Alphabete und Worte

Ein Alphabet A ist eine nicht-leere Menge.
Die Wortmenge A^* über A ist $A^* = \cup_{i=0}^{\infty} A^i$.

Beispiel:
- $A = \{\mathrm{A}, \ldots, \mathrm{Z}\}$, HALLO $\in A^*$
- $A = \{0, 1\}$, $101101 \in A^*$.
- $1^k = 11 \cdots 1$, k Einsen.

Leeres Wort: ε.
Aneinanderhängen von Worten: $m_1, m_2 \in A^* \to m_1 m_2 \in A^*$.
Länge $|m|$ eines Worts $m \in A^*$: $m \in A^{|m|}$.

Schreibweise in einigen Programmiersprachen: ’A’, “HALLO“, ““

Konzepte der Verschlüsselung

Klartexte sind aus Klartextraum: $m \in M \subseteq A_1^*$.
Chiffretexte sind aus Chiffretextraum: $c \in C \subseteq A_2^*$.
Schlüssel sind aus Schlüsselräumen: $e \in K_1 \subseteq A_3^*$, $d \in K_2 \subseteq A_4^*$.

Verschlüsselungsverfahren $\mathcal{E} : K_1 \times M \rightsquigarrow C$.
Entschlüsselungsverfahren $\mathcal{D} : K_2 \times C \to M$.

Verschlüsselung von Klartext m mit Schlüssel e: $c \leftarrow \mathcal{E}(e, m)$.
Entschlüsselung von Chiffretext c mit Schlüssel d: $m \leftarrow \mathcal{D}(d, c)$.

$\mathcal{E}$ und $\mathcal{D}$ sind effiziente Verfahren (z.B. Programme).
$\mathcal{E}$ und $\mathcal{D}$ dürfen probabilistisch sein (dürfen den Zufall verwenden).
$\mathcal{E}$ kann mehrdeutig sein ($\rightsquigarrow$).

Symmetrisch, asymmetrisch

Symmetrisches Verschlüsselungssystem (secret key):
- $d = e$ (oder d leicht aus e berechenbar).
- Sender und Empfänger haben e als gemeinsames Geheimnis.

Asymmetrisches Verschlüsselungssystem (public key):
- d geheim, e öffentlich.
- d schwer oder gar nicht aus e berechenbar.
- Empfänger besitzt e und d.
- Sender verwendet e zum Verschlüsseln.
- Empfänger verwendet d zum Entschlüsseln.

Prinzip von Kerkhoff

Grundlegende Anforderung:
Sicherheit eines Systems sollte sich nicht aus der Geheimhaltung der Algorithmen, sondern nur aus den geheimen Schlüsseln ableiten.

Gründe dafür:
- Geheime Schlüssel lassen sich schneller und häufiger austauschen, größere Flexibilität.
- Geheimhaltung funktionert nicht lange (siehe COMP128 im GSM Mobilfunk, RC4 von RSA), Umstellung auf neue Algorithmen teuer.
- Öffentliche Algorithmen können von unabhängiger Seite untersucht werden. Dies eliminiert Designfehler und baut Vertrauen bei Benutzern bzw. Kunden auf (siehe GSMK Cryptophone).

Prinzip von Kerkhoff

Gründe dagegen:

- Firmen wollen Geschäftsgeheimnisse bewahren.
- Staatliche Behörden wollen einen verbreiteten Gebrauch (z.B. durch Terroristen) des für eigene Zwecke entwickelten und benutzten, vielleicht sehr sicheren Kryptosystems verhindern.

Im allgemeinen wird die Meinung vertreten, daß das Prinzip von Kerkhoff befolgt werden sollte.

Angriffe und Sicherheitsmodelle

Es werden Angreifer (Programme, Menschen, etc.) betrachtet.

