

GBC083 - Segurança da Informação

Aula 3 - Sigilo Computacional

Prof. Marcelo Keese Albertini

Criptografia simétrica

- ▶ Sigilo perfeito
 - ▶ Nenhuma informação sobre o texto original é aparente
 - ▶ Custo alto e de difícil implementação
- ▶ Sigilo computacional
 - ▶ Pode vazar informação com probabilidade **mínima** para um atacante com poder computacional **limitado**
 - ▶ Pode falhar com probabilidade mínima
 - ▶ Considera atacantes polinomiais

Consideração 1: probabilidade de falha mínima

- ▶ Podemos controlar tal probabilidade: exemplo 2^{-60}
- ▶ Isso é um problema?
 - ▶ Mais fácil atacante ganhar na loteria todos os dias de um ano do que observar uma falha
 - ▶ Algo que ocorre com probabilidade 2^{-60} por segundo é esperado acontecer uma vez a cada 100 bilhões de anos

Consideração 2: atacante com poder computacional limitado

- ▶ Força-bruta
 - ▶ Uma chave por ciclo de CPU
 - ▶ Computador típico: avalia $\approx 2^{50}$ chaves por ano
 - ▶ Supercomputador: avalia $\approx 2^{80}$ chaves por ano
 - ▶ Supercomputador: avalia $\approx 2^{112}$ chaves desde o Big Bang
 - ▶ Métodos modernos usam espaço com a partir de 2^{128} chaves

Sigilo perfeito e Indistinguibilidade perfeita

- ▶ $\Pi = (Gen, Enc, Dec)$
- ▶ Cenário de ataque:
 - ▶ Temos 2 mensagens m_0 e m_1
 - ▶ Uma é escolhida aleatoriamente e cifrada com chave uniforme k :
$$c \leftarrow Enc_k(m_b)$$
 - ▶ Atacante A intercepta c e tenta saber qual das mensagens foi cifrada
- ▶ Π é seguro se nenhum adversário A tem $p > 0.5$ de adivinhar corretamente a mensagem

Indistinguibilidade perfeita

- ▶ Temos esquema criptográfico $\Pi = (Gen, Enc, Dec)$ e A um atacante
- ▶ Cenário do experimento aleatório $PrivK_{A,\Pi}$:
 - ▶ A escolhe m_0 e $m_1 \in \mathbf{M}$

- ▶ Qual é a chance do atacante A ser bem sucedido?

$$P(A(c) = b) = P(PrivK_{A,\Pi} = 1) = ?$$

Indistinguibilidade perfeita

- ▶ Temos esquema criptográfico $\Pi = (Gen, Enc, Dec)$ e A um atacante
- ▶ Cenário do experimento aleatório $PrivK_{A,\Pi}$:
 - ▶ A escolhe m_0 e $m_1 \in \mathbf{M}$
 - ▶ Obter chave $k \leftarrow Gen$

- ▶ Qual é a chance do atacante A ser bem sucedido?

$$P(A(c) = b) = P(PrivK_{A,\Pi} = 1) = ?$$

Indistinguibilidade perfeita

- ▶ Temos esquema criptográfico $\Pi = (Gen, Enc, Dec)$ e A um atacante
- ▶ Cenário do experimento aleatório $PrivK_{A,\Pi}$:
 - ▶ A escolhe m_0 e $m_1 \in \mathbf{M}$
 - ▶ Obter chave $k \leftarrow Gen$
 - ▶ Vítima gera bit $b \leftarrow U\{0, 1\}$ que seleciona a mensagem usada

- ▶ Qual é a chance do atacante A ser bem sucedido?

$$P(A(c) = b) = P(PrivK_{A,\Pi} = 1) = ?$$

Indistinguibilidade perfeita

- ▶ Temos esquema criptográfico $\Pi = (Gen, Enc, Dec)$ e A um atacante
- ▶ Cenário do experimento aleatório $PrivK_{A,\Pi}$:
 - ▶ A escolhe m_0 e $m_1 \in \mathbf{M}$
 - ▶ Obter chave $k \leftarrow Gen$
 - ▶ Vítima gera bit $b \leftarrow U\{0, 1\}$ que seleciona a mensagem usada
 - ▶ Obter texto cifrado $c \leftarrow Enc_k(m_b)$
- ▶ Qual é a chance do atacante A ser bem sucedido?

