max planck institut
informatik

Kryptograhie
Wie funktioniert Electronic Banking?

Kurt Mehlhorn

Adrian Neumann

Max-Planck-Institut für Informatik

Übersicht

- Zwecke der Krytographie
- Techniken
  - Symmetrische Verschlüsselung(One-time Pad, Caesar, moderne Blockchiffres)
  - Asymmetrische Verschlüsselung, Public-Key Kryptographie (1978)
  - Digitale Unterschriften
- Anwendungen: Electronic Banking, Sicherheitsinfrastrukturen

Kryptographie (geheim-schreiben)

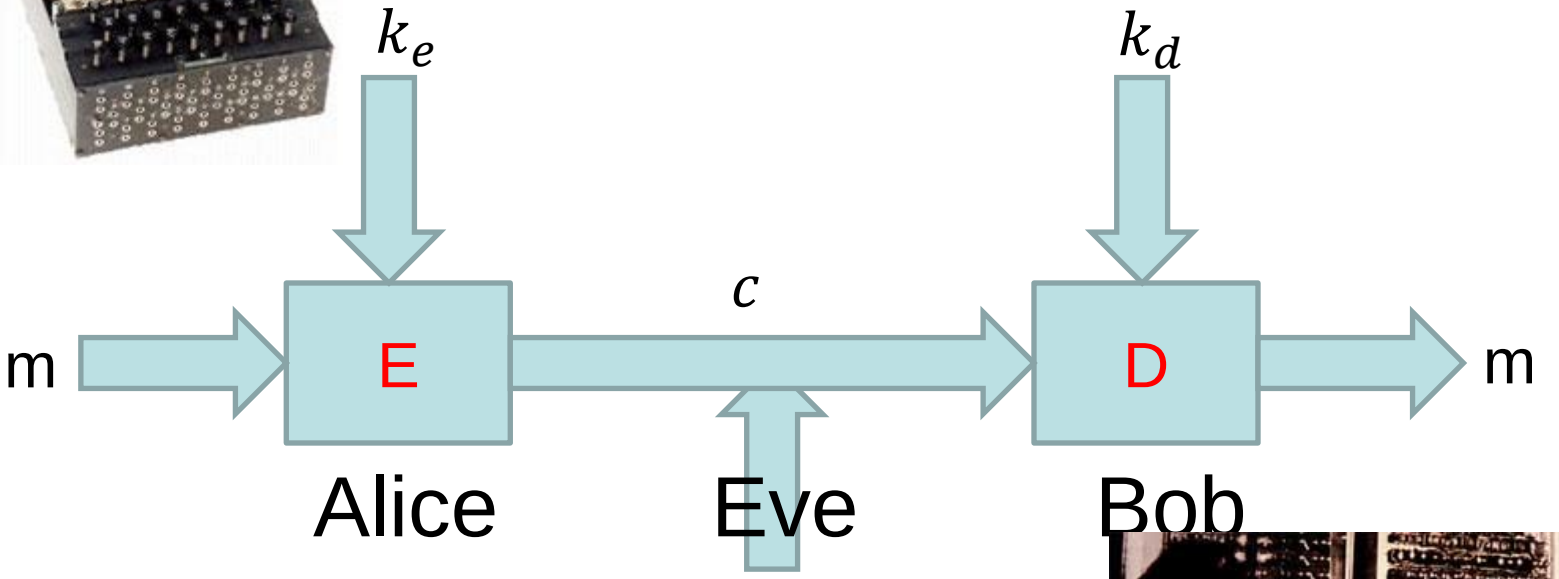
Hauptziele der modernen Kryptographie (Wolfgang Ertel)

- Vertraulichkeit / Zugriffsschutz: Nur dazu berechtigte Personen sollen in der Lage sein, die Daten oder die Nachricht zu lesen (auch teilweise).
- Integrität / Änderungsschutz: Die Daten müssen nachweislich vollständig und unverändert sein.
- Authentizität / Fälschungsschutz: Der Urheber der Daten oder der Absender der Nachricht soll eindeutig identifizierbar sein, und seine Urheberschaft sollte nachprüfbar sein.
- Verbindlichkeit / Nichtabstreitbarkeit: Der Urheber der Daten oder Absender einer Nachricht soll nicht in der Lage sein, seine Urheberschaft zu bestreiten.

