Geheime Botschaften

Wie funktioniert Verschlüsselung?

Wer bin ich?

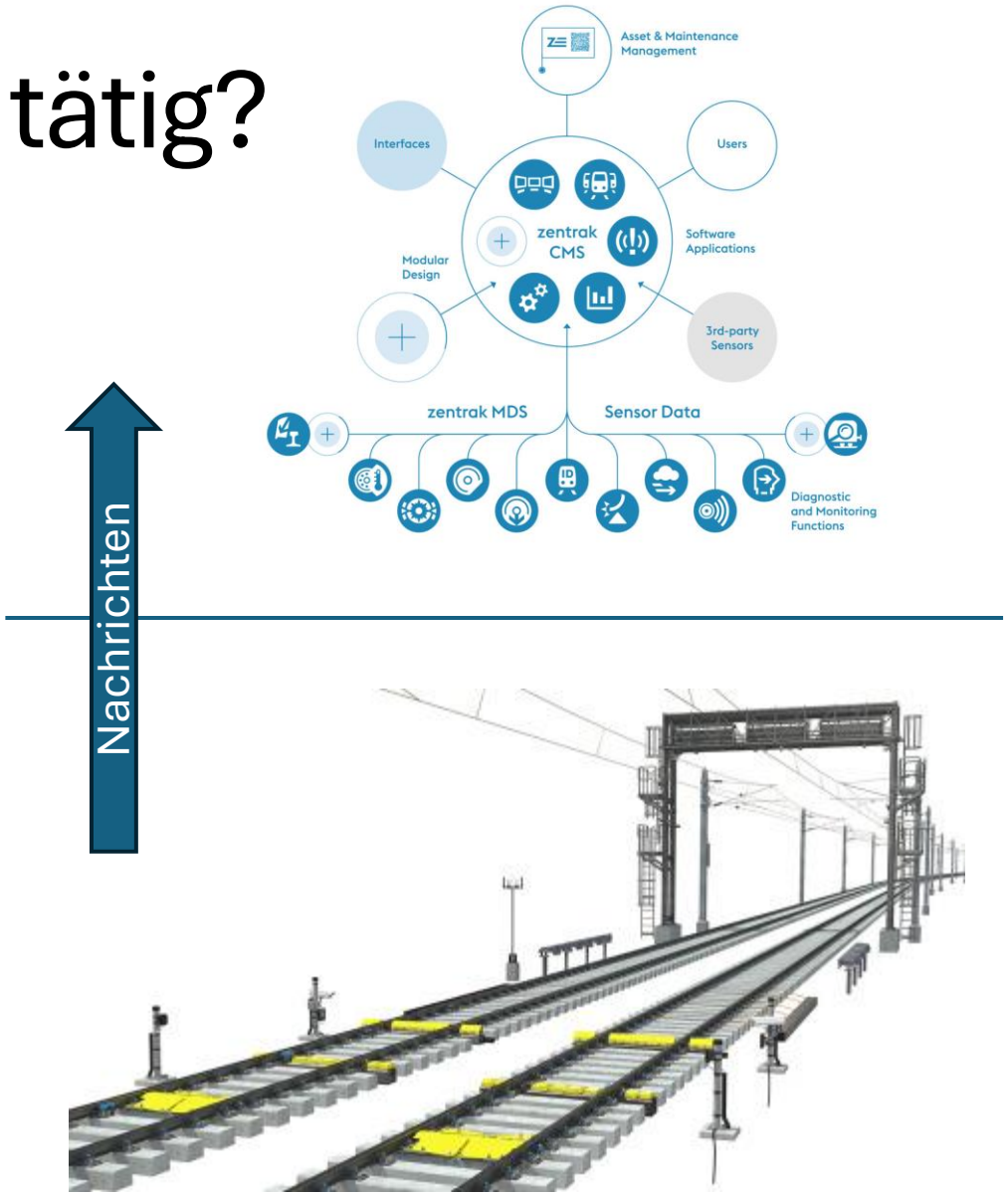
- Mario Demuth (Vater von Dexter Demuth 7a)
- Diplom Informatiker (in Koblenz studiert)
- Angestellt als Softwareentwickler bei *voestalpine Signaling Siershahn GmbH*
- Zuständig für die Ausbildung der Fachinformatiker für Anwendungsentwicklung

Wie wird man Softwareentwickler?