Im wesentlichen drei Aspekte:

- Was ist das Ziel eines Angriffs (wann erfolgreich)?
- Welche Informationen stehen dem Angreifer zur Verfügung?
- Welche Rechenleistung besitzt der Angreifer?

Trifft man für diese Aspekte eine Wahl, so legt man sich auf ein Sicherheitsmodell fest.

Gibt es unter den Aspekten keinen Angreifer, so ist das Kryptosystem sicher (im gewählten Modell).

Angriffe und Sicherheitsmodelle

Ziele eines Angreifers:

- Chiffretext entschlüsseln.
- Chiffretexte bekannten Nachrichten zuordnen.
- Gegebenen Chiffretext in neuen Chiffretext umwandeln, so daß neuer Klartext mit alten Klartext sinnvoll verbunden ist.
- Irgendeinen Chiffretext erzeugen (ohne Klartext zu kennen).

Die korrespondierenden (negierten) Eigenschaften des Verschlüsselungssystems:

- Einweg-Eigenschaft (onewayness, OW).
- Semantische Sicherheit, Nichtunterscheidbarkeit (indistinguishability, IND).
- Nicht-Modifizierbarkeit (non-malleability, NM).
- Plaintext-awareness (PA).

Angriffe und Sicherheitsmodelle

Potentiell gefährlich sind reguläre Eigenschaften innerhalb eines Verschlüsselungssystems, wie z.B. Homomorphieeigenschaften.

- Schlüssel bilden Gruppe.
- $\mathcal{E}(e,m)$ homomorph in m.
- $\ldots$

Angriffe und Sicherheitsmodelle

Informationen (Fähigkeiten) des Angreifers in aufsteigender Reihenfolge:

- Ciphertext-only Angriff: Der Angreifer erhält nur Chiffretexte.
- Known-plaintext Angriff: Der Angreifer erhält Klartexte und die zugehörigen Chiffretexte.
- Chosen-plaintext Angriff (CPA): Der Angreifer kann sich die Klartexte aussuchen und erhält die zugehörigen Chiffretexte.

Zusätzlich bei Public-Key Verschlüsselungssystemen:

- Chosen-ciphertext Angriff (CCA1): Der Angreifer kann sich Chiffretexte aussuchen und erhält die zugehörigen Klartexte.

Angriffe und Sicherheitsmodelle

Chosen-plaintext und Chosen-ciphertext Angriffe gibt es auch in adaptiver und kombinierter Form:

Der Angreifer darf Klar- und Chiffretexte in Abhängigkeit zuvor erhaltener Chiffre- bzw. Klartexte und den Ergebnissen von Zwischenrechnungen wählen.

CCA1 wird dann zu CCA2.

Angriffe und Sicherheitsmodelle

Rechenleistung eines Angreifers:

- Vergleichbar der von $\mathcal{E}$ und $\mathcal{D}$ (polynomiell).
- Unbeschränkt.

Die korrespondierenden (negierten) Eigenschaften des Verschlüsselungssystems:

- Komplexitätstheoretische Sicherheit (computational security).
- Perfekte Sicherheit (unconditional security, perfect secrecy).

Im allgemeinen betrachtet man nur komplexitätstheoretische Sicherheit.

Angriffe und Sicherheitsmodelle

Beispiel:

Das für Public-Key Systeme stärkste Sicherheitsmodell ist

Nichtunterscheidbarkeit unter einem adaptiven Chosen-ciphertext Angriff (IND-CCA2).

Hier versucht ein Angreifer die Chiffretexte zweier von ihm vorgegebener Nachrichten den Nachrichten zuzuordnen.

Gelingt dies mit Wahrscheinlichkeit signifikant besser als 1/2, so gilt der Angriff als erfolgreich.

NM ist sicherer als IND ist sicherer als OW.

CCA2 ist sicherer als CCA1 ist sicherer als CPA.

