CRIPTOGRAFIA RSA: FUNCIONAMENTO E LÓGICA

João Pedro Thomaz Kairalla dos Santos; Lucas Ferreira de Brito; Miguel Mussalam Silva;

Thiago Danilow de Araújo

RESUMO
A pesquisa diante o tema de criptografia do TDE propõe o estudo aprofundado do método criptográfico RSA (Rivest-Shamir-Adleman), enfocando sua operação e relevância num contexto em que a transmissão e recebimento de informações estão em constante evolução. Nos objetivos da pesquisa, planeja-se desenvolver um código em linguagem C que demonstre de maneira abrangente os passos para criptografar e descriptografar em menor escala, evidenciando a validade dos conceitos e fundamentos teóricos do RSA. Além disso, busca-se analisar a segurança e a eficácia do RSA atualmente, comparando-o com outros métodos criptográficos e considerando os cenários de aplicação.

Palavras-chave: RSA 1; Linguagem C 2; Criptografia 3; Segurança 4;

1. INTRODUÇÃO

Em 2020, a empresa de segurança Venafi realizou uma análise abrangente de 1 milhão de sites ao longo de 18 meses. O resultado desse estudo revelou que aproximadamente 51% dos sites examinados empregavam a tecnologia de criptografia RSA. A criptografia RSA é uma criptografia assimétrica (chave pública), que se utiliza de duas chaves distintas, uma pública que fica à disposição de qualquer um que queira criptografar uma mensagem e uma privada que fica em posse do proprietário, sendo o único a conseguir descriptografar a mensagem, o algoritmo foi explicado pelos criadores em 1977 publicamente. Constituindo uma das técnicas de segurança mais robustas da atualidade, é conhecida por ter sido desenvolvida com base na teoria dos números, especificamente na complexidade da fatoração de números em seus primos constituintes. Esta técnica foi utilizada pelos fundadores da empresa RSA Data Security, Inc., sendo eles Ron Rivest, Adi Shamir e Leonard Adleman, sendo o nome RSA derivado das primeiras letras de seus sobrenomes. À medida que a tecnologia continua a avançar, nossos dados pessoais estão cada vez mais armazenados nos servidores de grandes empresas. Com essa revolução digital, a importância da criptografia nunca foi tão evidente. Garantir que nossas informações permaneçam acessíveis apenas para as pessoas certas é uma prioridade crítica. Nesse cenário desafiador, a criptografia RSA tem sido uma ferramenta confiável, evoluindo ao longo dos anos. No entanto, a diferença entre uma criptografia segura e uma insegura é uma batalha constante, semelhante a um jogo de gato e rato. Isso significa que a criptografia deve estar sempre em constante evolução para se manter eficaz. Com um uso tão frequente da tecnologia RSA por diversas empresas, acabamos nos perguntamos se essa tecnologia é realmente segura, se o papel dela é realmente cumprido devidamente. Por essa problemática procuraremos responder o nível criptográfico do tipo RSA, suas limitações, eficácia e nível de segurança e como ele se compara com outras, se é ou não segura, como a criptografia de curva elíptica (ECSDA), reconhecida por sua segurança e eficiência computacional, a fim de estabelecer uma compreensão abrangente e atualizada da sua relevância na proteção de dados sensíveis num ambiente digital em constante mutação.

2. ANÁLISE E COMENTÁRIO DO CONTEÚDO

O código programado em linguagem C nos deu uma proporção de tamanho ao pensarmos nas chaves do método RSA. Mesmo seguindo passo a passo as equações providenciadas amplamente na internet, conseguimos recriar uma criptografia apenas até 32 bits, provavelmente por estourar o limite das variáveis na programação, mesmo usando tipos Long Long. Para Implementar uma RSA 31(RSA com chave pública de 31 bits), podemos usar as entradas “49333” e “50177”. Criptografando a mensagem “ola” neste sistema, temos a saída “806403262990996831731866227202537805 4”, com esse simples exemplo podemos ver o tamanho da mensagem comparado com a saída, em relação as entradas numéricas. Em algoritmos pequenos como este, não se torna um problema sério, a problemática real se torna visível apenas com a escalabilidade a longo termo de algoritmos maiores.

