Especificação do Trabalho 1
Criptografia

Engenharia de Segurança
Entrega: 07/06/2018 Grupo: 3 pessoas (no máximo)

Descrição:
Faça um estudo comparativo sobre **quatro** algoritmos de criptografia. Especifique o tipo de algoritmo, a força da chave, seu modo de compartilhamento e sua característica de funcionamento. É importante que os dois tipos de algoritmos, entre simétricos e assimétricos sejam descritos.

Em seguida, implemente **um** algoritmo, a sua livre escolha. A saída do programa deverá ser um único arquivo executável que deverá aceitar por parâmetros a seleção do algoritmo e a operação a ser efetuada. O programa gerado deverá ser executado **EXCLUSIVAMENTE** via linha de comando, sem iterações, pois os testes serão executados de forma automatizada. As saídas em tela serão ignoradas. A composição da linha de comando está especificada ao final do documento.

Finalmente, o trio deverá desenvolver um **breve** relatório com os resultados dos experimentos e a explicação do funcionamento do algoritmo.

Entrega:
Via moodle, upload de um arquivo único compactado contendo:
1) Código fonte documentado.
2) Breve relatório contendo as seguintes seções:

a) Resumo dos quatro algoritmos estudados, identificando o tipo, função da chave, estratégia para sua troca e funcionamento geral.

b) Algoritmo implementado:
i) Caso faça uma modificação em um algoritmo ou implemente seu próprio, será necessário descrever as características, modificações e justificá-las.
ii) Restrições do algoritmo implementado.

c) Instruções para compilação

i) Deve conter a versão do compilador utilizado

ii) No caso do uso de bibliotecas, especificar a fonte onde a mesma pode ser obtida e a versão utilizada

iii) Sequencia de passos para compilação, caso não exista um *Makefile*

Dúvidas importantes:

- **Bibliotecas:** ***Não deve ser utilizada nenhuma biblioteca*** / framework / código disponível na internet sobre qualquer formato que faça a criptografia em si. Podem ser utilizadas bibliotecas livres para qualquer outra função (ex: facilitar a leitura e escrita do arquivo, cálculo de números primos ou de números grandes, entre outros).
- **Compilação:** Se o código fonte não compilar o trabalho não será considerado entregue! (zero)
- Sobre restrições do algoritmo: A correção inicialmente é feita por meio da comparação do arquivo decrifrado com o arquivo original. Se seu algoritmo possui alguma restrição que impeça que eles sejam idênticos, especificar estas restrições no relatório.
- Os arquivos de entrada para testes contém acentos e pontuação. Caso o algoritmo escolhido não processe estes caracteres você pode:

1. Copiar o caractere na mesma posição no arquivo de saída, ignorando a função de criptografia.
2. Tratar a entrada, eliminando o que é indesejável, antes de processar o arquivo.

Ainda, os arquivos de teste podem ser muito pequenos, com menos de 100 bytes. O tamanho máximo é de 20 Mbytes.

Outros formatos: Opcionalmente você pode fazer criptografia de outro tipo de arquivo, como criptografia de imagem, áudio ou outros dados binários aleatórios. Todas as regras anteriores são aplicáveis.

Dica: Leitura, escrita e outras funções comuns poderão ser reaproveitadas. Se você tratar a criptografia ou decriptografia como uma função "BlocoCifrado = cifra(blocoOriginal, K)", você terá pouco código a reimplementar para dois algoritmos.

Critérios de avaliação:

- Originalidade do algoritmo. (baseie-se em um algoritmos pré-

existentes, mas faça alguma alteração sobre o mesmo para tentar torná-lo mais forte / ou escreva seu próprio algoritmo).
• Complexidade do algoritmo.
• Funcionamento correto (obtenção da mensagem original após as duas operações)
• Qualidade da documentação interna
• Relatório

Explicação da linha de comando:

Exemplos de linhas de comandos válidas:

```
C:\>programa.exe 1 C key01.txt arquivoentrada01.txt arquivocifrado01.txt
~]$ ./programa.out 1 D key01.txt arquivocifrado01.txt arquivodecifrado01.txt
~]$ ./programa.out 1 K key01.txt 01234567
```

Nestas linhas de exemplo, o primeiro item é programa.exe/programa.out, o arquivo de saída do código compilado,

• 1 ou 2 indicam quais algoritmos deverão ser utilizados (entre as duas opções de algoritmos implementados existentes).
• C ou D, em maiúsculo, indicam a operação de cifragem ou decifragem. Os arquivos na sequencia deverão ser criados e lidos na pasta do executável de saída. A opção K indica a geração do arquivo de chave key01.txt, se você precisar que o arquivo tenha um formato especial.

Atenção: os nomes dos arquivos poderão ser diferentes nos testes e mesmo o caminho completo ou relativo do arquivo poderá ser informado. Considere apenas como ordem dos parâmetros informados.

• key.txt é o arquivo que contém a chave ou par de chaves utilizados no processo de criptografia e decriptografia.
• arquivoentrada.txt é um arquivo de entrada. Para casos solicitados no relatório, poderão ser apresentados em outros formatos.
• arquivocifrado.txt é seu arquivo intermediário, contém dos dados criptografados.
• arquivodecifrado.txt é seu arquivo com a mensagem decifrada. Deve ser o mais similar ao arquivoentrada.txt possível.
• 01234567 é uma semente aleatória opcional que poderá ser utilizada para a geração do arquivo de chaves. Será sempre composta por uma sequencia de 8 dígitos não repetidos, variando entre 0 e 7.