proposto é composto pelas seguintes etapas: (*i*) extração do segmento de consumo alvo dos ataques, (*ii*) agrupamento de traços, (*iii*) filtro de ruído, (*iv*) alinhamento de traços e por fim o (*v*)ataque DPA propriamente dito.

Extração de segmentos: a primeira etapa é pesquisar nos traços o segmento de consumo de potência correspondente a ilha síncrona que contém a execução da função intermediária alvo do ataque DPA. Como as arquiteturas GALS pipeline podem ter 2 ou mais ilhas síncronas, o atacante deve observar o número de ilhas sob o traço para extrair a desejada. Além disso, deve-se distinguir computação e o ruído a fim de estabelecer um limiar capaz de delimitar e extrair o segmento desejado.

Agrupamento de traços: depois de extrair o segmento com o consumo de ilha síncrona desejada é necessário avaliar a frequência de relógio de cada traço através de uma transformada rápida de Fourier (do inglês, Fast Fourier Transform - FFT). Com esta informação, cada traço é classificado e agrupado de acordo com a frequência de operação. Um histograma é gerado para relacionar o número de traços e a frequência de operação. Deste modo é possível aplicar as técnicas clássicas encontradas na literatura para alinhamento de traços POC e DTW.

Filtro: esta etapa visa eliminar uma quantidade de ruído para aumentar a efetividade dos ataques DPA. Nesta etapa são utilizadas duas técnicas comuns de filtragem são utilizadas, o filtro de médias móveis e o filtro em frequência. A desvantagem de filtro de médias móveis é o tempo de processamento elevado em relação a filtro em frequência.

Alinhamento de traços: duas técnicas são usada com o propósito de alinhar traços no domínio do tempo: POC e DTW. Ambas mostram-se eficientes para alinhar os traços, porém possuem custos computacionais diferentes. POC exige menos tempo de computação ao custo de uso da etapa de agrupamento de traços. Por outro lado DTW, exige um grande esforço computacional, com a vantagem de conseguir alinhar traços de diferentes frequências eliminando a necessidade da etapa de agrupamento de traços. Segundo Real et al. (REAL, 2008), mesmo alinhado no tempo é necessário alinhar na amplitude, o que faz com que a divisão de frequência seja utilizada neste caso

.

Ataque DPA: depois de alinhados os traços é realizado o ataque DPA na tentativa de correlacionar os informações com o consumo de potência.

3. RESULTADOS E DISCUSSÃO

A Tabela 1 apresenta os resultados obtidos. O conjunto de traços foi dividido em 2 grupos, de acordo com a frequência de relógio de cada grupo. O grupo 1 é composto de 53.357 traços com frequências de relógio entre 38 e 42 MHz e o grupo 2 composto de 46.643 traços de frequências de relógio entre 55 e 60 MHz. O ataque DPA tem como alvo a subchave processada pelos módulos SBOXs do algoritmo DES que computa parte da chave criptográfica secreta informada pelo usuário, conforme (SOARES, 2011). A métrica comumente utilizada para determinar o sucesso de um ataque DPA é o ranking da subchave correta onde a posição 1 indica que a sub-chave foi corretamente identificada. Além disso, o número mínimo de traços necessários para a correta determinação da subchave é apresentado.

Tabela 1. Resultado dos ataques realizados com o fluxo proposto.

		Nenhum pré-processamento		POC		POC + Filter	
Função	Grupo	Nº Traços	Ranking	Nº Traços	Ranking	Nº Traços	Ranking
SBOX1	1	-	40	06943	01	01290	01
	2	-	29	02513	01	01382	01
SBOX2	1	-	21	22130	01	12932	01
	2	-	28	10468	01	02032	01
SBOX3	1	-	31	29752	01	18836	01
	2	-	15	03798	01	01077	01
SBOX4	1	-	08	12211	01	04790	01
	2	-	32	03510	01	02562	01
SBOX5	1	-	61	27238	01	21516	01
	2	-	12	-----	02	16023	01
SBOX6	1	-	25	15987	01	15987	01
	2	-	37	09202	01	02294	01
SBOX7	1	-	32	33511	01	13886	01
	2	-	38	02187	01	02097	01
SBOX8	1	-	05	-----	56	-----	22
	2	-	22	-----	02	13777	01

Nota-se que sem nenhum tipo de pré-processamento (coluna WPP) o ataque DPA não obtém êxito, enquanto que ao utilizar POC (coluna POC) o ataque DPA é eficaz em determinar quase todas as subchaves corretamente. Já a utilização

de filtros (coluna POC+Filter) mostra que é possível reduzir o número de traços necessários para a determinação correta da subchave.

4. CONCLUSÕES

O presente trabalho investigou a robustez de arquiteturas criptográficas GALS pipeline a ataques DPA dotados de etapas de pré-processamento. Notou-se que tais arquiteturas são possíveis de serem comprometidas utilizando realinhamento por correlação de fase e filtragem dos traços. Trabalhos futuros consistem em avaliar arquiteturas GALS com pipeline cheio e com 4 ilhas síncronas processando simultaneamente com pipeline cheio e vazio. Espera-se que tais arquiteturas sejam mais difíceis de ser comprometidas, visto que cada ilha processa menos informação, reduzindo as informações que um criptoanalista dispõe para realizar um ataque bem-sucedido.

5. REFERÊNCIAS BIBLIOGRÁFICAS

KOCHER, P, JAFFE, J, JUN, B. **Differential power analysis.** Springer-Verlag, 1999, pp. 388–397.

CLAVIER, C. J. CORON, S. Dabbous, N. **Differential power analysis in the presence of hardware countermeasures**, in CHES, ser. Lecture Notes in Computer Science, C
etin Kaya Koc
and C. Paar, Eds., vol. 1965. Springer, 2000, pp. 252–263.

LU, Y. O'Neill M. McCanny J. **FPGA Implementation and Analysis of Random Delay Insertion Countermeasure against DPA**. 2008

AVIRNENI N. D. P. SOMANI A. K., **Countering power analysis attacks using reliable and aggressive designs**, IEEE Transactions on Computers,
vol. 99, no. PrePrints, p. 1, 2013.

SOARES R. CALAZANS N. MORAES F. MAURINE P. ,TORRES L. **A robust architectural approach for cryptographic algorithms using gals pipelines**, Design Test of Computers, IEEE, vol. 28, no. 5, pp. 62 –71, sept.-oct. 2011.

NAGASHIMA S. HOMMA N. IMAI Y. AOKI T. SATOH, A. **DPA using phase-based waveform matching against random-delay countermeasure**, in ISCAS. IEEE, 2007, pp. 1807–1810.

WOUDENBERG J. WITTEMAN, M. BAKKER B. **Improving differential power analysis by elastic aligment** in CT-RSA. Springer, 2009, pp. 104–119.

REAL D. , CANOVAS C. , CLEDIERE J., DRISSI M. VALETTE F. , **Defeating classical hardware countermeasures: a new processing for side channel analysis**, in Design, Automation and Test in Europe, 2008. DATE ’08,
2008, pp. 1274–1279.