Com a análise feita de Nick sobre as chaves, vemos que:

> a common **RSA 2048-bit public key** provides a security level of **112 bits**. However, **ECDSA requires only 224-bit sized public keys** to provide the same **112-bit security level**. This striking difference in key size has two significant implications. Smaller key sizes require less bandwidth to set up an SSL/TLS stream, which means that ECDSA certificates are ideal for mobile applications. Moreover, such certificates can be stored into devices with much more limiting memory constraints, a fact that allows m/TLS stacks to be implemented in IoT devices without allocating many resources. (2018, p.02)

Comparando com a ECDSA, um algoritmo baseado em curva elíptica de criptografia assimétrica igualmente ao do tema escolhido, podemos ver o tamanho de bits das chaves públicas quando comparado com RSA, vemos que: uma chave pública RSA 2048 equivale ao mesmo que uma ECDSA 224, isso se torna ainda mais acentuado quando continuamos subindo as comparações, evidenciando que a RSA possui um problema de escalabilidade quando comparado com outras criptografias, isso também diz que enquanto continuarmos aumentando as chaves de algoritmos RSA, continuaremos aumentando significantemente a quantidade de processamento, o que pode acarretar em um grande consumo de recursos computacionais, diferente da implementação de um sistema ECDSA.

3. CONSIDERAÇÕES FINAIS

Com a constante imprescindível evolução tecnológica, faz-se útil a análise de quais criptografias serão seguras com o passar do tempo. Foi notado que a RSA é de fato, segura, mas não é a melhor opção em termos de recursos computacionais, implementação e segurança a longo termo. Percebemos que comparada a ECDSA, o método RSA tem um grande problema de escalabilidade e, se possível, é seguro que todas as empresas que usam o RSA, troquem para o ECDSA quando possível, para melhor otimização e segurança de dados. Também percebemos a reação do compilador GCC a contas com números gigantes, observando paradas com RSA além de 32 bits. É importante citar que, caso fosse usada uma biblioteca para a utilização da criptografia, não teríamos tais problemas. O fato é que o tratamento puro de grandes números acarretou problemas. Ao fim, temos um programa em C funcional que simula RSA, e respondemos se hoje em dia, esse método é forte o bastante.

REFERÊNCIAS BIBLIOGRÁFICAS

BARKER, Elaine. **Recommendation for key management**. 5. ed. rev. National Institute of Standards and Technology: Elaine Barker, 2020. 157 p. Disponível em: https://doi.org/10.6028/NIST.SP.800-57pt1r5. Acesso em: 5 out. 2023.

CASTRO, Felipe Lopes. RSA. *In*: CASTRO, Felipe Lopes. **Criptografia RSA: uma abordagem para professores do ensino básico**. Orientador: Alveri Alves Santana. 2014. Trabalho de conclusão de curso (Licenciatura em matemática) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2014. f. 65. Disponível em: https://lume.ufrgs.br/bitstream/handle/10183/110014/000951896.pdf. Acesso em: 25 set. 2023.

HELME, Scott. SSL. **TLS Crawler Report: Top 1 million Analysis 2021**, [*S. l.*], p. 1-1, 9 Dez. 2021. Disponível em: https://venafi.com/blog/crawler-report-top-1-million-analysis-2021/. Acesso em: 20 set. 2023.

NAZIRIDIS, Nick**. Comparing ECDSA vs RSA**. [S. l.], 27 jun. 2018. Disponível em: https://www.ssl.com/article/comparing-ecdsa-vs-rsa/#key-size-to-security-level-ratio. Acesso em: 8 out. 2023.

SAKURAI, Rafael Guimarães. **Criptografia de chave pública ou assimétrica**. [*S. l.*]: Rafael Guimarães Sakurai, 23 maio 2020. Disponível em: http://www.universidadejava.com.br/outros/criptografia-assimetrica/. Acesso em: 21 set. 2023.

SOUSA, Antonio Nilson Laurindo. **Criptografia de chave pública, criptografia RSA**. 2013. 57 f. Dissertação (Mestrado) - Universidade Estadual Paulista Júlio de Mesquita Filho, Rio Claro, 2013. Disponível em: https://repositorio.unesp.br/items/146d3877-ac1e-4ae6-95ea-9229699971f4. Acesso em: 10 out. 2023.

APÊNDICE A – Criptografia RSA em escala menor, produzido em C

Criptografia RSA em C