Криптографија
- трећи део -

доц. др Драган Ђокић

Математички факултет, Универзитет у Београду

dragan.djokic@matf.bg.ac.rs

15. март 2024.

Литература

- Neal Koblitz: *A course in number theory and criptography*, 2nd edition, Springer-Verlag, 1994.
- William Stein: *Elementary number theory: primes, conguences, and secrets*, 2017.
- Миодраг Живковић: *Криптографија*, 2020.

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат
  - Слање шифрата

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат
  - Слање шифрата
  - Декриптовање (дешифровање)

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат
  - Слање шифрата
  - Декриптовање (дешифровање)
  - Декодирање

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат
  - Слање шифрата
  - Декриптовање (дешифровање)
  - Декодирање
- У (де)кодирању нема ничег тајног.

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат
  - Слање шифрата
  - Декриптовање (дешифровање)
  - Декодирање
- У (де)кодирању нема ничег тајног.
- Криптосистем је пар: Алгоритам за криптовање и алгоритам за декриптовање

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат
  - Слање шифрата
  - Декриптовање (дешифровање)
  - Декодирање
- У (де)кодирању нема ничег тајног.
- Криптосистем је пар: Алгоритам за криптовање и алгоритам за декриптовање
  - Алгоритми су најчешће свима познати

- Слање поруке обухвата:
  - Кодирање - трансформише поруку (отворени текст, слика,...) у низ цифара или битова
  - Криптовање (шифровање) - трансформише кодирану поруку у шифрат
  - Слање шифрата
  - Декриптовање (дешифровање)
  - Декодирање
- У (де)кодирању нема ничег тајног.
- Криптосистем је пар: Алгоритам за криптовање и алгоритам за декриптовање
  - Алгоритми су најчешће свима познати
  - Увек зависе од параметра који се зове кључ, и који се чува у тајности (потпуно или делимично)

- Обично:
  - Алиса шаље поруку Бобану

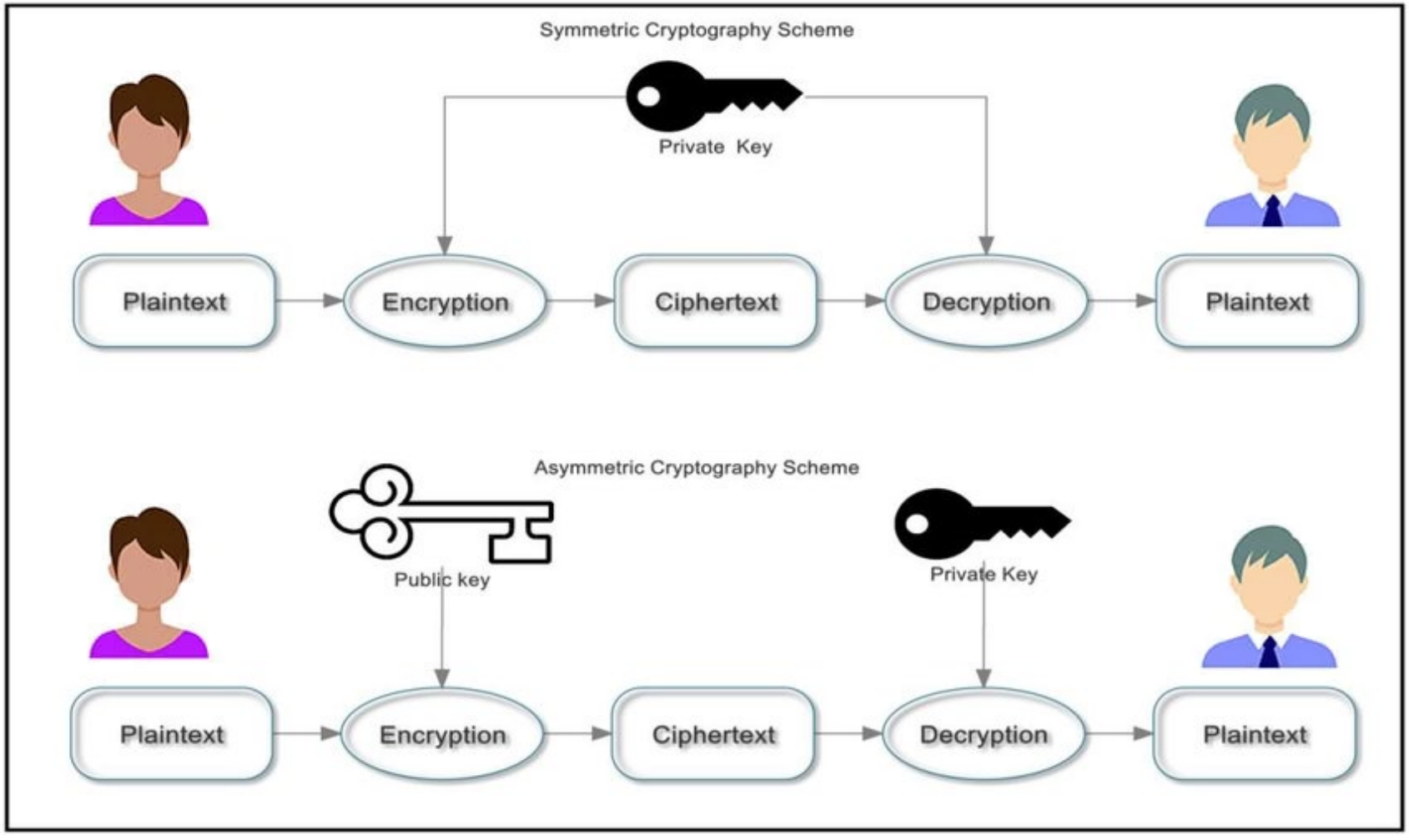
- Обично:
  - Алиса шаље поруку Бобану
  - Цица краде шифрат и покушава да га декриптује не знајући кључ (криптоанализа)

