Projeto 1

João Vítor Fonseca Pimenta
251031613

Felipe Augusto Ferreira de Almeida
251038534

Daniel Alves de Santana
251031382

***Resumo*—Este relatório apresenta a implementação e análise da cifra de Vigenère, explorando seu funcionamento e vulnerabilidades por meio de um ataque de criptoanálise. Os experimentos mostraram que seu mecanismo é suscetível a ataques estatísticos, evidenciando sua fragilidade frente a métodos modernos de criptografia.**

***Index Terms*—Vigenère, cifra, ataque, frequência**

I. Introdução

O estudo da segurança da informação é de suma importância para a área de Redes de Computadores e da Computação como um todo, para que haja garantia que a transmissão e recebimento de dados entre sistemas tenha confidencialidade, autenticidade e integridade. Nesse contexto, a criptografia se apresenta como uma ferramenta essencial para a proteção de mensagens contra acessos indevidos e ataques maliciosos. Com o passar dos anos, diversos métodos de criptografia foram propostos.

Entre eles, destaca-se a cifra de Vigenère, uma técnica de criptografia clássica que desempenhou papel importante na evolução dos sistemas criptográficos. Embora atualmente seja considerada vulnerável, o estudo dessa cifra permite compreender os fundamentos da criptografia e os mecanismos que inspiraram os algoritmos mais modernos utilizados em redes de computadores. Este trabalho tem como objetivo implementar e analisar a cifra de Vigenère, explorando o seu funcionamento e sua fragilidade diante de ataques de criptoanálise.

Este documento está disposto da seguinte forma: após a introdução, apresenta-se a **Fundamentação Teórica**, abordando o funcionamento da cifra e os princípios da análise de frequência. Em seguida, descreve-se o **Ambiente Experimental**, detalhando os métodos, ferramentas e linguagens utilizadas para a implementação. Na sequência, são discutidos os **Resultados Obtidos**, destacando a eficiência do algoritmo de cifragem/decifragem e a eficácia do ataque. Por fim, são expostas as **Conclusões**, relacionando os aprendizados do trabalho com os conceitos da disciplina de Segurança Computacional.

Os códigos de implementação podem ser acessados no link a seguir: LINK.

II. Fundamentação Teórica

A cifra de Vigenère é um método de criptografia semelhante à cifra de César, porém enquanto a cifra de César usa uma letra como senha, a cifra de Vigenère usa uma palavra.

O funcionamento é similar nos dois métodos quando observamos letra a letra: a letra é deslocada x casas para a frente, ou seja, se considerarmos:

$A = 0, B = 1, C = 2, ..., Z = 25,$

temos que um deslocamento de C (2), por exemplo, transformaria o alfabeto em:

$A => C$

$B => D$

$C => E$

...

$Z => B$

Na cifra de Vigènere a ideia é a mesma, porém ao invés de termos o mesmo deslocamento para todas as letras, temos um deslocamento para cada caracter da palavra-senha. Por exemplo: Vamos supor que temos o texto ”A CASA AMARELA ESTAVA TODA APAGADA.”e a palavra-senha ”COR”.

O primeiro passo é transformar a palavra-senha em uma senha do mesmo tamanho da mensagem. Faremos isso repetindo a mesma até que a condição esteja satisfeita. Uma vez que a mensagem tem 29 letras (vamos ignorar espaços e pontuações neste trabalho), a senha será: ”CORCORCORCORCORCORCORCORCORCO”.

Agora que a mensagem e a senha têm a mesma cardinalidade, podemos aplicar a cifra de César para cada letra da mensagem usando como deslocamento a letra que ocupa aquela posição na senha.

Por exemplo, as duas primeiras palavras da mensagem são ”A CASA”, e a senha nessas posições é composta pela substring ”C ORCO”, portanto a cifra desse trecho é:

A cesar(C) = C,

C cesar(O) = Q,

A cesar(R) = R,

S cesar(C) = U,

A cesar(O) = O.

O criptograma completo fica: ”C QRUO ROOIGZR GGKCJR VCUC OGCURFO.”.

Para descriptografar é usada a mesma ideia, porém ao invés de deslocar para frente, iremos deslocar para trás, assim como na cifra de César.

Apesar de parecer segura à primeira vista, a cifra de Vigenère é facilmente quebrada usando análise de frequência das letras.

