Desenvolvimento Histórico do RSA

Inicialmente, antes da introdução do RSA os sistemas criptográficos utilizados predominantemente eram simétricos, isto é, a mesma chave era utilizada tanto para codificar a mensagem quanto para decodificá-la, mas não havia um meio de distribuir a chave de forma segura e confiável em ambientes abertos, como rede de computadores, esse era o grande problema desse sistema na época.

Em 1976, Whitfield Diffie e Martin Hellman propuseram o conceito de criptografia de chave pública, no qual cada usuário possui um par de chaves: uma pública, divulgada livremente, e uma privada, mantida em segredo. Essa ideia revolucionou a segurança da informação.

O algoritmo que conhecemos hoje como RSA só foi publicado no ano seguinte em 1977 proposto pelos pesquisadores Ron Rivest, Adi Shamir e Leonard Adleman e a origem do nome RSA se dá pela iniciais do sobrenome de seus integrantes que na época estavam no MIT. Esse foi o primeiro sistema prático de criptografia de chave pública baseado em problemas matemáticos bem definidos e elaborados.

No entanto, mesmo que o RSA tenha sido publicado só em 1977, foi descoberto posteriormente que outra pessoa já havia chegado a uma aplicação prática de um algoritmo bem semelhante em 1973, desenvolvido pelo Clifford Cocks, um matemático do serviço de inteligência britânico (GCHQ). Contudo, seu trabalho permaneceu confidencial por motivos de segurança nacional e só foi divulgado publicamente anos depois.

A ideia era resolver problemas presentes na época que era a distribuição segura de chaves em qualquer canal ou meio e também torna possível a assinatura digital que garantia a autenticidade e integridade das mensagens.

As primeiras aplicações do RSA ocorreram em sistemas acadêmicos e militares. Posteriormente, o algoritmo passou a ser utilizado em softwares comerciais, tornando-se um dos pilares da segurança computacional moderna e ao longo dos anos o RSA foi amplamente adotado em protocolos de segurança da internet como por exemplo o HTTPS, em sistemas de email seguros (PGP) e em certificados digitais e infraestruturas de chave pública.

Atualmente o RSA ainda continua sendo bem utilizado, mesmo com o surgimento de novos algoritmos ditos mais eficientes, só mostra o quão impactante e bem construído é o RSA.

Conceitos de Matemática Discreta no RSA

A aritmética modular é a base do RSA. Todas as operações de cifragem e decifragem são realizadas mod n, onde n é o produto de dois números primos grandes. Esse tipo de

aritmética permite trabalhar com números grandes de forma controlada e garante propriedades matemáticas essenciais ao algoritmo.

O algoritmo depende da complexidade e dificuldade computacional de fatorar um número muito grande, dado por $n = p \cdot q$, onde p e q são números primos grandes, o processo de fatoração de números primos suficientemente grandes é praticamente inviável computacionalmente, mesmo que seja fácil multiplicar dois números primos, sendo assim, há garantia na segurança do sistema.

A Função Totiente de Euler, denotada por φ(n), é definida como o número de inteiros positivos menores que n que são coprimos com n. Formalmente:

$$\varphi(n) = |\{ k \in \mathbb{N} : 1 \leq k < n \, e \, gcd(k, n) = 1 \}|$$

Quando n é o produto de dois primos distintos p e q, tem-se:

$$\varphi(n) = (p - 1)(q - 1)$$

Essa função é fundamental no RSA, pois define o conjunto no qual são escolhidos os expoentes das chaves.

O Teorema de Euler afirma que, se gcd(a, n) = 1, então:

$$a^{\varphi(n)} \equiv 1 \, (mod \, n)$$

O Teorema do Pequeno Fermat é um caso particular desse resultado, válido quando nnn é primo. Esses teoremas, juntamente com a escolha adequada dos expoentes e e d, garantem a correção matemática do processo de cifragem e decifragem no algoritmo RSA.

A geração das chaves no RSA ocorre da seguinte forma:

1. Escolhem-se dois números primos grandes p e q;
2. Calcula-se n = p · q;
3. Determina-se φ(n) = (p − 1)(q − 1);
4. Escolhe-se um inteiro e tal que gcd(e, φ(n)) = 1;
5. Calcula-se d como o inverso multiplicativo de e módulo φ(n).

A chave pública é o par (n, e), enquanto a chave privada é o par (n, d). Embora a chave pública seja conhecida, a obtenção da chave privada sem o conhecimento de φ(n) é computacionalmente impraticável, pois requer a fatoração de n. Essa relação matemática é o fundamento da segurança do algoritmo RSA.