Signaturen und Null-Wissen-Beweis

# Königreich des Käse

Juni 2023

ubique

Geheimnis Bindung

# RSA Signaturen

ubique

RSA Signaturen

# Warum Signaturen?

Origin: Bordeaux

Vintage: 1964

Und jetzt digital?

# RSA Signatur

1. Wir kennen RSA-Verschlüsselung
2. Mit dem öffentlichen Schlüssel wird verschlüsselt
3. Mit dem privaten Schlüssel entschlüsselt

**Aber was passiert wenn man mit dem privaten Schlüssel verschlüsselt?**

**Jeder kann die Nachricht mit dem öffentlichen Schlüssel entschlüsseln!**

**(die beiden Schlüssel sind jeweils das Inverse des Anderen)**

# RSA Signatur

**Aber was bringt uns das jetzt?**

**Können wir als Fingerabdruck benutzen (Signatur!)**

{"typ":"JWT","alg":"RS256"}.
{"vintage":"Bordeaux","origin":"1964","exp":1686224874}

Kann nun mit öffentlichem Schlüssel entschlüsselt werden:

Liefert uns den “SHA-Hash” der Nachricht!

# RSA Signatur

- Alice erstellt Signatur mit privatem Schlüssel

$$S \equiv h^d \; mod \; n$$

- Bob benutzt öffentlichen Schlüssel um Signatur zu entschlüsseln

$$S^e \; mod \; n \equiv \left(h^d\right)^e mod \; n \equiv (h^e)^d \; mod \; n \equiv h \; mod \; n$$

- Bob kann nun den Hash der Nachricht erstellen und mit S vergleichen
- Da nur Alice eine Nachricht verschlüsseln kann, muss Nachricht von Alice sein

RSA Signaturen

# Aufgaben

- Erstelle ein RSA-Schlüsselpaar und signiere deinen Anfangs Buchstaben
- Publiziere deinen öffentlichen Schlüssel
- Verschlüssle deine Signatur mit einem anderen öffentlichen Schlüssel (sende eine Nachricht von dir)

