

GBC083 - Segurança da Informação

Aula 2 - Sigilo Perfeito

28 de Março de 2016

Encriptação segura

- ▶ Meta: “Independentemente de qualquer informação prévia que o atacante tem sobre o texto em claro, o texto cifrado não pode vaziar nenhuma informação adicional sobre o texto em claro
- ▶ Cenário: ataque usando somente um texto cifrado

Revisão sobre probabilidades

- ▶ Variável aleatória – variável que assume valores discretos de acordo com probabilidades
- ▶ Distribuição de probabilidades para uma variável aleatória – especifica as probabilidades com qual a variável assume cada possível valor
 - ▶ Cada probabilidade deve ser entre 0 e 1
 - ▶ As probabilidades somam 1

Revisão sobre probabilidades

- ▶ **Evento** - uma ocorrência específica em algum experimento
 - ▶ $P(E)$ - probabilidade do evento E
- ▶ Probabilidade condicional - probabilidade com que um evento ocorre, assumindo que algum outro ocorreu
 - ▶ $P(A|B) = P(A \text{ e } B)/P(B)$
- ▶ Duas variáveis aleatórias X e Y são independentes se para todo x, y , temos $P(X = x|Y = y) = P(X = x)$

Revisão sobre probabilidades

- ▶ Lei da probabilidade total - digamos E_1, \dots, E_n são uma partição de todas as possibilidades.
- ▶ Então para qualquer A :
 - ▶ $P(A) = \sum_i P(A \text{ e } E_i) = \sum_i P(A|E_i) \cdot P(E_i)$

Criptografia de chave privada

- ▶ Espaço de mensagens \mathbf{M} e algoritmos (Gen, Enc, Dec):
 - ▶ Gen (algoritmo de geração de chaves): gera k
 - ▶ Enc (algoritmo de encriptação): recebe chave k e mensagem $m \in \mathbf{M}$ como entrada; produz texto cifrado c

$$c \leftarrow Enc_k(m_0)$$

- ▶ Dec (algoritmo de descrição): recebe chave k e texto cifrado c como entrada; produz

$$m \leftarrow Dec_k(c)$$

Notação

- ▶ **K** espaço das chaves – conjunto de todas as possíveis chaves
- ▶ **C** espaço de texto cifrado – conjunto de todos os textos cifrados

Distribuições de probabilidades

- ▶ Seja M uma variável aleatória denotando o valor da mensagem
 - ▶ M varia de acordo com \mathbf{M}
 - ▶ Isto reflete a chance de diferentes mensagens sendo enviadas pelas partes, considerando o conhecimento prévio do atacante
 - ▶ Exemplo
 - ▶ $P(M = \text{atacar hoje}) = 0.7$
 - ▶ $P(M = \text{não atacar}) = 0.3$

Distribuições de probabilidades

- ▶ Seja K uma variável aleatória denotando a chave
 - ▶ K varia no espaço de chaves \mathbf{K}
- ▶ Escolher algum esquema de encriptação (Gen, Enc, Dec)
 - ▶ Gen define uma distribuição de probabilidade para K :
 - ▶ $P(K = k) = P(Gen \text{ produz chave } k)$

Distribuições de probabilidades

- ▶ Variáveis aleatórias M e K são independentes
 - ▶ a mensagem que a parte envia não depende da chave usado para encriptar aquela mensagem

Distribuições de probabilidades

- ▶ Escolher um esquema de encriptação (Gen, Enc, Dec) e alguma distribuição para M
- ▶ Considere o seguinte experimento aleatório:
 - ▶ Escolha uma mensagem m a partir de M
 - ▶ Gerar uma chave k usando Gen
 - ▶ Computar $c \leftarrow Enc_k(m)$
- ▶ Isto define uma distribuição no texto cifrado
- ▶ Seja C uma variável aleatória denotando o texto cifrado neste experimento aleatório

Exemplo 1

- ▶ Considere a cifra de deslocamento
 - ▶ Então para toda $k \in \{0, \dots, 25\}$, $P(K = k) = 1/26$
- ▶ Supondo $P(M = a) = 0.7$, $P(M = z) = 0.3$
- ▶ Quanto é $P(C = b)$?
 - ▶ Ou $M = 'a'$ e $K = 1$, ou $M = 'z'$ e $K = 2$
 - ▶ $P(C = b) = P(M = a) \cdot P(K = 1) + P(M = z) \cdot P(K = 2)$
 - ▶ $= 0.7 \cdot 1/26 + 0.3 \cdot 1/26 = 1/26$

