Troca de chaves de Diffie-Hellman-Merkle

Carlos Maziero

2 de junho de 2014

Um dos principais problemas no uso da criptografia simétrica para a criação de um canal de comunicação segura é a troca de chaves, ou seja, o estabelecimento de um segredo comum entre Alice e Bob. Caso eles não estejam fisicamente próximos, criar uma nova senha secreta comum pode ser complicado.

O protocolo de troca de chaves de Diffie-Hellman-Merkle (*Diffie-Hellman-Merkle Key Exchange Protocol*) [**?**] permite estabelecer uma chave secreta comum, mesmo usando canais de comunicação inseguros. Um atacante que estiver observando o tráfego de rede não poderá inferir a chave secreta a partir das mensagens em trânsito capturadas.

O protocolo de Diffie-Hellman é baseado em aritmética inteira modular e constitui um exemplo bastante didático dos mecanismos básicos de funcionamento da criptografia.

Sejam p um número primo e g uma raiz primitiva[1] módulo p:

passo	Alice	Mallory	Bob
1	escolhe p e g	$\xrightarrow{(p,g)}$	recebe p e g
2	escolhe a		escolhe b
3	$A = g^a mod\ p$		$B = g^b mod\ p$
4	envia A	$\xrightarrow{A}$	recebe A
5	recebe B	$\xleftarrow{B}$	envia B
6	$k = B^a mod\ p$		$k = A^b mod\ p$
	$k = g^{ba} mod\ p$		$k = g^{ab} mod\ p$

Como $g^{ba} mod\ p = g^{ab} mod\ p$, Alice e Bob possuem agora uma chave secreta comum k, que pode ser usada para cifrar e decifrar mensagens.

[1] Uma raiz primitiva módulo p é um número inteiro positivo com certas propriedades específicas em aritmética modular.

Durante o estabelecimento da chave secreta, a usuária Mallory pode observar as trocas de mensagens e obter as seguintes informações:

- O número primo p
- O número gerador g
- $A = g^a mod\ p$ (aqui chamado *chave pública* de Alice)
- $B = g^b mod\ p$ (aqui chamado *chave pública* de Bob)

Para calcular a chave secreta k, ela precisará encontrar a na equação $A = g^a mod\ p$ ou b na equação $B = g^b mod\ p$. Esse cálculo é denominado *problema do logaritmo discreto* e não possui nenhuma solução eficiente conhecida: a solução por força bruta tem complexidade em tempo exponencial em função do número de dígitos de p; o melhor algoritmo conhecido tem complexidade temporal subexponencial.

Em consequência, encontrar a ou b a partir dos dados capturados da rede por Mallory torna-se impraticável se o número primo p for muito grande. Por exemplo, caso seja usado o seguinte número primo de Mersenne[2]:

$$p = 2^{127} - 1 = 170.141.183.460.469.231.731.687.303.715.884.105.727$$

o número de passos necessários para encontrar o logaritmo discreto seria aproximadamente de $\sqrt{p} = 13 \times 10^{18}$, usando o melhor algoritmo conhecido. Um computador que calcule um bilhão (10^9) de tentativas por segundo levaria 413 anos para testar todas as possibilidades!

Apesar de ser robusto em relação ao segredo da chave, o protocolo de Diffie-Hellman-Merkle é suscetível a ataques do tipo *man-in-the-middle*[3], se Mallory puder interceptar as mensagens em trânsito e substituir os valores de p, g, A e B por valores que ela escolher, o que a permitiria estabelecer uma chave secreta $A \rightarrow M$ e uma chave chave secreta $M \rightarrow B$, sem que Alice e Bob percebam. Há versões modificadas do protocolo que resolvem este problema.

Informações mais detalhadas sobre o algoritmo de troca de chaves de Diffie-Helmann podem ser encontradas em:

- *Cryptography and Network Security – Principles and Practice, 4th edition*. William Stallings. Ed. Pearson, 2011.
- *Information Security: Principles and Practice, 2nd Edition*. Mark Stamp. Ed. Wiley, 2011.
- *Applied cryptography: protocols, algorithms, and source code in C, 2nd edition*. B. Schneier. Ed. Wiley, 1996.

[2]Um *número primo de Mersenne* é um número primo de forma $N_m = 2^m - 1$ com $m \geq 1$. Esta família de números primos tem propriedades interessantes para a construção de algoritmos de criptografia e geradores de números aleatórios.

[3]No caso de Mallory, *woman-in-the-middle*.