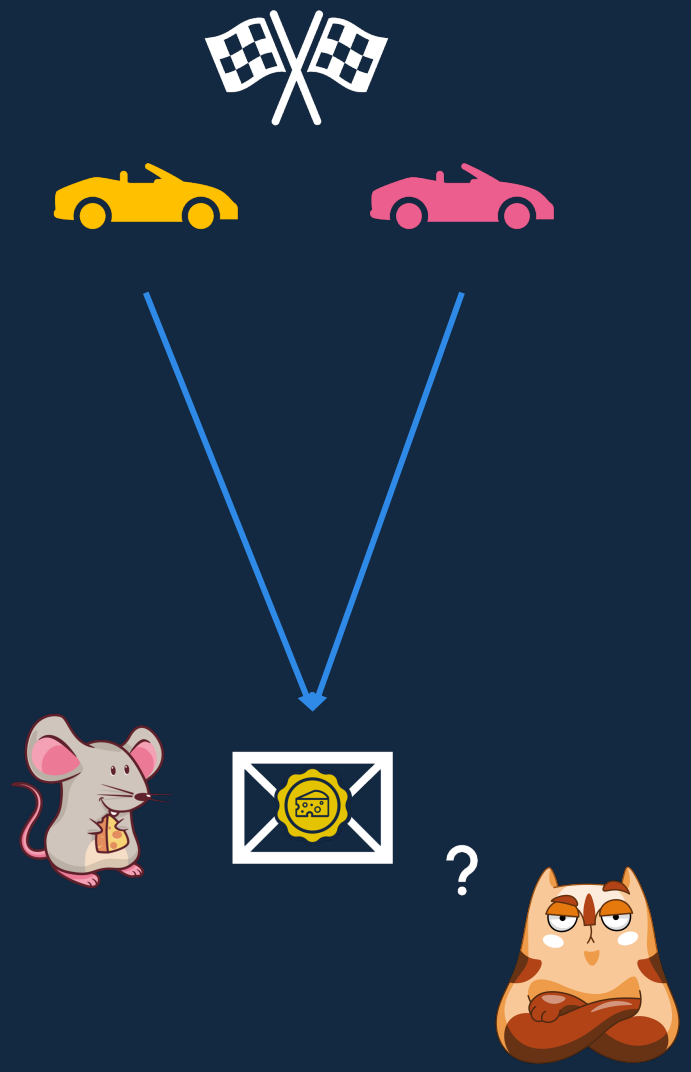
Geheimnis Bindung

# Pedersen Commitment

ubique

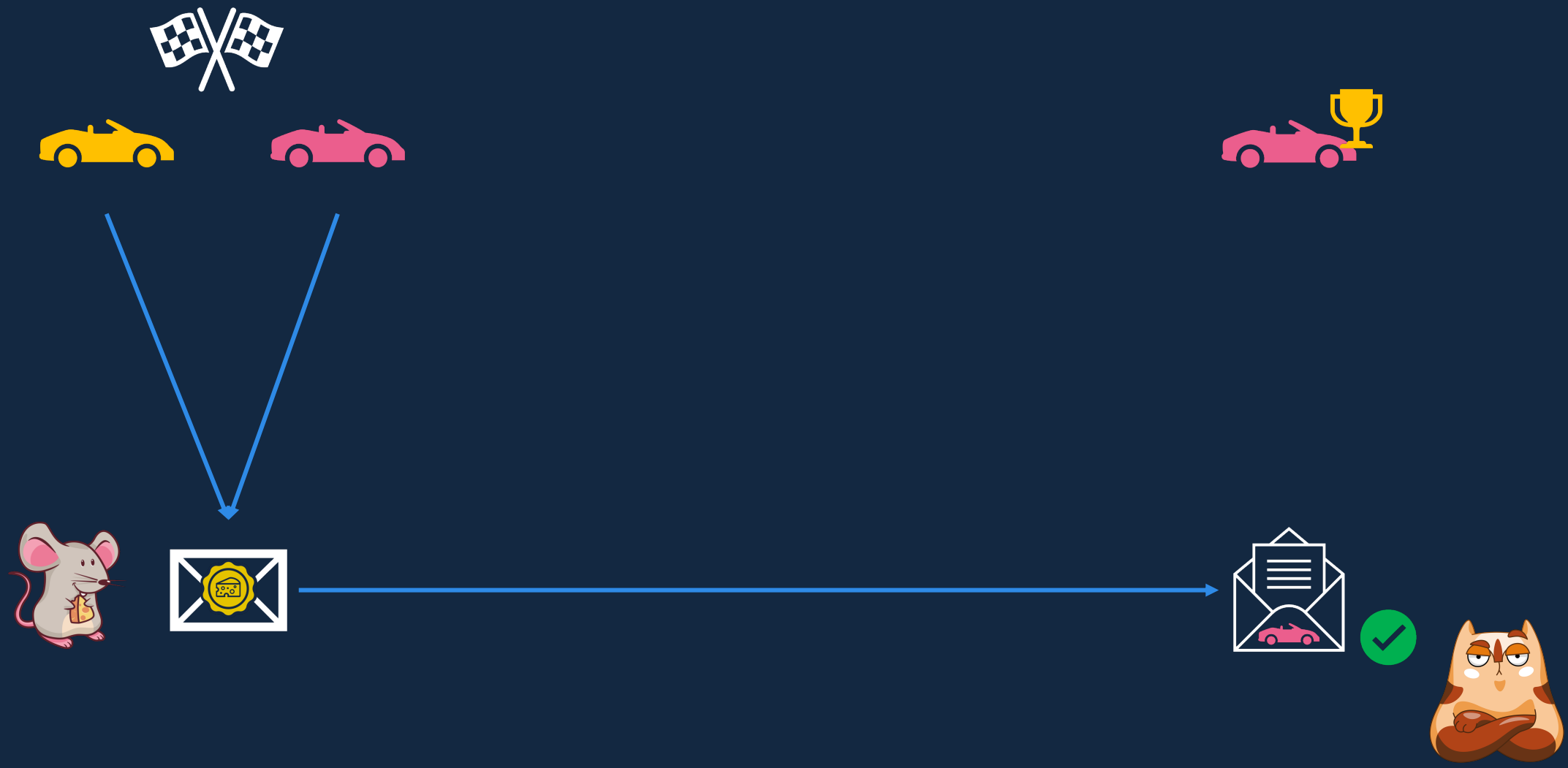
Geheimnis Bindung

# Pedersen Commitment



Linked-secrets

# Pedersen Commitment



Geheimnis Bindung

# Pedersen Commitment [1] (Algorithmus)

1. Wir brauchen einen “Ring” mit Ordnung gleich einer Primzahl
2. Wir suchen zwei “Generatoren” ($g, h \in \mathrm{R}$) → Öffentlicher Schlüssel
3. Wir bezeichnen $C = g^x h^r$ als **Bindung** zum Geheimnis x mit Versteckfaktor r zur Basis $(g, h)$

