Netzwerksicherheit & Kryptographie

Kryptographie I

M.Eng T. Yavas

cybermind-academy.com

Übersicht

- Geschichte der Kryptographie
- Grundlagen
- Sicherheitsanforderungen in der Kryptographie
- Symmetrische Verschlüsselung
- Asymmetrische Verschlüsselung
- Hybride Verschlüsselungsalgorithmen
- Public-Key-Infrastruktur (PKI)
- Diffie-Hellman-Schlüsselaustausch
- Transport Layer Security (TLS)
- Ende-zu-Ende-Verschlüsselung
- *Post-Quanten-Kryptographie (neu 2024)*

Zeitstrahl der Kryptographie

Datum	Ereignis
1900 v. Chr.	Ägypter verwenden Hieroglyphen
170 v. Chr.	Polybios entwickelt das Polybios-Quadrat
50–40 v. Chr.	Caesar-Chiffre
1585	Vigenère-Chiffre
1671	Leibniz verwendet das Binärsystem
1895	Erfindung des Radios macht Kryptographie notwendig
1933–1945	Enigma (Zweiter Weltkrieg)
1978	RSA-Algorithmus veröffentlicht
1991	PGP — Pretty Good Privacy
2013	Signal-Protokoll — modernste Ende-zu-Ende-Verschlüsselung
2022	NIST schließt Post-Quanten-Kryptographie-Standardisierung ab
2024	CRYSTALS-Kyber (FIPS 203) & Dilithium (FIPS 204) standardisiert

Grundlagen – Was ist Kryptographie?

Kryptos (griech.) : ich verberge
Graphe (griech.) : ich schreibe

Kryptographie ist die Wissenschaft des Verbergens von Informationen. Ziel der Verschlüsselung ist es, das Lesen von Nachrichten durch Unbefugte so schwierig wie möglich zu machen — *‚unmöglich'* ist dabei bewusst zu vermeiden. Nachrichten werden mit einem Verfahren verschlüsselt, das nur Sender und Empfänger kennen.

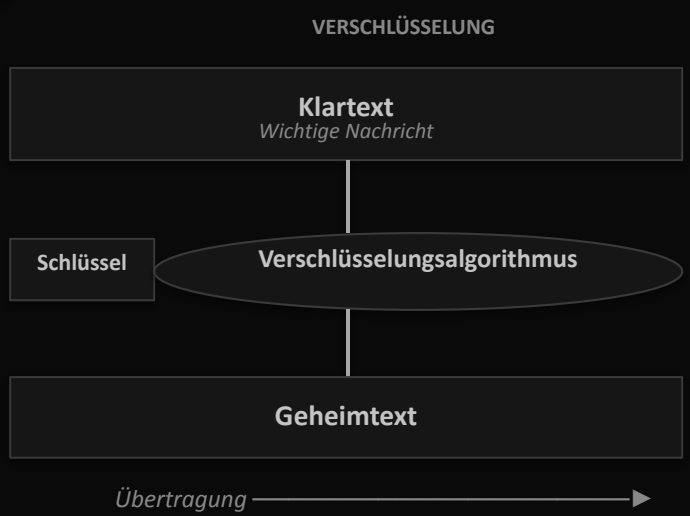
Verschlüsselung ist entscheidend für den Schutz sensibler Informationen — insbesondere bei Datenübertragungen über öffentliche Netzwerke.

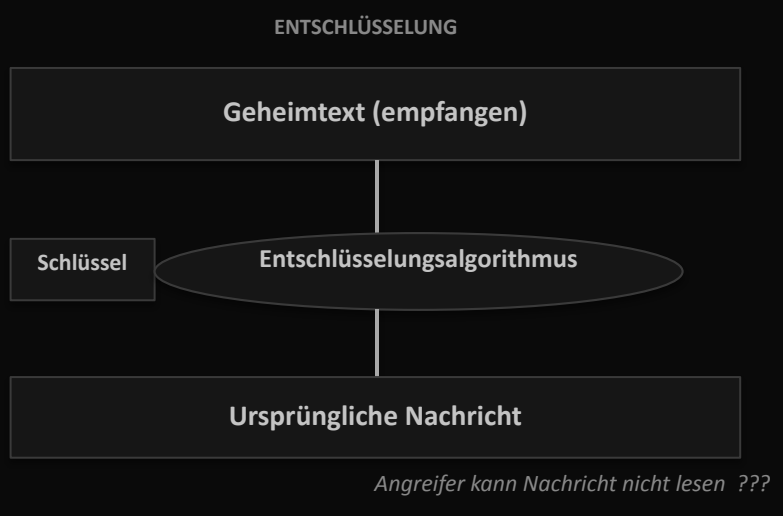
Aktuell 2025: KI-gestützte Kryptanalyse und Quantencomputer erhöhen den Bedarf an quantenresistenten Verfahren erheblich.

