Criptografia RSA gaussiana

Luis Antonio Coêlho

Trabalho de Conclusão de Curso - apresentado à
Faculdade de Tecnologia da
Universidade Estadual de Campinas

Orientadora: **Profa. Dra. Juliana Bueno**

20 de fevereiro de 2017

Resumo

O presente artigo expõe o resultado da pesquisa para TCC sobre o algoritmo de criptografia RSA gaussiano.

Sumário

1 Introdução **2**

2 O essencial em teoria dos números **7**

2.1 Ciclos e Restos 7
2.2 Números Primos e Fatoração 7
2.3 Inverso Multiplicativo 8

Bibliografia **9**

Capítulo 1

Introdução

O sigilo sempre foi uma arma explorada pelos seres humanos para vencer certas batalhas, mesmo que na cotidiana missão de se comunicar. Foi a partir dessa necessidade que se criou o que chamamos de *criptografia*, nome dado ao conjunto de técnicas usadas para se falar e escrever em códigos. Seu objetivo é garantir que apenas as pessoas envolvidas na comunicação possam compreender a mensagem codificada (ou criptogtafada), garantindo que terceiros não saibam o que foi conversado.

Para compreender como funciona o processo de codificação e decodificação faz-se necessário o uso de uma série de termos técnicos, para fins pedagógicos iremos introduzir tais conceitos apresentando um dos primeiros algorítmos criptogríficos que se tem conhecimento, a criptografia de César, além de seus sucessores.

A chamada *criptografia de César*, criada 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 da mensagem por outra em uma posição qualquer fixada. A chamada *criptografia de substituição monoalfabética* consite em substituir cada letra pela a 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*. Para podermos compreender a mensagem, precisamos substituir as letras que formam a mensagem criptografado pelas as que estão n posições antes.

O algoritmo monoalfabético tem a característica indesejada de ser de fácil decodificação, pois possui apenas 26 chaves possíveis, e isso faz com que no máximo em 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 posiçõ 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 aquele número de posições. Ao final do processo aparecem as letras encriptadas. Entre parênteses está a posição 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 que a monoalfabética uma vez que letras iguais não têm, necessariamente, a mesma encriptação. Observe que neste tipo de criptografia o emissor precisa passar a chave para o receptor da mensagem de forma segura para que o receptor possa decifrar a mensagem, isto é, a chave usada para encriptar a mensagem é a 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.

Durante a Primeira Guerra Mundial o contato físico para a criação de chaves era complicado, isso levou a criação de máquinas automáticas de cripotgrafia. 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 alemães 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 matematicamente que cifras teoricamente inquebráveis são semelhantes 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 ao conceito de *chave assimétrica*, 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. Neste mesmo artigo é apresentado o primeiro algoritmo de criptografia de chave assimétrica ou como é mais conhecido nos dias atuais *Algorito de Criptografia de Chave Pública*, o protocolo de Diffie-Hellman.

Um dos algoritmos mais famosos da criptografia 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 dois números primos de grandeza superior a 2^{512} multiplicados entre si.

Neste trabalho será feita a exposição detalhada da chamada criptografia RSA clássica, enfatizando a parte matemática relacionada a teoria dos números, necessária para a construção do algoritmo.

O maior objetivo deste é analisar a viabilidade em se propor um criptografia derivada do RSA, mas centrada no conjunto dos números primos de Gauss, que são todos os números complexos de forma $a + bi$ tais que a e b existam no conjunto dos inteiros e $a^2 + b^2$

resulte em um número primo, a qual chamamos de criptografia RSA gaussiana.

Como primeiro avanço necessário para este novo algoritmo se faz necessária a adaptação de uma série de resultados própios dos números primos para os números primos gaussianos, por isso grande parte do trabalho se centrará em demonstrar resultados matemáticos e discutir algumas dificuldades em obtê-los.

Como se trata de uma proposta inovadora, deixamos para trabalhos futuros a análise comparativa entre a RSA clássica e a RSA gaussiana.

Capítulo 2

O essencial em teoria dos números

2.1 Ciclos e Restos

Para podermos compreender a aritmética modular, precisamos começar entendendo o conceito de ciclicidade, que são os fatos que ocorrem sempre após um determinado período constante. Um bom exemplo deste conceito é o nascer do sol, que é um evento que ocorre sempre após um ciclo de 24 horas, assim como o dia de seu aniversário ocorre uma vez a cada ciclo de um ano.

O mesmo tipo de evento é observado com o resto dos números inteiros. Tomemos por exemplo os restos de divisão pelo número inteiro 4:

Inteiro	1	2	3	4	5	6	7	8	9	10	11	12
Resto	1	2	3	0	1	2	3	0	1	2	3	0

É visível que após 4 números o resto tende a se repetir. O mesmo feito ocorre a qualquer número inteiro n, onde o ciclo se repetirá sempre a cada n iterações. Os números que apresentam o resto 0 são conhecidos como múltiplos de n.

2.2 Números Primos e Fatoração

Existe um tipo especial de número que só é múltiplo, ou seja, possui resto 0, em duas condições, quando n é igual a 1 ou quando

ele é igual a n. A esse conjunto de números atribui-se o nome de *números primos*.

Todo o número que não é primo é chamado de *Número Composto*, sendo que este número composto pode ser escrito em uma combinação única de fatores primos. O processo de se descobrir estes faotres é chamado de *fatoração*.

2.3 Inverso Multiplicativo

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.