Vorlesung Sicherheit

Dennis Hofheinz

ITI, KIT

15.05.2017

Überblick

1. Hashfunktionen
   - Angriffe auf Hashfunktionen
   - Zusammenfassung Hashfunktionen

2. Asymmetrische Verschlüsselung
   - Idee
   - Beispiel: RSA
   - Sicherheit von RSA

Überblick

1. Hashfunktionen
   - Angriffe auf Hashfunktionen
   - Zusammenfassung Hashfunktionen
2. Asymmetrische Verschlüsselung
   - Idee
   - Beispiel: RSA
   - Sicherheit von RSA

Birthday Attack

- Idee: betrachte viele $Y_i := \mathrm{H}(X_i)$ für zufällige X_i
- Vereinfachend: $Y_i \in \{0,1\}^k$ unabhängig gleichverteilt

Theorem (Birthday Bound, ohne Beweis)

Sei $n \leq 2^{k/2}$ und $Y_1, \ldots, Y_n \in \{0,1\}^k$ unabhängig gleichverteilt. Dann gibt es $i \neq j$ mit $Y_i = Y_j$ mit Wkt. $p > (1/11) \cdot (n^2/2^k)$.

- Konsequenz: für $n = 2^{k/2}$ zufällige (verschiedene) X_i Kollisionen unter den Y_i mit Wahrscheinlichkeit $p > 1/11$
- Vorgehen (Aufwand ˆ$(k \cdot 2^{k/2})$ Schritte, ˆ$(k \cdot 2^{k/2})$ Bits):
  1. Schreibe (X_i, Y_i) in Liste ($X_i \in \{0,1\}^{2k}$ glv., $Y_i = \mathrm{H}(X_i)$)
  2. Sortiere Liste nach Y_i
  3. Untersuche Liste auf Y_i-Kollisionen

Weitere Angriffe, Konsequenzen

- Auch Meet-in-the-Middle-Angriff manchmal möglich
  - Setzt spezielle Struktur von Hashfunktionen voraus: Hashwert sollte sich „rückwärts“ berechnen lassen
  - Aufwand: asymptotisch wie Birthday Attack
- **Lehre:** Hash-Ausgabe $\geq k$ Bits für $k/2$ Bits „Sicherheit“
  - Ausgabelängen: MD5 128, SHA-1 160, SHA-3 variabel

Überblick

Zusammenfassung Hashfunktionen

- Hashwert „Fingerabdruck“ der Eingabe
- Kollisionsresistenz $\Rightarrow$ Einwegeigenschaft
- Populäre Strategie: Merkle-Damgård
- Ausgabelänge $\geq k$ Bits für $k/2$ Bits Sicherheit
- Populäre Verfahren: MD5, SHA-1, SHA-3 (Keccak)
- **Aber:** MD5, SHA-1 gebrochen

Aktuelle Forschung

- Hashfunktionen, deren Sicherheit auf gut untersuchten Problemen beruht (z.B. Berechnungsprobleme in Gittern)
- Für Passwortabfragen: „universelle“ Einwegfunktionen
- Untersuchung von generischen Strategien (wie Merkle-Damgård) in idealisierten Modellen
- Kryptoanalyse: Weitere „Anwendungen“ und Erweiterungen von bekannten Angriffen auf z.B. MD5, SHA-1, Beispiele:
  - gefälschte Webseiten-Zertifikate (Flame)
  - gefälschte Signaturen für (Postscript-/PDF-)Dokumente
  - Grundidee: gegeben (Signatur von) $\mathrm{H}(M)$, finde „sinnvolles“ M' mit $\mathrm{H}(M') = \mathrm{H}(M)$ (für das die Signatur dann gilt)

Überblick

Überblick

Motivation

- Symmetrische Verschlüsselung: gemeinsames Geheimnis K

$$\texttt{Alice}_K \xleftarrow{C:=\mathrm{Enc}(K,M)} \texttt{Bob}_K$$

- Bei n Benutzern $\binom{n}{2} = n \cdot (n-1)/2$ Schlüsselpaare
- Zudem kann Schlüsselverteilung problematisch sein
- Auftritt Merkle (1974) und Diffie und Hellman (1976):



Quelle: Wikipedia

Grundidee

- Asymmetrische (oder Public-Key-)Verschlüsselung:

