Resoluçao da Prova 2
Segurança da Informação

Prof. Márcio Moretto Ribeiro

4 de Julho de 2019

Exercício 1: *O protocolo MTProto é utilizado no aplicativo Telegram quando as partes optam por se comunicar por meio de um "chat seguro". Dentre as várias opções heterodoxas usadas no protocolo está o uso do modo "encrypt-and-mac" para garantir confidencialidade, autenticidade e integridade. Vimos na aula que o modo "encrypt-then-mac" possui vantagens sobre suas alternativas (em particular sobre o modo usado pelo Telegram).*

Descreva o modo "encrypt-then-mac" e enuncie as vantagens teóricas que esse modo possui.

O modo "encrypt-then-mac" consiste em primeiro criptografar a mensagem usando um sistema de criptografia $\Pi_E = \langle Gen_E, E', D' \rangle$ e em seguida gerar um código de autenticação (MAC) utilizando um sistema de autenticação $\Pi_M = \langle Gen_M, Mac, Ver \rangle$ gerando o seguinte sistema de criptografia:

- $Gen(1^n) := k = \langle k_E, k_M \rangle$ tal que $Gen_E(1^n) := k_E$ e $Gen_M(1^n) := k_M$
- $E(k, m) := \langle c, t \rangle$ tal que $E'(k_E, m) = c$ e $Mac(k_M, c) = t$
- $D(k, c) := D'(k_E, c)$ se $Ver(k_M, c, t) = 1$ e $\perp$ caso contrário

Caso Π_E seja seguro contra ataques "chosen plaintext" (CPA) e Π_M seja seguro contra falsificação, esse sistema é *seguro contra ataques "chosen ciphertext" (CCA)*. Os modo "encrypt-and-mac" usado pelo protocolo do Telegram não garante isso.

Exercício 2: *Prova de trabalho é uma medida para garantir que um determinado usuário tenha que executar uma certa quantidade de processamento durante a excecução de um protocolo. Essa ideia é usada na mineração de bitcoins e para evitar spams. No segundo caso, brevemente, a ideia é exigir uma quantidade mínima de processamento para um cliente que envie um email. Essa quantidade é desprezível para quem manda algumas dezenas de emails por dia, mas é muito cara para quem deseja mandar milhões de spams.*

Uma forma de prova de trabalho é entregar para o cliente a saída de um hash e pedir para que ele compute uma entrada que produza aquela saída. Argumente que, se a função de hash escolhida é segura contra colisão, o melhor que o cliente pode fazer é gerar valores aleatórios de entrada até encontrar um cuja a saída coincida com o resultado esperado.

Um hash H resistente contra colisão garante que qualquer adversário polinomial consegue gerar um par $\langle x, x' \rangle$ tal que $H(x) = H(x')$ apenas com probabilidade desprezível. A resistencia contra colisão garante, em particular que o hash seja resistente contra pré-imagem. Ou seja, dado y, qualquer adversário polinomial só é capaz de computar x tal que $H(x) = y$ com probabilidade desprezível. Note que isso é verdade pois se H não fosse resistente a pré-imagem então bastaria calcular $H(x) = y$ e então encontrar uma pré-imagem x' tal que $H(x') = y$ para encontrar uma colisão.

Uma vez que H é *resistente à pré-imagem*, o melhor que se pode fazer para encontrar um elemento x tal que $H(x) = y$ é um ataque força-bruta, ou seja, chutar valores de x até achar um tal que $H(x) = y$.

Exercício 3: *Considere as estruturas* $\langle \mathbb{Z}_n, + \rangle$ *formadas pelo conjunto* $\mathbb{Z}_n := \{0, \ldots, n-1\}$ *e a operação de soma* $(+)$ *módulo* n*. Mostre essa estrutura é um grupo cíclico para qualquer valor de* $n \geq 1$ *e que o número* 1 *é sempre um gerador nesses grupos. (Dica: Você precisa mostrar que a operação satisfaz fecho, associatividade, possui elemento neutro e inverso. Depois você deve mostrar que o elemento* 1 *gera todos os elementos do grupo.)*

Explique porque o grupo $\langle \mathbb{Z}_n, + \rangle$ *não é um bom candidato para ser usado no protocolo de Diffie-Hellmann.*

fecho: para qualquer $a, b \in \mathbb{Z}_n$ temos que $0 \leq a + b \ (mod\ n) \leq n$, portanto $a + b \ (mod\ n) \in \mathbb{Z}_n$
associatividade: a soma módulo n satisfaz associatividade, ou seja para todo $a, b, c \in \mathbb{Z}_n$ temos que $(a + b) + c \equiv a + (b + c) \ (mod\ n)$
elemento neutro: o zero 0 neste caso é o elemento neutro, pois para qualquer $a \in \mathbb{Z}_n$ temos que $a + 0 \equiv 0 + a \equiv a \ (mod\ n)$ **inverso:** para todo $a \in \mathbb{Z}_n$ temos que $a + (n - a) \equiv 0 \ (mod\ n)$, como $n - a \in \mathbb{Z}_n$ temos que $(n - a)$ é o inverso de a.

$\langle 1 \rangle := \{0, 1, 1 + 1, 1 + 1 + 1, \ldots n - 1\} = \mathbb{Z}_n$

Como 1 gera todos elementos do grupo, por definição, o grupo é cíclico.

Esse grupo não é adequado para ser usado no protocolo de Diffie-Hellman, pois *o problema do Logaritmo Discreto é um problema fácil*. Note que neste caso g^x seria um gerador g somado x vezes $(g + g + \cdots + g)$. Ou seja, seria $g \cdot x \ (mod\ n)$. Mas dado $g \cdot x \ (mod\ n)$, como g é conhecido, bastaria dividir esse valor por g para recuperar x.

Exercício 4: *O protocolo Pretty Good Privacy (PGP) criado nos anos 90 e usado até hoje utiliza o esquema de certificação baseado em rede de confiança.*

Explique com suas palavras o que é um certificado digital e como funciona o modelo de rede de confiança.

Um certificado digital é um documento que garante identitifica o "dono" de uma chave pública. Tal documento consiste em um arquivo *assinado digitalmente* por uma autoridade certificadora associando a identidade do dono com sua chave pública. No modelo rede de confiança, *qualquer um pode emitir certificado* cabendo aos usuários estabelecer a confiança dessas múltiplas entidades certificadoras.