CIC0201 - Segurança Computacional – 2024/1

Prof. João Gondim

Trabalho de Implementação

Parte I (3 pontos) - Cifra de bloco e modo de operação CTR

A parte I explora a cifra de bloco AES e o modo de operação CTR (contador), tendo três partes: implementação da cifra, do modo de operação e teste.

https://pt.wikipedia.org/wiki/Advanced_Encryption_Standard)

- EtapaI: implementação do AES (bloco= 128 bits, chave 128 bits)

A cifra AES deve ser implementada (cifração e decifração) de forma a ser possível especificar o número de rodadas que se deseja executar. Assim, deve-se implementar a rodada básica da cifra e também as manipulações específicas das rodadas inicial e final.

- Etapa II: implementação do modo de operação CTR

O modo CTR deve ser implementado para a cifra AES conforme especificada acima.

- **Extra 1 (1 ponto)**: Implemente o modo de cifração autenticada GCM - contador de Galois

Para checar a corretude da implementação, pode-se usar o openssl – apenas para verificação.

- **Testes**

O trabalho deve ser testado conforme segue: 0) cifre e decifre um arquivo

- **Extra 2 (1 ponto):**

1) tire uma selfie

2) cifre a selfie no modo CTR com 1, 5, 9 e 13 rodadas do AES implementado na parte 1. Renderize os resultados de cada execução e anexe ao relatório as imagens indicando o número de rodadas.

Parte II (4 pontos) - Gerador/Verificador de Assinaturas

Nesta parte, deve-se implementar um gerador e verificador de assinaturas RSA em arquivos. Assim, deve-se implementar um programa com as seguintes funcionalidades:

- Etapa I: Geração de chaves e cifra
  a. Geração de chaves (p e q primos com no mínimo de 1024 bits)
  b. Cifração/decifração assimétrica RSA usando OAEP.
- Etapa II: Assinatura
  1. Cálculo de hashes da mensagem em claro (função de hash SHA-3)
  2. Assinatura da mensagem (cifração do hash da mensagem)
  3. Formatação do resultado (caracteres especiais e informações para verificação em BASE64)
- Etapa III: Verificação:

1. Parsing do documento assinado e decifração da mensagem (de acordo com a formatação usada, no caso BASE64)
2. Decifração da assinatura (decifração do hash)
3. Verificação (cálculo e comparação do hash do arquivo)

O que deve ser entregue:

- **Relatório (3 pontos)** com:
  - descrição das cifras, modo de operação e operações implementados
  - descrição das implementações
  - selfie e resultados dos Testes (se houver)
  - Descritivo do RSA, do OAEP, da assinatura RSA e do programa.
    todo o código fonte

Observações:

1. Permite-se a utilização de bibliotecas públicas para aritmética modular e função de hash,
2. Não é permitida a utilização de bibliotecas públicas, como OpenSSL, para primitivas de criptográficas de cifração e decifração simétrica e assimétrica, bem como de geração de chaves.
3. A pontuação máxima será conferida os trabalhos que realmente implementarem as seguintes primitivas:
   Parte I:
   a. Implementação do AES
   b. Implementação do CTR

   Extras:

   a. Implementação do contador de Galois
   b. Resultados dos testes com selfie, conforme especificado

   Parte II:
   a. geração de chaves com teste de primalidade (Miller-Rabin)
   b. cifração e decifração RSA
   c. OAEP
   d. Formatação/parsing

4. A avaliação será mediante apresentação do trabalho, com a verificação das funcionalidades e inspeção do código.
5. Implementação preferencialmente individual, podendo ser em dupla. Linguagens preferenciais C, C++, Java e Python.

Data de Entrega: 02/09/2024, 10h. As instruções serão enviadas oportunamente.