- Autodidakt
- **Lehre** als *Fachinformatiker für Anwendungsentwicklung*
  - 3 Jahre Ausbildung im Betrieb mit begleitender Berufsschule
- **Studium** der *Informatik* (sowohl Uni als auch FH)
  - 3 Jahre Studium für den *Bachelor*
  - +2 Jahre auf dem Bachelor aufbauendes Studium für den *Master*

In welchem Bereich bin ich tätig?

- Wir bauen Sensorsysteme im Gleis, um diverse Eigenschaften (z.B. Temperaturen) von Zügen zu messen, die über unsere Anlagen fahren.
- Zusätzlich bauen wir eine Zentralensoftware, die Messergebnisse für Fahrdienstleiter graphisch aufbereitet.



Agenda

- Verschlüsselte Kommunikation im Allgemeinen
- Die Cäsar Chiffre im Speziellen

Was ist Verschlüsselung?

- Verschlüsselung ist die Umwandlung von **Klartext** in einen **Geheimtext** unter Zuhilfenahme eines geheimen **Schlüssels**.

  Verschlüsseln(Klartext, Schlüssel): Geheimtext

- Mit der Kenntnis über den **Schlüssel** kann mithilfe der Entschlüsselung der **Geheimtext** wieder in den **Klartext** zurückgewandelt werden.

  Entschlüsseln(Geheimtext, Schlüssel): Klartext

Wofür verwenden wir Verschlüsselung? (1/3)

- Auf Verschlüsselung greift man zurück, wenn ein **öffentliches Medium** zum **Nachrichtenaustausch** verwendet werden muss, man aber das **Mithören** unbefugter Dritte **verhindern** möchte.

Wofür verwenden wir Verschlüsselung? (2/3)

- *Problem:* **Alice** und **Bob** sind Lerner derselben Klasse. In der derzeitigen Sitzordnung sind sie keine direkten Tischnachbarn, zwischen ihnen sitzt **Eve**. Alice möchte eine Nachricht an Bob senden, jedoch soll Eve nicht mitlesen können.

Wofür verwenden wir Verschlüsselung? (3/3)

- *Lösung:* Symmetrisch verschlüsselte Kommunikation [1]

1. Alice und Bob vereinbaren einen geheimen Schlüssel.
2. Alice verschlüsselt die Nachricht, die sie an Bob senden möchte mithilfe des geheimen Schlüssels.
3. Alice versendet die verschlüsselte Nachricht an Bob.
   - Alice übergibt die verschlüsselte Nachricht an Eve.
   - Eve übergibt die verschlüsselte Nachricht an Bob.
4. Bob empfängt die verschlüsselte Nachricht.
5. Bob entschlüsselt die Nachricht mit dem vereinbarten Schlüssel.
6. Bob liest die im Klartext vorliegende Nachricht von Alice.