Exemplo 2

- ▶ Considere a cifra de deslocamento e a distribuição
 $P(M = \text{sim}) = 0.5$, $P(M = \text{não}) = 0.5$
- ▶ $P(C = \text{xnr}) = ?$
- ▶ $= P(C = \text{xnr} | M = \text{sim}) \cdot P(M = \text{sim}) + P(C = \text{xnr} | M = \text{não}) \cdot P(m = \text{não})$
- ▶ $1/26 \cdot 0.5 + 0 \cdot 0.5 = 1/52$

Sigilo perfeito (informal)

- ▶ “Independentemente de qualquer conhecimento prévio que o atacante tem sobre o texto em claro, o texto cifrado não deve vaziar nenhuma informação sobre o texto em claro”
 - ▶ Cenário: ataque usando somente um texto cifrado

Sigilo perfeito (informal)

- ▶ Informação do atacante sobre o texto em claro = distribuição do texto em claro conhecida pelo atacante
 - ▶ Sigilo perfeito significa que observando o texto cifrado não deve mudar o conhecimento do atacante sobre a distribuição do texto em claro

Sigilo perfeito (formal)

- ▶ Esquema de encriptação (Gen, Enc, Dec) com espaço de mensagens \mathbf{M} e espaço de texto cifrado \mathbf{C} é perfeitamente secreto se para toda distribuição sobre \mathbf{M} , todo $m \in \mathbf{M}$ e todo $c \in \mathbf{C}$ com $P(C = c) > 0$ vale que

$$P(M = m|C = c) = P(M = m)$$

Exemplo 3

- ▶ Considere uma cifra de deslocamento e a distribuição
 $P(M = \textit{sim}) = 0.5, P(M = \textit{nao}) = 0.5$
- ▶ Faça $m = \textit{nao}$ e $c = \textit{xnr}$
- ▶ $P(M = \textit{nao} | C = \textit{xnr}) = 0 \neq P(M = \textit{nao})$

Teorema de Bayes

- ▶ $P(A|B) = P(B|A) \cdot P(A)/P(B)$
- ▶ Por que isso é verdade?
- ▶ Como usar?

Exemplo 4

- ▶ Cifra de deslocamento
 - ▶ $P(M = hi) = 0.3$
 - ▶ $P(M = no) = 0.2$
 - ▶ $P(M = in) = 0.5$
- ▶ $P(M = hi) = P(M = hi|C = xy)?$
- ▶ $P(M = hi|C = xy) = ?$

Exemplo 4

- ▶ Cifra de deslocamento
 - ▶ $P(M = hi) = 0.3$
 - ▶ $P(M = no) = 0.2$
 - ▶ $P(M = in) = 0.5$
- ▶ $P(M = hi) = P(M = hi|C = xy)$?
- ▶ $P(M = hi|C = xy) = ?$ Usar teorema de Bayes

Exemplo 4

- ▶ Cifra de deslocamento
 - ▶ $P(M = hi) = 0.3$
 - ▶ $P(M = no) = 0.2$
 - ▶ $P(M = in) = 0.5$
- ▶ $P(M = hi) = P(M = hi|C = xy)$?
- ▶ $P(M = hi|C = xy) = ?$ Usar teorema de Bayes
- ▶ $= P(C = xy|M = hi) \cdot P(M = hi)/P(C = xy)$
- ▶ Resolver cada parte

Exemplo 4 continuado

- ▶ $P(C = xy | M = hi) = 1/26$

Exemplo 4 continuado

- ▶ $P(C = xy | M = hi) = 1/26$ porque só tem uma chave que pode transformar hi em xy

Exemplo 4 continuado

- ▶ $P(C = xy | M = hi) = 1/26$ porque só tem uma chave que pode transformar hi em xy
- ▶ $P(C = xy) = ?$ usar lei da probabilidade total

Exemplo 4 continuado

- ▶ $P(C = xy|M = hi) = 1/26$ porque só tem uma chave que pode transformar hi em xy
- ▶ $P(C = xy) = ?$ usar lei da probabilidade total
- ▶ $P(C = xy) = P(C = xy|M = hi) \cdot 0.3 + P(C = xy|M = no) \cdot 0.2 + P(C = xy|M = in) \cdot 0.5$
- ▶ $= 1/26 \cdot 0.3 + 1/26 \cdot 0.2 + 0 \cdot 0.5 = 1/52$

Exemplo 4 continuado

- ▶ $P(M = hi|C = xy) = ?$
- ▶ $P(C = xy|M = hi) \cdot P(M = hi)/P(C = xy)$
- ▶ $1/26 \cdot 0.3/(1/52) = 0.6 \neq P(M = hi)$
- ▶ Portanto a cifra de deslocamento **não** é perfeitamente secreta!
- ▶ Como construir um esquema perfeitamente secreto?