$$P(A(c) = b) = P(PrivK_{A,\Pi} = 1) = ?$$

Indistinguibilidade perfeita

- ▶ Temos esquema criptográfico $\Pi = (Gen, Enc, Dec)$ e A um atacante
- ▶ Cenário do experimento aleatório $PrivK_{A,\Pi}$:
 - ▶ A escolhe m_0 e $m_1 \in \mathbf{M}$
 - ▶ Obter chave $k \leftarrow Gen$
 - ▶ Vítima gera bit $b \leftarrow U\{0, 1\}$ que seleciona a mensagem usada
 - ▶ Obter texto cifrado $c \leftarrow Enc_k(m_b)$
 - ▶ Atacante tenta descobrir qual mensagem foi usada: $b' \leftarrow A(c)$
- ▶ Qual é a chance do atacante A ser bem sucedido?

$$P(A(c) = b) = P(PrivK_{A,\Pi} = 1) = ?$$

Indistinguibilidade perfeita

- ▶ Temos esquema criptográfico $\Pi = (Gen, Enc, Dec)$ e A um atacante
- ▶ Cenário do experimento aleatório $PrivK_{A,\Pi}$:
 - ▶ A escolhe m_0 e $m_1 \in \mathbf{M}$
 - ▶ Obter chave $k \leftarrow Gen$
 - ▶ Vítima gera bit $b \leftarrow U\{0, 1\}$ que seleciona a mensagem usada
 - ▶ Obter texto cifrado $c \leftarrow Enc_k(m_b)$
 - ▶ Atacante tenta descobrir qual mensagem foi usada: $b' \leftarrow A(c)$
 - ▶ Se atacante é bem sucedido $b = b'$ experimento resulta em 1
- ▶ Qual é a chance do atacante A ser bem sucedido?

$$P(A(c) = b) = P(PrivK_{A,\Pi} = 1) = ?$$

Indistinguibilidade perfeita

- ▶ Atacante é facilmente bem sucedido com probabilidade 0.5 se sua tentativa for uniformemente aleatória
- ▶ Π é perfeitamente indistinguível se para todos atacantes **A** vale

$$P(\text{Priv}K_{A,\Pi} = 1) = 0.5$$

Indistinguibilidade perfeita

- ▶ Π é perfeitamente indistinguível se e só se é perfeitamente secreto (One-Time Pad)
- ▶ Ou seja, uma definição alternativa de sigilo perfeito
- ▶ Mas inclui a definição do atacante que será usada para flexibilizar a noção de segurança

Indistinguibilidade computacional

- ▶ Ideia: flexibilizar a indistinguibilidade perfeita e evitar problemas do OTP
- ▶ Duas abordagens
 - ▶ Segurança concreta
 - ▶ Segurança assintótica

Indistinguibilidade perfeita – versão concreta

- ▶ Π é (t, ϵ) -indistinguível se para todos atacantes A rodando em tempo até t vale que

$$P(\text{PrivK}_{A,\Pi} = 1) \leq 0.5 + \epsilon$$

- ▶ ϵ é uma folga para flexibilizar e evitar problemas do OTP
- ▶ Criptografia simétrica moderna

Vantagens da segurança concreta

- ▶ Parâmetros t , ϵ são fatores reais importantes
- ▶ Mas não resulta em uma teoria limpa
 - ▶ Dependente do modelo computacional escolhido
 - ▶ Qual máquina usada? Máquina, paralela, distribuída, quântica?
- ▶ É desejável ter esquemas em que usuários ajustem nível de segurança

Indistinguibilidade computacional concreta

- ▶ (t, ϵ) -indistinguibilidade:
 - ▶ Segurança pode falhar com probabilidade $\leq \epsilon$
 - ▶ Atenção restrita a atacantes rodando em tempo $\leq t$
- ▶ Exemplo
 - ▶ Nenhum atacante rodando até $t = 200$ anos pode ter sucesso em quebrar segurança com probabilidade maior que $\epsilon = 2^{-30}$

Exemplo

- ▶ Definição: Π é (t, ϵ) -indistinguível se para todos atacantes A rodando em tempo até t vale que

$$P(\text{PrivK}_{A,\Pi} = 1) \leq 0.5 + \epsilon$$

- ▶ Para dado parâmetro de segurança n , quanto tempo ele precisa atacar para obter uma vantagem ϵ de quanto?
 - ▶ Seja um adversário polinomial que ataca por n^3 minutos
 - ▶ Durante o ataque ele tem probabilidade negligível ϵ de quebrar igual a $2^{40} \cdot 2^{-n}$
 - ▶ E se $n = 50$?
 - ▶ E se $n = 500$?

