Departamento de Ciência da Computação

Introdução à Criptografia

Prof. Charles F. de Barros

ATIVIDADES 2 E 3

EXPONENCIAÇÃO RÁPIDA

- Uma das operações mais frequentes em Criptografia de chave pública é a exponenciação modular.
- Esta operação é empregada no acordo de chaves Diffie-Hellman, e também nas cifras El Gamal e RSA.
- Trata-se de calcular

$$a^x \pmod{p}$$

para valores conhecidos de a, x e p.

- No entanto, pensada da forma usual, esta operação torna-se extremamente custosa do ponto de vista computacional, devido ao número de iterações.
- De fato, ao todo devem ser realizadas x-1 multiplicações, o que para valores muito grandes de x se torna inviável.
- No entanto, há uma maneira eficaz de se efetuar a exponenciação modular.

- A ideia é tirar proveito da representação binária do expoente x.
- Escrevendo

$$x = b_0.2^0 + b_1.2^1 + b_2.2^2 + \cdots + b_n.2^n$$

onde cada

$$b_i \in \{0, 1\}$$

podemos calcular

$$a^x = a^{b_0.2^0+b_1.2^1+b_2.2^2+\cdots+b_n.2^n}$$

$$= a^{b_0.2^0} \times a^{b_1.2^1} \times a^{b_2.2^2} \times \cdots \times a^{b_n.2^n}$$

$$= (a^{2^0})^{b_0} \times (a^{2^1})^{b_1} \times (a^{2^2})^{b_2} \times \cdots \times (a^{2^n})^{b_n}$$

- Por exemplo:

$$3^{11} \mod 17 = 3^{1.2^0+1.2^1+0.2^2+1.2^3} \mod 17$$

$$= (3^{2^0})^1.(3^{2^1})^1.(3^{2^2})^0.(3^{2^3})^1 \mod 17$$

- Observe agora que a quantidade de fatores é da ordem de log (x).
- Além disso, os únicos fatores que efetivamente contribuem para o resultado são aqueles que correspondem aos bits não-nulos de x.
- Observe ainda que cada um dos fatores entre parênteses é igual ao quadrado do anterior. De fato,

$$a^{2^i} = (a^{2^{i-1}})^2$$

- Como a exponenciação é modular, a cada multiplicação pode-se fazer a redução módulo p, o que impede os números envolvidos no cálculo de crescerem demasiadamente.
- Com base nestas observações, podemos pensar em um algoritmo que execute a exponenciação modular com O(log (x)) multiplicações.

Universidade Federal de São João del-Rei

Dcomp

Departamento de Ciência da Computação

IDEIA DO ALGORITMO

- **O algoritmo recebe como entrada os valores de a, x e p, e deve retornar o valor de**

$$a^x \pmod{p}$$

- Uma variável RESULTADO acumula os valores parciais das multiplicações
- Uma variável FATOR recebe os fatores entre parênteses que são multiplicados por RESULTADO a cada iteração
- A variável RESULTADO é inicializada com o valor 1
- A variável FATOR é inicializada com o valor de

$$a^{2^0} = a$$

- A cada iteração deve-se dividir o expoente por 2 e verificar o resto, a fim de saber se o bit correspondente da representação binária é 0 ou 1

- RESULTADO = 1
- FATOR = a
- enquanto x > 0:
  - se x mod 2 == 1:
    - RESULTADO * = FATOR mod p
  - FATOR *= FATOR mod p
  - x /= 2

EXEMPLO DE EXECUÇÃO

- Calcular

$$3^{11} \pmod{17}$$

- RESULTADO = 1
- FATOR = 3
- 1ª iteração:
  - 11 MOD 2 = 1
    - RESULTADO = 1.3 mod 17 = 3
  - FATOR = 9
  - x = 11/2 = 5
- 2ª iteração:
  - 5 MOD 2 = 1
    - RESULTADO = 3.9 mod 17 = 10
  - FATOR = 81 mod 17 = 13
  - x = 5/2 = 2
- 3ª iteração:
  - 2 MOD 2 = 0
  - FATOR = 169 mod 17 = 16
  - x = 2/2 = 1
- 4ª iteração:
  - 1 MOD 2 = 1
    - RESULTADO = 10.16 mod 17 = 7
  - FATOR = 256 mod 17 = 1
  - x = 1/2 = 0

- Resposta:

$$3^{11} \pmod{17} = 7$$

- **ATIVIDADE 2:** implemente o algoritmo de exponenciação rápida
- **ATIVIDADE 3**: utilizando o algoritmo de exponenciação rápida, implemente a cifra RSA. (Para geração dos números primos, utilize **from Crypto.Util import number**, e em seguida **number.getPrime(N)**, onde N é o número de bits desejados.
- Chaves do RSA utilizam primos de até 2048 bits

UFSJ Universidade Federal de São João del-Rei