

O Algoritmo Usado No Programa de Criptografia PASME

Péricles Lopes Machado

Abstract—This work will present the main encryption algorithm of the PASME tool, PASME allows encrypt and hide information in various types of files. The algorithm uses the fact that factoring large numbers is a difficult issue in terms of computational performing to make the main steps of the encryption.

Resumo—Neste trabalho será apresentado o principal algoritmo de criptografia da ferramenta PASME, a qual permite encriptação e ocultamento de informações em diversos tipos de arquivos. O algoritmo utiliza o fato da fatoração de números grandes ser um problema difícil do ponto de vista computacional, efetuando assim, os principais passos da encriptação.

Palavras-chave—Criptografia, Teoria dos números, Teoria da informação

I. INTRODUÇÃO

A ideia fundamental de qualquer algoritmo de criptografia é modificar a representação de uma informação para garantir proteção contra acesso indevidos.

No decorrer dos anos, muitos algoritmos de criptografia foram desenvolvidos. Um dos mais antigos realiza uma permutação no alfabeto que contém todos os símbolos da mensagem que será encriptada. Contudo, este algoritmo apresenta grande vulnerabilidade a uma análise da frequência de ocorrência de determinados símbolos, principalmente quando aplicados à textos escritos.

Outro método clássico, usado em mensagens binárias, consiste em inverter certos bits e armazenar a posição dos bits que foram invertidos em outra palavra, a folha-chave, com o mesmo tamanho da mensagem que foi encriptada. Um problema desse método é que o tamanho da folha-chave pode ser muito grande, inviabilizando o processo de encriptação.

Muitos algoritmos modernos utilizam estratégias envolvendo teoria dos números através da utilização de problemas que atualmente são intratáveis do ponto de vista computacional. Um exemplo clássico desta classe de algoritmo é o RSA [2] [3].

A ideia do presente trabalho é utilizar a intratabilidade da fatoração de inteiros grandes para realizar os passos-chave de sua encriptação. Nas próximas seções, serão descritos os passos realizados pelo algoritmo de encriptação PASME, além de serem comentados alguns detalhes de sua implementação [1].

II. ALGUMAS FUNÇÕES FUNDAMENTAIS

A. A função \mp (inflar)

A ideia fundamental do algoritmo PASME é a mudança na base de representação de um número. Mudar a base de

representação de um número inteiro $n = a_0a_1a_2...a_k$ para a base b consiste em realizar a operação descrita na equação 1

$$T(n, b) = a_0b^k + a_1b^{k-1} + ... + a_kb^0 \quad (1)$$

A função \mp descrita em 2 é uma mudança de base onde a cada dígito é adicionado um "lixo".

$$\mp(n, b, v) = (a_0 + c_0)b^1 + (a_1 + c_1)b^2 + ... + (a_k + c_k)b^{k+1} \quad (2)$$

Onde c_i é descrito na equação 3.

$$c_i = \begin{cases} \triangleright(v) & , se \quad i = 0 \\ \triangleright(c_{i-1}) & , se \quad i > 0 \end{cases} \quad (3)$$

Nas equações 2 e 3, $\triangleright(x)$ é o próximo primo depois de x , v é um inteiro qualquer, n é a informação representada na forma de um inteiro, a_k é um dígito de n na base original e b é a base alvo.

B. A função \pm (sujar)

\pm é semelhante a função \mp , só que o "lixo" v usado é o mesmo em todos dígitos, conforme pode ser visto na equação 4.

$$\pm(n, b, v) = (a_0 + v)b^0 + (a_1 + v)b^1 + ... + (a_k + v)b^k \quad (4)$$

III. O ALGORITMO DE ENCRIPÇÃO PASME

A seguir, serão descritos os procedimentos para encriptar ou desencriptar uma mensagem usando o algoritmo PASME. O algoritmo $PASME(n, key)$ encripta uma mensagem n usando a frase-chave key .

A. Encriptando uma mensagem

O processo de encriptação inicia com a geração de 7 números aleatórios (de preferência, grandes) $r_i, i = 1...7$. Em seguida, são definidos 4 números $K_i = \triangleright(r_i)$, para $i = 1...5$ e $i \neq 3$, $K_3 = \triangleright(K_5 + d_{max} + r_3 + 1)$, d_{max} é o maior dígito da base em que a informação originalmente está representada.

Para continuar o processo de encriptação, uma frase-chave key tem de ser fornecida. Usando-se a frase-chave, são gerados os números $W = \mp(key, K_3, K_2) + K_1$, $Q = \triangleright(\pm(n, K_3, K_5) + r_7)$, $P = WQ + K_4$, e $X = \pm(n, K_3, K_5) \text{ xor } Q$. X é a mensagem n encriptada.

As informações divulgadas são os números $K_i (i = 1...5)$, P e X .

B. Descriptando uma mensagem

Para descriptar uma mensagem, é preciso que sejam fornecidos os números $K_i (i = 1 \dots 5)$, P e X , além da frase-chave key .

