Descritivo do trabalho prático de AES usando os modos de operação CBC e CTR.

Felipe Guedes [1]

[1]Escola Politécnica – Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

felipe.guedes@edu.pucrs.br

Abstract. *This paper describes the solution applied by the students on the second assessment for the systems security discipline given by the professor Avelino Zorzo. The main aspect to be explored on this paper is the use of the AES algorithm and its operational modes CTR and CBC.*

Resumo. *Este artigo descreve a solução utilizada pelos alunos no primeiro trabalho da disciplina de segurança de sistemas ministrada pelo professor Avelino Zorzo. O principal aspecto a ser explorado neste artigo é o uso do algoritmo de criptografia AES e seus modos de operação CBC e CTR.*

1. Introdução

Este é o segundo trabalho da disciplina de segurança de sistemas, ministrada pelo professor Avelino Zorzo durante o segundo semestre de 2019. O objetivo do trabalho é a criação de um programa que permita os alunos explorarem o uso do algoritmo AES usando dois modos de operação sendo eles o CBC e o CTR.

2. Descrição do problema

Foi dado aos alunos a tarefa de implementar um programa que fosse possível cifrar e decifrar textos usando o algoritmo AES, junto com os modos de operação CBC ou CTR, conforme especificado.

Abaixo está um exemplo de especificação de uma tarefa, na qual é informado o modo de operação, se o texto é cifrado ou não, e a chave que deve ser usada no algoritmo.

- CBC key: 140b41b22a29beb4061bda66b6747e14
- CBC Ciphertext:
  4ca00ff4c898d61e1edbf1800618fb2828a226d160dad07883d04e008a7897ee2e4
  b7465d5290d0c0e6c6822236e1daafb94ffe0c5da05d9476be028ad7c1d81

3. Descrição da solução

Para solucionar o problema proposto foi implementado um programa em Java. O programa aceita quatro parâmetros, sendo o primeiro se a operação desejada é de criptografia ou descriptografia, o segundo argumento informa o algoritmo (CBC ou CTR), terceiro é a chave que deve ser utilizada e por último deve-se informar o texto relativo a operação em hexadecimal.

3.1. Encriptar

Para encriptar o programa primeiro cria uma instância de do *Cipher* requisitando uma instância relativa ao modo de operação escolhido. Em seguida o programa gera um *iv* (vetor de inicialização) aleatório através do *SecureRandom* do Java. O próximo passo é inicializar a instância do Cipher passando os parâmetros para informar que se deseja criptografar, informando a chave e o *iv*. Logo após é feita a criptografia do texto em hexadecimal requerido. Por fim concatena-se ao início do texto o respectivo *iv* para enfim retornar.

3.2. Decriptar

Para desencriptar o processo é diferente. A instância do Cipher é requerida da mesma forma que para o modo de encriptação, no entanto as diferenças iniciam na hora de resgatar o *iv* que já está contido no texto, sendo os primeiros 16 bytes do mesmo. A inicialização do Cipher se dá com os mesmos parâmetros, a única divergência é que desta vez informamos que desejamos descriptografar. Por fim um valor em Hexadecimal é informado para o algoritmo que então retorna o texto claro em um array de bytes que podem então serem convertidos para uma string.

4. Conclusão

Pode-se perceber que apesar da complexidade do algoritmo AES apresenta, sua utilização através de bibliotecas de alto nível como o *Cipher* do Java é extremamente fácil, bem como os modos de operação já pré-definidos dentro da biblioteca.

Este nível de padronização fez com que me sentisse confiante de que o que está sendo encriptado segue padrões internacionais e é seguro, uma vez que foi necessário apenas fazer pequenas configurações e não foi necessário implementar todas as etapas necessárias do AES descritas no "*Federal information processing standards publication 197*" (2001).

Referências

STANDARD, Advance Encryption. Federal information processing standards publication 197. **FIPS PUB**, p. 46-3, 2001.