- Обично:
  - Алиса шаље поруку Бобану
  - Цица краде шифрат и покушава да га декриптује не знајући кључ (криптоанализа)
- Врсте шифре:
  - Проточна - трансформише симбол по симбол или бит по бит

- Обично:
  - Алиса шаље поруку Бобану
  - Цица краде шифрат и покушава да га декриптује не знајући кључ (криптоанализа)
- Врсте шифре:
  - Проточна - трансформише симбол по симбол или бит по бит
  - Блоковска - групише симболе у биграфе, триграфе,... и њих трансформише

- Обично:
  - Алиса шаље поруку Бобану
  - Цица краде шифрат и покушава да га декриптује не знајући кључ (криптоанализа)
- Врсте шифре:
  - Проточна - трансформише симбол по симбол или бит по бит
  - Блоковска - групише симболе у биграфе, триграфе,... и њих трансформише
- Врсте криптосистема:
  - Симетричан - Алиса и Бобан користе исти кључ за (де)криптовање. Унапред договоре кључ и чувају га у тајности. Кључ се периодично мења. Непрактично када имамо велики број корисника и свако комуницира са сваким.

- Обично:
  - Алиса шаље поруку Бобану
  - Цица краде шифрат и покушава да га декриптује не знајући кључ (криптоанализа)
- Врсте шифре:
  - Проточна - трансформише симбол по симбол или бит по бит
  - Блоковска - групише симболе у биграфе, триграфе,... и њих трансформише
- Врсте криптосистема:
  - Симетричан - Алиса и Бобан користе исти кључ за (де)криптовање. Унапред договоре кључ и чувају га у тајности. Кључ се периодично мења. Непрактично када имамо велики број корисника и свако комуницира са сваким.
  - Асиметричан (криптосистем са јавним кључем) - Алиса и Бобан праве сопствене кључеве, кључ за криптовање објављују, док кључ за декриптовање чувају у тајности.


Symmetric Cryptography Scheme
Private Key
Plaintext
Encryption
Ciphertext
Decryption
Plaintext
Asymmetric Cryptography Scheme
Public key
Private Key
Plaintext
Encryption
Ciphertext
Decryption
Plaintext

- Недостаци криптосистема:
  - код симетричних: Алиса и Бобан морају да имају додатни канал комуникације за размену кључа, дуже коришћење истог кључа смањује сигурност, лакши су за криптоанализу

- Недостаци криптосистема:
  - код симетричних: Алиса и Бобан морају да имају додатни канал комуникације за размену кључа, дуже коришћење истог кључа смањује сигурност, лакши су за криптоанализу
  - код асиметричних: споро извршавање. Обично се користе за пренос кратке поруке када је потребна висока сигурност

- Недостаци криптосистема:
  - код симетричних: Алиса и Бобан морају да имају додатни канал комуникације за размену кључа, дуже коришћење истог кључа смањује сигурност, лакши су за криптоанализу
  - код асиметричних: споро извршавање. Обично се користе за пренос кратке поруке када је потребна висока сигурност
- Најбоље резултате даје комбиновање симетричних и асиметричних

- Недостаци криптосистема:
  - код симетричних: Алиса и Бобан морају да имају додатни канал комуникације за размену кључа, дуже коришћење истог кључа смањује сигурност, лакши су за криптоанализу
  - код асиметричних: споро извршавање. Обично се користе за пренос кратке поруке када је потребна висока сигурност
- Најбоље резултате даје комбиновање симетричних и асиметричних
  - Обично се за слање поруке користи симетричан криптосистем, а његов кључ се размењује асиметричим криптосистемом.
  - По том принципу ради HTTPS протокол (Hypertext Transfer Protocol Secure)

Пример: Цезарова шифра

Слова $A - Z$ кодирамо са $\{0, \ldots, 25\} = \mathbb{Z}_{26}$, проточна шифра:

$$f(P) = P + b \bmod 26$$

где је $\in \mathbb{Z}_{26}$ кодирани симбол, $b = 16$ тајни кључ.

Пример: Цезарова шифра

Слова $A - Z$ кодирамо са $\{0, \ldots, 25\} = \mathbb{Z}_{26}$, проточна шифра:

$$f(P) = P + b \bmod 26$$

где је $\in \mathbb{Z}_{26}$ кодирани симбол, $b = 16$ тајни кључ.
Алиса хоће да пошаље поруку $'PAYMENOW'$, она кодира

$$'PAYMENOW' \longrightarrow 15\ 0\ 24\ 12\ 4\ 13\ 14\ 22$$
$$\xrightarrow{f} 5\ 16\ 14\ 2\ 20\ 3\ 4\ 12 \longrightarrow 'FQOCUDEM'$$

Пример: Цезарова шифра

Слова $A - Z$ кодирамо са $\{0, \ldots, 25\} = \mathbb{Z}_{26}$, проточна шифра:

$$f(P) = P + b \bmod 26$$

где је $\in \mathbb{Z}_{26}$ кодирани симбол, $b = 16$ тајни кључ.
Алиса хоће да пошаље поруку $'PAYMENOW'$, она кодира

$$'PAYMENOW' \longrightarrow 15\ 0\ 24\ 12\ 4\ 13\ 14\ 22$$
$$\xrightarrow{f} 5\ 16\ 14\ 2\ 20\ 3\ 4\ 12 \longrightarrow 'FQOCUDEM'$$

Бобан може да прочита поруку помоћу алгоритма

$$f^{-1}(C) = C - b \bmod 26$$

Пример: Цезарова шифра

Слова $A - Z$ кодирамо са $\{0, \ldots, 25\} = \mathbb{Z}_{26}$, проточна шифра:

$$f(P) = P + b \bmod 26$$

где је $\in \mathbb{Z}_{26}$ кодирани симбол, $b = 16$ тајни кључ.
Алиса хоће да пошаље поруку $'PAYMENOW'$, она кодира

$$'PAYMENOW' \longrightarrow 15\ 0\ 24\ 12\ 4\ 13\ 14\ 22$$
$$\xrightarrow{f} 5\ 16\ 14\ 2\ 20\ 3\ 4\ 12 \longrightarrow 'FQOCUDEM'$$

