1 Kryptologie - Allgemeines

Kryptographie	Entwicklung und Anwendung von Verschlüsselungen.
Kryptoanalyse	Testen und knacken von Verschlüsselungen.
Steganographie	Verbergen von geheimen Botschaften.

1.1 Entwicklung von Algorithmen

Bewährtes Vorgehen: Öffentliche Entwicklung von Algorithmen. Große Zahl von Schlüsseln. Nur Schlüssel geheim.

symmetrisch	gleicher Schlüssel für Ver- und Entschlüsselung.
asymmetrisch	öffentlicher und privater Schlüssel.

1.2 Kryptoanalyse

Brute-Force	Vollständiges Durchsuchen des Schlüsselraums.
Ciphertext-only	einer oder mehrere Chiffretexte sind bekannt.
Known-plaintext	Einige Chiffre wie auch Klartext bekannt.
Chosen-plaintext	wählbarer zu verschlüsselnder Text.
Häufigkeitsanalyse	Buchstabenhäufigkeit einer Sprache.

1.3 perfekte Sicherheit

ist gewährleistet, wenn $|M| = |C| = |K|$ die Länge der Nachricht, des Chiffrats und des Schlüssels gleich lang ist und alle Chiffre gleich wahrscheinlich sind.

Kerckhoffs'sches Prinzip: Die Sicherheit eines Kryptosystems darf nicht von der Geheimhaltung des Algorithmus abhängen, sondern nur von der Geheimhaltung des Schlüssels!

Shannon:

Große Konfusion: Jede Chiffrekombination soll gleich wahrscheinlich auftreten

Große Diffusion: Jede kleine Änderung am Klartext oder Schlüssel soll eine große Änderung im Chiffre bewirken. Sicherheit gegen bekannte Angriffe.

2 Mathematische Grundlagen

Euklidscher Satz: $a = q \cdot n + r$

$a \mod n = a \mod (-n)$

$a + i \cdot n \mod n = a \mod n = r$

$(a_1 + a_2) \mod n = (a_1 \mod n + a_2 \mod n) \mod n$

$(a_1 \cdot a_2) \mod n = (a_1 \mod n \cdot a_2 \mod n) \mod n$

2.1 größter gemeinsamer Teiler ggT

$\mathrm{ggT}(a, 0) = |a| \quad \mathrm{ggT}(a + i \cdot n, n) = \mathrm{ggT}(a, n)$

Rekursive Definition: $\mathrm{ggT}(a, n) := \mathrm{ggT}(n, a \mod n)$

Erweiterter Euklidische Algorithmus wichtig!

2.2 Algebraische Systeme

Eine Algebraische Struktur ist eine Menge S mit einer oder mehreren binären Verknüpfungen $* : S \times S \to S$, die bestimmte Axiome erfüllen.

Struktur	Definition
Halbgruppe $(S, *)$	$*$ ist assoziativ
Monoid $(S, *)$	Halbgruppe mit neutralem Element e
Gruppe $(S, *)$	Monoid mit Inversem $^{-1}$
Abelsche Gruppe $(S, *)$	Gruppe, so dass $*$ kommutativ ist.
Ring $(S, +, \cdot)$	$(S, +)$ ist abelsche Gruppe, $(S, \cdot)$ ist Halbgruppe und es gilt das Distributivgesetz
Körper(Field) $(S, +, \cdot)$	Ring, so dass $(S \setminus \{0\}, \cdot)$ abelsche Gruppe ist. (Multiplikativ Inverse)

Kommutativer Ring: abelsche Gruppe $\langle S, \oplus \rangle$ und Monoid $\langle S, \odot \rangle$

2.3 Modulo-Arithmetik

Additiv Inverses $-a = n - a$

Ein Element a hat nur dann ein multiplikativ inverse Element a^{-1} wenn $\mathrm{ggT}(a, n) = 1$

$(a_1 + a_2) \mod n = (a_1 \mod n + a_2 \mod n) \mod n$

$(a_1 \cdot a_2) \mod n = (a_1 \mod n \cdot a_2 \mod n) \mod n$

$??(a_1 \cdot a_2) \mod n = (n - r) \cdot (n - r) \mod n??$

Der Galois-Körper $GF(p) = \langle \mathbb{Z}_p, + \mod p, \cdot \mod p \rangle$ ist besonders wichtig.

