Avaliando a encriptação especulativa em CPUs multicore

Vinícius C. Oliveira de Andrade[1], **Wagner M. Nunan Zola** [1]

[1]Departamento de Informática – Universidade Federal do Paraná (UFPR)
Curitiba – PR – Brasil

{vcoandrade, wagner}@inf.ufpr.br

Resumo. *Arquiteturas modernas possuem instruções especiais AES-NI, possibilitando encriptação acelerada em hardware. Neste trabalho apresentamos os testes de implementação da biblioteca WAESlib para CPUs multicore. Observamos a viabilidade de utilização de criptografia especulativa usando aceleração AES-NI e múltiplas threads.*

1. Introdução e fundamentos teóricos

A encriptação de dados é requisito necessário para o funcionamento de diversos sistemas utilizados atualmente. Explorando características de alguns modos de operação de algoritmos de criptografia é possível conseguir melhorias no desempenho de aplicações clientes [Eduardo et al. 2019]. Neste trabalho apresentamos a implementação e testes preliminares da biblioteca WAESlib em CPU, que explora a criptografia especulativa com múltiplas *threads*, com possibilidade de geração paralela antecipada de máscaras de criptografia no modo AES CTR (*Advanced Encryption Standard, Counter mode*).

Algoritmos de cifragem simétrica se utilizam de modos de operação. Durante muito tempo foi utilizado o modo CBC (*Cipher Block Chaining* ou criptografia de blocos encadeada) como padrão, porém o modo CTR, sigla para contador, possui o mesmo nível de segurança [Rogaway 2011] além de várias vantagens adicionais como a possibilidade de uso de paralelismo. Uma dessas vantagens é a capacidade de pré-processamento da criptografia antes de se saber o texto em claro, como apresentado em [Rogaway 2011] e explorado em [Nunan Zola and De Bona 2012] e [Eduardo et al. 2019]. A encriptação especulativa se baseia em estimar quais os próximos blocos a serem encriptados e realizar o pré-processamento dos mesmos.

2. Resultados e discussão

Foi implementada uma versão da WAESlib para o processamento de criptografia especulativa em CPU. Embora possua a desvantagem de uma menor capacidade de processamento quando comparado com GPUs a versão em CPU compensa com sua versatilidade, podendo ser utilizada em sistemas que não possuem placas de vídeo dedicadas para este fim. Entretanto CPUs modernas possuem instruções para aceleração de criptografia, como AES-NI (Advanced Encryption Standard New Instructions). Assim como a versão apresentada por [Nunan Zola and De Bona 2012] a versão em CPU é capaz de realizar encriptações especulativas utilizando o algoritmo AES nos modos CTR de 128 e 256 bits e é segura para utilização com *threads*, podendo receber requisições de várias fontes diferentes simultaneamente.

Serão apresentado os valores referentes aos testes preliminares de funcionamento da biblioteca. Para os testes foram realizadas simulações, em RAM, de requisição para

encriptação de blocos de um arquivo. Cada bloco requisitado possui 4KiB, mesmo tamanho dos contextos pré-processados, e em cada ciclo de testes é encriptado 1 GiB. Embora os testes sejam feitos em memória são adicionados delays de modo a simular possíveis pequenas demoras quando a requisição dos dados é feita em disco rápido (tipo SSD M.2).

Na Figura 1, à esquerda, é apresentada a comparação da vazão para testes com e sem o uso de especulação e, à direita, é apresentado o resultado preliminar dos tempos para as funções WAES_ctx, responsável por agendar o pré-processamento dos contextos, e da função WAES_encrypt, responsável por encriptar um bloco do arquivo, além de mostrar a vazão total alcançada pelo processo. Os testes foram realizados em um computador com um Intel i7-10700, de 8 núcleos e 16 *threads*, com 2.90GHz e 16GB de RAM.

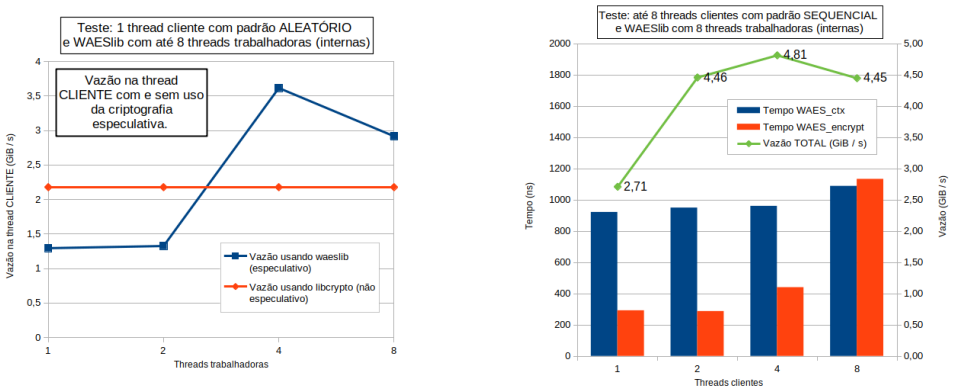

Figura 1. Resultados preliminares para a biblioteca WAESlib em CPU.

Verificamos que a função de agendamento do pré-processamento possui tempo praticamente constante devido à sua simplicidade, enquanto a função de encriptação tem um aumento no tempo conforme o número de requisições aumenta, devido à maior carga de trabalho, o que atrasa o término do processamento do contexto. Também é possível ver o valor relativo à vazão total do processo, que possui máximo com 4 clientes uma vez que com menos existe pouca carga de trabalho e com mais o processador fica sobrecarregado (caso com 8 *threads* internas e 8 clientes, sendo um processador de 8 núcleos).

3. Conclusão

Verificamos o comportamento esperado das funções e da vazão para a encriptação, baixa latência de operações de cifragem, mantendo alta vazão. São necessárias revisões de modo a tentar melhorar o desempenho porém os resultados mostram a viabilidade da utilização de criptografia especulativa em CPU. Como trabalho futuro a versão em CPU da WAESlib será primeiramente utilizada na implementação de sistemas de arquivos criptografados, sendo incorporada ao sistema de arquivos EncFS++ [Eduardo et al. 2019].

Referências

Eduardo, V., De Bona, L. C. E., and Nunan Zola, W. M. (2019). Speculative encryption on GPU applied to cryptographic file systems. In *17th USENIX Conference on File and Storage Technologies (FAST 19)*, pages 93–105, Boston, MA. USENIX Association.

Nunan Zola, W. M. and De Bona, L. C. E. (2012). Parallel speculative encryption of multiple AES contexts on GPUs. In *2012 Innovative Parallel Computing*, pages 1–9.

Rogaway, P. (2011). Evaluation of some blockcipher modes of operation. Technical report, Dept. of Computer Science - University of California, Davis, California, USA.