Fachbegriffe der Kryptographie

- **Klartext** Eine ursprüngliche Nachricht oder Datei, die jeder mit Zugriff lesen kann.
- **Geheimtext** Durch einen Algorithmus verschlüsselte Daten — für Dritte unlesbar.
- **Verschlüsselung** Der Prozess der Umwandlung von Klartext in Geheimtext.
- **Hash-Funktion** Berechnet aus beliebigen Daten einen Wert fester Länge — dient der Integritätsprüfung.
- **Digitale Signatur** Verschlüsselter Hash einer Nachricht mit dem privaten Schlüssel — beweist Echtheit.
- **Algorithmus** Mathematische Vorschrift für Verschlüsselung, Entschlüsselung und Signaturen.
- **Entschlüsselung** Umwandlung von Geheimtext zurück in lesbaren Klartext.
- **Kryptanalyse** Angriff auf ein Kryptosystem mit dem Ziel, den Schlüssel zu ermitteln.
- **Zero-Knowledge-Beweis** Beweist eine Aussage, ohne weitere Informationen preiszugeben (modern, 2024).

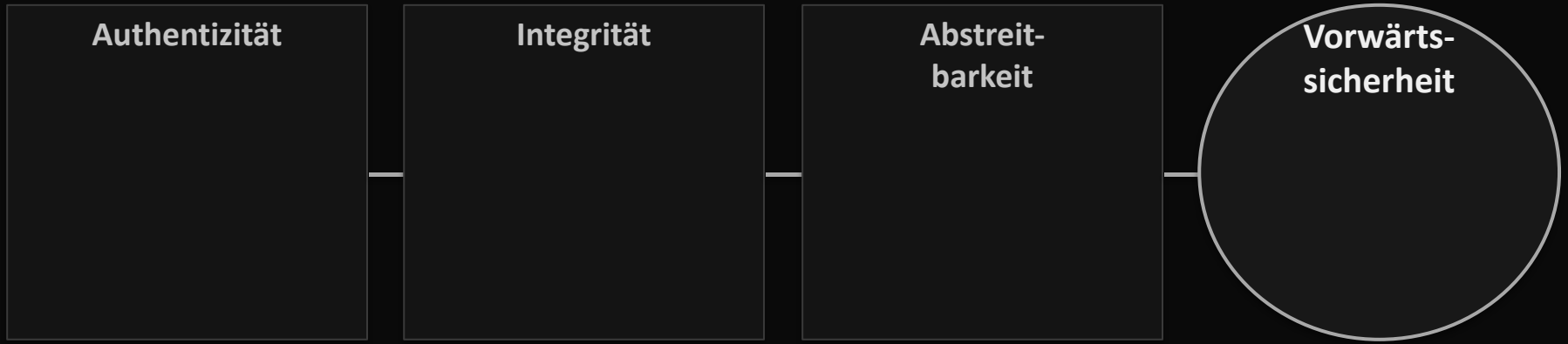
Ver- und Entschlüsselung – Ablauf





Angreifer kann Nachricht nicht lesen ???

Anforderungen an Kryptographie



Authentizität: Identität bestätigt · Integrität: Nachricht unverändert · Abstreitbarkeit: Sender kann leugnen · Vorwärtssicherheit: vergangene Sitzungen sicher

Modernes CIA-Triad: Vertraulichkeit · Integrität · Verfügbarkeit — plus Nicht-Abstreitbarkeit als 5. Anforderung (BSI 2025)

Schlüsselbegriffe der Kryptographie

- **Verschlüsselungsschlüssel** Zeichenfolge, die mit einem Algorithmus Daten ver- oder entschlüsselt bzw. digitale Signaturen erstellt.
- **Schlüsselverschlüsselungsschlüssel** Ein Verschlüsselungsschlüssel, der einen anderen Schlüssel verschlüsselt.
- **Schlüssellänge** Größe eines Schlüssels in Bit. Längere Schlüssel erschweren Angriffe exponentiell.
- **Blockchiffre** Verschlüsselungsalgorithmus, der auf festen Datenblöcken arbeitet (z. B. AES).
- **Stromchiffre** Verschlüsselungsalgorithmus für kontinuierliche Datenströme (z. B. Audio, Video).