One-Time Pad – Cifra de uso único

- ▶ Esquema de encriptação (Gen, Enc, Dec) com espaço de mensagens \mathbf{M} e espaço de texto cifrado \mathbf{C} é perfeitamente secreto se para toda distribuição de probabilidade sobre \mathbf{M} , todo $m \in \mathbf{M}$ e todo $c \in \mathbf{C}$ com $P(C = c) > 0$, vale



$$P(M = m|C = c) = P(M = m)$$

One-Time Pad

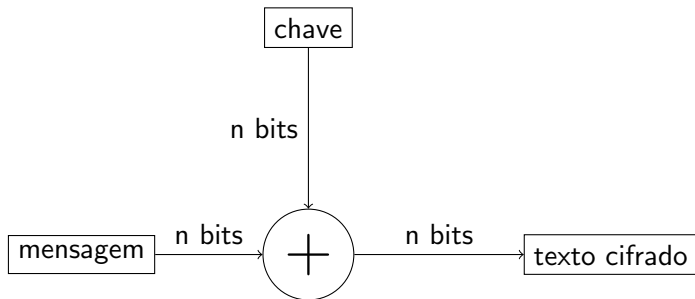
- ▶ Patentado em 1917 por Vernam
 - ▶ Pesquisas históricas indicam que foi inventado pelo menos 35 anos antes que isso
- ▶ Provado como **perfeitamente secreto** por Shannon em 1949

One-Time Pad

- ▶ Seja $\mathbf{M} = \{0, 1\}^n$
- ▶ Gen: escolher chave uniforme $k \in \{0, 1\}^n$
- ▶ $Enc_k(m) = k \oplus m$, sendo \oplus a operação xor bit-a-bit
- ▶ $Dec_k(c) = k \oplus c$
- ▶ Corretude: $Dec_k(Enc_k(m)) = k \oplus (k \oplus m) = (k \oplus k) \oplus m = m$

```
#include <stdio.h>
int main() {
    int a = 25 ^ 10;
    printf("a ^= %d\n", a);
    printf("a^10 ^= %d\n", a^10);
    printf("a^25 ^= %d\n", a^25);
}
```

One-Time Pad



Implementação da cifra OTP

- ▶ Seja $\mathbf{M} = \{0, 1\}^n$
- ▶ *Gen*: escolher chave uniforme $k \in \{0, 1\}^n$
- ▶ $Enc_k(m) = k \oplus m$, XOR bit-a-bit
- ▶ $Dec_k(c) = k \oplus c$

Representação dos dados

- ▶ Texto limpo – em ASCII
- ▶ Chave – em hexa escrito em ASCII
- ▶ Texto cifrado – em hexa escrito em ASCII

Sigilo perfeito do One-Time Pad

- ▶ Determinar distribuição arbitrária sobre $\mathbf{M} = \{0, 1\}^n$ e um m arbitrário, $c \in \{0, 1\}^n$
- ▶ $P(M = m|C = c) = ?$
 - ▶ $= P(C = c|M = c) \cdot P(M = m)/P(C = c)$
- ▶ $P(C = c)$
 - ▶ $\sum_{m'} P(C = c|M = m') \cdot P(M = m')$

Sigilo perfeito do One-Time Pad

- ▶ Determinar distribuição arbitrária sobre $\mathbf{M} = \{0, 1\}^n$ e um m arbitrário, $c \in \{0, 1\}^n$
- ▶ $P(M = m|C = c) = ?$
 - ▶ $= P(C = c|M = c) \cdot P(M = m)/P(C = c)$
- ▶ $P(C = c)$
 - ▶ $\sum_{m'} P(C = c|M = m') \cdot P(M = m')$
 - ▶ $\sum_{m'} P(k = m' \oplus c) \cdot P(M = m')$

Sigilo perfeito do One-Time Pad

- ▶ Determinar distribuição arbitrária sobre $\mathbf{M} = \{0, 1\}^n$ e um m arbitrário, $c \in \{0, 1\}^n$
- ▶ $P(M = m|C = c) = ?$
 - ▶ $= P(C = c|M = c) \cdot P(M = m)/P(C = c)$
- ▶ $P(C = c)$
 - ▶ $\sum_{m'} P(C = c|M = m') \cdot P(M = m')$
 - ▶ $\sum_{m'} P(k = m' \oplus c) \cdot P(M = m')$
 - ▶ $\sum_{m'} 2^{-n} \cdot P(M = m')$

Sigilo perfeito do One-Time Pad

- ▶ Determinar distribuição arbitrária sobre $\mathbf{M} = \{0, 1\}^n$ e um m arbitrário, $c \in \{0, 1\}^n$
- ▶ $P(M = m|C = c) = ?$
 - ▶ $= P(C = c|M = c) \cdot P(M = m)/P(C = c)$
- ▶ $P(C = c)$
 - ▶ $\sum_{m'} P(C = c|M = m') \cdot P(M = m')$
 - ▶ $\sum_{m'} P(k = m' \oplus c) \cdot P(M = m')$
 - ▶ $\sum_{m'} 2^{-n} \cdot P(M = m')$
 - ▶ $= 2^{-n}$

