Criptografia RSA gaussiana

Luis Antonio Coêlho

Relatório para Trabalho de Conclusão de Curso - parte I apresentado à
Faculdade de Tecnologia da
Universidade Estadual de Campinas

Orientador: **Profa. Dra. Juliana Bueno**

29 de novembro de 2016

Resumo

O presente relatório expõe o resultado parcial do projeto para TCC sobre o algoritmo de criptografia RSA gaussiano.

Sumário

Capítulo 1

Justificativa

Faz parte da sobrevivência humana manter segredo acerca de alguns fatos e quando se trata de comunicação, seja escrita ou falada, essa prática se intensifica ainda mais à medida que se expandem os meios de comunicação. A necessidade de manter algumas informações em segredo levou o homem a se comunicar através de códigos e à medida em que tais códigos são descobertos há a necessidade de se criar novos códigos para que a comunicação seja antida em sigilo. Foi a partir desse tipo de necessidade que se inaugurou o que chamamos de *criptografia*, isto é, ao conjunto de técnicas usadas para se comunicar em códigos. O objetivo da comunicação em códigos é garantir que apenas as pessoas interessadas (ou envolvidas na comunicação) possam compreender a mensagem codificada (ou criptogtafada), garantindo que a mensagem seja lida apenas pelos interessados.

Para compreender como funciona o processo de codificação e decodificação fazemos uso de uma série de termos técnicos, com intuito pedagógico iremos introduzir tais conceitos apresentando um dos primeiros algorítmos criptográficos que se tem conhecimento, a criptografia de César, e suas sucessoras.

A chamada *criptografia de César*, criado pelo imperador romano César Augusto, consistia em substituir cada letra por outra que estivesse a três posições a frente, como, por exemplo, a letra **A** era substituída pela letra **D**.

Uma forma muito natural de se generalizar o algoritmo de César é fazer a troca da letra por outra em uma posição qualquer fixada. A chamada *criptografia de substituição monoalfabética* consite em substituir cada letra pela letra que ocupa n posições a sua frente, sendo que o número n é conhecido apenas pelo emissor e pelo receptor da mensagem. Chamamos este

número n de *chave criptográfica*.

O algoritmo monoalfabético tem a característica indesejada de ser de fácil decodificação, pois possui apenas 26 chaves, e isso faz com que no máximo 26 tentativas o código seja decifrado. Com o intuito de dificultar a quebra do código monoalfabético foram propostas as *cifras de substituição polialfabéticas* em que a chave criptográfica passa a ser uma *palavra* ao invés de um número. A ideia é usar as posições ocupadas pelas letras da chave para determinar o número de posições que devemos avançar para obter a posição da letra encriptada. Vejamos, por meio de um exemplo, como funciona esse sistema criptográfico.

Sejam "SENHA" a nossa chave criptográfica e "ABOBORA" a mensagem a ser encriptada. Abaixo colocamos as letras do alfabeto com suas respectivas posições. Observe que repetimos a primeira linha de letras para facilitar a localização da posição da letra encriptada e usamos a barra para indicar que estamos estamos no segundo ciclo.

1	2	3	4	5	6	7	8	9	10	11	12	13
A	B	C	D	**E**	F	G	**H**	I	J	K	L	M

14	15	16	17	18	19	20	21	22	23	24	25	26
N	O	P	Q	R	**S**	T	U	V	X	Y	W	Z

27	28	29	30	31	32	33	34	35	36	37	38	39
$\overline{A}$	$\overline{B}$	$\overline{C}$	$\overline{D}$	$\overline{E}$	$\overline{F}$	$\overline{G}$	$\overline{H}$	$\overline{I}$	$\overline{J}$	$\overline{K}$	$\overline{L}$	$\overline{M}$

Vejamos como encriptar a palavra "ABOBORA". Iniciamos o processo escrevendo a mensagem. Ao lado de cada letra da mensagem aparece entre parênteses o número que indica a sua posição. Abaixo da mensagem escrevemos as letras da chave criptográfica, repetindo-as de forma cíclica quando necessário. Analogamente, ao lado de cada letra da chave aparece entre parênteses o número da posição ocupada de cada letra, e o sinal de soma indica que devemos avançar o aquele número de posições. Ao final do processo aparecem as letras encriptadas. Entre parênteses está a posições resultante da combinação das posições da mensagem e da chave.

