Cryptography and Security Protocols
Paulo Mateus MMA–MEIC

Previously on CSP

Symmetric Cryptosystems
• Basic concepts: Symmetric cryptosystem, Caeser
• Historical systems: Substitution, Viginère, One-time pad,
• Perfect cryptography: Shannon Theorem for perfect cryptography

Today

• Symmetric Cryptosystems: Hill
• Modern cryptosystem: S-box, Substitution-Permutation, Feistel Ciphers, AES
• Cryptoanalysis: Linear and differential

Modern cryptosystems

Hill cryptosystem
Before finishing this section we shall look into an interesting cryptosystem that can be easily attacked

Definition 6. Hill Cryptosystem \( \mathcal{H} = (X, Y, K, e, d) \)
• \( X = Y = \mathbb{Z}_n^m \)
• \( K = \text{GL}(m, \mathbb{Z}_n) \) called the general linear group, that is, the set of all invertible \( m \times m \) matrices modulo \( n \);
• \( e_k(x) = k \cdot x \)
• \( d_k(x) = k^{-1} \cdot x \)

Hill implemented mechanically the cipher for \( n = 26 \) and \( m = 6 \) in 1929, and it was the first cipher that could operate on more than three symbols at once. However, his machine did not sell!

As we shall see, for \( n = 26 = 13 \times 2 \), thanks to the Chinese Remainder Theorem, a matrix \( A \) is is
invertible iff it is invertible modulo 13 and modulo 2. The order of GL(n, q) when q is prime

$$|GL(k, q)| = \prod_{i=0}^{m-1} (q^m - q^i)$$

The first column can be any but the zero vector;
The second column can be anything but multiples of the first column;
The kth column can be any vector not in the linear span of the first k – 1 columns.

Thus, $$|GL(k, \mathbb{Z}_{26})| = \prod_{i=0}^{k-1} (2^k - 2^i) \prod_{i=0}^{k-1} (13^k - 13^i)$$. For the case k=6 we have:

$$\prod_{i=1}^{k-1} (2^6 - 2^i) \prod_{i=0}^{k-1} (13^6 - 13^i)$$

The above construction gives an idea on how to construct a key:
The first column can be any column but the zero vector;
The second column can be anything but a multiple of the first column modulo 2 and 13;
The kth column can be any vector not in the linear span of the first k – 1 columns modulo 2 and 13.

Example: for K=3, a useful result is that:
A matrix K is invertible mod n iff gcd(det(K), n) = 1

$$\begin{bmatrix} 1 & 2 & 4 \\ 9 & 8 & 1 \\ 0 & 25 & 13 \end{bmatrix}$$

MatrixForm[k]

$$\begin{bmatrix} 1 & 2 & 4 \\ 9 & 8 & 1 \\ 0 & 25 & 13 \end{bmatrix}$$

GCD[Mod[Det[k], 26], 26]

1

b = Inverse[k, Modulus -> 26]

$$\begin{bmatrix} 23 & 12 & 12 \\ 13 & 13 & 25 \\ 1 & 23 & 4 \end{bmatrix}$$

k.(1, 2, 3)

$$\begin{bmatrix} 17 & 28 & 89 \end{bmatrix}$$

m = "the path of the righteous man is beset on all sides by the iniquities of the selfish and the tyrannoy of evildoers. Blessed is he whose name is in the name of charity and goodwill. He is the shepherd's friend, even through the valley of darkness for he is truly his brothers' keeper and the finder of lost children. And he will strike down upon thee with great vengeance and furious anger those who would attempt to poison and destroy my brothers and you will know my name, the Lord when I lay my vengeance upon thee;"
tonumber[raw_] :=
    Select[ToCharacterCode[raw], Function[x, x ≥ 97 && x ≤ 97 + 25]] - 97;
tostring[cod_] := FromCharacterCode[cod + 97];
enc[k_, m_] := Mod[Flatten[Map[Function[x, k.x], Partition[m, 3]]], 26]
dec[k_, m_] :=
    Mod[Flatten[Map[Function[x, Inverse[k, Modulus -> 26].x], Partition[m, 3]]], 26]
tostring[dec[k, enc[k, tonumber[m]]]]