NM-CCA2 = IND-CCA2

(In spezieller Situation: PA ist sicherer als IND-CCA2, NM-CCA2)

Brute-Force (exhaustive search) Angriff

1. Alle Entschlüsselungsschlüssel ausprobieren.
2. Schauen, ob das Entschlüsselte Sinn macht (d.h. Ausnutzen von Redundanz in beispielsweise geschriebener Sprache etc.)

Gegenmaßnahme:

- Ausreichend großen Schlüsselraum haben.

Konzepte der Signatur

Nachrichten aus Nachrichtenraum: $M \in \mathcal{M} \subseteq A_1^*$.
Signaturen aus Signaturenraum: $\sigma \in S \subseteq A_2^*$.
Schlüssel sind aus Schlüsselräumen: $d \in K_1 \subseteq A_3^*$, $e \in K_2 \subseteq A_4^*$.

Signierungsverfahren $\mathcal{S} : K_1 \times \mathcal{M} \rightsquigarrow S$.
Verifizierungsverfahren $\mathcal{V} : K_2 \times \mathcal{M} \times S \to \{0,1\}$.

Signatur von Nachricht M mit Schlüssel d: $\sigma \leftarrow \mathcal{S}(d,M)$.
Verifizierung von M,σ mit Schlüssel e: $f \leftarrow \mathcal{V}(e,M,\sigma)$.

$\mathcal{S}$ und $\mathcal{V}$ sind effiziente Verfahren (z.B. Programme).
$\mathcal{S}$ und $\mathcal{V}$ dürfen probabilistisch sein (dürfen den Zufall verwenden).
$\mathcal{S}$ kann mehrdeutig sein ($\rightsquigarrow$).

Symmetrisch

Symmetrisches Signatursystem (secret key, MAC):

- $d = e$ (oder d leicht aus e berechenbar).
- Sender und Empfänger haben e als gemeinsames Geheimnis.
- Gemeinhin als Message Authentication Code (MAC) bezeichnet, meistens deterministisch.

Erweiterte Nachrichten werden verschickt: (M,σ) wo $\sigma \leftarrow \mathcal{S}(d,M)$.
Integrität der erweiterten Nachrichten (M,σ) wird von $\mathcal{V}$ mit $\mathcal{S}(d,M) = \sigma$ überprüft.

Hashfunktion: MAC ohne geheimen Schlüssel.

Asymmetrisch

(Asymmetrisches, public key) Signatursystem:

- d geheim, e öffentlich.
- d schwer oder gar nicht aus e berechenbar.
- Signierer besitzt e und d.
- Signierer verwendet d zum Signieren.
- Empfänger verwendet e zum Verifizieren.

Angriffe und Sicherheitsmodelle

Das Prinzip von Kerkhoff soll gelten
(ein Grund dagegen weniger hier).

Ziele des Angreifers:

- Existentielle Fälschung. Der Angreifer berechnet eine Signatur für eine Nachricht.
- Universelle Fälschung: Der Angreifer kann Signaturen für jede beliebige Nachricht berechnen.
- Total break: Der Angreifer berechnet den geheimen Schlüssel des Signierers.

Angriffe und Sicherheitsmodelle

Informationen (Fähigkeiten) des Angreifers in aufsteigender Reihenfolge:

- Key-only Angriff: Der Angreifer kennt nur den öffentlichen Schlüssel des Signierers.
- Known-Signature Angriff: Der Angreifer erhält Nachrichten und die zugehörigen Signaturen.
- Chosen-Message Angriff: Der Angreifer kann sich die Nachrichten aussuchen und erhält die zugehörigen Signaturen.

Den letzte Variante gibt es auch in adaptiver Form.

Stärkstes Sicherheitsmodell: Sicherheit bezüglich existenzieller Fälschung unter adaptiven Chosen-Message Angriffen.

Bemerkungen

Im allgemeinen wird nur ein Hashwert $H(M)$ und nicht M selbt signiert.