Бобан може да прочита поруку помоћу алгоритма

$$f^{-1}(C) = C - b \bmod 26$$

Уопштење: Афина шифра

$$f(P) = aP + b \bmod 26 \qquad \text{и} \qquad f^{-1}(C) = a'C + b' \bmod 26,$$

где је $a' = a^{-1}$ у $\mathbb{Z}_{26}^*$ и $b' = -a^{-1}b$

Криптоанализа

$$f(P) = aP + b \bmod 26 \qquad \text{и} \qquad f^{-1}(C) = a'C + b' \bmod 26,$$

Криптоанализа

$$f(P) = aP + b \bmod 26 \qquad \text{и} \qquad f^{-1}(C) = a'C + b' \bmod 26,$$

- Познато: Најфреквентније слово у тексту на енглеском језику је $'E'$.

Криптоанализа

$$f(P) = aP + b \bmod 26 \qquad \text{и} \qquad f^{-1}(C) = a'C + b' \bmod 26,$$

- Познато: Најфреквентније слово у тексту на енглеском језику је $'E'$.
- Цица проналази најфреквентније слово у шифрату, нпр. нека је $'K'$, и претпоставља да је $f('E') = 'K'$, тј. $f^{-1}('K') = 'E'$

Криптоанализа

$$f(P) = aP + b \bmod 26 \qquad \text{и} \qquad f^{-1}(C) = a'C + b' \bmod 26,$$

- Познато: Најфреквентније слово у тексту на енглеском језику је $'E'$.
- Цица проналази најфреквентније слово у шифрату, нпр. нека је $'K'$, и претпоставља да је $f('E') = 'K'$, тј. $f^{-1}('K') = 'E'$
- Слично, упоређује друго најфреквентније слово и закључује $f^{-1}('D') = 'T'$

Криптоанализа

$$f(P) = aP + b \bmod 26 \qquad \text{и} \qquad f^{-1}(C) = a'C + b' \bmod 26,$$

- Познато: Најфреквентније слово у тексту на енглеском језику је $'E'$.
- Цица проналази најфреквентније слово у шифрату, нпр. нека је $'K'$, и претпоставља да је $f('E') = 'K'$, тј. $f^{-1}('K') = 'E'$
- Слично, упоређује друго најфреквентније слово и закључује $f^{-1}('D') = 'T'$
- Цица решава систем

$$\begin{aligned} 10a' &+ b' &&= 4 \bmod 26 \\ 3a' &+ b' &&= 19 \bmod 26 \end{aligned}$$

и закључује да је кључ највероватније $a' = 9$, $b' = 18$

Криптоанализа

$$f(P) = aP + b \bmod 26 \qquad \text{и} \qquad f^{-1}(C) = a'C + b' \bmod 26,$$

- Познато: Најфреквентније слово у тексту на енглеском језику је $'E'$.
- Цица проналази најфреквентније слово у шифрату, нпр. нека је $'K'$, и претпоставља да је $f('E') = 'K'$, тј. $f^{-1}('K') = 'E'$
- Слично, упоређује друго најфреквентније слово и закључује $f^{-1}('D') = 'T'$
- Цица решава систем

$$\begin{aligned} 10a' &+ b' &&= 4 \bmod 26 \\ 3a' &+ b' &&= 19 \bmod 26 \end{aligned}$$

  и закључује да је кључ највероватније $a' = 9$, $b' = 18$
- Ако систем нема (јединствено) решење: уместо 2. најфреквентнијег слова може користити 3., 4., ...

- Закључак: боље радити са већим блоковима слова

- Закључак: боље радити са већим блоковима слова

Пример: Диграфови

Пар слова се кодира нечим из $\{0, 1, \ldots, 26^2 - 1 = 675\}$.

- Закључак: боље радити са већим блоковима слова

Пример: Дигрофови

Пар слова се кодира нечим из $\{0, 1, \ldots, 26^2 - 1 = 675\}$.
Диграф $'NO'$ се кодира са $26 \cdot {}'N' + {}'0' = 26 \cdot 13 + 14 = 352$,
затим се криптује $\underbrace{159}_{\text{кључ}} \cdot 352 + \underbrace{580}_{\text{кључ}} = 440 \bmod 676$ што је
еквивалент $'QY'$.

- Нека је $A \in \mathcal{M}_2(\mathbb{Z}_n)$ инвертибилна матрица, тј. таква да је $\det A$ инвертибилно у $\mathbb{Z}_n$

  Нпр. $n = 26$ и $A = \begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix}$

- Нека је $A \in \mathcal{M}_2(\mathbb{Z}_n)$ инвертибилна матрица, тј. таква да је $\det A$ инвертибилно у $\mathbb{Z}_n$

  Нпр. $n = 26$ и $A = \begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix}$

- Алгоритам за криптовање/декриптовање диграфа

$$\begin{pmatrix} x \\ y \end{pmatrix} \xrightarrow{f} A \begin{pmatrix} x \\ y \end{pmatrix} \qquad \text{и} \qquad \begin{pmatrix} x \\ y \end{pmatrix} \xrightarrow{f^{-1}} A^{-1} \begin{pmatrix} x \\ y \end{pmatrix}$$

- Нека је $A \in \mathcal{M}_2(\mathbb{Z}_n)$ инвертибилна матрица, тј. таква да је $\det A$ инвертибилно у $\mathbb{Z}_n$
  Нпр. $n = 26$ и $A = \begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix}$
- Алгоритам за криптовање/декриптовање диграфа

$$\begin{pmatrix} x \\ y \end{pmatrix} \xrightarrow{f} A \begin{pmatrix} x \\ y \end{pmatrix} \qquad \text{и} \qquad \begin{pmatrix} x \\ y \end{pmatrix} \xrightarrow{f^{-1}} A^{-1} \begin{pmatrix} x \\ y \end{pmatrix}$$

