



Dining Cryptographers Networks

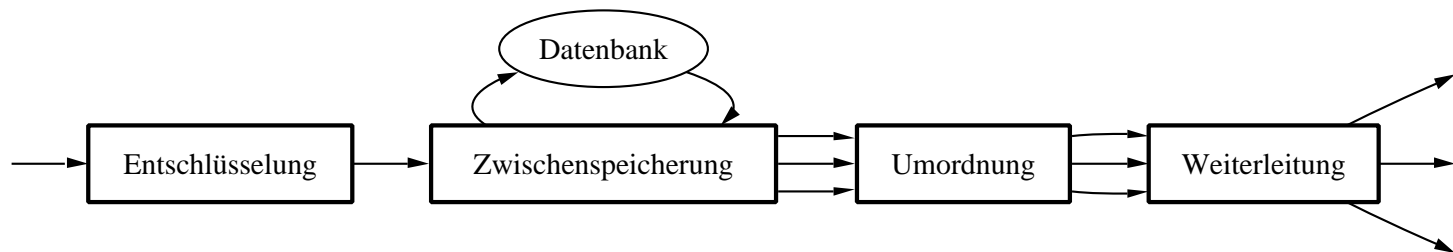
Robuste Anonymität durch Kryptographie

Heiko Stamer <stamer@theory.informatik.uni-kassel.de>

76F7 3011 329D 27DB 8D7C 3F97 4F58 4EB8 FB2B E14F

MIX-Netzwerk

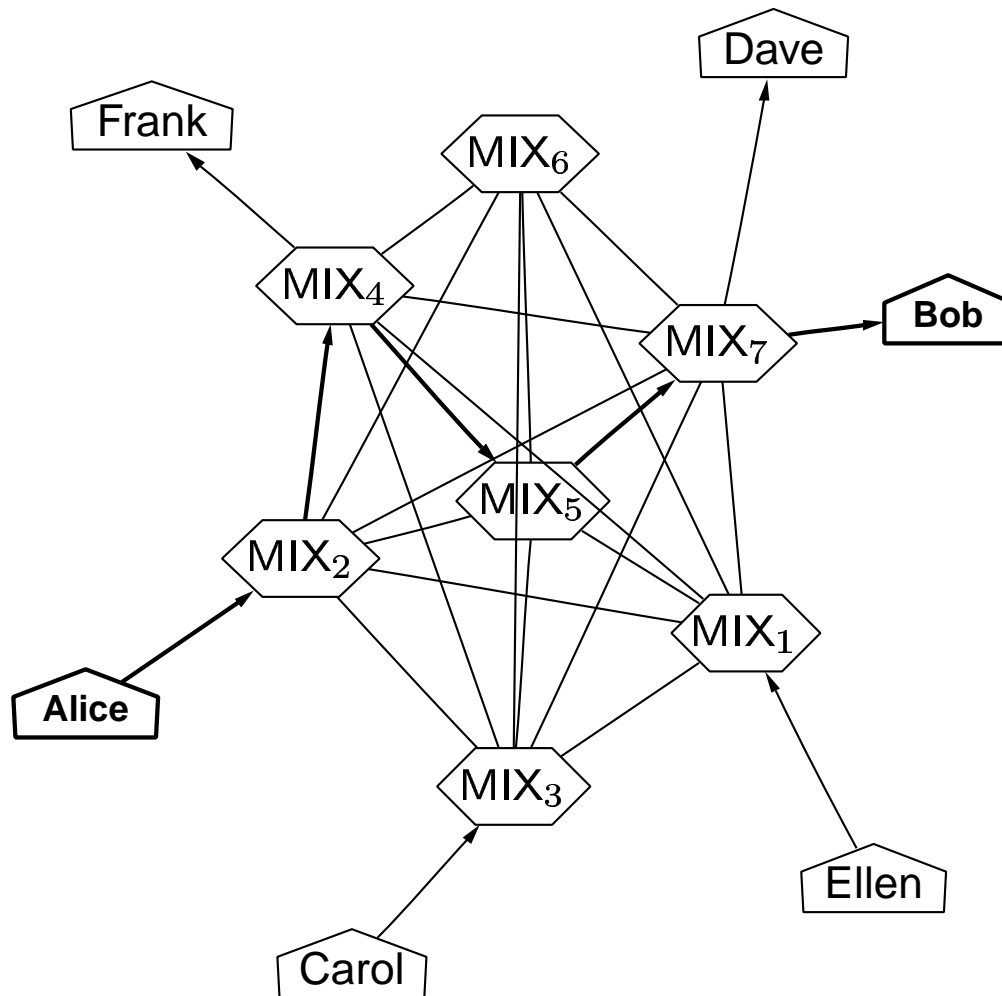
- Netzwerk autonomer Knoten MIX_1, \dots, MIX_n (öffentliche Schlüssel $K_{PUB}^{MIX_i}$ sind bekannt)
- Verarbeitungsschritte im Knoten MIX_i :