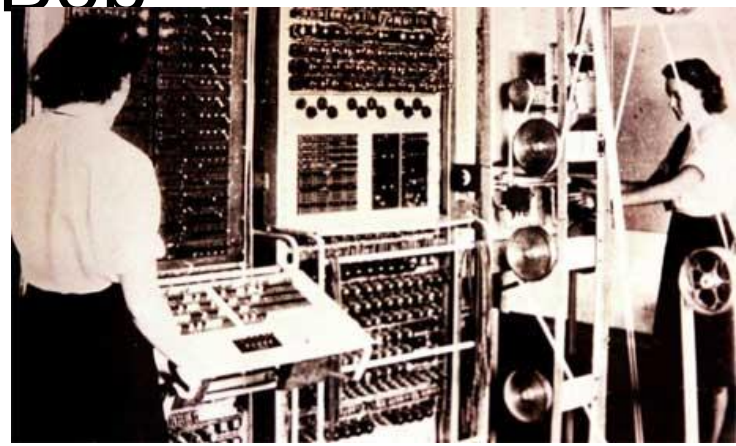
Ver- und Entschlüsselung

Vorhängeschloss

$$m = D(k_d, E(k_e, m))$$



Eve = Eavesdropper



Ver- und Entschlüsselung

- m, Nachricht, Text, Klartext, message
- k_e Schlüssel zum Verschlüsseln, encode
- k_d Schlüssel zum Entschlüsseln, decode
- $k = k_e = k_d$ symmetrisches Verfahren
- $c = E(k_e, m)$, E ist Verschlüsselungs-funktions, c ist Nachricht, cipher,
- $m = D(k_d, c)$, D ist Entschlüsselungs-funktions, $m = D(\ , E(\ , m))$!!!!!!!

Symmetrische Verfahren

- Sender (Alice) und Empfänger (Bob) benutzen den **gleichen** Schlüssel
- Dieser Schlüssel muss geheim bleiben
- Wie einigt man sich auf einen Schlüssel?
  - Früher: physisches Treffen zum Schlüsselaustausch, rotes Telefon
  - Heute: asymmetrisches Verfahren zum Schlüsselaustausch

One-Time Pad

- m ist Folge von Großbuchstabe + ZR
- k ist zufällige Folge über diesem Alphabet
- m_i, k_i , jeweilige i-te Buchstaben $\{0, \ldots, 26\}$
- $c_i = m_i + k_i \; modulo \; 27$
- Beispiel: 5 + 24 mod 27 = 2
- Decoding: $m_i = c_i - k_i) \; mod \; 27$
- Absolut sicher, aber Schlüssel muss genauso lang wie Nachricht sein, rotes Telefon

Caesar

- m ist Folge von Großbuchstabe + ZR
- k ist Zahl zwischen 0 und 26
- m_i, i-ter Buchstaben $\{0, \ldots, 26\}$
- $c_i = m_i + k \; modulo \; 27$
- Beispiel: 5 + 24 mod 27 = 2
- Decoding: $m_i = c_i - k \; mod \; 27$
- Sehr unsicher, aber kurzer Schlüssel

Blockchiffrierung

- Nachricht wird in Blöcke der Länge b zerlegt. Jeder Block wird getrennt kodiert.
- Alle mit dem gleichen Schlüssel.
- Typisch Blocklänge 64, 128, 256 Bits
- Schlüssellänge ähnlich
- Populäre Verfahren: DES (Data-Encryption-Standard), AES (Nachfolger)
- Sicherheit: nicht gebrochen, aber …

Blockchiffrierung: Prinzip der Vorgehensweise

- Kodierung eines Blocks der Länge b
- Verknüpfe mit dem Schlüssel (wie im One-Time Pad)
- Wende Substitution auf Paare

benachbarter Buchstaben an

- Permutiere die Positionen
- Wiederhole 16 Mal.