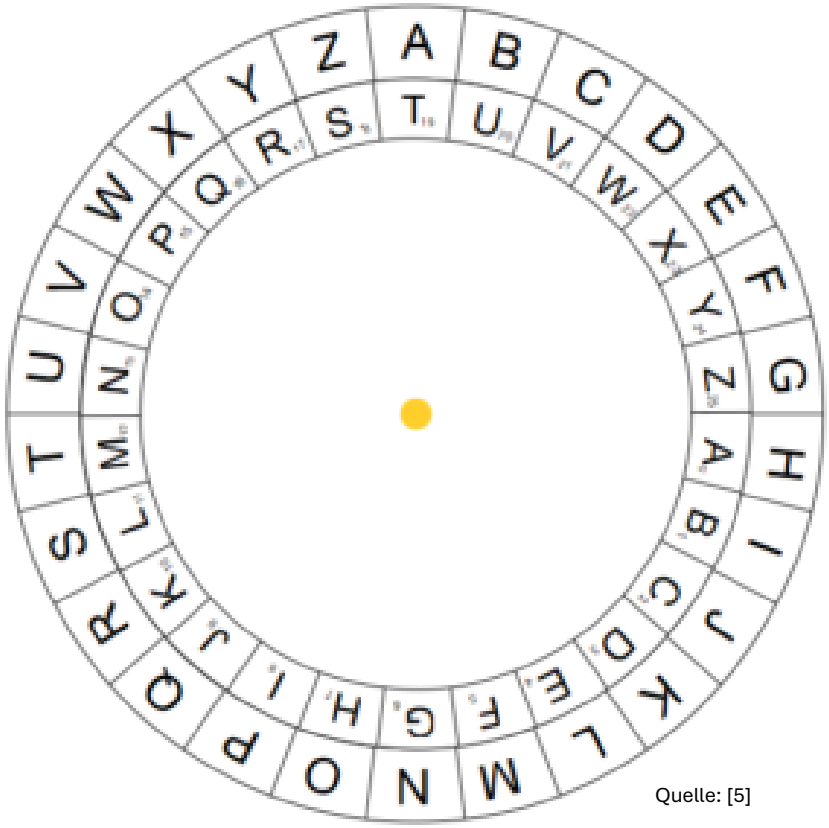
Exkurs: Teile-und-herrsche-Ansatz

- Der gezeigte Lösungsansatz für die symmetrisch verschlüsselte Kommunikation wurde in sechs Teilschritten präsentiert.
- Das eigentliche schwierig erscheinende Problem wurde so lange in kleinere Teilprobleme zerlegt, bis diese beherrschbar (trivial lösbar) sind.
- Anschließend wird aus diesen Teillösungen eine Lösung für das Gesamtproblem rekonstruiert.
- Diesen Lösungsansatz bezeichnet man in der Informatik als Teile-und-herrsche-Ansatz (*engl. divide and conquer, lat. divide et impera*) [2] und findet auch in vielen anderen Bereichen des Lebens und der Politik Anwendung.

Wie funktioniert die Cäsar-Chiffre?

- Die Cäsar-Chiffre [3][4] ist ein **symmetrisches** Verschlüsselungs-verfahren, in dem ein **geordnetes Alphabet** verwendet wird um jeden Buchstaben des Klartexts um eine bestimmte **Anzahl** an **Positionen** nach rechts zu **verschieben**. Der daraus entstandene Geheimtext kann entschlüsselt werden, indem man jeden Buchstaben um dieselbe Anzahl an Positionen nach links verschiebt.
- Ein nützliches Hilfsmittel, um die Ver- und Entschlüsselung der Cäsar-Chiffre per Hand durchzuführen ist die Cäsar-Scheibe [5].

Praxis: Verschlüsseln mit Hilfe der Cäsar-Scheibe (1/2)



Quelle: [5]

Praxis: Verschlüsseln mit Hilfe der Cäsar-Scheibe (2/2)

Verschlüsseln(„RAIFFEISEN-CAMPUS“, 3)
= „UDLIIHLVHQ-FDPSXV “

Wie kann man die Lösung mit Hilfe des Computers umsetzen? (1/2)

```
Verschlüsslen(Klartext, Schlüssel): Geheimtext {

        Geheimtext = „“

         für jeden Buchstaben b aus dem Klartext {
                Geheimtext = Geheimtext + VerschlüsseleBuchstabe(b, Schlüssel)
         }

         gebe Geheimtext zurück;
}

VerschlüsseleBuchstaben(Buchstabe, Schlüssel) Geheimbuchstabe {
      gebe rotierten Buchstaben zurück;
}
```

Wie kann man die Lösung mit Hilfe des Computers umsetzen? (2/2)