Quadratzahlen: $y = a^x \mod p$

Zu jedem Quadratischen Rest gibt es zwei mögliche Wurzeln

Jacobitest: $J(a, p) = a^{\frac{p-1}{2}} \mod p$

$J(a, p) = 1 \mod p \Rightarrow$ Quadratischer Rest

$J(a, p) = -1 \mod p \Rightarrow$ Quadratischer NichtRest (keine Wurzel)

Trick: $a = \pm a^{\frac{p+1}{2}} = \pm (a^2)^{\frac{p+1}{4}} \mod p$ falls $p \mod 4 = 3$

Falls p nicht Prim, dann Durchlauf für jede Teilprimzahl. Ergebnis miteinander multiplizieren. 2 Teilprimzahlen $\Rightarrow$ 4 Ergebnisse

3 Sicherheitsdienste

- Verbindlichkeit, Nachweisbarkeit (Nicht-Abstreitbarkeit)
- Authentifikation (Identitätsnachweis)
- Integrität (Unverändert?)
- Vertraulichkeit (Geheimhaltung)
- Anonymität
- Berechtigung (Zugangskontrolle)

3.1 OSI Sicherheitsarchitektur

Was kann passieren?	Bedrohungen	Abhören,Manipulation
Sicherheitsgrad?	Regeln	
Gegenmaßnahmen?	Sicherheitsdienste	Vertraulichkeit
Hilfmittel?	Mechanismen	Verschlüsselung
Verfahren?	Algorithmen	AES

3.2 MAC – Message Authentification Code

64,128 oder 160 bit Wert um Integrität und Authentizität nachweisen.

Initwert I, $MAC = g(m, k, g(m, k, ...))$

3.3 Einwegfunktionen

Bsp: Multiplikation von zwei großen Primzahlen.

3.4 Krypto-Hash-Funktionen

Nicht bijektive Einwegfunktion um lange Nachrichten auf kurze Hashwerte abzubilden. $h(m)$

4 Verschlüsselungsverfahren

Verschlüsselung ist eine bijektive Abbildung: $f : M \leftrightarrow C$

m: Klartext, c: Chiffre, k: Schlüssel

symmetrisch: $c = f(k, m) \qquad m = f^{-1}(k, c)$

Anzahl der Schlüssel bei n Teilnehmern: $|K| = \frac{n(n-1)}{2}$

asymmetrisch: $c = f_e(m) \qquad m = f_d(c)$

Sender **ver**schlüsselt mit **fremden** öffentlichen Schlüssel. Empfänger **ent**schlüsselt mit **eigenem** privaten Schlüssel.

4.1 Permutationsalgorithmen

Vertauschen der Buchstabenpositionen in einem Text.

4.1.1 Skytala

500 v. Chr. in Sparta: Aufwickeln eines Gürtels auf einen Stock.

4.2 Substitutions-Algorithmen

Jeweils ein Klartextbuchstabe wird durch einen Chiffrebuchstaben ersetzt. Kann oft durch Häufigkeitsanalyse (HA) gebrochen werden.

Ceasarcode	1 Fester Verschiebungsschlüssel	26
Vigenère-Chiffre	r periodische Schlüssel	26^r
Verman-Chiffre	binär, einmalig, Strom-Chiffre	∞
Enigma	3 aus 5 Rotationsscheiben	10^{23}

Caesar: $C_k = in + k \mod 26$

Caesar arbeitet auf einer Gruppe $\langle \mathbb{Z}_26, + \mod 26 \rangle$

Vigenère: Polyalphabetische Substitution.

Angriff: Wiederholung von Buchstabenfolgen, Primfaktoren ergeben Schlüssellänge, dann HA

Schlüssellänge h für Koinzidenzindex k: $h \approx (k_d - k_r)n/[(n-1)k - k_r n + k_d]$

4.3 Blockchiffre

Blockweise Verschlüsselung. Typische Blocklänge: 64Bit, typische Schlüssellänge 128Bit – 256Bit.

4.4 DES

Digital Encryption Standard 64-Schlüssel bestehend aus 56 Bit + 8 Parity Bits

Permutation und Shifts nötig, damit jedes Schlüsselbit jedes Textbit beeinflussen kann. Insgesamt 28 Bit Schlüsselshifts

F-Funktion: Expansion von 32 Textbits auf 48 durch Bitverdopplung. Aufteilung in 8 mal 6 Bit: S-Funktion: 1. und 6. Bit bestimmen Zeile:

	0000	0001	...
00			
01			S-Bos ist einzige nichtlinearität
10			
11			

S-Box hat 6 inputs und 4 outputs

Verschlüsselung: $R_{16} = L_{15} \otimes F(R_{15}, k_{16})$

4.4.1 IDEA

4.4.2 AES

Blocklänge: 128Bit; Schlüssellänge: 128Bit - 256Bit.

Modulpolynom $M(x) = x^8 + x^4 + x^3 + x + 1 = \{100011011\}$

Substitution ist die einzige nichtlineare Operation.