Angriffe

- Caesar: Buchstabenhäufigkeit
- DES 56: brute-force mit Spezialhardware
- ENIGMA: Alan Turing und einer der ersten Computer
- Siehe Wikipedia: Cryptanalysis für weitere Beispiele
- AES 128 gilt als sicher für die nächsten 10 Jahre

Asymmetrische Verfahren (seit 78)

- Sender (Alice) und Empfänger (Bob) benutzen verschiedene Schlüssel
- Bob erzeugt Schlüssel k_e und k_d, hält k_d geheim, veröffentlicht k_e
- Alice benutzt k_e zum Verschlüsseln
- Aus k_e kann man k_d nach heutiger mathematischer Kenntnis nicht berechnen
- Vorhängeschloss

Sicherheit

- RSA (Rivest-Shamir-Adleman, Turing Award), Rabin (Turing Award) : Faktorisierung von Zahlen mit 2000 Ziffern braucht nach Stand der Kunst Jahrzehnte (unter Nutzung aller Rechner)
- El Gamal: das gleiche gilt für diskreten Logarithmus bezüglich 2000 stelliger Primzahl

Baby-Version von ElGamal

- Folge Bongartz/Unger (Alg der Woche)
- Annahme: Wir können multiplizieren und addieren, aber dividieren ist sehr sehr schwer, also
- Aus p und f kann man $P = p \times f$ berechnen, aber aus
- f und $P = p \times f$ kann man p nicht berechnen

Baby-Version von ElGamal

- **Empfänger** wählt p und f; veröffentlicht f und $P = p \times f$; p bleibt geheim
- **Sender** möchte m schicken, $m < P$
- Wählt eine Zahl s und schickt öffentlich

$$s \times f \text{ und } N = m + s \times P$$

Er hält s geheim.

- **Eve** kann s nicht berechnen und weiß nur

$$m \in \{N, N - P, N - 2P, N - 3P, \ldots\}$$

Baby-Version von ElGamal

- Empfänger wählt p und f und veröffentlicht f und $P = p \times f$
- Sender möchte m schicken, $m < P$
- Wählt eine Zahl s und schickt öffentlich

$$s \times f \text{ und } N = m + s \times P$$

- **Empfänger** berechnet

$$p \times (s \times f) = s \times P$$

und dann $m = N - s \times P$

Rechnen mod n

- Grundmenge = $\{0,1,\ldots,n-1\}$, etwa $n = 7$
- Addition, Subtraktion, Multiplikation mod n

Bringe Ergebnis durch Restbildung wieder in die Grundmenge

$$4 \times 6 = 36 \equiv 1 \; mod \; 7$$

$$3 + 4 \times 2 = 11 \equiv 4 \; mod \; 7$$

- n prim, dann gibt es zu jedem $a \neq 0$ ein b so dass $a \times b \equiv 1 \; mod \; n$ und es gibt ein g so dass $\{g, g^2, \ldots, g^{n-1}\} = \{1, \ldots, n-1\}$

Multiplikationstafel mod 7

- 3 ist Erzeuger mod 7, aber 2 ist keiner.

ElGamal

- Empfänger wählt Primzahl p, Erzeuger g und $x, 2 \leq x \leq p-1$ und veröffentlicht
$$(p, g, y = g^x \bmod p)$$
- Berechnung von y aus x ist leicht, aber von x aus y ist unmöglich
- Sender möchte m schicken, wählt s und schickt
$$(z = g^s \bmod p, N = m \times y^s \, mod \, p)$$

ElGamal

- Empfänger wählt Primzahl p, Erzeuger g und $x, 2 \leq x \leq p-1$ und veröffentlicht

$$(p, g, y = g^x \bmod p)$$

- Sender möchte m senden, wählt s, sendet

$$(z = g^s \bmod p, N = m \times y^s \, mod \, p)$$

- Eve kann s nicht berechnen und weiß nur $m \in \left\{N, \frac{N}{y}, N/y^2, N/y^3, \ldots\right\}$

