



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

RSA

Trabalho Prático

Criptografia

2016/2017

Tiago Viana nº45043

Pedro Santos nº45245

Prof^a Isabel Maria Oitavem Fonseca da Rocha Kahle

19/05/2017

Índice

História	3
Funcionamento	6
Função Totiente de Euler	7
Algoritmo	8
Computação de Cifragem e Decifragem.....	9
Vulnerabilidades	11
Conclusão	12
Bibliografia.....	13

História

Até certa altura Criptografia era baseada apenas em chaves Simétricas, ou seja, um emissor envia a mensagem com uma certa chave e o recetor descripta usando uma chave igual.

Sabemos então que encriptar é uma conversão de dados com um código especial tornando-os incompreensíveis. Para descriptar recorria-se ao uso da mesma chave.

Existe assim a necessidade de haver uma troca de chave, infelizmente esta troca só poderia ser feita a partir de duas formas. Teriam de se encontrar pessoalmente, o que nem sempre é possível, ou usando comunicação extra usando Deffie Hellman. Um cenário um pouco pior ainda é imaginar que o emissor é uma grande superfície como um banco. Iria existir a necessidade de possuir uma chave, distinta, para cada um dos seus clientes.

Em 1970 James Ellis, engenheiro britânico trabalhou num conceito para recorrer ao uso de uma encriptação publica, baseando se na ideia de que trancar e destrancar são operações inversas. Exemplificando:



Alice compra
cadeado, abre-o e publica-o.



Bob fecha o cadeado com a mensagem e devolve á Alice.



Alice é a única que contém a chave e por isso, a única que consegue abrir o cadeado.

No entanto, praticamente, a ideia consiste em partir a chave em duas partes, chave de encriptação e chave de desencriptação (chave inversa á chave de encriptação).

Precisavam então de uma solução matemática. Esta solução foi encontrada por outro matemático britânico, Clifford Cocks que apresentou uma função de sentido único (*one way function*). Este tipo de função no fundo é uma função que é fácil de computar com qualquer *input* mas difícil de inverter dada uma imagem de um *input* qualquer. Para além de ser uma função de sentido único, ainda existia uma particularidade, a informação privilegiada (*trapdoor*). Esta informação privilegiada era um auxílio para tornar possível inverter a imagem dada por qualquer input.

Em 1976 Diffie Hellman como foi falado, apresentaram um método de criptografia específico para troca de chaves, em que duas partes que não se conhecem, num canal inseguro, conseguem efetuar uma troca de chaves.

Logo após esta publicação, criptógrafos começaram a tentar desenvolver um algoritmo que conseguisse corresponder a essas especificações. Foi então que em 1978 em resposta á necessidade três professores do MIT desenvolveram o RSA. Este nome, RSA toma partido dos seus nomes, Ronald Rivest, Adi Shamir, e Leonard Adleman.

Funcionamento

RSA é uma implementação de sistemas de chaves assimétricas. Envolve um par de chaves, a chave pública (n , e) e uma chave privada (p , q , e). A chave pública, como o próprio nome indica é conhecida por todos e a chave privada deve ser mantida em segredo. Ou seja, toda a gente pode cifrar e enviar uma mensagem usando a chave pública, mas esta mensagem só pode ser decifrada pela respectiva chave privada. Este algoritmo fundamenta-se numa das áreas mais clássicas da matemática, a Teoria dos Números. Baseia-se em conseguir factorizar o produto de dois números primos de grandes dimensões, o que é considerado computacionalmente complexo, sendo que o tempo estimado para o conseguir fazer ronda as centenas ou até milhares de anos. Isto faz o RSA praticamente inquebrável e um dos algoritmos criptográficos mais seguros.

Função Totiente de Euler

Função totiente de Euler ou função phi, representa-se por $\Phi(x)$ e é igual à quantidade de números menores ou iguais a x que sejam coprimos de x .

Números coprimos(ou primos entre si): Dois números são coprimos ou primos entre si se o único divisor comum entre os dois é o 1.

A função é então definida matematicamente:

$$\Phi(x) = \#\{k \in \mathbb{N} \mid k \leq x \wedge \text{mdc}(k, x) = 1\}$$

Exemplo:

$\Phi(8) = 4$, pois no conjunto dos números menores ou iguais a 8, **1 2 3 4 5 6 7 8**, só o 1, 3, 5 e 7 é que são coprimos de 8.

Duas propriedades desta função que serão úteis para a sua utilização no algoritmo RSA:

- Para todos os números primos \mathbf{P} , $\Phi(\mathbf{P}) = \mathbf{P} - 1$, pois os números primos só têm dois divisores, o 1 e ele próprio, o que faz com que todos os números menores que \mathbf{P} sejam coprimos a respeito dele.
- $\Phi(A * B) = \Phi(A) * \Phi(B)$

Algoritmo

Este algoritmo tem três processos:

- **Geração de chaves**, onde são geradas tanto as chaves públicas como as chaves privadas (n , e , p , q e d).
- **Cifragem**, onde é cifrada a mensagem que se pretende enviar
- **Decifragem**, onde é decifrada a mensagem que foi enviada

Geração de Chaves:

1º Passo: Escolher dois números primos **p** e **q** de grandes dimensões, na ordem dos 10^{100} . É importante que estes números sejam secretos, ou seja, só a pessoa que vai descriptar a mensagem é que os pode conhecer.

2º Passo: Calcular a chave pública **n**, $n = pq$.

3º Passo: Determinar a função totiente de n , $\Phi(n) = (p - 1)(q - 1)$.