$A(1)$	$B(2)$	$O(15)$	$B(2)$	$O(15)$	$R(18)$	$A(1)$	Mensagem
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	
$S(+19)$	$E(+5)$	$N(+14)$	$H(+8)$	$A(+1)$	$S(+19)$	$E(+5)$	Chave
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	
$T(20)$	$G(7)$	$C(29)$	$J(10)$	$P(16)$	$K(37)$	$F(6)$	Mensagem encriptada

Observe que a encriptação polialfabética é mais difícil de ser quebrada do que a monoalfabética uma vez que letras iguais não têm, necessariamente, a mesma encriptação. Observe que nesse tipo de criptografia o emissor precisa passar a chave para o receptor da mensagem de forma segura para que o receptor possa decifra a mensagem, isto é, a chave usada para encriptar a mensagem é mesma que deve ser usada para decifrar a mensagem. Veremos que esse é justamente o ponto fraco nesse tipo de encriptação pois usa a chamada *chave simétrica*, ou seja, a chave usada pelo emissor para codificar a mensagem é a mesma usada pelo receptor para decodificar a mensagem. Nesse processo, a chave deve ser mantida em segredo e bem guardada para garantir que o código não seja quebrado e isso requer algum tipo de contato físico entre emissor e receptor.

O contato pessoal era um problema durante a Primeira Guerra Mundial, época em que houve muitos investimentos em máquinas de encriptação automática. O *Enigma* foi uma destas máquinas e era utilizada pelos alemães tanto para criptografar como para descriptografar códigos de guerra. Semelhante a uma máquina de escrever, os primeiros modelos foram patenteados por Arthur Scherbius em 1918. Essas máquinas ganharam popularidade entre as forças militares alemãs devido a facilidade de uso e sua suposta indecifrabilidade do código.

O matemático Alan Turing foi o responsável por quebrar o código dos alemães durante a Segunda Guerra Mundial. A descoberta de Turing mostrou a fragilidade da criptografia baseada em chave simétrica e colocou novos desafios à criptografia. O grande problema passou a ser a questão dos protocolos, isto é, como transmitir a chave para o receptor de forma segura sem que haja contato físico entre as partes?

Em 1949, com a publicação do artigo *Communication Theory of Secrecy Systems* [Sha49] de Shannon, temos a inauguração da criptografia moderna. Neste artigo ele escreve metematicamente que cifras teoricamente inquebráveis tem os mesmo requisitos que as cifras polialfabéticas. Com isso ele transformou a criptografia que até então era uma arte em uma ciência.

Em 1976 Diffie e Hellman publicaram *New Directions in Cryptography* [DH76]. Neste artigo há a introdução do conceito da *chave assimetica*, onde há chaves diferentes entre o emissor de mensagens e seu receptor. Com a assimetria de chaves não era mais necessário um contato tão próximo entre emissor e receptor, que já havia sido problema no passado.

Um dos algoritmos deste modelo de chave assimétrica é o RSA(RIVEST et al, 1983) [RSA78], algoritmo desenvolvido por Rivest, Shamir e Adleman. Este algoritmo está presente em muitas aplicações de alta segurança, como bancos, sistemas militares e servidores de internet, e ele utiliza para a

geração de chaves os números primos naturais de grandeza superior a 2^{512} multiplicados entre si.

O grande problema no uso de primos naturais consiste na Hipótese de Riemann (Riemann,1859) [Rie59], um dos sete desafios matemáticos do milênio. A hipótese diz que informando um número teto, podemos saber quantos e quais são os primos presentes naquele conjunto, este tipo de dado é desastroso para modelos criptográficos centrados nos primos naturais, como o RSA, pois tornaria extremamente mais fácil a realização da descoberta das chaves.

Uma forma para se evitar este problema seria uma substituição dos primos naturais por um conjunto de primos não-naturais, como os primos de Gauss (Gauss, 1815) [Gau15]. Estes números são números complexos da forma $a + bi$, onde a e b são diferentes de 0, com $a^2 + b^2$ resultando em um primo natural.

Por isso, com esta pesquisa, vamos testar a capacidade de um algoritmo criptográfico centrado nos primos de gauss de ser tão efetivos quanto os algoritmos criptográficos centrados em primos naturais, para que em um futuro, caso ele seja necessário ele possa ser imediatamente e amplamente utilizado.