ElGamal

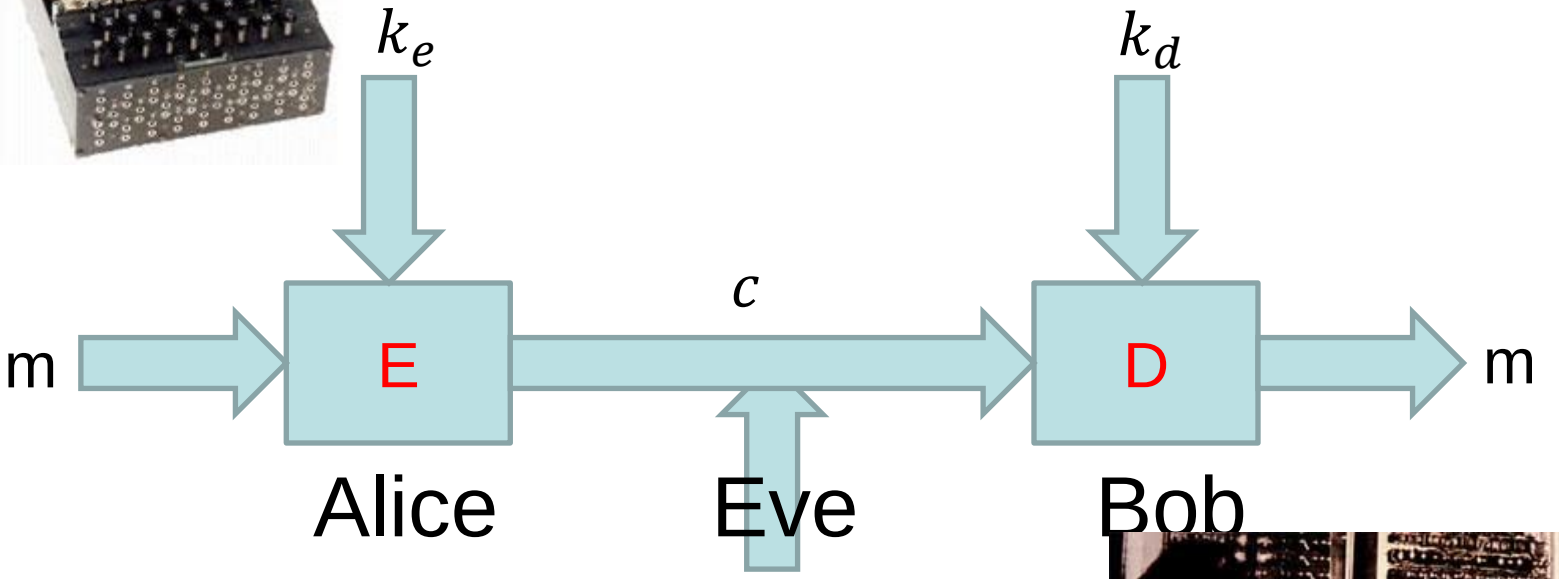
- Empfänger wählt Primzahl p, Erzeuger g und $x, 2 \leq x \leq p-1$ und veröffentlicht
$$(p, g, y = g^x \bmod p)$$
- Sender möchte m senden, wählt s, sendet
$$(z = g^s \bmod p, N = m \times y^s \, mod \, p)$$
- Empfänger berechnet $z^x = g^{sx} = y^s$ und dann $m = N / y^s \, mod \, p.$

Electronic Banking

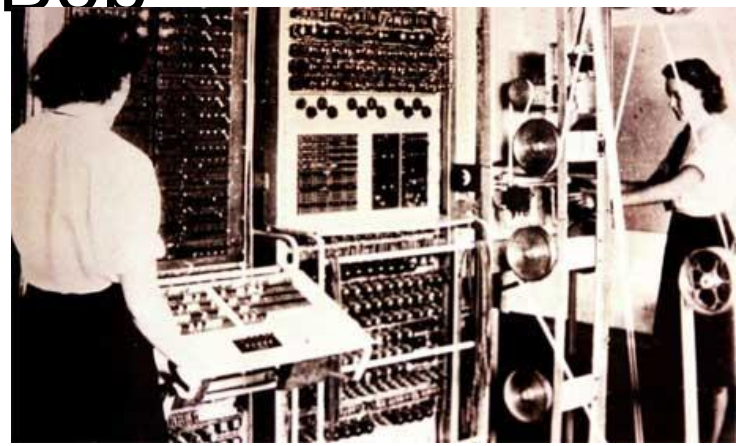
- Kunde sucht öffentlichen Schlüssel k_e der Bank
- Kunde erfindet geheimen Schlüssel k (256 Bit Zufallszahl) für symmetrisches Verf.
- Kunde verschlüsselt k mit k_e und schickt den verschlüsselten Schlüssel an die Bank
- Bank entschlüsselt mit Hilfe ihres privaten Schlüssels k_d
- Nun symmetrisches Verfahren mit k.