- Für jeden Buchstabe: for-Schleife
  - Nehme einen Buchstabe: text.CharAt(…)
  - Rotiere Buchstabe: function (Buchstabe, Rotation) -> Buchstabe
    - Index eines Buchstaben im Text = text.indexOf(…)
    - (Index des Buchstaben + Rotation) mod Alphabetlänge = Index des Geheimbuchstabens
  - Modulo ist der Rest der ganzzahligen Division. Modulo wird fast täglich von euch benutzt, z.B. bei der Berechnung der Uhrzeit. Oft wird zur Uhrzeit 15:00 Uhr meist 3 Uhr (nachmittags) gesagt.
    Das ist die Modulo-Rechnung mit der Zahl 12:
    `15 mod 12 = 3` da
    `15 : 12 = 1 R 3`. (3. Klasse Grundschule)
  - Verbinde Buchstaben: Geheimtext += Buchstabe oder Geheimtext = Geheimtext + Buchstabe

Hat diese Art der Verschlüsselung Schwachstellen?

- **S1**: Der Schlüsselraum hat nur eine Größe von 25 >> Brute Force.
- **S2**: Es gibt nur wenige verschiedene Wörter mit zwei oder drei Buchstaben >> Educated Guess.
- **S3**: Die in der natürlichen Sprache ungleiche Verteilung der Buchstaben wird nicht verborgen, so dass eine Häufigkeitsanalyse das Wirken der Rotation enthüllt (Häufigkeit von E > N > I >...).

Die Sicherheit der Cäsar-Verschlüsselung bestand nicht auf der Geheimhaltung des Schlüssels, sondern im Wesentlichen auf der Geheimhaltung des Verfahrens. Heutige Verschlüsselungsverfahren beruhen auf dem Prinzip von Kerckhoffs [6]: Die Sicherheit des Verschlüsselungsverfahrens soll auf der Geheimhaltung des Schlüssels beruhen, anstatt auf der Geheimhaltung des Algorithmus.

Wie kann man diese Schwachstellen beseitigen?

- **L1**: Das Alphabet erweitern um Kleinbuchstaben, Umlaute, etc.
- **L2**: Das Alphabet erweitern um Leerzeichen, Satzzeichen, etc.
- **L3**: Erweitern des Algorithmus um ein Passwort:
  - Wähle anstatt einer Rotation ein Passwort bestehend aus Buchstaben des Alphabets.
  - Der erste Buchstabe des Klartextes wird um den Index des ersten Buchstaben des Passworts rotiert.
  - Der zweite Buchstabe des Klartextes wird um den Index des zweiten Buchstaben des Passworts rotiert.
  - Wiederhole bis alle Buchstaben des Passworts aufgebraucht sind und fahre fort mit dem ersten Buchstaben des Passworts.
  - Dabei gilt:
    - je länger das Passwort, desto sicherer.
    - Im Idealfall sollte das Passwort mindestens genau so lang sein wie der zu verschlüsselnde Klartext.
    - Verwende niemals das Passwort ein zweites Mal: One Time Pad [7]
    - Die Buchstaben des One Time Pad sollten zufällig gewählt werden.
    - Unter den Bedingungen eines zufällig generierten One Time Pads kann die Verschlüsselung nicht mehr gebrochen werden.

Exkurs: Wie kann man einen geheimen Schlüssel in der Öffentlichkeit vereinbaren?

- Die Wissenschaftler Whitfield Diffie und Martin Hellman haben 1976 festgestellt, wie es geht [8].

Quellen

1. Luber, Stefan (2021-12-22). Wer sind Alice und Bob?
2. Wikipedia (2024-09-23). Teile-und-herrsche-Verfahren
3. Wikipedia (2024-09-23). Caesar Verschlüsselung
4. Shah, Sunny (2023-03-16). An introduction to Caesar Cipher in Cryptography
5. Diehl, Peter (2020-10-26). Die Caesar-Scheibe
6. Wikipedia (2024-09-23). Kerckhoffs‘ Prinzip
7. Wikipedia (2024-09-23). One-Time-Pad
8. inf-schule.de (2024-09-17). Diffie-Hellman-Schlüsselaustausch

Kontakt

mario.demuth@voestalpine.com

Material: github.com/mdemuth/emc-2024/