- Алиса жели да пошаље поруку $'NO|AN|SW|ER'$ тј.
  $\begin{pmatrix} 13 \\ 14 \end{pmatrix} \begin{pmatrix} 0 \\ 13 \end{pmatrix} \begin{pmatrix} 18 \\ 22 \end{pmatrix} \begin{pmatrix} 4 \\ 17 \end{pmatrix}$

$$\begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix} \begin{pmatrix} 13 & 0 & 18 & 4 \\ 14 & 13 & 22 & 17 \end{pmatrix} = \begin{pmatrix} 68 & 39 & 102 & 59 \\ 203 & 104 & 302 & 164 \end{pmatrix} = \begin{pmatrix} 16 & 13 & 24 & 7 \\ 21 & 0 & 16 & 8 \end{pmatrix}$$

што је $'QVNAYQHI'$

- Нека је $A \in \mathcal{M}_2(\mathbb{Z}_n)$ инвертибилна матрица, тј. таква да је $\det A$ инвертибилно у $\mathbb{Z}_n$

  Нпр. $n = 26$ и $A = \begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix}$

- Алгоритам за криптовање/декриптовање диграфа

$$\begin{pmatrix} x \\ y \end{pmatrix} \xrightarrow{f} A \begin{pmatrix} x \\ y \end{pmatrix} \qquad \text{и} \qquad \begin{pmatrix} x \\ y \end{pmatrix} \xrightarrow{f^{-1}} A^{-1} \begin{pmatrix} x \\ y \end{pmatrix}$$

- Ако Бобан добије поруку $'FW|MD|IQ'$, он ће је помоћу $A^{-1} = \begin{pmatrix} 14 & 11 \\ 17 & 10 \end{pmatrix} \mod 26$ прочитати као

$$\begin{pmatrix} 14 & 11 \\ 17 & 10 \end{pmatrix} \begin{pmatrix} 5 & 12 & 8 \\ 22 & 3 & 16 \end{pmatrix} = \begin{pmatrix} 0 & 19 & 2 \\ 19 & 0 & 10 \end{pmatrix}$$

што је $'ATTACK'$

Једнократна шифра (One-Time Pad)

- Ово је најједноставнија проточна шифра
- порука M се кодира бинарно
- кључ K (који је исто записан бинарно) мора да буде исте дужине као M

Једнократна шифра (One-Time Pad)

- Ово је најједноставнија проточна шифра
- порука M се кодира бинарно
- кључ K (који је исто записан бинарно) мора да буде исте дужине као M
- криптовање се врши бит по бит, сабирањем бита из M и бита из K по модулу 2

Једнократна шифра (One-Time Pad)

- Ово је најједноставнија проточна шифра
- порука M се кодира бинарно
- кључ K (који је исто записан бинарно) мора да буде исте дужине као M
- криптовање се врши бит по бит, сабирањем бита из M и бита из K по модулу 2
- идентично се ради и декриптовање (истим кључем)

Једнократна шифра (One-Time Pad)

- Ово је најједноставнија проточна шифра
- порука M се кодира бинарно
- кључ K (који је исто записан бинарно) мора да буде исте дужине као M
- криптовање се врши бит по бит, сабирањем бита из M и бита из K по модулу 2
- идентично се ради и декриптовање (истим кључем)
- кључ сме да се користи само једном

Једнократна шифра (One-Time Pad)

- Ово је најједноставнија проточна шифра
- порука M се кодира бинарно
- кључ K (који је исто записан бинарно) мора да буде исте дужине као M
- криптовање се врши бит по бит, сабирањем бита из M и бита из K по модулу 2
- идентично се ради и декриптовање (истим кључем)
- кључ сме да се користи само једном
- Пример: Порука $'Go'$ се кодира са 0100011101101111

шифровање		*дешифровање*
0100011101101111		0011101011100010
$\oplus\ 0111110110001101$	*низ кључа*	$\oplus\ 0111110110001101$
0011101011100010		0100011101101111
		Go

Савремене ланчане шифре

- Проблем: Када Алиса и Бобан размењују кључ за OTP они (можда) не знају дужину поруке. Шта ако је порука дужа од кључа?
  - Понављање (дела) кључа није сигурно

Савремене ланчане шифре

- Проблем: Када Алиса и Бобан размењују кључ за OTP они (можда) не знају дужину поруке. Шта ако је порука дужа од кључа?
  - Понављање (дела) кључа није сигурно
- Једно решење су савремени ланчани криптосистеми. Они решавају поменути проблем тако што
  - Алиса и Бобан размене неки (мали) кључ K

Савремене ланчане шифре

- Проблем: Када Алиса и Бобан размењују кључ за OTP они (можда) не знају дужину поруке. Шта ако је порука дужа од кључа?
  - Понављање (дела) кључа није сигурно
- Једно решење су савремени ланчани криптосистеми. Они решавају поменути проблем тако што
  - Алиса и Бобан размене неки (мали) кључ K
  - Приликом (де)криптовања OTP-ом они неће користити кључ K, већ други кључ L који добијају уз помоћ генератора псеудослучајних бројева. Генератор покрећу независно једно од другог, али за исто почетно стање K.