Capítulo 2

Atividades executadas

Durante estes quatro meses foi feito o levantamento bibliográfico voltado para os números primos de Gauss e os algoritmos de criptografia, com enfoque no RSA.

A pesquisa está voltada para a compreensão dos tipos de protocolos envolvidos em criptografia RSA. O objetivo da pesquisa é analisar a dificuldade em se propor protocolos baseados em primos de Gauss para a criptografia RSA.

Foram estudados algoritmos para a criptografia RSA clássica. Neste estudo foi observado que é necessário ter uma lista dos números primos para que a criptografia seja executada. Dessa forma, um primeiro avanço no sentido de obter uma criptografia baseada em números primos de Gauss desenvolvemos um código em Python para gerar números primos de Gauss.

Capítulo 3

Principais resultados

A partir dos problemas levantados no decorrer da pesquisa sobre criptografia RSA observamos que uma parte importante do processo para criptografar uma mensagem é ter a disposição uma lista de números primos, como um primeiro avanço para a criptografia RSA gaussiana desenvolvemos um código em linguagem Python para gerar uma lista de números primos de Gauss. Resta, agora, estudar os principais resultados matemáticos usados na criptografia RSA com o intuito de adaptá-los aos primos de Gauss.

Capítulo 4

Organização para conclusão

A proposta para o desenvolvimento do trabalho consiste em estudar detalhadamente os teoremas matemáticos envolvidos na criptografia RSA com o intuito de adaptá-los ao que chamamos de criptografia RSA gaussiana.

O trabalho será estruturado da seguinte forma:

1. Introdução contendo um histórico sobre o desenvolvimento da criptografia focando no papel dos protocolos envolvidos nesse processo.

2. O primeiro capítulo será reservado para descrever a criptografia RSA, focando nos resultados matemáticos necessários para a construção dos protocolos e seus respectivos problemas, tais como o problema P=NP.

3. No segundo capítulo iremos descrever a história e a definição dos números primos de gauss, objetivando adaptar os teoremas vistos na Capítulo 1. Ainda neste capítulo levantaremos os principais problemas envolvidos na criptografia RSA gaussiana.

4. Discussão sobre a criptografia RSA gaussiana enfocando nos principais desafios a serem atacados.

Capítulo 5

Cronograma

ITEM	ATIVIDADES	PERÍODOS (meses)									
		ago	set	out	nov	dez	jan	mar	abr	mai	jun
1	Levantamento de literatura		X								
2	Montagem do Projeto			X							
3	Apresentação do Projeto				X						
4	Desenvolvimento de aplicações				X						
5	Elaboração do Relatório Parcial				X						
6	Apresentação do Relatório Parcial				X						
7	Estudo da viabilidade do RSA gaussiano					X	X	X	X		
8	Análise de Resultados							X	X		
9	Elaboração do Relatório Final								X	X	
10	Revisão do texto									X	
11	Entrega do trabalho final										X
12	Apresentação do trabalho final										X

Figura 5.1: Cronograma com atividades realizadas e futuras

Capítulo 6

Organização para conclusão

1. Compreender os teoremas usados na criptografia RSA.
2. Descrever um algoritmo para obtenção dos primos de Gauss.

Referências Bibliográficas

[DH76] Diffie, Whitfield e Martin Hellman: *New directions in cryptography*. IEEE transactions on Information Theory, 22(6):644–654, 1976.

[Gau15] Gauss, Carl Friedrich: *Methodus nova integralium valores per approximationem inveniendi*. apvd Henricvm Dieterich, 1815.

[mil] *Millennium Problems — Clay Mathematics Institute*. `http://www.claymath.org/millennium-problems`. Acessado em 15/11/2016.

[Rie59] Riemann, Bernhard: *Ueber die Anzahl der Primzahlen unter einer gegebenen Grosse*. Ges. Math. Werke und Wissenschaftlicher Nachlaß, 2:145–155, 1859.

[RSA78] Rivest, Ronald L, Adi Shamir e Leonard Adleman: *A method for obtaining digital signatures and public-key cryptosystems*. Communications of the ACM, 21(2):120–126, 1978.

[SF09] Sinkov, Abraham e Todd Feil: *Elementary cryptanalysis*, volume 22. MAA, 2009.

[Sha49] Shannon, Claude E: *Communication theory of secrecy systems*. Bell system technical journal, 28(4):656–715, 1949.