Ver- und Entschlüsselung

$$m = D(k_d, E(k_e, m))$$

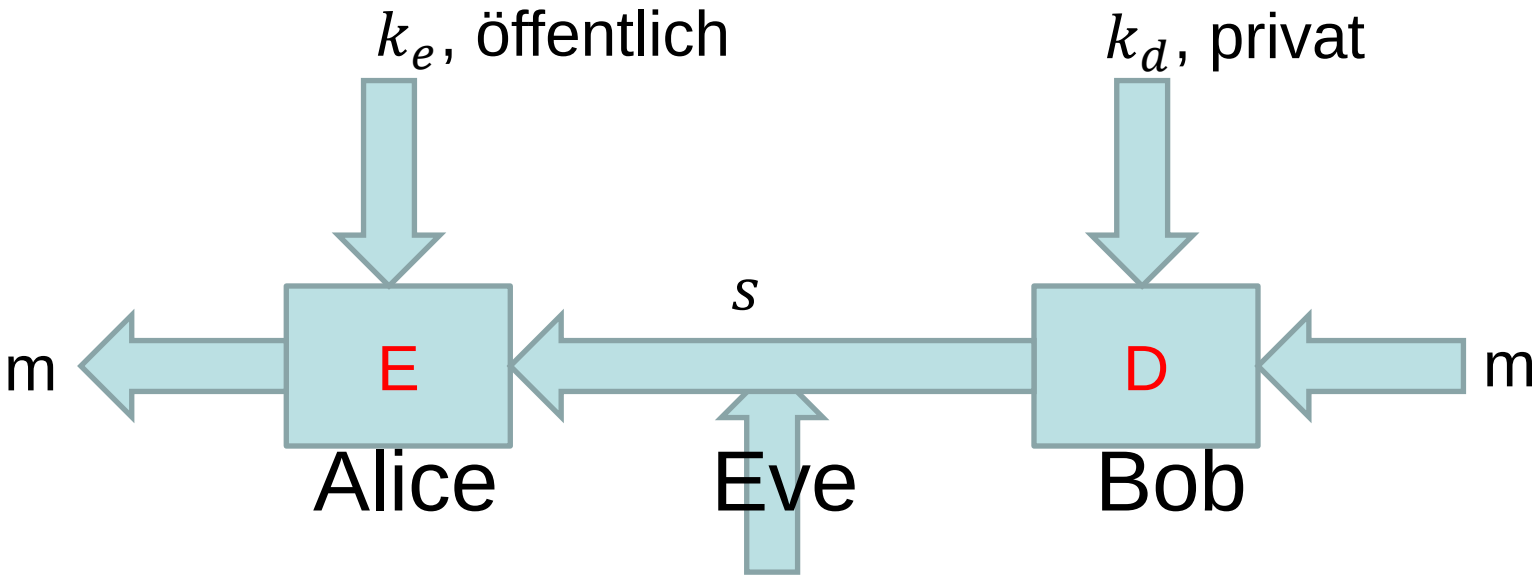


Eve = Eavesdropper



Digitale Signaturen
Signatur = etwas, das nur ich kann

$$m = E\left(k_e, D(k_d, m)\right)$$



Eve = Eavesdropper s = Signatur von m

Electronic Banking, Schritt 1

- Bank hinterlegt ihren öffentlichen Schlüssel k_e bei einem Trustcenter
- Kunde kennt (fest eingebaut im Browser) den öffentlichen Schlüssel des TC und fragt nach Schlüssel der Bank
- TC signiert k_e und schickt an Kunden
- Kunde verifiziert und benutzt dann k_e

Zusammenfassung

- Electronic Banking, Einkaufen im Netz nutzt symmetrische und asymmetrische Kryptographie
- Kommunikation mit der Bank ist damit geschützt https://my.hypovereinsbank.de/
- Aber Vorsicht: für die Qualität ihrer PIN und Passwörter sind sie selbst verantwortlich

Gleiche Geburtstage

- Haben zwei Personen in diesem Raum den gleichen Geburtstag?

- Version 1: Tag Monat
- Version 2: Tag Monat Jahr