Reverse-Engineering der Transposition-Chiffre

Tamer, Yusuf
78159@studmail.htw-aalen.de

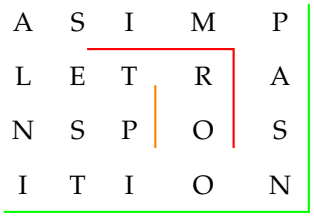
June 15, 2025

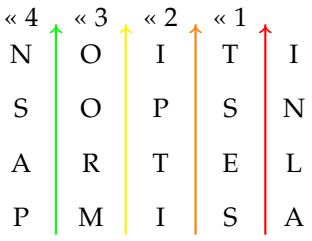
Abstract

Dieses Paper befasst sich mit der Transposition-Chiffre, bei der es um ein symmetrisches Kryptosystem handelt.

Einleitung

Moderne sowohl auch Chiffren aus der Antike basieren auf das Transposition-Verfahren. Dabei handelt es sich um eine Permutation mit der die einzelnen Zeichen vertauscht werden, sodass der eigentliche Klartext damit verborgen wird. Dieses Verfahren ist Zweidimensional, weshalb die einzelnen Zeichen auf einen 2D-Feld verteilt wird. Beispielsweise soll der folgende Klartext:
ASIMPLETRANSPOSITION mit der Schlüssellänge 5 verschlüsselt werden.





N M T S I
S O I E N
A O I S L
P R P T A

Programmierung der Transposition-Chiffre in C++

Formel

Mit folgenden C-Code lässt sich die Verschlüsselung sowohl auch die Entschlüsselung der Transposition-Chiffre implementieren:

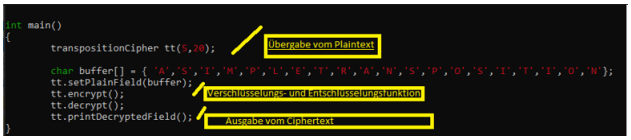
```
for (int i = 0; i< plainTextSize; i++)
{
this->cipherField[i % this->key]
[i % (this->size/this->key)] = plainTextField[i];
}
```

Klasse

```
class transpositionCipher
{
 public:
  int key;
  int size;
  char * plainField;
  char ** cipherField;
  char * decryptedField;
  transpositionCipher(int a, int b);
  void readPlainField();
  void setPlainField(char *buf);
  void encrypt();
  void decrypt();
  void readCipherField();
}
```

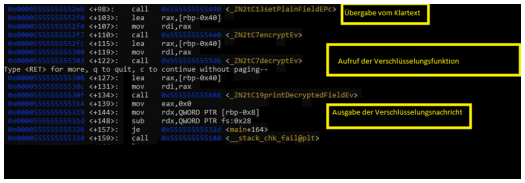
Das Hauptprogramm

Das Hauptprogramm lässt sich folgendermaßen implementieren:



Assembler-Code vom Hauptprogramm

Anhand des Konstruktors wird die Größe vom CipherField und ebenso auch die KeySize festgelegt. In diesem Fall enthaltet das CiperField 20 Zeichen. Die KeySize beträgt 5 und damit passen in die vertikale Zeile dementsprechend nur 5 Zeichen.
Daher wird der Plaintext einer Funktion übergeben, um die Verschlüsselungs-, Entschlüsselungs- und auch die Ausgabefunktion nacheinander aufzurufen.



Chronologischer Ablauf der einzelnen Funktionen, mit der die Transposition-Chiffre implementiert werden kann.

1. transpositionCipher(CipherField,KeySize)
2. setPlainField(buffer)
3. encrypt()
4. decrypt()
5. printEncryption()