Criptografia com Números Irracionais

Foz-2006

Fábio Borges

LNCC – Laboratório Nacional de Computação Científica

Ataque

- $\mathbb{M} = \{M_1, \ldots, M_n\}$

Ataque

- $\mathbb{M} = \{M_1, \ldots, M_n\}$
- $P(M_1), \ldots, P(M_n)$

Ataque

- $\mathbb{M} = \{M_1, \ldots, M_n\}$
- $P(M_1), \ldots, P(M_n)$
- $\mathbb{E} = \{E_1, \ldots, E_n\}$

$$E = T_i M$$

Ataque

- $\mathbb{M} = \{M_1, \ldots, M_n\}$
- $P(M_1), \ldots, P(M_n)$
- $\mathbb{E} = \{E_1, \ldots, E_n\}$

$$E = T_i M$$

- Criptoanalista intercepta E

$$P_E(M)$$

Definições

- Definimos *Segredo Perfeito* pela condição

$$P_E(M) = P(M)$$

para todo $M \in \mathbb{M}$ e todo $E \in \mathbb{E}$

Definições

- Definimos *Segredo Perfeito* pela condição

$$P_E(M) = P(M)$$

para todo $M \in \mathbb{M}$ e todo $E \in \mathbb{E}$

- Definimos *one-time-pad* como um tipo de algoritmo cuja chave é maior ou igual à mensagem e só pode ser usada uma vez, isto é, existe uma relação biunívoca entre as mensagens e as chaves

Teo. One-Time-Pad

Teorema: One-time-pad é um segredo perfeito.

Teo. One-Time-Pad

Teorema: One-time-pad é um segredo perfeito.
Prova: Considere um alfabeto com n símbolos e

$$TM = E$$

Teo. One-Time-Pad

Teorema: One-time-pad é um segredo perfeito.
Prova: Considere um alfabeto com n símbolos e

$$TM = E$$

então

$$P(M_i) = \frac{1}{n} \; \forall i$$

Teo. One-Time-Pad

Teorema: One-time-pad é um segredo perfeito.
Prova: Considere um alfabeto com n símbolos e

$$TM = E$$

então

$$P_E(M_i) = \frac{1}{n} \; \forall i$$

Teo. One-Time-Pad

Teorema: One-time-pad é um segredo perfeito.
Prova: Considere um alfabeto com n símbolos e

$$TM = E$$

Portanto

$$P(M) = P_E(M)$$

Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅

Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅
- 00,20,15,25,14,09,13,03,05,25,22,19,00,05,00

Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅
- 00,20,15,25,14,09,13,03,05,25,22,19,00,05,00
- A∅MENINA∅BRINCA

Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅
- 00,20,15,25,14,09,13,03,05,25,22,19,00,05,00
- A∅MENINA∅BRINCA
- 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01

Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅
- 00,20,15,25,14,09,13,03,05,25,22,19,00,05,00
- A∅MENINA∅BRINCA
- 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
- 01,07,25,07,00,00,01,25,22,04,23,17,14,25,01

Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅
- 00,20,15,25,14,09,13,03,05,25,22,19,00,05,00
- A∅MENINA∅BRINCA
- 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
- 01,07,25,07,00,00,01,25,22,04,23,17,14,25,01
- ATACAR∅DE∅MANHA

Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅
- 00,20,15,25,14,09,13,03,05,25,22,19,00,05,00
- A∅MENINA∅BRINCA
- 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
- 01,07,25,07,00,00,01,25,22,04,23,17,14,25,01
- ATACAR∅DE∅MANHA
- 01,20,01,03,01,18,00,04,05,00,13,01,14,08,01