BSI-Empfehlungen 2025: AES-256 (symmetrisch) · RSA-3072+ / ECC-384+ · CRYSTALS-Kyber Level 3 oder 5 (Post-Quanten)

Steganographie

- **Was ist das?**
- Kunst des Verbergens von Informationen in anderen, unverdächtigen Daten
- Historischer Hintergrund — seit der Antike bekannt (Geheimtinte, etc.)
- Heute in digitalen Medien verbreitet (Bilder, Audio, Videodateien)
- Beispiel: versteckte Daten in Bild- oder Audiodateien einbetten
- Auch in Druckern eingesetzt — Machine Identification Codes (MIC-Punkte)
- *2025: Deep-Learning-Steganalyse kann versteckte Daten zuverlässig aufspüren*

Siehe: https://null-byte.wonderhowto.com/how-to/steganography-hide-secret-data-inside-image-audio-file-seconds-0180936/

Erste Algorithmen – Caesar-Chiffre (Monoalphabetische Substitution)

- Benannt nach dem römischen Feldherrn Gaius Julius Caesar (100 v. Chr. – 44 v. Chr.)
- Caesar nutzte diese einfache Verschiebungschifffre für militärische Nachrichten

Das Alphabet wird zweimal geschrieben; die untere Zeile wird um den Schlüsselwert verschoben:

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
```

Untere Zeile um 3 Positionen nach rechts verschieben (Schlüssel = 3):

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
```

A→D B→E C→F … Das Wort „Beispiel" wird zu: „EHLVSLHO"

Interaktiv: https://www.cryptool.org/en/cto-ciphers/caesar

Erste Algorithmen – Caesar-Chiffre – Beispiel

Interaktives Tool: https://www.cryptool.org/en/cto-ciphers/caesar

Klartext (Schlüssel = 3):

```
Cybermind Academy ist wirklich cool
```

↓ Verschlüsseln mit K = 3

Geheimtext:

```
FABHUPLQG DFDGHPB LVW ZLUNOLFK FRRO
```

```
Klartextalphabet:      A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Geheimtextalphabet: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
```

Schlüssel = 3 | Alternativ: Rot-13 (K = 13) | 26 mögliche Schlüssel → leicht brutal zu brechen

Kodierung

- **Was ist das?**
- Übersetzung von Daten von Format A in Format B
- Kodieren und Dekodieren von Bytes
- Beispiele: Barcode · JSON · Base64 · QR-Code · ASCII
- *Hinweis: Kodierung ist KEINE Verschlüsselung — kein Geheimschlüssel, keine Sicherheit!*

Siehe: https://www.cryptool.org/en/cto-codings/base64

ASCII

- ASCII (American Standard Code for Information Interchange) ist ein Zeichenkodierungsstandard zur Darstellung von Text in Computern und anderen Geräten.
- Erweitertes ASCII (8-Bit) umfasst 256 Zeichen; Unicode (UTF-8) kodiert über 1,1 Millionen Zeichen.

Siehe: https://de.wikipedia.org/wiki/ASCII · https://www.cryptool.org/en/cto-codings/ascii

Funktionsweise — Klartext → Dezimal → Binär:

```
Klartext:  A       B       C       D       E       F
Dezimal:  65      66      67      68      69      70
Binär: 01000001 01000010 01000011 01000100 01000101 01000110
```

Kryptanalyse

- Ziel: Erlangung von Kenntnissen über den verschlüsselten Text ohne Besitz des Schlüssels.

Unterschiedene Angriffsszenarien in der Kryptanalyse:

- **Nur-Geheimtext-Angriff** Nur der Geheimtext ist bekannt; Klartext und Schlüssel werden gesucht.
- **Wahrscheinlicher Klartext** Es wird angenommen, dass die Nachricht einem bestimmten Muster folgt.
- **Bekannter Klartext** Ein Geheimtext und sein zugehöriger Klartext sind bekannt; Ziel: Schlüssel bestimmen.
- **Gewählter Klartext** Angreifer verschlüsselt selbst gewählte Klartexte und analysiert den Geheimtext.
- **Gewählter Geheimtext** Angreifer kann beliebige Geheimtexte entschlüsseln und das Ergebnis analysieren.