- Problem: Vertrauenswürdigkeit der Betreiber
- Lösung: **Kaskadierung** von MIX-Knoten

∴ Indirektionspfad (MIX-Netzwerk)

Alice \hookrightarrow MIX₂ \hookrightarrow MIX₄ \hookrightarrow MIX₅ \hookrightarrow MIX₇ \hookrightarrow **Bob**



$$\hat{m}_1 = E \left(K_{\text{PUB}}^{\text{MIX}_7}, [\text{Bob}, m] \right)$$

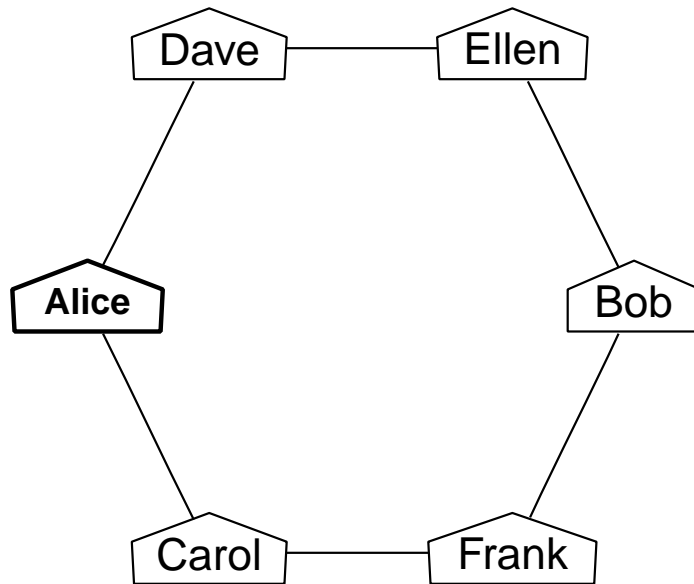
$$\hat{m}_2 = E \left(K_{\text{PUB}}^{\text{MIX}_5}, [\text{MIX}_7, \hat{m}_1] \right)$$

$$\hat{m}_3 = E \left(K_{\text{PUB}}^{\text{MIX}_4}, [\text{MIX}_5, \hat{m}_2] \right)$$

$$\hat{m}_4 = E \left(K_{\text{PUB}}^{\text{MIX}_2}, [\text{MIX}_4, \hat{m}_3] \right)$$

Alice sendet \hat{m}_4 an MIX₂

⋮⋮⋮ DC-Netzwerk (Funktionsweise)



1. Idealer Münzwurf hinter der Speisekarte (auch rechter Nachbar sieht das Ergebnis)
2. Jeder Kryptograph i kennt nun das Ergebnis zweier unabhängiger Münzwürfe:

$$K_i^L, K_i^R \in \{\text{Wappen, Zahl}\}$$

$$K_1^R = K_2^L, K_2^R = K_3^L, \dots, K_6^R = K_1^L$$

3. Bekanntgabe von $O_i = P_i \oplus M_i$ mit

$$P_i = \begin{cases} 1 & \text{falls } K_i^L = K_i^R \\ 0 & \text{sonst} \end{cases}$$

$$M_i = \begin{cases} 1 & \text{falls } i \text{ hat bezahlt} \\ 0 & \text{sonst} \end{cases}$$