- Effizienter, da $H(M)$ viel kürzer als M ist.
- Beweisbare Sicherheit von in der Praxis relevanten Verfahren (allerdings im Zufallsorakelmodell, RO).

Offenbar muß H kollisionsfrei sein, man kann keine zwei Nachrichten M_1, M_2 mit $H(M_1) = H(M_2)$ berechnen.

Vergleich Public-Key und Secret-Key Kryptographie

Public-Key:

- größere Funktionalität, z.B. Schlüsselaustausch, Signaturen, etc.
- basiert auf mathematischen Problemen (aus der Zahlentheorie).

Secret-Key:

- effizienter in Verschlüsselung (ebenso MAC und Hashfunktionen).

Wahrscheinlichkeitstheorie

X endliche Menge, $p : X \to \mathbb{R}^{\geq 0}$, $\sum_{x \in X} p(x) = 1$.

1. p heißt Wahrscheinlichkeitsverteilung.
2. (X, p) heißt Wahrscheinlichkeitsraum.
3. $A \subseteq X$ heißt Ereignis. Die Wahrscheinlichkeit von A ist $\Pr(A) = \sum_{x \in A} p(x)$.
4. Komplementärereignis von A: $\bar{A} = X \setminus A$.
5. $0 \leq \Pr(A) \leq 1$, $\Pr(\{\}) = 0$, $\Pr(X) = 1$, $\Pr(\bar{A}) = 1 - \Pr(A)$.
6. $A, B \subseteq X$: $\Pr(A \cup B) = \Pr(A) + \Pr(B) - \Pr(A \cap B)$.
7. $A \subseteq B \Rightarrow \Pr(A) \leq \Pr(B)$.
8. Gleichverteilung: $p(x) = 1/\#X$, $\Pr(A) = \#A/\#X$.

Wahrscheinlichkeitstheorie

(X, p) Wahrscheinlichkeitsraum, $A, B \subseteq X$, $\Pr(B) > 0$.

1. Bedingte Wahrscheinlichkeit: $\Pr(A \mid B) := \Pr(A \cap B) / \Pr(B)$.
2. A und B heißen unabhängig, wenn $\Pr(A \cap B) = \Pr(A)\Pr(B)$.
3. Satz von Bayes: $\Pr(A \mid B) = \Pr(A)\Pr(B \mid A) / \Pr(B)$.

Y endliche Menge, $f : X \to Y$.

1. f heißt Y-wertige Zufallsvariable auf X.
2. Induzierte Verteilung $f(p)$ auf Y: $f(p)(y) := p(\{x \in X \mid f(x) = y\})$
3. $\Pr(f = y) := f(p)(y)$, $\Pr(f \in A) := \Pr(f^{-1}(A))$ für $A \subseteq Y$.
4. Erwartungswert von f für $Y = \mathbb{R}$: $E(f) := \sum_{y \in Y} y \Pr(f = y) = \sum_{x \in X} f(x)p(x)$.

Wahrscheinlichkeitstheorie

Konstruktionen mit W-Räumen und Z-Variablen.

(X, p), $f_i : X \to Y_i$, $1 \leq i \leq n$.

1. $\Pr(f_1 = y_1, \ldots, f_n = y_n) := \Pr(\{x \in X \mid f_i(x) = y_i\})$
2. f_i unabhängig: $\Pr(f_1 = y_1, \ldots, f_n = y_n) = \prod_i \Pr(f_i = y_i)$

(X_i, p_i), $f_i : X_i \to Y_i$, $1 \leq i \leq n$, verschiedene X_i.

1. $X_1 \times \cdots \times X_n$ wird durch $p((x_1, \ldots, x_n)) = \prod_i p_i(x_i)$ zum Wahrscheinlichkeitsraum.
2. $\Pr(f_i = y_i) = \Pr(f_i \circ \pi_i = y_i)$, wobei π_i die Projektion $X_1 \times \cdots \times X_n \to X_i$ ist.