Савремене ланчане шифре

- Проблем: Када Алиса и Бобан размењују кључ за OTP они (можда) не знају дужину поруке. Шта ако је порука дужа од кључа?
  - Понављање (дела) кључа није сигурно
- Једно решење су савремени ланчани криптосистеми. Они решавају поменути проблем тако што
  - Алиса и Бобан размене неки (мали) кључ K
  - Приликом (де)криптовања OTP-ом они неће користити кључ K, већ други кључ L који добијају уз помоћ генератора псеудослучајних бројева. Генератор покрећу независно једно од другог, али за исто почетно стање K.
  - Алиса види поруку, Бобан види шифрат, тако да обоје имају дужину поруке и на основу тога одређују која им је дужина кључа L потребна

RC4 криптосистем (Rivest cipher 4)

- RC4 је пример савременог ланчаног криптосистема (OTP криптосистем који користи генератор псеудослучајних бројева), има бројне унапређене верзије

RC4 криптосистем (Rivest cipher 4)

- RC4 је пример савременог ланчаног криптосистема (OTP криптосистем који користи генератор псеудослучајних бројева), има бројне унапређене верзије
- Најпре се изабере природан број $m = 2^n$, најчешће $n = 8$

RC4 криптосистем (Rivest cipher 4)

- RC4 је пример савременог ланчаног криптосистема (OTP криптосистем који користи генератор псеудослучајних бројева), има бројне унапређене верзије
- Најпре се изабере природан број $m = 2^n$, најчешће $n = 8$
- Кључ K (записан бинарно) се издели на блокове од по n битова и од тих блокова се формира низ K_0, K_1, K_2, ..., K_{m-1} (по потреби се блокови периодично понављају све док се не попуни свих m позиција)

RC4 криптосистем (Rivest cipher 4)

- RC4 је пример савременог ланчаног криптосистема (OTP криптосистем који користи генератор псеудослучајних бројева), има бројне унапређене верзије
- Најпре се изабере природан број $m = 2^n$, најчешће $n = 8$
- Кључ K (записан бинарно) се издели на блокове од по n битова и од тих блокова се формира низ K_0, K_1, K_2, ..., K_{m-1} (по потреби се блокови периодично понављају све док се не попуни свих m позиција)

Пример

$n = 3$. *Нека је кључ* 011001100001101, *односно* 011 001 100 001 101, *односно* $[3, 1, 4, 1, 5]$. *Периодичним проширивањем добијамо низ дужине* 2^n

$$[3, 1, 4, 1, 5, 3, 1, 4] = [K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7].$$

Припрема генератора за рад (сва сабирања су по модулу $m = 2^n$):

- У овој фази се користи проширени кључ K_0, K_1, K_2, ..., K_{m-1}

Припрема генератора за рад (сва сабирања су по модулу $m = 2^n$):

- У овој фази се користи проширени кључ K_0, K_1, K_2, ..., K_{m-1}
- Формира се низ $S_0, S_1, S_2 \ldots, S_{m-1}$ и иницијално се постави $S_i = i$, за све i
- Бројач j је на почетку 0

Припрема генератора за рад (сва сабирања су по модулу $m = 2^n$):

- У овој фази се користи проширени кључ K_0, K_1, K_2, ..., K_{m-1}
- Формира се низ $S_0, S_1, S_2 \ldots, S_{m-1}$ и иницијално се постави $S_i = i$, за све i
- Бројач j је на почетку 0
- За свако $i = 0, 1, 2, \ldots, m-1$:
  - на бројач j се додаје $S_i + K_i$
  - S_i и S_j мењају места у низу

Припрема генератора за рад (сва сабирања су по модулу $m = 2^n$):

- У овој фази се користи проширени кључ K_0, K_1, K_2, ..., K_{m-1}
- Формира се низ $S_0, S_1, S_2 \ldots, S_{m-1}$ и иницијално се постави $S_i = i$, за све i
- Бројач j је на почетку 0
- За свако $i = 0, 1, 2, \ldots, m-1$:
  - на бројач j се додаје $S_i + K_i$
  - S_i и S_j мењају места у низу

Приметимо:

- Сваки члан низа ће променити место бар једном
- Сабирају се индекси и чланови низа - практично је немогуће пратити шта се догађа (псеудослучајност)

Пример (наставак):

$n = 3.$

$[3, 1, 4, 1, 5, 3, 1, 4] = [K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7].$

i	j	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
	0	0	1	2	3	4	5	6	7
0	3	3	1	2	0	4	5	6	7
1	5	3	5	2	0	4	1	6	7
2	3	3	5	0	2	4	1	6	7
3	6	3	5	0	6	4	1	2	7
4	7	3	5	0	6	7	1	2	4
5	3	3	5	0	1	7	6	2	4
6	6	3	5	0	1	7	6	2	4
7	6	3	5	0	1	7	6	4	2

- Порука се исто дели на n-битне блокове и нека је l број тих блокова, генератор треба да да кључ $L_0, L_1, \ldots, L_{l-1}$

- Порука се исто дели на n-битне блокове и нека је l број тих блокова, генератор треба да да кључ $L_0, L_1, \ldots, L_{l-1}$

Покретање генератора (сва сабирања су по модулу m):

- Иницијално $i, j = 0$ и користи се низ $S_0, S_1, \ldots, S_{m-1}$ (испермутован у фази припреме)

- Порука се исто дели на n-битне блокове и нека је l број тих блокова, генератор треба да да кључ $L_0, L_1, \ldots, L_{l-1}$

Покретање генератора (сва сабирања су по модулу m):

- Иницијално $i, j = 0$ и користи се низ $S_0, S_1, \ldots, S_{m-1}$ (испермутован у фази припреме)
- За свако $r = 0, 1, 2, \ldots, l-1$:
  - на бројач i се додаје 1
  - на бројач j се додаје S_i
  - S_i и S_j мењају места у низу
  - у L_r се уписује S_t, где је $t = S_i + S_j$

- Порука се исто дели на n-битне блокове и нека је l број тих блокова, генератор треба да да кључ $L_0, L_1, \ldots, L_{l-1}$

Покретање генератора (сва сабирања су по модулу m):

- Иницијално $i, j = 0$ и користи се низ $S_0, S_1, \ldots, S_{m-1}$ (испермутован у фази припреме)
- За свако $r = 0, 1, 2, \ldots, l-1$:
  - на бројач i се додаје 1
  - на бројач j се додаје S_i
  - S_i и S_j мењају места у низу
  - у L_r се уписује S_t, где је $t = S_i + S_j$