4. Gesamtergebnis: $M = \bigoplus_{i=1}^n O_i$

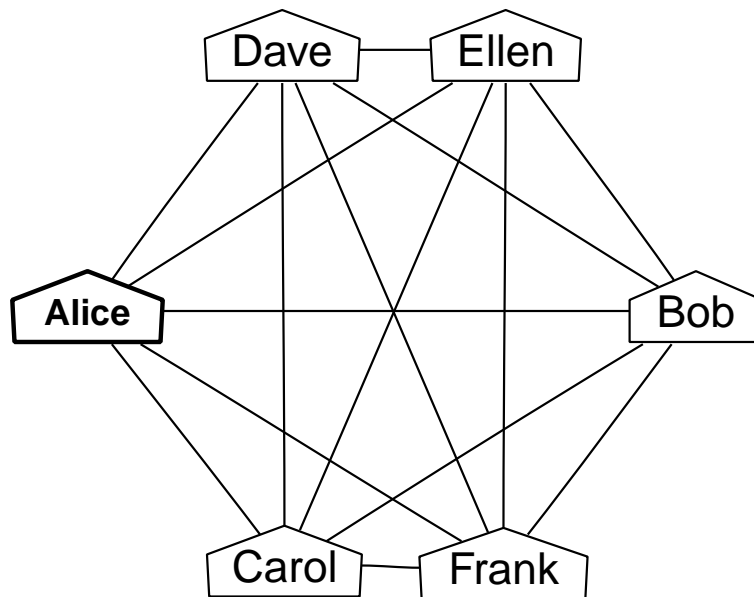
$M = 0$: NSA, zwei, vier, sechs

$M = 1$: ein, drei, fünf Bezahler

⋮⋮⋮ DC-Netzwerk (Angriffe, Probleme)

Angriff: Falls Carol und Dave kollaborieren (d. h. Münzwurfergebnisse austauschen), dann wäre die Anonymität von Alice verloren.

Lösung: vollständige Münzwurf-Topologie



Technische Probleme:

- Ideale Münzen
~> PRBG (BLUM, BLUM, SHUB)
- Kollisionen beim Senden
~> Reservierungsprotokoll (CHAUM et al.)
- Robustheit (böartige Störer)
~> Fallenprotokoll (PFITZMANN, WEIDNER)
- Zuverlässige Verteilung (Konsistenz O_i)
~> Byzantinische Übereinstimmung
ABBA (CACHIN, KURSAWE, SHOUP 2000)
 $t < n/3$ (n Teilnehmer, t Angreifer)

⋮⋮⋮ Schwellenwertsignaturschema

Definition 7 ((n, k, t) -Schwellenwertsignaturschema)

Von insgesamt n Teilunterschriften (bis zu t können vom Angreifer stammen) müssen mindestens k gesammelt werden, um eine gültige Gesamtunterschrift zu erzeugen.

- Bedingung $t < k \leq n - t$ sinnvoll; praktisch meist $k = n - t$.
- Vermittleralgorithmus (Trusted Dealer): Schlüsselerzeugung
- $\forall i$: Teilsignaturerzeugung, Teilsignaturverifikation, Teilsignaturkombination, Gesamtsignaturverifikation
- **Robustheit:** Angreifer kann bei der *Teilsignaturkombination* keine ungültige Gesamtunterschrift durch k gültige Teilsignaturen berechnen.
- **Unfälschbarkeit:** Angreifer kann keine gültige Gesamtunterschrift berechnen, ohne wenigstens $k - t$ gültige Teilsignaturen zu besitzen.

⋮⋮⋮ Kryptographische Annahmen

Bezeichner	Beschreibung	gegeben	gesucht
DLOG	diskreter Logarithmus (in \mathbb{Z}_p^*)	$\alpha^r \bmod p, \alpha, p$	r
CDH	Computational Diffie-Hellman	$\alpha^a \bmod p, \alpha^b \bmod p, \alpha, p$	$\alpha^{ab} \bmod p$
FAKTOR	Primfaktorisierung	$n = p \cdot q$	p, q
RSA	Rivest-Shamir-Adleman	$m^e \bmod n, e, n$	m

$$\text{RSA} \leq_{\mathcal{P}} \text{FAKTOR} \leq_{\mathcal{P}} \text{CDH} \leq_{\mathcal{P}} \text{DLOG}$$

Definition 8 (Decisional Diffie-Hellman assumption)

Seien die Gruppe G , ein Erzeuger $\alpha \in G$ und Elemente $\alpha^a, \alpha^b, \alpha^c \in G$ mit $a, b \stackrel{R}{\in} \mathbb{Z}$ gegeben. Die Fragestellung $ab \stackrel{?}{=} c$ ist schwierig.

$$\text{DDH} \leq_{\mathcal{P}} \text{CDH} \leq_{\mathcal{P}} \text{DLOG}$$

⋮⋮⋮ RSA-Schwellenwertsignaturschema

■ Schlüsselerzeugung: (Fortsetzung)

5. Bildet das Teilungspolynom $F(x) = \sum_{i=0}^{k-1} a_i x^i$ aus $\mathbb{Z}_\mu[x]$ mit zufälligen Koeffizienten $a_i \stackrel{R}{\in} \mathbb{Z}_\mu$ für $1 \leq i \leq k-1$ und $a_0 = d$ (geheimer Exponent). Nun werden die geheimen Schlüsselteile aller Teilnehmer $1 \leq i \leq n$ berechnet:

$$s_i = F(i)\Delta^{-1} \pmod{\mu} \quad \text{mit} \quad \Delta = n!$$

6. Wählt zufälligen Quadratrest^a $v \stackrel{R}{\in} \text{QR}_\nu$ und berechnet Verifikationsteile $v_i = v^{s_i}$ für alle i .
7. Wählt $u \stackrel{R}{\in} \mathbb{Z}_\nu^*$ mit Jacobi-Symbol $\left(\frac{u}{\nu}\right) = -1$, d. h. $u \in \text{INQR}_\nu$.
8. Verteilt $K_{\text{PUB}} = (\nu, e)$, v , u und v_i öffentlich und s_i geheim.