Usando as propriedades da função temos,

$$\Phi(n) = \Phi(pq) = \Phi(p) * \Phi(q) = (p-1)(q-1)$$

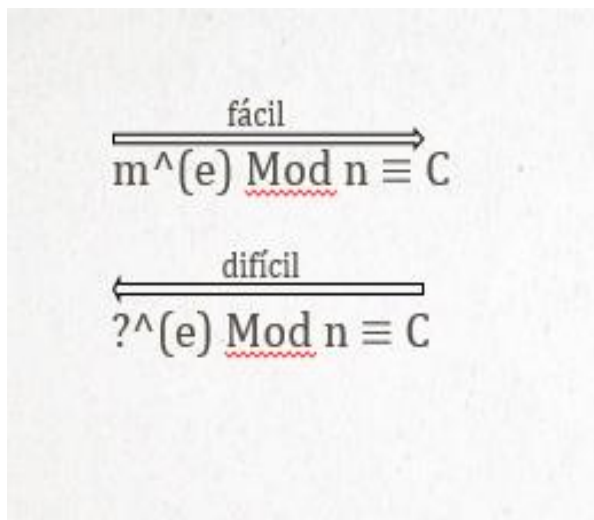
4º Passo: Escolher um inteiro **e** tal que $1 < e < \Phi(n)$ e $\Phi(n)$ e **e** sejam primos entre si.

5º Passo: Calcular a chave privada **d** tal que $de \equiv 1(\text{mod}\Phi(n))$ ou seja **d** é o inverso multiplicativo de e em $\text{mod}\Phi(n)$.

Computação de Cifragem e Decifragem

Na criptografia de chave pública é bastante usado a exponenciação modular que no fundo trata-se de exponenciar módulos. Para facilitar a compreensão desta cifragem e decifragem iremos utilizar um pequeno exemplo.

Imaginando que o Bob possui uma mensagem, que é convertida para um numeral, m . Então este realiza:

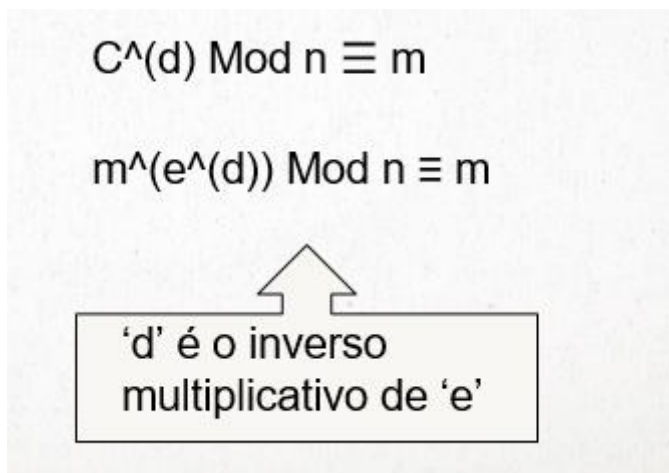


The diagram illustrates the encryption and decryption processes using modular exponentiation. It consists of two parts, each with a horizontal double arrow. The top part has a right-pointing arrow labeled 'fácil' (easy) above it, with the equation $m^e \pmod n \equiv C$ below it. The bottom part has a left-pointing arrow labeled 'difícil' (difficult) above it, with the equation $?^e \pmod n \equiv C$ below it. In both equations, the modulus n is underlined in red.

Aqui conseguimos perceber que a cifragem é uma operação fácil de realizar, mas bastante difícil de converter para descobrir o m . Ao falar de fácil e difícil referimo-nos á computação por de trás da operação.

É neste ponto em que a operação inversa é bastante complicada, que se usa a informação privilegiada.

Assim temos de elevar neste caso, C , a algum numeral, d , que irá reverter o efeito inicial aplicado a m , e retorne esse mesmo m .



No fundo 'd' ser inverso multiplicativo de 'e' implica que:



$$e.d \equiv 1 \text{ mod } \phi(n)$$

Vulnerabilidades

RSA é um sistema muito utilizado por isso vem tendo vulnerabilidades ao longo dos anos. Existe uma lista completa de todos os ataques que se conhecem num artigo de um professor, Dan Boneh. Mas aqui só vamos fazer uma breve explicação de dois ataques comuns:

- **Expoente Privado pequeno** – Isto acontece quando, na tentativa de tornar o processo de descriptação mais rápido, podemos acabar por escolher um expoente privado “pequeno”. Isto pode levar à ruptura total da segurança do RSA, ou seja, consegue-se determinar o expoente privado d e consequentemente a factorização de n .
- **Expoente Público pequeno** – Tal como o expoente privado pequeno, pode-se cair na tentação de escolher um expoente público pequeno e na tentativa de acelerar processo de encriptação. Ao contrário do ataque anterior, este não leva à ruptura da segurança do RSA, mas o objectivo é descriptar uma mensagem sem precisar de descobrir a chave privada.

Conclusão

Concluimos assim que os professores Rivest, Shamir e Adleman são os principais responsáveis pelo algoritmo de criptografia de dados.

Até hoje, é considerado a mais bem sucedida implementação de sistemas de chaves assimétricas.

É importante recordar que é um algoritmo que funciona com um par de chaves, uma pública e privada.

Esta implementação é utilizada diretamente na internet em locais como emails e até compras on-line.

Bibliografia

- <https://pt.wikipedia.org/wiki/RSA>
- https://www.youtube.com/watch?v=wXB-V_Keiu8
- <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.182.9999&rep=rep1&type=pdf>