Низ S се и даље пермутује - ако се некад понови индекс $S_i + S_j$, не значи да ће се поновити и члан низа $S_{S_i+S_j}$

Пример (наставак):

i	j	t	L_t	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
0	0			3	5	0	1	7	6	4	2
1	5	3	1	3	6	0	1	7	5	4	2
2	5	5	0	3	6	5	1	7	0	4	2
3	6	5	0	3	6	5	4	7	0	1	2
4	5	7	2	3	6	5	4	0	7	1	2
5	4	7	2	3	6	5	4	7	0	1	2
6	5	1	6	3	6	5	4	7	1	0	2
7	7	4	7	3	6	5	4	7	1	0	2
0	2	0	5	5	6	3	4	7	1	0	2
1	0	3	4	6	5	3	4	7	1	0	2
2	3	7	2	6	5	4	3	7	1	0	2
3	6	3	0	6	5	4	0	7	1	3	2
4	5	0	6	6	5	4	0	1	7	3	2

Низ кључа добија се од тробитних репрезентација бројева $1, 0, 0, 2, 2, 6, 7, 5, 4, 2, 0, 6$, *односно*
001 000 000 010 010 110 111 101 100 010 000 110
(без размака).

Један могући напад на RC4:

- Заснива се на томе да Цица зна један део Алисине нешифроване поруке (нпр. Цица претпоставља да порука почиње са 'Поштовани' или 'Драги' или да се завршава са 'поздрав, Алиса').

Један могући напад на RC4:

- Заснива се на томе да Цица зна један део Алисине нешифроване поруке (нпр. Цица претпоставља да порука почиње са 'Поштовани' или 'Драги' или да се завршава са 'поздрав, Алиса').
- Поређењем тога и одговарајућег дела шифрата Цица може да реконструише један део L' кључа L (који је добијен генератором)

Један могући напад на RC4:

- Заснива се на томе да Цица зна један део Алисине нешифроване поруке (нпр. Цица претпоставља да порука почиње са 'Поштовани' или 'Драги' или да се завршава са 'поздрав, Алиса').
- Поређењем тога и одговарајућег дела шифрата Цица може да реконструише један део L' кључа L (који је добијен генератором)
- Да ли Цица може да реконструише кључ K помоћу L'? Тиме би аутоматски дошла до целог кључа L
  - требало би да буде неизводљиво у реалном времену
  - али чињеница је да L' јединствено одређује K, ако је L' довољно дугачак

- Као додатна заштита приликом криптовања: за i-ти бит шифрата c_i (поред i-тог бита поруке p_i и i-тог бита кључа l_i) користити још неки бит поруке нпр. p_{i-1}, p_{i-2}, p_{i+1} итд.

- Као додатна заштита приликом криптовања: за i-ти бит шифрата c_i (поред i-тог бита поруке p_i и i-тог бита кључа l_i) користити још неки бит поруке нпр. p_{i-1}, p_{i-2}, p_{i+1} итд.
- Пример таквог криптовања и декриптовања је

$$c_i = p_i +_2 l_i +_2 \begin{cases} p_{i-2} & ако\ је\ p_{i-1} = 0 \\ p_{i-3} & ако\ је\ p_{i-1} = 1 \end{cases} \qquad p_{-1} = p_0 = 0.$$

$$p_i = c_i +_2 l_i +_2 \begin{cases} p_{i-2} & ако\ је\ p_{i-1} = 0 \\ p_{i-3} & ако\ је\ p_{i-1} = 1 \end{cases}$$

Коначна поља (подсећање)

- Ако је p прост број, тада је $(\mathbb{Z}_p, +_p, \cdot_p)$ је поље, где је $\mathbb{Z}_p = \mathbb{Z}/(p\mathbb{Z}) = \{0, 1, 2, \ldots, p-1\}$

Коначна поља (подсећање)

- Ако је p прост број, тада је $(\mathbb{Z}_p, +_p, \cdot_p)$ је поље, где је $\mathbb{Z}_p = \mathbb{Z}/(p\mathbb{Z}) = \{0, 1, 2, \ldots, p-1\}$
- Мултипликативна група $(\mathbb{Z}_p \setminus \{0\}, \cdot_p)$ је циклична. тј. постоји генератор (примитивни корен) $g \in \mathbb{Z}_p \setminus \{0\}$ тд. се сви елементи $\mathbb{Z}_p \setminus \{0\}$ могу видети као степени g

Коначна поља (подсећање)

- Ако је p прост број, тада је $(\mathbb{Z}_p, +_p, \cdot_p)$ је поље, где је $\mathbb{Z}_p = \mathbb{Z}/(p\mathbb{Z}) = \{0, 1, 2, \ldots, p-1\}$
- Мултипликативна група $(\mathbb{Z}_p \setminus \{0\}, \cdot_p)$ је циклична. тј. постоји генератор (примитивни корен) $g \in \mathbb{Z}_p \setminus \{0\}$ тд. се сви елементи $\mathbb{Z}_p \setminus \{0\}$ могу видети као степени g
  - Пример: 3 је генератор $\mathbb{Z}_7 \setminus \{0\}$ јер је $3^1 = 3$, $3^2 = 2$, $3^3 = 6$, $3^4 = 4$, $3^5 = 5$, $3^6 = 1$,

Коначна поља (подсећање)

- Ако је p прост број, тада је $(\mathbb{Z}_p, +_p, \cdot_p)$ је поље, где је $\mathbb{Z}_p = \mathbb{Z}/(p\mathbb{Z}) = \{0, 1, 2, \ldots, p-1\}$
- Мултипликативна група $(\mathbb{Z}_p \setminus \{0\}, \cdot_p)$ је циклична. тј. постоји генератор (примитивни корен) $g \in \mathbb{Z}_p \setminus \{0\}$ тд. се сви елементи $\mathbb{Z}_p \setminus \{0\}$ могу видети као степени g
  - Пример: 3 је генератор $\mathbb{Z}_7 \setminus \{0\}$ јер је $3^1 = 3$, $3^2 = 2$, $3^3 = 6$, $3^4 = 4$, $3^5 = 5$, $3^6 = 1$, али 2 није генератор јер је $2^1 = 2$, $2^2 = 4$, $2^3 = 1$