the path of the righteous man is beset on all sides by the iniquities of the selfish and the tyrannous evil men. Blessed is he who keeps the way of righteousness and walks in the path of the righteous man, who is not turned aside from it. When he is tempted to sin, he will not stumble, for he will keep his heart pure and his hands clean. He will not fall into the hands of evil men, for he will not associate with them. If he is tempted and does not fall, he will be rewarded with double the joy. If he is tempted and does fall, he will be punished with double the severity. For the righteous man is blessed in all his ways, and his enemies will be ashamed and disgrace him. And the righteous man will be praised in all the world, for he is a son of God. And he will be given a crown of life, and he will inherit the kingdom of heaven. And the saints will praise him in heaven, and they will call him blessed.

Cryptoanalysis

Until now we have assumed that the attacker has no knowledge about the plaintext, but in general this is not the case. Hill cipher is trivially attacked if the attacker has some knowledge about the plaintexts. In cryptography four types of attacks are considered:

- **Ciphertext only attack**: The attacker only knows the ciphertext.

- **Know plaintext attack**: The attacker knows some plaintexts and the corresponding ciphertexts.

- **Chosen plaintext attack**: The attacker can choose some plaintexts for which he obtains corresponding ciphertexts (She had access to the encryption function).

- **Chosen ciphertext attack**: The attacker can choose some ciphertext for which he obtains corresponding ciphertexts (She had access to the decryption function).

The previous ciphers were broken using ciphertext only attack, Hill cipher can be easily broken with the know plaintext attack, or any of the others.

To realize this, notice that we just have to know the ciphertext of m independent plaintexts and we are able to extract the key by solving a system of linear equations (modulo 26)!

We exemplify the procedure for m=3;

\[
\begin{bmatrix}
k_{11} & k_{12} & k_{13} \\
k_{21} & k_{22} & k_{23} \\
k_{31} & k_{32} & k_{33}
\end{bmatrix}
\]

So our variable is the key k, with 9 variables. Assume we know the encryption of some 3 linearly independent plaintexts, p1, p2 and p3, say
Given \( S \) the a

vector

\[
\begin{align*}
\{ & p1 = (0, 1, 1); \\
& c1 = \text{enc}[K, p1] \\
& \{ 6, 9, 12 \} \\
& p2 = (3, 2, 1); \\
& c2 = \text{enc}[K, p2] \\
& \{ 11, 18, 11 \} \\
& p3 = (14, 5, 2); \\
& c3 = \text{enc}[K, (14, 5, 2)] \\
& \{ 6, 12, 21 \}
\end{align*}
\]

We can confirm if the know plaintexts are linearly independent:

\[
\text{GCD}[\text{Mod}[\text{Det}([p1, p2, p3]), 26], 26]
\]

1

The system of linear equations can be extracted as follows:

\[
\text{sys} = \{ \text{kx}.p1 = c1, \\
& \text{kx}.p2 = c2, \\
& \text{kx}.p3 = c3 \}
\]

\[
\begin{align*}
\{ & \{ k12 + k13, k22 + k23, k32 + k33 \} = \{ 6, 9, 12 \}, \\
& \{ 3 k11 + 2 k12 + k13, 3 k21 + 2 k22 + k23, 3 k31 + 2 k32 + k33 \} = \{ 11, 18, 11 \}, \\
& \{ 14 k11 + 5 k12 + 2 k13, 14 k21 + 5 k22 + 2 k23, 14 k31 + 5 k32 + 2 k33 \} = \{ 6, 12, 21 \}\}
\end{align*}
\]

We can now use Gaussian elimination, adapted to \( \mathbb{Z}_{26} \), to extract the keys

\[
\text{sol} = \text{Solve}[\text{sys}, \text{Flatten}[\text{kx}], \text{Modulus} \to 26]
\]

\[
\begin{align*}
\{ & \{ k11 \to 1, k12 \to 2, k13 \to 4, k21 \to 9, k22 \to 8, k23 \to 1, k31 \to 0, k32 \to 25, k33 \to 13 \} \}
\end{align*}
\]

\[
\text{sol} = \text{sol}[[1]] \text{ (* There is only one solution *)}
\]

\[
\{ k11 \to 1, k12 \to 2, k13 \to 4, k21 \to 9, k22 \to 8, k23 \to 1, k31 \to 0, k32 \to 25, k33 \to 13 \}
\]

\[
\text{kx} /. \text{sol}
\]

\[
\{ (1, 2, 4), (9, 8, 1), (0, 25, 13) \}
\]