^a $\text{QR}_\nu = \{a \in \mathbb{Z}_\nu^* \mid \exists x \in \mathbb{Z}_\nu : x^2 \equiv a \pmod{\nu}\}$, $\text{INQR}_\nu = \mathbb{Z}_\nu^* \setminus \text{QR}_\nu$

⋮⋮⋮ RSA-Schwellenwertsignaturschema

■ *Teilsignaturkombination* $S = \{i_1, \dots, i_k\} \subset \{1, \dots, n\}$:

1. Hinsichtlich der Teilsignaturen gilt: $\forall i_j \in S : x_{i_j}^2 = x^{4s_{i_j}}$
2. Interpolation an der Stelle 0 ($4d$ im Exponenten):

$$w = x_{i_1}^{2\lambda_{0,i_1}^S} \cdot x_{i_2}^{2\lambda_{0,i_2}^S} \cdot \dots \cdot x_{i_k}^{2\lambda_{0,i_k}^S} \pmod{\nu} \quad \lambda_{0,j}^S := \Delta \frac{\prod_{j' \in S \setminus \{j\}} (-j')}{\prod_{j' \in S \setminus \{j\}} (j - j')}$$

3. Bestimme ganze Zahlen $a, b \in \mathbb{Z}$ mit $4a + eb = 1$.
4. Gesamtsignatur: $y = w^a x^b \pmod{\nu}$ [$y \equiv x^{4da+b} \pmod{\nu}$]

■ *Gesamtsignaturverifikation (wie bei RSA)*: $y^e \stackrel{?}{\equiv} x \pmod{\nu}$

Satz 1 (SHoup 2000)

Für $k \geq t + 1$ ist obiges Schema (im Random-Orakel-Modell für h, h') sicher (d. h. robust und unfälschbar), vorausgesetzt RSA- und DDH-Problem sind jeweils schwer.

⋮⋮⋮ Schwellenwertmünzwurfschema

Definition 9 ($/n, k, t/$ -Schwellenwertmünzwurfschema)

Von insgesamt n Teilnehmern (bis zu t können Angreifer sein) müssen mindestens k zusammenarbeiten, um den Ausgang (Wappen oder Zahl) eines unvorhersagbaren Münzwurfs mit der Bezeichnung $C \in \{0, 1\}^*$ zu erfahren.

- Bedingung $t < k \leq n - t$ sinnvoll; praktisch meist $k = n - t$.
- Münzwurf ist unvorhersagbare Funktion $f : \{0, 1\}^* \rightarrow \{0, 1\}$
- $\forall i$: Münzteilherzeugung, Münzteilverifikation, Münzteilkombination
- **Robustheit:** Angreifer kann keine k gültigen Münzteile zu C erzeugen lassen, bei denen die *Münzteilkombination* eine Münze mit anderem Wert als $f(C)$ liefert.
- **Unvorhersagbarkeit:** Angreifer kann den Wert $f(C)$ einer unbekanntes Münze C nicht mit höherer Wahrscheinlichkeit als $1/2$ vorhersagen, ohne wenigstens $k - t$ gültige Münzteile von C zu besitzen.

⋮⋮⋮ DC-Netzwerk (Komplexität)

	Nachrichten- komplexität	optimale Runde $n = 25, t = 8$ $ \nu = 1024$ Bit	Nachrichten- komplexität	gestörte Runde $n = 25, t = 8$ $ \nu = 1024$ Bit
ABBA	exp. $O(n^2)$		exp. $O(n^2)$	
DC-PfWa	$O(n^3)$		$O(n^3)$	
ABBA ○ DC-PfWa	$O(n^5)$		$O(n^5)$	
Nutzlast		1 – 25 KB		25 KB
GnuDCN	$O(n^3)$	182 MB	$O(n^4)$	988 MB
+ Broadcast	$O(n^4)$	4.4 GB	$O(n^5)$	24.1 GB