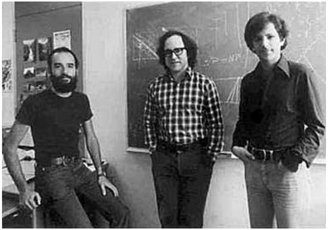
$$\mathtt{Alice}_{sk} \quad \xleftarrow{C:=\mathrm{Enc}(pk,M)} \quad \mathtt{Bob}_{pk}$$

- Verschlüsselung mit pk: $C \leftarrow \mathrm{Enc}(pk, M)$
- Entschlüsselung mit sk: $M \leftarrow \mathrm{Dec}(sk, C)$
- pk und sk gemeinsam generiert: $(pk, sk) \leftarrow \mathrm{Gen}(1^k)$
- pk darf veröffentlicht werden, sk muss geheim bleiben

Erste Eigenschaften

$$\texttt{Alice}_{sk} \xleftarrow{C:=\mathrm{Enc}(pk,M)} \texttt{Bob}_{pk}$$

- Keine (geheime) Schlüsselverteilung, *pk* öffentlich
- Bei *n* Benutzern *n* öffentliche (und *n* geheime) Schlüssel
- **Problem:** Diffie und Hellman hatten kein Verfahren
- Auftritt Rivest, Shamir und Adleman (1977): (Quelle: `ams.org`)



Überblick

Überblick RSA

$$pk = (N, e) \qquad sk = (N, d)$$

- $N = PQ$ für (hinreichend große) Primzahlen $P \neq Q$
- Rechnung in $\mathbb{Z}_N := \mathbb{Z}/N\mathbb{Z}$ (d.h. Rechnung modulo N)
- e und d sind „zueinander inverse" Exponenten
  - Genauer: $e \cdot d = 1 \bmod \varphi(N)$ (mit $\varphi(N) = (P-1)(Q-1)$)
- Nachrichtenraum ist $\mathcal{M} := \mathbb{Z}_N$

$$\mathrm{Enc}(pk, M) = M^e \bmod N \qquad \mathrm{Dec}(sk, C) = C^d \bmod N$$

RSA-Schlüsselgenerierung

- Gesucht:

$$pk = (N, e) \qquad sk = (N, d)$$

- Gen wählt P und Q zufällig von vorgegebener Bitlänge k:
  - Gängig: wähle gleichverteiltes *ungerades* $P \in \{2^k, \dots, 2^{k+1}\}$
  - ... bis P prim (Primalitätstest z.B. Miller-Rabin), analog für Q
- Um e und d zu erhalten:
  - Wähle zufällig gleichverteilt $e \in \{3, \dots, \varphi(N) - 1\}$
  - ... bis $\mathrm{gcd}(e, \varphi(N)) = 1$
  - Berechne $d = e^{-1} \bmod \varphi(N)$ mit erweitertem Euklid:
    - $\mathrm{EE}(e, \varphi(N)) = (\alpha, \beta)$ mit $\alpha \cdot e + \beta \cdot \varphi(N) = \mathrm{gcd}(e, \varphi(N)) = 1$
    - Damit ist $\alpha \cdot e = 1 \bmod \varphi(N)$, setze also $d := \alpha \bmod \varphi(N)$

Korrektheit von RSA

- Für Korrektheit sollte gelten: $(M^e)^d = M^{ed} = M \bmod N$
- Zutaten (ohne Beweis):

Theorem (Kleiner Satz von Fermat, ohne Beweis)

Für primes P und $M \in \{1, \ldots, P-1\}$ gilt $M^{P-1} = 1 \bmod P$.

- Konsequenz: $\forall M \in \mathbb{Z}_P, \alpha \in \mathbb{Z} : \ (M^{P-1})^\alpha \cdot M = M \bmod P$

Theorem (Chinesischer Restsatz, ohne Beweis)

Sei $N = PQ$ mit P, Q teilerfremd. Dann ist die Abbildung $\mu : \mathbb{Z}_N \to \mathbb{Z}_P \times \mathbb{Z}_Q$ mit $\mu(M) = (M \bmod P, M \bmod Q)$ bijektiv.

- Also: $(X = Y \bmod P) \wedge (X = Y \bmod Q) \Longrightarrow X = Y \bmod N$

Korrektheit von RSA

Theorem (Korrektheit von RSA)

Sei N, e, d wie oben. Dann ist $M^{ed} = M \bmod N$ für alle $M \in \mathbb{Z}_N$.

Beweis.

