Vorlesung am 5.5.2015

6 Datenauthentisierung

Grundidee: Detektion von Datenmanipulation

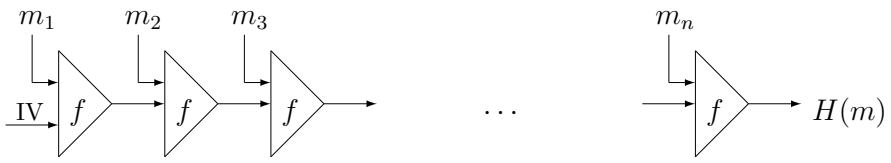
- Beweisender (Sender/Erzeuger der Daten) berechnet Prüfsumme.
- Prüfer (Empfänger der Daten) überprüft.

Message Authentication Codes (symmetrische Verfahren)
Erste Idee: basierend auf einer Hashfunktion H

- Beweisender (B) und Prüfer (P) vereinbaren Schlüssel $k \in \{0,1\}^{128}$.
- B authentisiert Nachricht $m \in \{0,1\}^n$:
  - Berechne $s = H(k||m)$, sende m, s an P.
- P prüft Authentizität von m:
  - Berechne $s' = H(k||m)$, prüfe, ob $s = s'$.

Ist unsicher:

- Erinnerung: Konstruktion von Hashfunktionen nach Merkle-Damgård



- Angreifer hängt weitere Blöcke an m und berechnet Hashwert weiter.
  Schützt also nur Anfang der Nachricht, nicht deren Ende.
- Auch $H(m||k)$ keine gute Idee.
  Schützt nur Ende der Nachricht, nicht deren Anfang.

Sichere MAC-Verfahren:

- CMAC: Basiert auf Blockchiffren (C=Chiffre)
- HMAC: Basiert auf Hashfkt. (H=Hash)

$$\mathrm{HMAC}(k,m) := \underbrace{H((k \oplus opad)|| \underbrace{H((k \oplus ipad)||m)}_{\substack{sichert\ Anfang \\ der\ Nachricht\ ab}})}_{\substack{sichert\ auch\ Ende\ der \\ Nachricht\ ab}}.$$

opad, ipad sind Konstanten:

$$opad := \underbrace{0x5C \cdots 0x5C}_{\text{Schlüssellänge}} \quad \text{und} \quad ipad := \underbrace{0x36 \cdots 0x36}_{\text{Schlüssellänge}}$$

Übung: Beschreiben Sie das MAC-Verfahren CMAC.

Verschlüsselung-und MAC-Verfahren werden häufig zusammen eingesetzt:

- Secure Messaging (vertraulicher und authentischer Kanal)

Zwei Nebenbedingungen

- Verschiedene Schlüssel für Verschlüsselung und Authehtisierung:
  - Wesentliches Grundprinzip: Trenne wo du trennen kannst. Verschiedene Schlüssel erhöhen die Sicherheit.
- Erst verschlüsseln, dann verschlüsselte Daten authentisieren:
  - Vertraulichkeit ist ein anderes Schutzziel als Authentizität: MAC-Verfahren müssen nicht Vertraulichkeit garantieren.
  - Weiterverarbeitung von Daten nur, wenn Sender bekannt ist: Verhindert z.B. Entschlüsselung von Schadsoftware.

MAC-Verfahren erfüllen nicht das Schutzziel Nichtabstreitbarkeit:

- Beide nutzen den selben Schlüssel.
- Auch der Prüfer könnte die Daten authentisiert haben.
- Gegenüber Dritten nicht nachweisbar, wer Prüfsumme berechnet hat.

Signaturverfahren (asymmetrische Verfahren)

- Beweisender nutzt geheimen Schlüssel zur Berechnung der Prüfsumme.
- Prüfer prüft mit assoziiertem öffentlichen Schlüssel.
- Also: Erfüllt auch zusätzlich Schutzziel Nichtabstreitbarkeit:
  Prüfsumme nur vom Inhaber des geheimen Schlüssels berechenbar.

RSA-Signaturen: Modul n, Schlüsselpaar (e, d) (e öffentlich, d geheim)

- B nutzt d, um Prüfsumme $s = m^d \bmod n$ zu berechnen
- P nutzt e, um $m' = s^e \bmod n$ zu berechnen
  Wegen $s^e = (m^d)^e = m^{de} = m \bmod n$ akzeptiert P, wenn $m' = m$ gilt.

Problem: Es können nur Nachrichten $m < n$ signiert werden.

Für $n \approx 2^{2000}$ also Bitlänge von $m < 2000$.

Lösung:

- Nachricht m wird zunächst gehasht, signiert und geprüft wird $H(m)$
- Dazu wichtig: Hashfunktion muss kollisionsresistent sein.
  - Gilt für zwei Nachrichten $m, m' : H(m) = H(m')$,
    dann ist eine gültige Signatur von m auch eine für m'.

Allgemein:

- Signaturverfahren besteht aus
  - Hashfunktion $H : \{0,1\}^n \longrightarrow \{0,1\}^n$ $(n \geq 200)$
  - Signaturalgorithmus

Übung: Welche Angriffe auf das RSA-Signaturverfahren gibt es.
Wie können diese verhindert werden.