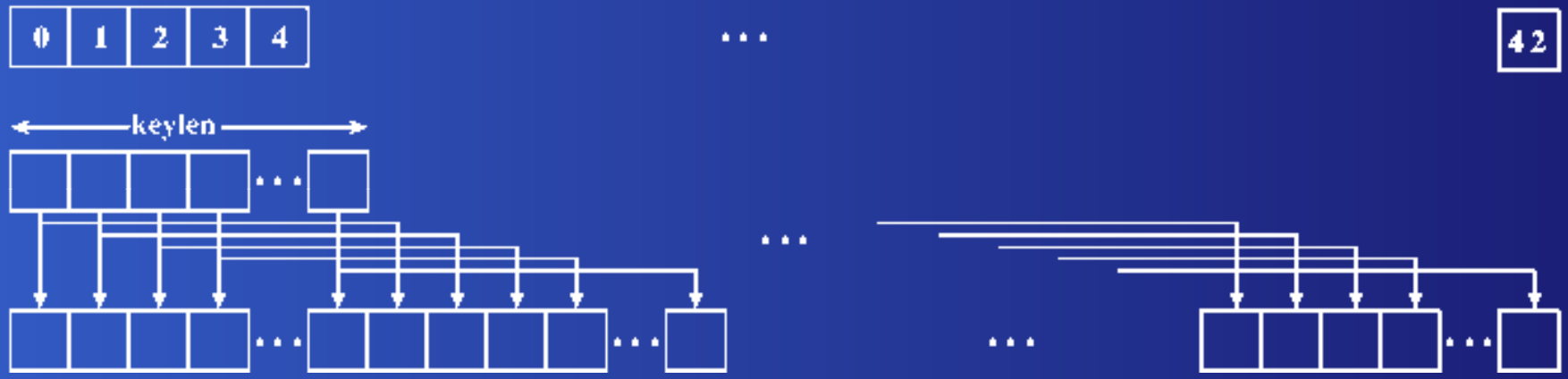
Vigenère-Vernam

- ∅TOYNIMCEYVS∅E∅
- 00,20,15,25,14,09,13,03,05,25,22,19,00,05,00
- A∅MENINA∅BRINCA
- 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
- 01,07,25,07,00,00,01,25,22,04,23,17,14,25,01
- ATACAR∅DE∅MANHA
- 01,20,01,03,01,18,00,04,05,00,13,01,14,08,01
- 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01

Possibilidades

- ESTUDANDO$\not\emptyset$MUITO
- CADEIRA$\not\emptyset$AMARELA
- RENATO$\not\emptyset$PORTUGAL
- IR$\not\emptyset$EMBORA$\not\emptyset$AGORA
- EU$\not\emptyset$TO$\not\emptyset$COM$\not\emptyset$FOME$\not\emptyset$
- EU$\not\emptyset$AMO$\not\emptyset$O$\not\emptyset$FABIO$\not\emptyset$
- O$\not\emptyset$SAPATO$\not\emptyset$FURADO
- TERMINAR$\not\emptyset$A$\not\emptyset$AULA

Esquema Vigenère



Vigenère

- Gerando uma chave do tamanho do texto, a partir de uma chave menor (Keystream)

Vigenère

- Gerando uma chave do tamanho do texto, a partir de uma chave menor (Keystream)
  - A␣MENINA␣BRINCA

Vigenère

- Gerando uma chave do tamanho do texto, a partir de uma chave menor (Keystream)
  - A␣MENINA␣BRINCA
  - 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01

Vigenère

- Gerando uma chave do tamanho do texto, a partir de uma chave menor (Keystream)
  - A␣MENINA␣BRINCA
  - 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
  - SENHASENHASENHA

Vigenère

- Gerando uma chave do tamanho do texto, a partir de uma chave menor (Keystream)
  - A␣MENINA␣BRINCA
  - 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
  - SENHASENHASENHA
  - 19,05,14,08,01,19,05,14,08,01,19,05,14,08,01

Vigenère

- Gerando uma chave do tamanho do texto, a partir de uma chave menor (Keystream)
  - A␣MENINA␣BRINCA
  - 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
  - SENHASENHASENHA
  - 19,05,14,08,01,19,05,14,08,01,19,05,14,08,01
  - 20,05,00,13,15,01,19,15,08,03,10,14,01,11,02