Wahrscheinlichkeitstheorie

(X, p), $(W_x, p_x)_{x \in X}$.

1. $XW := \cup_{x \in X} \{x\} \times W_x$, $p_{XW}(x, w) := p(x)p_x(w)$.

$f : X \to Y$, $g : XW \to Y$.

1. $\Pr(f(x) = y : x \leftarrow X) := \Pr(f = y)$.
2. $\Pr(g(x, w) = y : x \leftarrow X, w \leftarrow W_x) := \Pr(g = y)$.

Prop: $A, A_i \subseteq X$, $\Pr(A_i) > 0$, so daß X disjunkte Vereinigung der A_i ist. Dann $\Pr(A) = \sum_i \Pr(A_i)\Pr(A \mid A_i)$.

Bew: $\Pr(A) = \sum_i \Pr(A \cap A_i) = \sum_i \Pr(A_i)\Pr(A \mid A_i)$.

Wahrscheinlichkeitstheorie

Prop: $\mathbb{R}$-wertige f, g.
1. $E(f+g) = E(f) + E(g)$.
2. Sind f, g unabhängig, so $E(fg) = E(f)E(g)$.

X kann als zufälliges Experiment aufgefaßt werden.
$X^n = X \times \cdots \times X$ als n-fache, unabhängige Ausführung des Experiments.

Prop: $A \subseteq X$, $\Pr(A) > 0$. Sei f die Zufallsvariable, welche angibt, wie häufig X wiederholt werden muß, um das erste mal A zu erhalten. Dann gilt $E(f) = 1/\Pr(A)$.

Wahrscheinlichkeitstheorie

Prop (Markov Ungleichung): f nicht negativ und $v > 0$.
Dann $\Pr(f \geq v) \leq E(f)/v$ bzw. $\Pr(f \geq rE(f)) \leq 1/r$.

Einfache Anwendung: Bei 12mal Würfeln erhalten wir mindestens eine 6 mit Wahrscheinlichkeit $\geq 1/2$.

Algorithmen

Unter einem Algorithmus verstehen wir einen Text von elementaren Anweisungen, die beispielsweise auf einem Computer oder einer Turing Maschine ausgeführt werden können.

Ein Algorithmus erwartet eine Eingabe, tätigt eine Ausgabe und endet dann.

Ein Algorithmus heißt deterministisch, wenn die auszuführenden Anweisungen eindeutig durch die Eingabe bestimmt sind. Ein solcher Algorithmus verhält sich wie eine Funktion.

Ein Algorithmus heißt probabilistisch, wenn die auszuführenden Anweisungen neben der Eingabe auch von zufälligen Entscheidungen während des Laufs abhängen.

Algorithmen

Zu vorgegebener Eingabe ist die Laufzeit eines Algorithmus die Anzahl der bis zum Ende des Algorithmus ausgeführten Anweisungen, inklusive der Zufallsabfragen. Im Zusammenhang mit Computern werden häufig Bitoperationen gezählt.

Die Laufzeit und die Ausgabe eines probabilistischen Algorithmus hängen damit ebenfalls vom Zufall ab und stellen Zufallvariablen dar.

Ist A ein probabilistischer Algorithmus, so können wir daraus einen deterministischen Algorithmus A_D machen, indem wir die von A benutzte Zufallsquelle als Eingabe von A_D auffassen.

Ein Algorithmus ist polynomiell, wenn es ein Polynom $f \in \mathbb{Z}[t]$ gibt, so daß seine Laufzeit für die Eingabe x durch $f(|x|)$ beschränkt ist.

Algorithmen

Beispiel:

Aufgabe ist, eine 6 zu würfeln. Lösung ist, einfach wiederholt zu würfeln, bis eine 6 erscheint.

In vielen Fällen sind probabilistische Algorithmen deterministischen Algorithmen zur Lösung der gleichen Probleme überlegen (einfacher und schneller).