4.4.3 Block-Operations-Modi

- ECB (electronic Codeblock): Jeder Block wird unabhänig verschlüsselt. Sollte nur für Nachrichten < 1 Block verwendet werden.
- CBC (Cipher Block Chaining): Cipherblock wird mit nächsten Klartextblock verkettet. (XOR)
- CFB (Cipher Feedback): Selbstsynchronisierende Stromchiffre.
- OFB (Output Feedback): keine Selbssync.
- CTR (Counter):

nonce: number used only once

4.5 Stromchiffre

Zeichenweise Verschlüsselung. z.B. XOR: $c_i = m_i \otimes k \qquad m_i = c_i \otimes k$

Kryptoanalyse: Chiffre mit 0-Folge.

5 Asymmetrische Verfahren

5.1 Potenzen in Arithmetik modulo n

$y = a^x \mod p$

$x \backslash q$	0	1	2	3	4
1	0	1	2	3	4
2	0	1	4	4	1
3	0	1	3	2	4
4	0	1	1	1	1

Kleiner Satz von Fermat: $\boxed{a^{p-1} \mod p = 1}$, falls $\mathrm{ggT}(a, p) = 1$

$a = 2 \vee a = 3$ sind Generatorelemente.

Eulersche Φ-Funktion: $\Phi(n) = |\{z \in [1, n-1] \mid \mathrm{ggT}(n, z) = 1\}|$

Es gilt $\Phi(p) = p - 1$ und $\Phi(p \cdot q) = (p-1)(q-1)$ für $p \neq q$

Euler's Satz: $a^{\Phi(n)} \equiv 1 \pmod n \qquad a^{k\Phi(n)+1} \equiv a \pmod n$

$a^j \mod n = a^{j \mod \Phi(n)} \mod n$

5.2 Miller-Rabin-Test

$(n-1)/a^s = d \in \mathbb{N}$ größtes s, so dass $d \in \mathbb{N}$

Falls n prim, dann entweder $a^d = l \mod n$ oder $\exists r \leq s - 1 : a^{a^r d} = -1$

5.3 RSA

Erzeugung eines Schlüsselpaares:

1. $n = p \cdot q$ mit $p \neq q$
2. Wähle ersten Schlüssel e zufällig mit $1 < e < \Phi(n)$ und $\mathrm{ggT}(e, \Phi(n)) = 1$
3. Berechne privaten Schlüssel d durch $e \cdot d \equiv 1 \mod \Phi(n)$

Verschlüsselung $c = (m^e) \mod n$

Entschlüsselung $(c^d) \mod n = (m^e)^d \mod n = m$

Signatur

5.4 Diffie-Hellman-Schlüsselvereinbarung (1976)

Protokoll um öffentlich einen geheimen Schlüssel k zu vereinbaren. Bietet jedoch keine Authentifizierung. Vereinbarung kann öffentlich abgehört werden ohne Sicherheit zu gefährden.

Beide Partner Alice und Bob vereinbaren eine 1024Bit Primzahl p und eine Basis $g \in \mathrm{GF}(p)$ daraus wird der gemeinsame Schlüssel k abgeleitet.

Alice	Bob	
wählt geheime Zufallszahl a	wählt geheime Zufallszahl b	
berechnet $\alpha = g^a \mod p$	berechnet $\beta = g^b \mod p$	Es gilt
schickt α an Bob	schickt β an Alice	
berechnet $k = \beta^a \mod p$	berechnet $k = \alpha^b \mod p$	

$\beta^a = \alpha^b = g^{ba}$

5.5 ElGamal-Verfahren (1984)

Öffentlich bekannt: Primzahl p und Basis $g \in \mathrm{GF}(p)$

Jeder Teilnehmer wählt einen privaten Schlüssel d und berechnet daraus den öffentlichen Schlüssel $e = g^d \mod p$

6 Elliptische Kurven, ECC Kryptographie

Sicherheitsvergleich: 160Bit ECC entspricht 1024Bit RSA.

Elliptische Kurve: $\boxed{y^2 = x^3 + ax + b}$

6.1 Protokolle

Definition: Satz von Regeln für den Austausch von Daten zwischen Kommunikationspartnern.

Fiat-Shamir Authentifizierung:

Schlüsselbank: $n = p \cdot q$; n öffentlich; p, q geheim
Für jeden Teilnehmer Zufallszahl z_i und Geheimnis s_i
Kerberos (sym.) Trustet Third Party TTP:

Needham Schroeder Protokoll (asym.)

7 IT-Sicherheit

- Safety: Sicher für den Menschen bei Fehlfunktion.
- Secure: Sicher gegen Angriffe.

7.1 Datensicherheit

Quantencomputer: Faktorisierungsprobleme lösbar -¿ RSA unsicher AES 128 unsicher, 256Bit noch als sicher erachtet.