O primeiro passo da descriptação é a validação da chave, para realizar essa operação, gera-se o número $W' = \mp(key, K_3, K_2) + K_1$ e é verificado se $P \bmod W' = K_4$. Efetuada a validação, pode-se recuperar $Q = (P - K_4)/W'$.

Com Q recuperado, a mensagem n oculta em X poderá ser revelada. Para revelar a mensagem n , gera-se o número $Y = X \text{ xor } Q$ e o procedimento descrito a seguir tem de ser efetuado.

- 1) $X' = \emptyset$, X' é uma palavra vazia
- 2) Enquanto $Y \neq 0$:
 - a) $a \leftarrow Y \bmod K_3$
 - b) $Y \leftarrow Y - a$
 - c) $Y \leftarrow Y/K_3$, efetua-se a divisão inteira de Y por K_3 .
 - d) $a \leftarrow a - K_5$
 - e) $X' \leftarrow X' \oplus a$, \oplus é a operação de concatenação, ou seja, a união de duas palavras (por exemplo, $33 \oplus 5 = 335$).
- 3) X' é a mensagem descriptada

IV. COMENTÁRIOS SOBRE A IMPLEMENTAÇÃO DE PASME DISPONÍVEL EM [1]

Em [1] está disponível uma implementação do algoritmo de criptografia descrito na seção III. Essa implementação utiliza a biblioteca GMP [4] para realizar as operações envolvendo inteiros presentes no algoritmo PASME. Como a ferramenta [1] permite encriptar arquivos com tamanho variáveis, usar o algoritmo PASME nem sempre é uma boa escolha, já que dependendo do tamanho da mensagem o tempo de execução pode ser alto. Então, por questões de eficiência, a implementação [1] utiliza o processo de encriptação de dois passos descrito a seguir para encriptar uma mensagem n .

- 1) Gera-se uma folha-chave fc com um tamanho de $L(fc)$ bytes.
- 2) Cria-se aleatoriamente uma frase-chave key com $L(key)$ bytes de tamanho.
- 3) Utiliza-se o algoritmo descrito em III para encriptar a folha-chave fc .
- 4) Quebra-se a mensagem n em $L(n)$ bytes,
- 5) $i \leftarrow 0$
- 6) $k \leftarrow 0$
- 7) $X \leftarrow \emptyset$
- 8) Enquanto $i \leq L(n)$:
 - a) $X \leftarrow X \oplus (n_i \text{ xor } fc_k)$, n_i é i -ésimo byte da mensagem n e fc_k é o k -ésimo byte da folha-chave fc .
 - b) $i \leftarrow i + 1$
 - c) $k \leftarrow (k + 1) \bmod L(fc)$

Para descriptar, o passo (1) do algoritmo anterior não é executado, no passo (2) é fornecido a frase-chave que "abre" a mensagem e no passo (3) é chamado o algoritmo de descriptação descrito em III.

Na implementação [1], cada símbolo (dígito num número) tem 1 byte (8 bits) de comprimento.

A implementação [1] armazena em um arquivo-alvo as informações públicas geradas pelo algoritmo III e a mensagem X gerada pelo procedimento anterior.

Para ocultar informações em arquivos, [1] primeiramente verifica o tamanho, em bytes, da informação que será ocultada. Após isso, a informação é concatenada ao arquivo e, por fim, concatena-se o tamanho da informação (em [1], um inteiro com 4 bytes de comprimento). O procedimento para recuperar a informação é semelhante, só que, primeiramente, recupera-se o tamanho L (em [1], os 4 últimos bytes do arquivo) da informação que está oculta, depois recua-se $L - 4$ bytes a partir do fim do arquivo, no caso de [1], e armazena-se os L bytes seguintes em um arquivo-alvo.

A interface gráfica da implementação [1] foi criada utilizando-se o framework QT4 [5].

V. CONCLUSÕES

Este trabalho apresentou um algoritmo de encriptação que usa o fato da mesma informação ter significados distintos dependendo da base em que está representada e de, atualmente, certos problemas em teoria dos números serem intratáveis. Tal algoritmo faz parte da ferramenta PASME que permite a encriptação e ocultamento da informação em arquivos nos mais diversos formatos.

VI. AGRADECIMENTOS

O autor agradece a Diego Aranha por apontar uma falha no algoritmo inicial, a João Augusto Palmitesta Neto por sugestões e testes na implementação [1] do algoritmo e Fabio Lobato por revisar o artigo.

REFERENCES

- [1] "Projeto pasme," <http://sourceforge.net/projects/pasme/>.
- [2] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Algoritmos*. Editora Campus, 2002.
- [3] L. Lovász, J. Pelikán, and K. Vesztegombi, *Matemática Discreta*. Sociedade Brasileira de Matemática, 2003.
- [4] "The gnu multiple precision arithmetic library," <http://gmplib.org/>.
- [5] "qt - cross platform and ui framework," <http://qt.nokia.com/>.