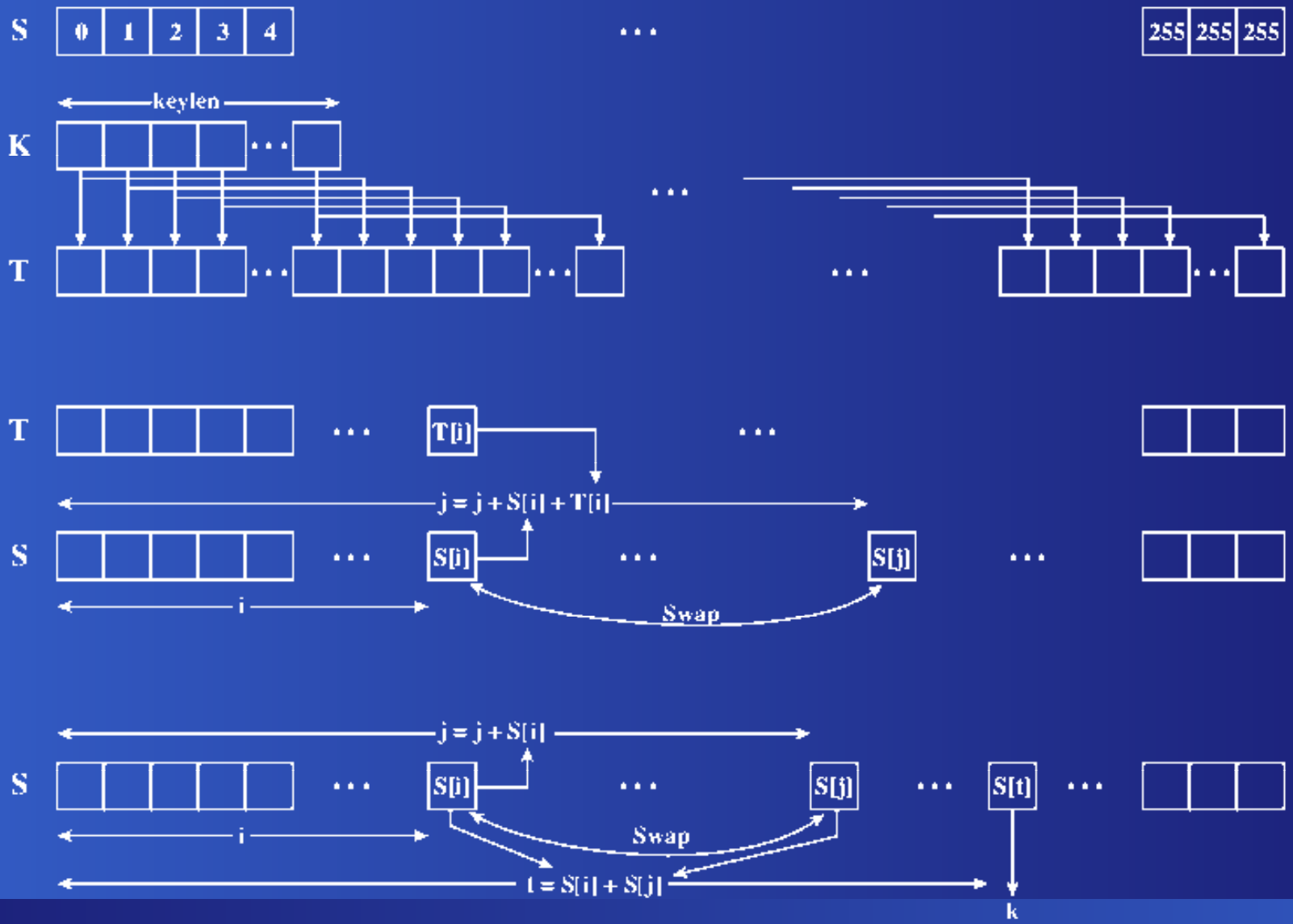
Vigenère

- Gerando uma chave do tamanho do texto, a partir de uma chave menor (Keystream)
  - A␣MENINA␣BRINCA
  - 01,00,13,05,14,09,14,01,00,02,18,09,14,03,01
  - SENHASENHASENHA
  - 19,05,14,08,01,19,05,14,08,01,19,05,14,08,01
  - 20,05,00,13,15,01,19,15,08,03,10,14,01,11,02
  - TE␣MOASOHCJNAKB

Desempenho

Cifra	Comprimento da chave	Mbps
DES	56	9
3DES	168	3
RC2	Variável	0,9
RC4	Variável	45

RC4


S
0
1
2
3
4
255
255
255
keylen
K
T
T
T[i]
j = j + S[i] + T[i]
S
S[i]
S[j]
i
Swap
j = j + S[i]
S
S[i]
S[j]
S[t]
i
Swap
t = S[i] + S[j]
k

Grau de Segurança

Algoritmos	Segurança
assimétricos	computacional
simétricos	probabilística
segredo perfeito	matemática

Questões

- existe algum outro algoritmo que seja um segredo perfeito sem ser one-time-pad?

Questões

- existe algum outro algoritmo que seja um segredo perfeito sem ser one-time-pad?
  - parece que sim

Questões

- existe algum outro algoritmo que seja um segredo perfeito sem ser one-time-pad?
  - parece que sim
- por que encontra-lo?