Quantenbedrohung (Shors Algorithmus): Quantencomputer können RSA & ECC in polynomialer Zeit brechen. Post-Quanten-Kryptographie (NIST 2024) liefert quantenresistente Alternativen.

Siehe: https://www.cryptool.org/en/cto-cryptanalysis

Vigenère-Chiffre

- Entwickelt im 16. Jahrhundert vom französischen Kryptologen Blaise de Vigenère (* 15. April 1523 in Saint-Pourçain – † 1596)
- Basiert auf der Caesar-Chiffre, verwendet jedoch wechselnde Alphabete (polyalphabetisch)
- Galt lange als unknackbar — bekannt als 'le chiffre indéchiffrable'
- Friedrich Wilhelm Kasiski veröffentlichte 1863 eine Methode zur Entschlüsselung (Kasiski-Test)

Vigenère-Chiffre – Beschreibung

- Ein Schlüssel beliebiger Länge wird gewählt; Text und Schlüssel verwenden dasselbe Alphabet (A–Z)
- Der Schlüssel wird wiederholt unter den Klartext geschrieben, bis die volle Länge gedeckt ist:

```
Klartext:    D I E S  I S T  E I N  G E H E I M E R  T E X T
Schlüssel:   K E Y K  E Y K  E Y K  E Y K E Y K E Y  K E Y K
```

- Mit der Caesar-Chiffre: jedes Klartextzeichen wird durch das entsprechende Schlüsselzeichen kodiert (K = 11. Buchstabe → Verschiebung um 10):

```
Klartext:    A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Chiffretext: K L M N O P Q R S T U V W X Y Z A B C D E F G H I J
```

D (4. Buchstabe) + K (Verschiebung 10) = N → Erstes kodiertes Zeichen: N

Vigenère-Chiffre – Beschreibung Teil II

- Das zweite Zeichen I wird durch Schlüssel E (5. Buchstabe → Verschiebung um 4) kodiert:

```
Klartext:    A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Chiffretext: E F G H I J K L M N O P Q R S T U V W X Y Z A B C D
```

I wird zu M → Bisher kodierter Text: NM

Kodierung der vollständigen Nachricht nach demselben Verfahren:

```
Klartext:    DIES IST EIN GEHEIMER TEXT
Schlüssel:   KEYK EYK EYK EYKEYKEY KEYK
Chiffretext: NMCC MQD IGX KCRIGWIP DIVD
```

- Sicherheit abhängig von der Schlüssellänge — kurze oder wiederholte Schlüssel sind durch den Kasiski-Test (1863) angreifbar. Heute durch Verfahren wie AES ersetzt.

Vigenère-Chiffre – Interaktives Beispiel

Interaktives Tool: https://www.cryptool.org/en/cto-ciphers/vigenere

Beispiel — Schlüssel: K (K = 11. Zeichen → Verschiebung um 10)

Klartext:

```
Hallo
```

↓ Verschlüsseln mit Schlüssel = K

Verschlüsselter Text:

```
Rkvvy
```

```
H+10=R   a+10=k   l+10=v   l+10=v   o+10=y   →   R k v v y
```

Hinweis: Groß-/Kleinschreibung wird berücksichtigt — 'H' (Großbuchstabe) → 'R'

Quellen

- https://www.cryptool.org/en/cto-codings/base64
- https://www.cryptool.org/en/cto-cryptanalysis
- https://null-byte.wonderhowto.com/how-to/steganography-hide-secret-data-inside-image-audio-file-seconds-0180936/
- https://www.cryptool.org/en/cto-ciphers/vigenere
- NIST PQC-Standards (2024): https://csrc.nist.gov/projects/post-quantum-cryptography
- BSI – Post-Quanten-Kryptographie: https://www.bsi.bund.de/quantenkryptographie
- Signal-Protokoll-Spezifikation: https://signal.org/docs/
- TLS 1.3 (RFC 8446): https://www.rfc-editor.org/rfc/rfc8446