Neste trabalho implementamos um ataque à cifra de Vigenère em linguagem C++.

O ataque foi dividido em duas partes:

A. Descobrir tamanho da senha

O método utilizado para descobrir o tamanho da senha foi o **teste de Kasiski**. Essa técnica permite encontrar o comprimento provável da chave, a partir da análise de repetições de padrões no texto cifrado.

1. Repetições de sequências:

Na cifra de Vigenère, quando uma sequência de letras do texto criptografado coincide com as posições da chave, acontece uma repetição no texto cifrado. Assim, ao procurar por sequências repetidas no criptograma, é possível identificar padrões associados à repetição da chave.

Ex.: `GXCFZFOKQATUIFUFERQTEWZFOKMWO`

A substring `ZFOK` aparece em mais de uma posição (por exemplo, nas posições 5 e 23), isso indica que essa parte do texto foi cifrada com o mesmo trecho da chave.

2. Distâncias entre repetições:

Após identificar essas sequências repetidas, calcula-se a distância entre suas ocorrências no texto criptografado. Essas distâncias são múltiplos do comprimento da chave, já que as repetições acontecem quando os alinhamentos da chave coincidem. No caso da substring `ZFOK` encontrada nas posições 5 e 23, a distância entre as ocorrências é $23 - 5 = 18$. Isso sugere que o comprimento da chave pode ser um divisor de 18.

3. Cálculo dos divisores:

O próximo passo é calcular os divisores dessas distâncias. Como as distâncias tendem a ser múltiplos do comprimento da chave, os divisores mais recorrentes fornecem indicações do comprimento da chave. A distância 18 tem como divisores: $1, 2, 3, 6, 9, 18$. Se outras repetições também gerarem distâncias com o número 6 como divisor, isso reforça a hipótese de que a chave pode ter comprimento 6.

4. Análise de frequência dos divisores:

Ao contar a frequência de cada divisor encontrado, pode-se estimar qual é o comprimento da chave mais provável. Em geral, o divisor mais frequente (ou um dos mais frequentes) é considerado o valor candidato para o tamanho da chave. Se as distâncias analisadas produzem os divisores $[2, 3, 6, 10, 6, 15, 6]$, é provável que 6 seja o comprimento da chave.

B. Descobrir senha

Após descoberto um provável tamanho da palavra-senha, o usamos para fazer a análise de frequência das letras.

Assumindo que o tamanho da palavra-senha é n, montamos um mapa de frequência das letras para cada posição $i \ mod \ n$, e comparamos cada um deles ao mapa de frequência das letras no idioma da mensagem (neste trabalho: inglês ou português). Ao achar um deslocamento que mais se assemelha à frequência das letras no idioma, assumimos aquele deslocamento como a letra a ser colocada naquela posição.

A semelhança deslocamento-alfabeto foi calculada multiplicando a frequência de cada letra, no idioma da mensagem, por sua respectiva frequência naquele deslocamento e somando todos os produtos daquele deslocamento. O deslocamento que obtiver a maior soma será o escolhido para compor a palavra-senha naquela posição.

Por exemplo:

Assumindo que o tamanho da palavra-senha é três (3), ao observar o criptograma ”C QRUO ROOIGZR GGKCJR VCUC OGCURFO.”, podemos notar que a frequência das primeiras letras $mod 3$, isto é, as letras nas posições $\{0, 3, 6, ...\}$ formam a seguinte distribuição:

$C(4) => 4\%$

$U(1) => 1\%$

$O(1) => 1\%$

$G(2) => 2\%$

$V(1) => 1\%$

$F(1) => 1\%$.

É fácil notar que a letra 'C' é a mais frequente. Assumindo que o texto original foi escrito em português, isso é um forte indício de que a primeira letra da palavra-senha é 'C', pois, a letra 'A' é a mais frequente na língua portuguesa e, ao mapear todos os 'A's que aparecem em posições congruentes a zero $mod3$, é mais provável que a freqüência se mantenha.

Isso está correto, porque esse criptograma é o criptograma que criamos no início desta seção, usando a palavra-senha "COR".

Observe que é PROVÁVEL que isso esteja correto, não é uma certeza, devido ao fato desse algoritmo ser probabilístico, e não determinístico. Mas de fato, quanto maior o tamanho do criptograma, maior a probabilidade do resultado ser positivo.

