PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS

FACULDADE DE ENGENHARIA DE COMPUTAÇÃO

Sistemas Operacionais B

Projeto 2

Ananda Rodrigues - 13034707

Marcelino Noguero - 16011538

*Introdução*

O segundo projeto de Sistemas Operacionais B tem como objetivo principal conhecer e se familiarizar com detalhes de implementação em chamadas do sistema em um Kernel, particularmente chamadas que permitam armazenamento e leituras de arquivos de forma cifrada através de uma API criptográfica do próprio Kernel.

Para que o desenvolvimento deste trabalho, é possível a utilização de atividades anteriores, assim como o primeiro projeto.

*Desenvolvimento*

O segundo projeto, como mencionado na introdução, a implementação de duas chamadas de sistema no Kernel: ***write\_crypt*** e ***read\_crypt.***

***write\_crypt:***

A primeira chamada necessária permite que programas dentro do espaço do usuário consigam guardar arquivos de maneira cifrada. Diferente do projeto anterior, dessa vez será utilizado o algoritmo AES em modo ECB para que seja possível cifrar os dados desses arquivos.

A chave simétrica usada na cifra é definida no código fonte da chamada de sistema, o que significa que ela só pode ser modificada. Essa chamada é feita no sistema enviando três informações necessárias:

Um inteiro para descrever o tipo de arquivo obtido através do open, um ponteiro que aponta para um vetor de caracteres que armazena o que deve ser escrito no arquivo e por fim o número de bytes desse vetor.

Consequentemente a chamada é apresentada da forma:

*ssize\_t write\_crypt (int fd, const void \*buf, size\_t nbytes);*

Exemplo de chamada do write\_crypt:

Arquivo cifrado:

Exemplo de chamada de write p/ comparação:

O valor de retorno e os códigos de erro retornados pela chamada implementada seguem o mesmo modelo da chamada *write*, já existente no Linux.

***read\_crypt:***

Por fim, a segunda chamada a ser implementada no projeto permite que os programas em espaço de usuário consigam ler arquivos que já estejam cifrados utilizando a chamada write\_crypt anterior. O algoritmo utilizado é o AES em modo ECB, como mencionado anteriormente.

Assim como a primeira chamada, essa também tem sua chave simétrica definida no código fonte para decifrar os dados. Além disso, a chamada é feita com três variáveis semelhantes a de escrita, como algumas pequenas diferenças:

Um inteiro para descrever o arquivo obtido, um ponteiro que aponta para um vetor de caracteres no qual são armazenados os bytes lidos do arquivo cifrado e por fim o número de bytes que são lidos do arquivo referenciado.

Concluindo, a chamada é feita dessa forma:

*ssize\_t read\_crypt(int fd, void \*buf, size\_t nbytes);*

Exemplo de chamada do read\_crypt:

Arquivo decifrado:

Exemplo de chamada de read p/ comparação:

O valor de retorno e os códigos de erro retornados pela chamada implementada seguem o mesmo modelo da chamada *read*, já existente no Linux.

*Conclusão*

É possível concluir que, ao finalizar o projeto dois de Sistemas Operacionais, houve um entendimento mais aprofundado sobre criação de chamadas no Linux Kernel além de modos de cifragem e decifragem dentro delas. Podemos observar a complexidade dos algoritmos que envolvem esse conceito assim como o funcionamento de uma cifra bem sucedida, em que, obtendo-se apenas a mensagem cifrada, sem a chave de criptografia, seja impraticável descobrir a mensagem original.