Zusammenfassung

Bisher symmetrische Kryptographie behandelt.

Verschlüsselung - Blockchiffren:

- $E : K \to S(\{0,1\}^b)$, $\mathcal{D} : K \to S(\{0,1\}^b)$.
- Ideal: Für zufälliges $k \in K$ ist $E(k, \cdot)$ nicht von zufällig gewählter Funktion aus $S(\{0,1\}^b)$ effizient zu unterscheiden.
- Lange Nachrichten: Blockweiser Betrieb (CBC, CTR).

Verschlüsselung - Stromchiffren:

- Nachrichtenstrom wird durch $\oplus$ mit Schlüsselstrom verschlüsselt, initialisiert durch Schlüssel $k \in K$.
- Ideal: Schlüsselstrom zufällig (one time pad).
- Abschwächung: Pseudozufallszahlen.

Angestrebte Sicherheit: $\#K$, exhaustive Keysearch.

Zusammenfassung

Hashfunktionen - ohne Schlüssel (MDC):

- $h : \{0,1\}^* \to \{0,1\}^n$.
- Ideal: Zufallsfunktion, speziell kollisionsfrei und Einwegfunktion.
- Kurze Nachrichten: Kompressionsfunktion.
- Lange Nachrichten: Iterierung, Merkle-Damgard Konstruktion.
- Angestrebte Sicherheit: ca. 2^n Aufwand für Urbilder, ca. $2^{n/2}$ Aufwand für Kollision.

Hashfunktionen - mit geheimem Schlüssel (MAC):

- $h : K \times \{0,1\}^* \to \{0,1\}^n$.
- Ideal: Zufallsfunktion, niemand kann ohne Schlüssel k Fälschungen (x,y) mit $y = h_k(x)$ bestimmen.
- Konstruktion durch Blockchiffre (CBC-MAC) oder Kombination (HMAC).
- Angestrebte Sicherheit: ca. $2^{n/2}$ Aufwand für Fälschung.

Zusammenfassung

Pseudozufallszahlen bzw. -generatoren:

- Durch Blockchiffre im CBC oder CTR Mode.
- Durch Hashfunktionen.
- Ideal: Nicht effizient von echten Zufallszahlen unterscheidbar.
- Für den seed muß man genug Entropie aus dem System sammeln ...

Man kann zeigen, daß die folgenden Aussagen im wesentlichen äquivalent sind:

- Es gibt Einwegfunktionen.
- Es gibt ideale Pseudozufallsgeneratoren.
- Es gibt ideale Blockchiffren.

Public-Key Kryptographie

Zwei Probleme mit der symmetrischen Kryptographie:

- Wie geheime Schlüssel austauschen?
- n Leute haben paarweise verschiedene Schlüsselpaare: Also muß jeder $n(n-1)/2$ viele geheime Schlüssel speichern. Ist nicht praktikabel.

Diese und weitere Probleme werden durch die Public-Key Kryptographie gelöst.

Jeder Teilnehmer hat einen öffentlichen und einen geheimen Schlüssel:

- Zum Verschlüsseln wird der öffentliche und zum Entschlüsseln der geheime Schlüssel verwendet.
- Zum Unterschreiben wird der geheime und zum Verifizieren der öffentliche Schlüssel verwendet.

Hybridverschlüsselung

Public-Key Kryptographie liefert viel neue Funktionalität.

Public-Key Verschlüsselung ist aber auch viel langsamer als symmetrische Verschlüsselung.

Daher Hybridverschlüsselung:

- Nachricht mit symmetrischen Verfahren verschlüsseln.
- Den zugehörigen symmetrischen Schlüssel mit Public-Key Verfahren verschlüsseln.
- Alles an Empfänger schicken.
- Der entschlüsselt zunächst den symmetrischen Schlüssel, und dann die Nachricht.

Mann-in-der-Mitte Angriffe

Angenommen, Bob will mit Alice kommunizieren und Eve kontrolliert die Verbindung. Dann kann folgendes passieren:

- Bob fragt Alice nach ihrem öffentlichen Schlüssel.
- Eve fängt die Nachricht ab und schickt ihren eigenen öffentlichen Schlüssel an Bob.
- Bob verschlüsselt seine Nachricht damit und schickt sie an Alice.
- Eve fängt die Nachricht ab und entschlüsselt. Dann verschlüsselt sie die Nachricht mit Alice's öffentlichem Schlüssel und schickt dies an Alice.
- Weder Bob noch Alice schöpfen Verdacht.

Dies ist ein prinzipielles Problem, was auch mit anderen Public-Key Verfahren auftritt.

Zertifikate

Also Problem: Wie kann Bob sicher sein, daß ein öffentlicher Schlüssel auch Alice gehört?

Lösung: Zertifizierungsbehörde (Certificate Authority, CA).

Die CA erstellt für Alice eine digitale Unterschrift ihres öffentlichen Schlüssels und identifizierender Information, z.B. Alice's email Adresse oder Paßnummer. Bob überprüft dann die Unterschrift unter Verwendung des öffentlichen Schlüssels der CA.

Woher weiß Bob, daß der öffentliche Schlüssel der CA richtig ist?

„Den kennt ja jeder". Problem ist auf nur wenige öffentliche Schlüssel reduziert.

- Manchmal in Browsern fest eingebaut.

Führt auf Public Key Infrastructure (PKI) ...

Einwegfunktionen mit Falltür

Sei $f : M \to C$ eine Funktion des endlichen Nachrichten- und Chiffretextraums M bzw. C.

- f soll eine Einwegfunktion sein.
- Mit Hilfe gewisser zusätzlicher Informationen d soll es leicht sein, Urbilder unter f zu berechnen.

Dann nennt man diese Informationen d Falltür-Informationen und f eine Falltür-Einwegfunktion.

Ein injektives f liefert Kryptosystem:

- Verschlüsseln durch $c \leftarrow f(m)$.
- Entschlüsseln durch $m \leftarrow f^{-1}(c)$, mittels des geheimen d.

Einwegfunktionen mit Falltür basieren auf algorithmischen, zahlentheoretischen Problemen.