Ако је $f(x) \in \mathbb{Z}_p[x]$ нерастављив полином (у $\mathbb{Z}_p[x]$) степена d, тада је $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ поље

- Кренули смо од прстена $\mathbb{Z}_p[x] =$ сви полиноми по x са коефицијентима у $\mathbb{Z}_p = \{0, 1, \ldots, p-1\}$
- Затим су идентификовани полиноми чија је разлика дељива са $f(x)$

Ако је $f(x) \in \mathbb{Z}_p[x]$ нерастављив полином (у $\mathbb{Z}_p[x]$) степена d, тада је $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ поље

- Кренули смо од прстена $\mathbb{Z}_p[x]$ = сви полиноми по x са коефицијентима у $\mathbb{Z}_p = \{0, 1, \ldots, p-1\}$
- Затим су идентификовани полиноми чија је разлика дељива са $f(x)$
- У $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ је $f(x) = 0$. Ако је $f(x) = a_d x^d + a_{d-1}x^{d-1} + \cdots + a_1 x + a_0$, $a_d \neq 0$, тада је $x^d = -\frac{a_{d-1}}{a_d}x^{d-1} - \cdots - \frac{a_1}{a_d}x - \frac{a_0}{a_d}$, а самим тим се и сви степени x^k, $k \geqslant d$ могу расписати преко $x^{d-1}, \ldots, x, 1$

Ако је $f(x) \in \mathbb{Z}_p[x]$ нерастављив полином (у $\mathbb{Z}_p[x]$) степена d, тада је $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ поље

- Кренули смо од прстена $\mathbb{Z}_p[x]$ = сви полиноми по x са коефицијентима у $\mathbb{Z}_p = \{0, 1, \ldots, p-1\}$
- Затим су идентификовани полиноми чија је разлика дељива са $f(x)$
- У $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ је $f(x) = 0$. Ако је $f(x) = a_d x^d + a_{d-1}x^{d-1} + \cdots + a_1 x + a_0$, $a_d \neq 0$, тада је $x^d = -\frac{a_{d-1}}{a_d}x^{d-1} - \cdots - \frac{a_1}{a_d}x - \frac{a_0}{a_d}$, а самим тим се и сви степени x^k, $k \geqslant d$ могу расписати преко $x^{d-1}, \ldots, x, 1$
- Зато је $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ састављен од свих полинома степена мањег од $d = \deg f$, па има p^d елемената

Ако је $f(x) \in \mathbb{Z}_p[x]$ нерастављив полином (у $\mathbb{Z}_p[x]$) степена d, тада је $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ поље

- Кренули смо од прстена $\mathbb{Z}_p[x]$ = сви полиноми по x са коефицијентима у $\mathbb{Z}_p = \{0, 1, \ldots, p-1\}$
- Затим су идентификовани полиноми чија је разлика дељива са $f(x)$
- У $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ је $f(x) = 0$. Ако је $f(x) = a_d x^d + a_{d-1}x^{d-1} + \cdots + a_1 x + a_0$, $a_d \neq 0$, тада је $x^d = -\frac{a_{d-1}}{a_d}x^{d-1} - \cdots - \frac{a_1}{a_d}x - \frac{a_0}{a_d}$, а самим тим се и сви степени x^k, $k \geqslant d$ могу расписати преко $x^{d-1}, \ldots, x, 1$
- Зато је $\mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$ састављен од свих полинома степена мањег од $d = \deg f$, па има p^d елемената
- Сабира се и множи по модулу $f(x)$ (и по модулу p)

Теорема

1. Кардиналност коначног поља мора бити степен простог броја
2. Два коначна поља су изоморфна акко имају исти број елемената

Теорема

1. Кардиналност коначног поља мора бити степен простог броја
2. Два коначна поља су изоморфна акко имају исти број елемената

- $\mathbb{F}_q =$ коначно поље са q елемената (јединствено одређено до на изоморфизам) где је $q = p^d$ степен простог броја

Теорема

1. Кардиналност коначног поља мора бити степен простог броја
2. Два коначна поља су изоморфна акко имају исти број елемената

- $\mathbb{F}_q =$ коначно поље са q елемената (јединствено одређено до на изоморфизам) где је $q = p^d$ степен простог броја
- $\mathbb{F}_p \cong \mathbb{Z}_p$ за прост p

Теорема

1. Кардиналност коначног поља мора бити степен простог броја
2. Два коначна поља су изоморфна акко имају исти број елемената

- $\mathbb{F}_q =$ коначно поље са q елемената (јединствено одређено до на изоморфизам) где је $q = p^d$ степен простог броја
- $\mathbb{F}_p \cong \mathbb{Z}_p$ за прост p
- Ако је $q = p^d$ тада је $\mathbb{F}_q \cong \mathbb{Z}_p[x]/(f\mathbb{Z}_p[x])$, где је $f(x) \in \mathbb{Z}_p[x]$ нерастављив полином степена d

Теорема

1. Кардиналност коначног поља мора бити степен простог броја
2. Два коначна поља су изоморфна акко имају исти број елемената

- $\mathbb{F}_q =$ коначно поље са q елемената (јединствено одређено до на изоморфизам) где је $q = p^d$ степен простог броја
- $\mathbb{F}_p \cong \mathbb{Z}_p$ за прост p
- Ако је $q = p^d$ тада је $\mathbb{F}_q \cong \mathbb{Z}_p[x] / (f\mathbb{Z}_p[x])$, где је $f(x) \in \mathbb{Z}_p[x]$ нерастављив полином степена d
- $\mathbb{F}_q^* = \mathbb{F}_q \setminus \{0\} =$ мултипликативна група састављена од елемената из $\mathbb{F}_q$ који имају инверз у односу на $\cdot_f$

