Uma proposta para sistemas de arquivos criptográficos em *Kernel Space*

Jorge Pires Correia[1], **Wagner M. Nunan Zola**[1]

[1]Universidade Federal do Paraná (UFPR)
Curitiba/PR

{jpcorreia,wagner}@inf.ufpr.br

Resumo. *Este artigo apresenta uma proposta de utilização de encriptação especulativa através de* hooks *em sistemas de arquivos em* kernel-space*, o que permite aproveitar o tempo das operações de entrada e saída para realizar operações criptográficas, mantendo a quantidade de troca de contextos e dispondo da flexibilidade na arquitetura do sistema.*

1. Introdução

Sistemas de arquivos criptográficos (SAC) formam uma das principais camadas de software no processo de armazenamento de dados privados. Contudo, a latência das operações nos SACs é aumentada devido à encriptação e decriptação de dados, tornando-os mais lentos que os sistemas de arquivos convencionais. Instruções de criptografia implementadas em hardware [Gueron 2010] permitem diminuição da latência das operações criptográficas, mas ainda exercem grande influência quando aplicadas em SACs.

[Nunan Zola and de Bona 2012] apresentaram uma forma de realizar operações criptográficas de forma especulativa e paralela através da WAESlib, uma biblioteca que implementa o algoritmo AES no modo de operação Counter (CTR) para execução em GPUs. A característica especulativa permite a execução das operações criptográficas antes do dado estar pronto em memória, de modo a esconder a latência destas operações durante o acesso ao dispositivo de armazenamento. A característica paralela permite a encriptação e decriptação independente de cada bloco, além de possibilitar a execução simultânea destas operações em diferentes núcleos de processamento. [Eduardo et al. 2019, de Andrade and Nunan Zola 2022] aplicaram o método citado anteriormente em sistemas de arquivos implementados em *User Space* e comprovaram a sua eficiência. Contudo, implementações em *User Space* sofrem com o aumento das trocas de contextos e com a rigidez na arquitetura, tendo em vista que essas implementações devem ser empilhadas sobre outros sistemas de arquivos existentes.

2. Proposta

O presente artigo propõe a utilização da encriptação especulativa em sistemas de arquivos presentes em *Kernel Space* através de *hooks*. *Hooks* são chamadas de funções que modificam o fluxo normal de execução do sistema. Nesse caso, os *hooks* são as chamadas para as funções que realizam as operações do SAC proposto. Por meio dessa abordagem, as operações de criptografia podem ser inseridas em diferentes níveis dos sistemas de arquivos sem aumentar o número de trocas de contexto ou modificar interfaces.

Quando uma requisição de leitura é realizada, o sistema de arquivos deve buscar o *nonce* referente ao dado em questão e submeter a operação criptográfica que gerará uma

máscara para este *nonce*. Essa máscara será produzida enquanto a operação de entrada e saída (E/S) estiver ocorrendo, sendo aplicada sobre o dado quando estiver finalizada. O processo de aplicação da máscara é somente uma operação XOR. As máscaras de criptografia para as requisições de escrita podem ser construídas antes mesmo das requisições propriamente ditas, pois estas usarão novos *nonces*. A interface inferior da camada do sistema de arquivos se mostra promissora para receber as funções de criptografia, tendo em vista que requisições à camada de E/S de blocos são realizadas independente do caminho de execução dentro do sistema de arquivos. Tal abordagem também permite que operações que acessem somente a Page Cache não sofram qualquer *overhead*, pois as operações criptográficas serão executadas somente quando existir um acesso ao dispositivo de armazenamento.

Além dos pontos em que as operações criptográficas serão executadas, deve-se considerar o gerenciamento de metadados e a forma de utilização da WAESlib. Para isso, operações como abertura, fechamento, criação e remoção de arquivos também devem realizar funções de gerenciamento, a fim de manter a consistência do sistema de arquivos. A forma de utilização da WAESlib não será similar à abordagem dos trabalhos anteriores. A flexibilidade na arquitetura proposta abre uma série de novos cenários de execução, permitindo novas abordagens e, assim, necessitando um novo gerenciamento dos contextos e *threads* trabalhadoras disponibilizadas pela WAESlib.

Para demonstrar a eficiência da abordagem proposta, planeja-se adicionar funções criptográficas no sistema de arquivos padrão do *kernel* Linux, EXT4, e comparar o desempenho com o fscrypt [Kernel Development Community 2023], um SAC implementado em *kernel space* através de *hooks* no EXT4, F2FS e UBIFS, e o eCryptfs [Halcrow 2005], um SAC implementado em *kernel space* através de empilhamento. Ao final deste trabalho, espera-se que o sistema proposto apresente ganhos de desempenhos significativos com relação aos sistemas comparados, tendo em vista a diminuição da latência das operações através da criptografia especulativa.

Referências

de Andrade, V. C. O. and Nunan Zola, W. M. (2022). Aplicabilidade da encriptação especulativa em CPUs multicore para sistema de arquivo no espaço de usuário. In *Computer on the Beach 2022*, pages 282–289, Itajaí - Brasil.

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 - USA.

Gueron, S. (2010). Intel advanced encryption standard (AES) new instructions set.

Halcrow, M. A. (2005). ecryptfs: An enterprise-class encrypted filesystem for Linux. In *2005 Linux Symposium*, pages 201–218, Ottawa - Canada.

Kernel Development Community (2023). Filesystem-level encryption (fscrypt). https://www.kernel.org/doc/html/v6.1/filesystems/fscrypt.html. Acessado em 09/03/2023.

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 (InPar)*, pages 1–9, San Jose - USA.