III. Ambiente Experimental e Análise de Resultados

A. Descrição do cenário

As técnicas neste trabalho foram realizadas utilizando programação em **C++**, no editor **Visual Studio Code (VS-Code)**.

Para simular um ataque à **cifra de Vigenère**, foram criados diferentes códigos em **C++**, cada um responsável por uma etapa do ataque:

- **cifrador.cpp**: responsável por cifrar a mensagem original, utilizando a senha informada;
- **tamanho.cpp**: aplica o **Teste de Kasiski** para estimar o comprimento da chave, analisando padrões repetidos no criptograma;
- **ataque.cpp**: após estimado o tamanho da senha, realiza a análise de frequência em diferentes idiomas (português ou inglês) para deduzir a chave;
- **decifrador.cpp**: implementa o processo inverso do **cifrador.cpp**, recuperando o texto original a partir do criptograma e da senha;

Durante os testes, foram utilizadas diferentes mensagens (plaintext) para verificar o ataque, sabendo que criptogramas maiores costumam gerar distribuições mais confiáveis da frequência das letras do criptograma com as frequências das letras do idioma escolhido.

B. Análise de Resultados

Quanto à segunda parte do ataque (usar o tamanho para descobrir a senha), os resultados foram bem satisfatórios.

A princípio foi difícil escolher a função que calcularia a semelhança deslocamento-alfabeto, mas após definida a função, o algoritmo obteve sucesso em todos os testes feitos para textos grandes (acima de 250 letras) com senhas entre 3 e 10 letras.

Na descoberta do tamanho da senha, foi implementado o teste de Kasiski, que identifica sequências repetidas no criptograma, calcula as distâncias entre suas ocorrências e os divisores mais frequentes. Uma dificuldade encontrada foi que em alguns casos, o algoritmo indicava como candidato ao tamanho da chave um divisor menor, mas com frequência relativamente alta (como 2 ou 3), ignorando um divisor maior que também possuía alta frequência e correspondia ao verdadeiro tamanho da chave. A solução encontrada foi realizar uma análise manual dos resultados impressos no arquivo tamanho.txt. Quando vários divisores tinham frequência elevada, o maior valor tendia a ser o comprimento mais provável da senha.

IV. Conclusão

A realização deste trabalho permitiu compreender de forma prática o funcionamento da cifra de Vigenère e suas limitações diante de ataques baseados em análise de frequência. A implementação do cifrador e decifrador demonstrou a simplicidade do algoritmo, enquanto a aplicação do Teste de Kasiski auxiliou a estimar o tamanho da chave com uma precisão razoável. Apesar de seu valor histórico, os experimentos confirmaram que a cifra de Vigenère não oferece segurança frente às técnicas atuais de criptoanálise. Dessa forma, este estudo contribuiu para consolidar os conhecimentos de Segurança Computacional.

Referências

[1] Kurose Jim, Ross Keith. "Security"em *Computer networking: A top-down approach edition.*. 9a. Pearson. 2025

[2] Udacity, *Vigênere Cipher.* (2015). Acessado: 06 de Setembro, 2025. Disponível em: https://www.youtube.com/watch?v=SkJcmCaHqS0&ab_channel=Udacity

[3] Veitch Brian, *Cryptography - Breaking the Vigenere Cipher* (2014). Acessado: 06 de Setembro, 2025. Disponível em: https://www.youtube.com/watch?v=P4z3jAOzT9I&ab_channel=BrianVeitch

[4] Theoretically, *Vigenere Cipher - Decryption (Unknown Key)* (2015). Acessado: 06 de Setembro, 2025. Disponível em: https://www.youtube.com/watch?v=LaWp_Kq0cKs&ab_channel=Theoretically

[5] Rodriguez-Clark Daniel, "Kasiski Analysis: Breaking the Code"crypto.interactive-maths.com. https://crypto.interactive-maths.com/kasiski-analysis-breaking-the-code.html Acessado: 06 de Setembro, 2025.

[6] Michel Polak Jean, "Vigenère Analysis"legacy.cryptool.org. https://legacy.cryptool.org/en/cto/vigenerebreak Acessado: 06 de Setembro, 2025.