Segurança assintótica

- ▶ Incluir parâmetro de segurança $n \in \mathbb{Z}^+$
 - ▶ Estabelecido pelas partes na inicialização
 - ▶ Permite usuário definir nível de segurança
 - ▶ Por agora, relacionar n com o comprimento da chave
 - ▶ Conhecido por atacante
- ▶ Avaliar tempos de execução das partes e a probabilidade de sucesso do atacante como função de n

Indistinguibilidade computacional assintótica

- ▶ Indistinguibilidade computacional
 - ▶ Segurança pode falhar com probabilidade negligível em n
 - ▶ Restringir atenção a atacantes rodando em tempo polinomial em n

Definições

- ▶ Uma função $f : Z^+ \rightarrow Z^+$ é polinomial (limitada por polinômios) se existe $\{c_i\}$ tal que $f(n) < \sum_i c_i n^i$ para todo $n > 0$
- ▶ Uma função $f : Z^+ \rightarrow R^{+,0}$ é negligível se para todo polinômio p existe um N tal que $f(n) < 1/p(n)$ para $n > N$
- ▶ Isto é. $f(n)$ é assintoticamente negligível se é assintoticamente menor que qualquer polinômio invertido
 - ▶ $f(n) = \text{poly}(n) \cdot 2^{-cn}$
 - ▶ $f(n) = 2^{-\sqrt{n}}$
 - ▶ $f(n) = n^{-\log n}$

Porque essas escolhas?

- ▶ Escolhas arbitrárias
- ▶ Eficiente = tempo polinomial probabilístico (TPP) visto em teoria de complexidade
- ▶ Propriedades convenientes
 - ▶ $\text{Poly} * \text{poly} = \text{poly}$
 - ▶ Número polinomial a subrotinas polinomiais continua polinomial
 - ▶ $\text{Poly} * \text{negligível} = \text{negligível}$
 - ▶ Número polinomial de chamadas a subrotina que falha com probabilidade negligível também tem probabilidade total negligível
 - ▶ $\text{negligível} + \text{negligível} = \text{negligível}$

Redefinindo criptografia

- ▶ Um esquema de criptografia de chave privada é definido por três algoritmos TPP (Gen, Enc, Dec):
 - ▶ Gen: recebe parâmetro de segurança unário 1^n e produz k (assumir $|k| \geq n$)
 - ▶ Enc: recebe chave k e mensagem $m \in \{0, 1\}^*$ e produz texto-cifrado $c \leftarrow Enc_k(m)$
 - ▶ Dec: recebe chave k e texto-cifrado c e produz mensagem m ou “erro”
- ▶ Um esquema é seguro se para todo atacante probabilístico \mathcal{A} de tempo polinomial $p(\cdot)$, existe um inteiro N tal que a probabilidade de sucesso de ataque é negligível e menor que $\frac{1}{p(n)}$ para todo $n > N$.

Indistinguibilidade computacional assintótica

- ▶ Escolher Π, A
- ▶ Montar experimento aleatório $PrivK_{A,\Pi}(n)$:
 - ▶ $A(1^n)$ produz $m_0, m_1 \in \{0, 1\}^*$ de mesmo comprimento
 - ▶ $k \leftarrow Gen(1^n), b \leftarrow \{0, 1\}, c \leftarrow Enc_k(m_b)$
 - ▶ $b' \leftarrow A(c)$ e A é bem sucedido se $b = b'$ e experimento resulta em 1 neste caso

Indistinguibilidade computacional assintótica

- ▶ Π é indistinguível se para todo atacante TPP A existe uma função negligível ϵ tal que:

$$P(\text{PrivK}_{A,\Pi}(n) = 1) \leq 0.5 + \epsilon(n)$$

Exemplo

- ▶ Considere um esquema em que o melhor ataque é por força-bruta sobre o espaço de chaves e $Gen(1^n)$ gera uma chave uniforme de n bits
 - ▶ Então se A tem tempo $t(n)$ então
 - ▶ $P(\text{Priv}_{K_{A,n}}(n) = 1) = 0.5 + t(n)/2^n$
- ▶ Este esquema é indistinguível: para qualquer polinômio t a função $t(n)/2^n$ é negligível

Próximos assuntos

- ▶ Geração de números aleatórios e pseudo-aleatórios
- ▶ Pseudo One-Time Pad