Sigilo perfeito do One-Time Pad

- ▶ Determinar distribuição arbitrária sobre $\mathbf{M} = \{0, 1\}^n$ e um m arbitrário, $c \in \{0, 1\}^n$
- ▶ $P(M = m|C = c) = ?$
 - ▶ $= P(C = c|M = m) \cdot P(M = m)/P(C = c)$
 - ▶ $= P(k = m \oplus c) \cdot P(M = m)/2^{-n}$
 - ▶ $= 2^{-n} \cdot P(M = m)/2^{-n}$
 - ▶ $= P(M = m)$

Encriptação One-Time Pad (OTP)

- ▶ Temos
 - ▶ Texto em claro= sequência de caracteres ASCII
 - ▶ Chave = sequência de dígitos em hexa, escritos em ASCII
- ▶ Encriptação
 - ▶ Ler os dois, aplicar XOR entre eles e obter o texto cifrado

Deciptação OTP

- ▶ Reverter encriptação
- ▶ Ler texto cifrado e chave; aplicar XOR para obter mensagem original

Otimidade de tamanho mínimo de chave do OTP

- ▶ Teorema: se $(\text{Gen}, \text{Enc}, \text{Dec})$ com mensagem \mathbf{M} é perfeitamente secreto então $|\mathbf{K}| \geq |\mathbf{M}|$
- ▶ Intuição:
 - ▶ Dado texto cifrado, tentar decriptar com cada chave em \mathbf{K}
 - ▶ Lista de até $|\mathbf{K}|$ possíveis mensagens
 - ▶ Se $|\mathbf{K}| < |\mathbf{M}|$ alguma mensagem não está na lista
 - ▶ Vazamento de informação sobre o que é mensagem legítima ou não

Prova: Otimalidade de OTP

- ▶ Teorema

- ▶ se $(\text{Gen}, \text{Enc}, \text{Dec})$ no espaço \mathbf{M} é perfeitamente secreto então $|\mathbf{K}| \geq |\mathbf{M}|$

- ▶ Prova

- ▶ Assumir $|\mathbf{K}| < |\mathbf{M}|$
 - ▶ Mostrar que existe distribuição em \mathbf{M} , mensagem m e texto cifrado c tal que
 - ▶ $P(M = m|C = c) \neq P(M = m)$

Continuação da Prova: Otimidade de OTP

- ▶ Continuação da Prova:
 - ▶ Usar distribuição uniforme em \mathbf{M}
 - ▶ Usar qualquer texto cifrado c
 - ▶ Considerar o conjunto $M(c) = \{Dec_k(c)\}_{k \in \mathbf{K}}$
 - ▶ Essas são as únicas mensagens possíveis que gerariam o texto cifrado c
 - ▶ $|M(c)| \leq |\mathbf{K}| < |\mathbf{M}|$ então existe algum m que não está em $M(c)$
 - ▶ $P(M = m | C = c) = 0 \neq P(M = m)$
 - ▶ Então não é uniforme.

Limitações One-Time Pad

- ▶ Limitações
 - ▶ Chave do tamanho da mensagem
 - ▶ Somente segura se chave for usada só uma vez
- ▶ Limitações inerentes a qualquer método seguro como o One-Time Pad

Ataque ao OTP com reuso de chave

- ▶ Explorando tabela ASCII
 - ▶ Letras começam com 01
 - ▶ Caractere espaço inicia com 00
 - ▶ XOR de duas letras resultam em 00
 - ▶ XOR de letra e espaço resulta 01
 - ▶ Embora espaço XOR espaço também
 - ▶ Pontuação pode complicar
 - ▶ É fácil identificar XOR de entre letra e espaço

Exemplo - OTP com mesma chave

Cifra 1: 10 | 01 | 01 | 11

Cifra 2: 10 | 01 | 01 | 10

$1 \oplus 2$: 00 | 00 | 00 | 01010000

Como o caractere espaço é em hexa 0x20 e em binário 0010 0000,
a letra é ...

Exemplo - OTP com mesma chave

Cifra 1: 10 | 01 | 01 | 11

Cifra 2: 10 | 01 | 01 | 10

$1 \oplus 2$: 00 | 00 | 00 | 01010000

Como o caractere espaço é em hexa 0x20 e em binário 0010 0000,
a letra é . . . **p**.

Conclusões sobre OTP

- ▶ Definiu-se o que é perfeitamente secreto
- ▶ OTP é perfeitamente secreto
- ▶ OTP usa a menor chave possível para ser perfeitamente secreto

Próximos passos

- ▶ Relaxar definição de segurança
- ▶ Utilizar chave menor