[1] https://link.springer.com/content/pdf/10.1007/3-540-46766-1_9.pdf

Geheimnis Bindung

# Pedersen Commitment (Properties)

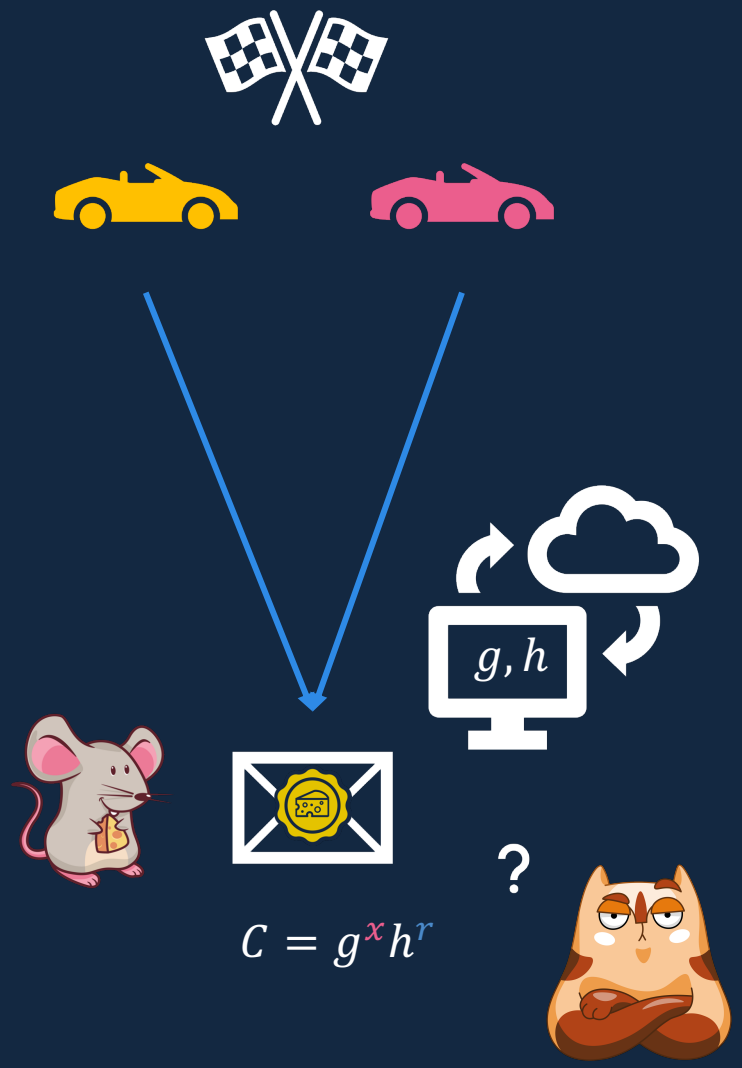
- Es ist bindend $C' \neq C \; für \; x' \neq x$
- Es ist versteckend weil $C$ alleine keine Informationen über $x$ preisgibt
- (additiv) homomorph
  - Benutzt man die Gruppenstruktur die dem Ring $R$ zu Grunde liegt kann man zeigen dass:

$$C_1/C_2 = (g^{x_1}h^{r_1})/(g^{x_2}h^{r_2}) = g^{x_1-x_2}h^{r_1-r_2} \xrightarrow[x_1=x_2]{} C_1/C_2 = h^{r_1-r_2}$$

  - Wenn Jenny $r_1 - r_2$ preisgibt, kann Tomas prüfen, dass $C_1, C_2$ zwei Bindungen zum gleichen x sind

Geheimnis Bindung

# Pedersen Commitment



Geheimnis Bindung

# Pedersen Commitment


$C = g^x h^r$

$x, r$

# Aufgaben (Zahlenbingo)

- N = 403, g= 256, h=225
- Bildet dreier Gruppen
- Wählt eine Zahl und erstellt eine Bindung dazu (beachtet dass die Gruppensumme 31 sein muss)
- Berechnet den Versteckungsfaktor für die Summe der Zahlen
- Publiziert die vier Bindungen und den Versteckungsfaktor für die Summe

Proof of knowledge

# Schnorr Protocol

ubique

Null Wissen Beweis

# Schnorr Protocol

1. Um zu beweisen, dass wir das Geheimnis kennen (ohne es zu zeigen) kann man das Schnorr Protokoll benutzen

2. Mathematisch beweisen wir, dass wir den (diskreten) Logarithmus kennen

3. Das Schnorr-Protokoll ist ein interaktives Protokoll

Null Wissen Beweis

# Schnorr Protocol

1. Wir starten mit der Bindung $C = g^x h^r$

2. Der Beweisende wählt $\beta$ und $\gamma$ zufällig und sendet $a = a_1 a_2 = g^\beta h^\gamma$ zum Prüfer

3. Der Prüfer erstellt eine zufällige Challenge c und schickt sie zum Beweisenden

4. Der Beweisende berechnet $s_1 = \beta - cx \pmod{q}, s_2 = \gamma - cr \pmod{q}$ und schickt diese Werte zum Prüfer

5. Der Prüfer kann nun folgendes berechnen

$$C' = g^{s_1} h^{s_2} C^c = g^\beta g^{-cx} h^\gamma h^{-cr} g^{cx} h^{cr} = g^\beta h^\gamma = a_1 a_2 = a$$