Questões

- existe algum outro algoritmo que seja um segredo perfeito sem ser one-time-pad?
  - parece que sim
- por que encontra-lo?
  - com a chave menor que a mensagem podemos combinar uma nova chave em cada mensagem

Questões

- existe algum outro algoritmo que seja um segredo perfeito sem ser one-time-pad?
  - parece que sim
- por que encontra-lo?
  - com a chave menor que a mensagem podemos combinar uma nova chave em cada mensagem
  - entender melhor a segurança dos algoritmos

Teo. Chaves

Teorema: Dado uma mensagem M fixa e uma chave K, se $M_i, K_j \in |\mathcal{A}| \; \forall M_i, K_j$ e $E = T_k M$ então one-time-pad é o único segredo perfeito.

Prova: Temos que T_k é uma transformação biunívoca, como $|K| < |M|$ temos criptogramas que não são gerados por T_k.

Solução

- $|\mathcal{A}_M| < |\mathcal{A}_K|$

Solução

- $|\mathcal{A}_M| < |\mathcal{A}_K|$
  - *impraticável*

Solução

- $|\mathcal{A}_M| < |\mathcal{A}_K|$
  - *impraticável*
- atribuir uma semântica à chave
  - *novo paradigma*
  - *como?*

Solução

- $|\mathcal{A}_M| < |\mathcal{A}_K|$
  - *impraticável*
- atribuir uma semântica à chave
  - *novo paradigma*
  - *expressões matemáticas*

Solução

- $|\mathcal{A}_M| < |\mathcal{A}_K|$
  - *impraticável*
- atribuir uma semântica à chave
  - *novo paradigma*
  - *expressões matemáticas*
- transferência de custo do tamanho da chave para um custo computacional

Falta algo?

- gerar todas as seqüências do tamanho da mensagem

Falta algo?

- gerar todas as seqüências do tamanho da mensagem
  - $\frac{a}{b}\sqrt[r]{p_1 \cdots p_n}$

Falta algo?

- gerar todas as seqüências do tamanho da mensagem
  - $\frac{a}{b}\sqrt[r]{p_1 \cdots p_n}$
- todas devem ser equiprováveis

Falta algo?

- gerar todas as seqüências do tamanho da mensagem
  - $\frac{a}{b}\sqrt[r]{p_1 \cdots p_n}$
- todas devem ser equiprováveis
  - irracionais não têm ciclos

Falta algo?

- gerar todas as seqüências do tamanho da mensagem
  - $\frac{a}{b}\sqrt[r]{p_1 \cdots p_n}$
- todas devem ser equiprováveis
  - irracionais não têm ciclos
  - irracionais são não enumeráveis

Falta algo?

- gerar todas as seqüências do tamanho da mensagem
  - $\frac{a}{b}\sqrt[r]{p_1 \cdots p_n}$
- todas devem ser equiprováveis
  - irracionais não têm ciclos
  - irracionais são não enumeráveis
  - raízes quadradas não inteiras são normais na base 2

Algoritmo

Recebe uma mensagem m
Recebe uma chave $r, a, b, e_1, \ldots, e_n$
$p_1 = próximo\_primo(e_1)$
$\vdots$
$p_n = próximo\_primo(e_n)$
$I = \frac{a}{b}\sqrt[r]{p_1 \cdots p_n}$
k recebe $|m|$ casas decimais da mantissa de I
Para $i := 1$ até $|m|$
$C[i] = k[i] \oplus m[i]$
Retorne C

Teo. Aproximação

Teorema: Se $\sqrt[r]{p_{m+1}} - \sqrt[r]{p_m} < 1$, com p_m e p_{m+1} primos consecutivos, então todo número pode ser aproximado através da raiz de um produto de primos.

Prova: Seja k o número que desejamos aproximar, então $k^r = p_1 \dots p_n(f_1 \dots f_s)$, onde f_i são fatores primos com potências maiores que um. Seja p_m o maior primo menor que $f_1 \dots f_s$

$$p_m < f_1 \dots f_s < p_{m+1}$$

$$\sqrt[r]{f_1 \dots f_s} - \sqrt[r]{p_m} < 1.$$

Dificuldade

A equação da hipótese do teorema anterior é uma generalização da conjectura de Andrica, isto é,

$$\sqrt{p_{m+1}} - \sqrt{p_m} < 1.$$

Neste caso, o espaço de busca das chaves é maior que a mensagem, além de termos todas as combinações de criptograma equiprováveis, assim se a conjectura de Andrica for satisfeita temos um segredo perfeito diferente do one-time-pad.

Exemplo

Podemos passar $pr\acute{o}ximo\_primo(5^{604})$, tal número tem 423 dígitos decimais, isto é, 1403 bits versus 40 bits.

Último Slide

- Obrigado.
- Quaisquer sugestões serão bem-vindas.

www.lncc.br/borges

Fábio Borges de Oliveira