\((\text{kx} /. \text{sol}) = k\)

True

**Substitution Box**

**One-time pad is quite unpractical** because the key size is overwhelming and managing such keys is beyond mundane applications. Therefore, the need to **come up with efficient and usable symmetric** cryptosystems has not vanish with Shannon’s theorem.

Shannon suggested two concepts to develop cryptosystems, that are (quite) informally stated as:

**Confusion**: what makes the relationship between the ciphertext and the symmetric key as complex
and involved as possible.

**Diffusion:** what refers to dissipating the statistical structure of plaintext over the bulk of ciphertext.

Due to the digital revolution, the community has dropped the interest on $\mathbb{Z}_{26}^m$ (natural language messages) and started looking into $\mathbb{Z}_2^m$ (bitstrings), some cool people also use $\text{GF}(2^k)$. There are several mathematical advantages of considering $\mathbb{Z}_p^m$ with $p$ prime, one of them is that $\mathbb{Z}_p^m$ forms is a vector space and not just a module.

As illustrated by the Hill cipher, encryption based on linear transformations can be easily attacked using known plaintexts. To defeat this attack one has to incorporate non-linear maps in the ciphers, which were called called substitution boxes, or shortly S-Box. That is,

$$\text{S-Box}: \mathbb{Z}_2^m \rightarrow \mathbb{Z}_2^m$$

Given that there should not be an easy representation of an S-Box, they have to be memorized into table, that is the S-box can be represented by a list of outputs, where the index (starting at 0) is the input. For instance, for the case $m=4$

![S-box diagram](image)

we can define an S-Box has the list with 16 values ($2^4$).

$$\text{Sbox} = \{14, 4, 13, 1, 2, 15, 11, 8, 3, 10, 6, 12, 5, 9, 0, 7\}$$

$$\{14, 4, 13, 1, 2, 15, 11, 8, 3, 10, 6, 12, 5, 9, 0, 7\}$$

and the map it represents is

$$\text{Sf}[\text{Sbox}, i_] := \text{S}[[i + 1]]$$

14

$$\text{Sf}[\text{Sbox}, 4]$$

2

Because ciphers have to be used in embedded and hardware systems, S-boxes should not be very large, otherwise they require a large amount of memory to store. It is common to consider S-boxes of size 8 and at most of size 16. In some cryptosystems (like Feistel Networks - DES), S-Boxes do not need to be bijective.

**Substitution-permutation network**

Some modern cryptosystems (like AES) are based on substitution permutation networks: The idea is the following:
1) Split the encryption in rounds $i=0...r-1$.

2) Consider a function called the key scheduler ($ks$), that for each round $i$ and given the key $k$ generates a key round.

   $ks(k, i) \mapsto k_i$

   and so we will get key rounds $k_0, k_1..k_{r-1}$.

3) Encryption of the message $x$ is the composition of the each round encryption, that is

   $\text{enc}(k, m) = \text{renc}(k_{r-1}, \text{renc}(k_{r-2}, ... \text{renc}(k_0, x) ...))$.

4) Each round encryption is the decomposed in 3 parts:

   a) The key round is summed (modulo 2) to the message

   b) S-boxes are applied to the result:

      - usually the text is split into several blocks and an S box is applied to each block.

   c) A P permutation is applied to the result (which is a linear transformation).

   that is $\text{renc}(k_i, y) = P \cdot Sf(k_i \oplus y)$.

5) Sometimes the last round (or some rounds) might be different from others in order the simplify the decryption method.

An illustration of substitution permutation network is given below (extracted form Wikipedia).

We note that the only non-linear step of the encryption is that based on the S-Boxes, both the permutation and the summation of the key induce a linear transformation.

**Exercise:** Write informally the decryption method for the substitution-permutation network.