- Es gilt $ed = 1 \bmod (P-1)(Q-1)$ nach Definition, deshalb:

$$
\begin{aligned}
(P-1)(Q-1) \mid ed - 1 \quad &\Rightarrow \quad P - 1 \mid ed - 1 \\
&\Rightarrow \quad ed = \alpha(P-1) + 1 \qquad \text{für } \alpha \in \mathbb{Z} \\
&\Rightarrow \quad M^{ed} = (M^{P-1})^{\alpha} \cdot M \overset{\text{Fermat}}{=} M \bmod P
\end{aligned}
$$

- Analog: $M^{ed} = M \bmod Q$
- Chinesischer Restsatz $\Rightarrow M^{ed} = M \bmod N$

□

Überblick

Sicherheit?

- **Diskussion:** Was wollen wir eigentlich?
  Wieder Einschränkung auf passive Sicherheit/Angriffe

Semantische Sicherheit für Public-Key-Verschlüsselung

Definition (Semantische Sicherheit für Public-Key-Verfahren)

Ein Public-Key-Verschlüsselungsschema ist semantisch sicher, wenn es für jede M-Verteilung von Nachrichten gleicher Länge, jede Funktion f und jeden PPT-Algorithmus $\mathcal{A}$ einen PPT-Algorithmus $\mathcal{B}$ gibt, so dass

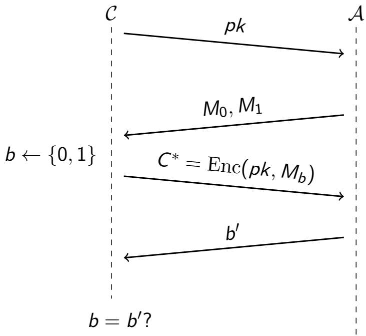
$$\Pr\left[\mathcal{A}(1^k, pk, \mathrm{Enc}(pk, M)) = f(M)\right] - \Pr\left[\mathcal{B}(1^k) = f(M)\right]$$

vernachlässigbar (als Funktion im Sicherheitsparameter) ist.

- Unhandlich, aber (wie symmetrisch) äquivalent zu IND-CPA

IND-CPA für asymm. Verschlüsselung

- Herausforderer $\mathcal{C}$ erzeugt Schlüsselpaar $(pk, sk) \leftarrow \mathrm{Gen}(1^k)$.
- Kein Enc-Orakel, stattdessen erhält der Angreifer pk.



- Formal: analog zu IND-CPA für symm. Verfahren (mit entsprechenden Anpassungen).

Sicherheit von RSA

- RSA nicht semantisch sicher
  - $f(M) = M^e \bmod N$ kann mit Chiffrat berechnet werden
  - Aber ohne Chiffrat keine Information über M
  - „Angriff“ nutzt aus, dass RSA deterministisch
- Intuitiv überzeugender: beispielsweise

$$\mathrm{Enc}(pk, \texttt{annehmen}) \quad \text{und} \quad \mathrm{Enc}(pk, \texttt{ablehnen})$$

bei RSA effizient unterscheidbar (keine IND-CPA-Sicherheit)

Weitere Angriffe auf RSA

- Was, wenn $e = 3$ (aus Effizienzgründen) für alle Benutzer?
  - Problem, wenn M an ≥ 3 Benutzer gesendet wird
  - Angreifer kennt Chiffrate $M^3 \bmod N_i$ für $1 \leq i \leq 3$
  - Chinesischer Restsatz $\rightsquigarrow$ $M^3 \bmod N_1 N_2 N_3$
  - Wegen $0 \leq M \leq N_1, N_2, N_3$ ist $M^3 \bmod N_1 N_2 N_3 = M^3 \in \mathbb{Z}$
  - „Wurzelziehen“ über $\mathbb{Z}$ liefert M
- Könnte mit probabilistischem Enc behoben werden
- Weitere schlechte Idee: gleiches N für mehrere Benutzer

Homomorphie von RSA

- Es gilt (Rechnung modulo N):

$$\begin{aligned}\mathrm{Enc}(pk, M) \cdot \mathrm{Enc}(pk, M') &= M^e \cdot {M'}^e \\ &= (M \cdot M')^e = \mathrm{Enc}(pk, M \cdot M')\end{aligned}$$

- Problem z.B. bei Auktionen:
  - Auktionator A veröffentlicht pk, behält sk
  - Bieter B_1, B_2 senden verschlüsselte Gebote $C_i := \mathrm{Enc}(pk, M_i)$ an Auktionator
  - A entschlüsselt C_i, Bieter mit höchstem Gebot erhält Ware
  - Angriff: B_2 wartet B_1's Gebot ab, setzt

$$C_2 := C_1 \cdot \mathrm{Enc}(pk, 2) \bmod N$$