Теорема

1. Кардиналност коначног поља мора бити степен простог броја
2. Два коначна поља су изоморфна акко имају исти број елемената

- $\mathbb{F}_q =$ коначно поље са q елемената (јединствено одређено до на изоморфизам) где је $q = p^d$ степен простог броја
- $\mathbb{F}_p \cong \mathbb{Z}_p$ за прост p
- Ако је $q = p^d$ тада је $\mathbb{F}_q \cong \mathbb{Z}_p[x] / (f\mathbb{Z}_p[x])$, где је $f(x) \in \mathbb{Z}_p[x]$ нерастављив полином степена d
- $\mathbb{F}_q^* = \mathbb{F}_q \setminus \{0\} =$ мултипликативна група састављена од елемената из $\mathbb{F}_q$ који имају инверз у односу на $\cdot_f$
- $\left(\mathbb{F}_q^*, \cdot\right)$ је циклична група

Пример: Таблица множења у $\mathbb{F}_8^* \cong (\mathbb{Z}_2[x]/f\mathbb{Z}_2[x])^*$, где је $f(x) = x^3 + x^2 + 1$

$\cdot$	1	x	$x+1$	x^2	x^2+1	x^2+x	x^2+x+1
1	1	x	$x+1$	x^2	x^2+1	x^2+x	x^2+x+1
x	x	x^2	x^2+x	x^2+1	x^2+x+1	1	$x+1$
$x+1$	$x+1$	x^2+x	x^2+1	1	x	x^2+x+1	x^2
x^2	x^2	x^2+1	1	x^2+x+1	$x+1$	x	x^2+x
x^2+1	x^2+1	x^2+x+1	x	$x+1$	x^2+x	x^2	1
x^2+x	x^2+x	1	x^2+x+1	x	x^2	$x+1$	x^2+1
x^2+x+1	x^2+x+1	$x+1$	x^2	x^2+x	1	x^2+1	x

- Коришћено $x^3 = -x^2 - 1 = x^2 + 1$ у $\mathbb{F}_8$

Пример: Таблица множења у $\mathbb{F}_8^* \cong (\mathbb{Z}_2[x]/f\mathbb{Z}_2[x])^*$, где је $f(x) = x^3 + x^2 + 1$

$\cdot$	1	x	$x+1$	x^2	x^2+1	x^2+x	x^2+x+1
1	1	x	$x+1$	x^2	x^2+1	x^2+x	x^2+x+1
x	x	x^2	x^2+x	x^2+1	x^2+x+1	1	$x+1$
$x+1$	$x+1$	x^2+x	x^2+1	1	x	x^2+x+1	x^2
x^2	x^2	x^2+1	1	x^2+x+1	$x+1$	x	x^2+x
x^2+1	x^2+1	x^2+x+1	x	$x+1$	x^2+x	x^2	1
x^2+x	x^2+x	1	x^2+x+1	x	x^2	$x+1$	x^2+1
x^2+x+1	x^2+x+1	$x+1$	x^2	x^2+x	1	x^2+1	x

- Коришћено $x^3 = -x^2 - 1 = x^2 + 1$ у $\mathbb{F}_8$
- Генератор ове групе је x јер је

k	0	1	2	3	4	5	6	7
x^k	1	x	x^2	x^2+1	x^2+x+1	$x+1$	x^2+x	1

За $q = p^d$

- можемо поистоветити полиноме

$$a_{d-1}x^{d-1} + a_{d-2}x^{d-2} + \cdots + a_1 + a_0$$

$(0 \leqslant a_i \leqslant d-1$, за све $i)$ са бројем записаним у систему са основом p:

$$\overline{a_{d-1}a_{d-2}\ldots a_1a_0}$$

са највише d цифара

За $q = p^d$

- можемо поистоветити полиноме

$$a_{d-1}x^{d-1} + a_{d-2}x^{d-2} + \cdots + a_1 + a_0$$

$(0 \leqslant a_i \leqslant d-1$, за све $i)$ са бројем записаним у систему са основом p:

$$\overline{a_{d-1}a_{d-2}\ldots a_1a_0}$$

са највише d цифара

- $+$ у $\mathbb{F}_q$ одговара ,,сабирању бројева без преноса“

За $q = p^d$

- можемо поистоветити полиноме

$$a_{d-1}x^{d-1} + a_{d-2}x^{d-2} + \cdots + a_1 + a_0$$

$(0 \leqslant a_i \leqslant d-1$, за све $i)$ са бројем записаним у систему са основом p:

$$\overline{a_{d-1}a_{d-2}\ldots a_1a_0}$$

са највише d цифара

- $+$ у $\mathbb{F}_q$ одговара ,,сабирању бројева без преноса“
- Специјално, $\mathbb{F}_{2^d}$ садржи све бројеве са највише d бинарних цифара и $+$ у $\mathbb{F}_{2^d}$ одговара ексклузивној дисјункцији бит по бит

За $q = p^d$

- можемо поистоветити полиноме

$$a_{d-1}x^{d-1} + a_{d-2}x^{d-2} + \cdots + a_1 + a_0$$

$(0 \leqslant a_i \leqslant d-1$, за све $i)$ са бројем записаним у систему са основом p:

$$\overline{a_{d-1}a_{d-2}\ldots a_1a_0}$$

са највише d цифара

- $+$ у $\mathbb{F}_q$ одговара „сабирању бројева без преноса“
- Специјално, $\mathbb{F}_{2^d}$ садржи све бројеве са највише d бинарних цифара и $+$ у $\mathbb{F}_{2^d}$ одговара ексклузивној дисјункцији бит по бит
- Пример: У $\mathbb{F}_8$ можемо поистоветити $x+1$ са 011 и x^2+x са 110, тада је $011 \cdot 110 = 111$ (из таблице)