Null Wissen Beweis

# Schnorr Protocol

1. Der Beweisende muss jedesmal neue Zufallsfaktoren r bestimmen

2. Ansonsten kann das Geheimnis einfach gefunden werden

$$s_1 = \beta - c_1 x, \qquad s_2 = \beta - c_2 x$$

$$s_1 - s_2 = (c_2 - c_1)x$$

$$x = \frac{s_1 - s_2}{c_2 - c_1}$$

Gewählte Veröffentlichung

# Camenisch-Lysyanskaya

ubique

Kingdom of Cheese

# Gewählte Veröffentlichungs Problem


Origin: Bordeaux

Vintage: 1964

# Camenisch-Lysyanskaya

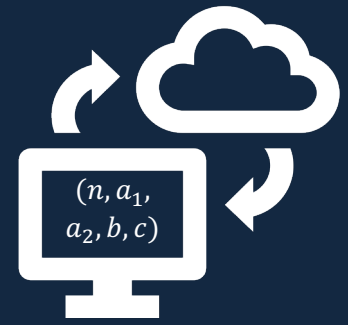
- Die Sicherheit des Verfahrens basiert auf der Gleichen Annahme wie bei RSA
- Das Verfahren ermöglicht:
  - Teilweise Veröffentlichung (Signaturbesitzer entscheidet)
  - Nullwissens Beweis der Signatur

# Camenisch-Lysyanskaya

1. Schlüsselerstellung
2. Signierungsprozess
3. Verifizierungsprozess

Camenisch-Lysyanskaya

# Schlüsselerstellung


1. Der Signaturaussteller wählt einen RSA-Modulos n = p*q

2. Wir wählen zufällig Element $(a_1, a_2, b, c)$ aus den sogenannten Quadratic Residues (QR) ($QR = \{a, b \in \mathbb{Z}_n^* | b^2 = a\}$)

3. Der öffentliche Schlüssel entspricht nun $(n, a_1, a_2, b, c)$ wobei der private Schlüssel der Faktorisierung von n entspricht

Camenisch-Lysyanskaya

# Signierungsprozess

1. Für $m = (m_1, m_2)$, wählt man eine zufällige Primzahl e (prime) und ein s
2. Berechne v
3. Die Signatur: $(s, e, v)$

$$v^e = a_1^{m_1} a_2^{m_2} b^s c \bmod n$$

$$v = \left(a_1^{m_1} a_2^{m_2} b^s c\right)^{\frac{1}{e}} \bmod n$$

$m_1$ = Bordeaux

$m_2$ = 1980

Camenisch-Lysyanskaya

# Verifizierungsprozess


1. Man prüft $v^e \equiv a_1^{m_1} a_2^{m_2} b^s c \; mod \; n$ und bestimmte Bedingungen von e
2. In diesem Fall braucht Tomas alle Eigenschaften $(m_1, m_2)$

Camenisch-Lysyanskaya

# Verifizierungsprozess (einzelne Eigenschaften)

$m_1$ = Bordeaux

$m_2$ = *****

Bordeaux Club

Mit Bindung $C = a_2^{m_2} b^r$ können wir die Signatur anpassen $(s - r, e, v)$ →

$$v^e \equiv a_1^{m_1} C b^{s-r} c = a_1^{m_1} a_2^{m_2} b^r b^{s-r} c = a_1^{m_1} a_2^{m_2} b^s c \bmod n$$

Was immer noch eine gültige Signatur ist $(s, e, v)$

Null-Wissen Beweis

# Lagrange 4 Quadrate

ubique

# Lagrange 4 Quadrate

Origin: Bordeaux

Vintage: zwischen **1960-1970**

# Lagrange 4 Quadrate

1. Alter muss zwischen 1960 und 1970 liegen
2. Sprich: $x - 1960 > 0 \wedge 1970 - x > 0$
3. Lagrange's vier Quadrate Satz besagt folgendes: Jede natürliche Zahl lässt sich als Summe von vier Quadraten schreiben

$$p = a^2 + b^2 + c^2 + d^2$$

4. Wenn man also zeigen kann, dass es vier Zahlen gibt so dass $x - 1960 = a^2 + b^2 + c^2 + d^2$
5. Sowie vier weitere $1970 - x = a^2 + b^2 + c^2 + d^2$
6. Dann weiss man das x zwischen 1960 und 1970 liegen muss

Null-Wissen Beweis

# Bereichsbeweis

ubique