Biometric Crypto-Systems

Felix Baumann, Ravinder Sangar, Jonas Winkler

25. Juni 2020

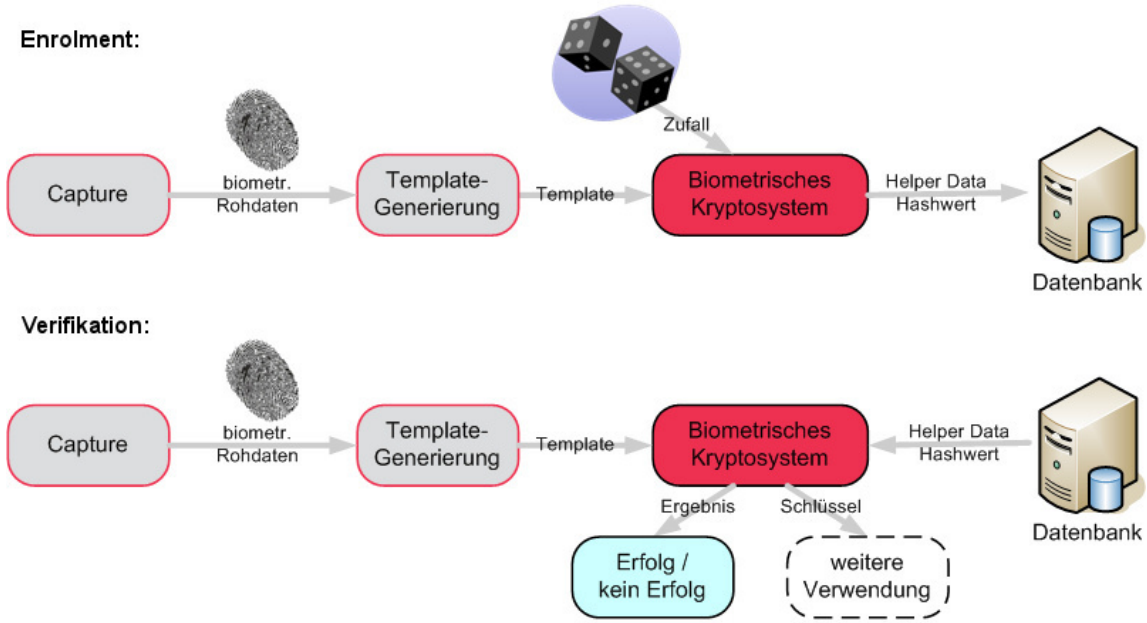
Inhalt

Biometrische Kryptosysteme

- Biometrische Daten sind biologische Messwerte, die Personen eindeutig identifizieren
- Bsp: Fingerabdruckscan, Gesichtserkennung, Irisscan
- Biometrische Kryptosysteme verbinden Kryptographie mit Biometrie

Quelle: *https://www.cancom.info/2019/03/gesichtserkennung-technologie/*

Funktionsweise

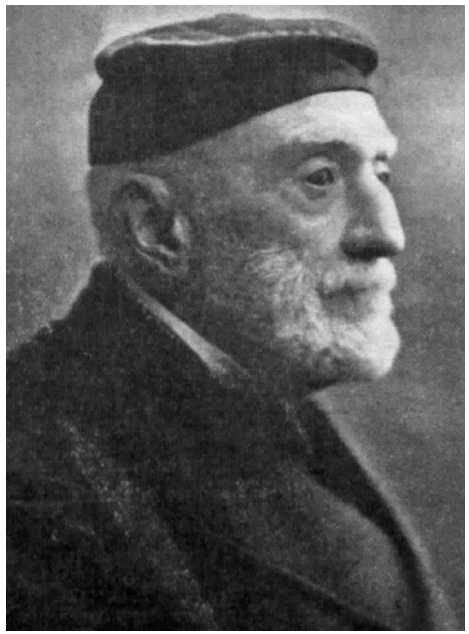


Quelle: *http://2014.kes.info/archiv/online/images/08-6-052-1-400.jpg*

Inhalt

Idee hinter Biometrie

- Erste Biometrie war der Fingerabdruck
- Vor 4000 Jahren wurde mit Fingerabdrücken unterzeichnet
- Henry Fauld fand heraus, dass Fingerabdrücke individuell sind



Quelle: *de.wikipedia.org*

Heutige Verwendung

- Fingerabdrücke werden heute in Forensik eingesetzt
- Gesichtserkennung bei einigen Smartphone und in Flughäfen
- Schlüssel für Gebäude werden von Fingerabdruckscane abgelöst
- Biometrischer Pass wird weltweit eingesetzt: Bild des Gesichts und 2 Fingerabdruckbilder

Quelle: *wikimedia.org*

Problematik

- Alle biometrischen Daten, die gesammelt wurden, können gehackt werden
- Biometrische Daten können nicht verändert werden, wie ein Passwort
- Duplikation von Daten: Fingerabdrücke können im Alltag unbemerkt abgenommen werden, Hochauflösende Fotos von Gesichtern

Biometric Template Protection

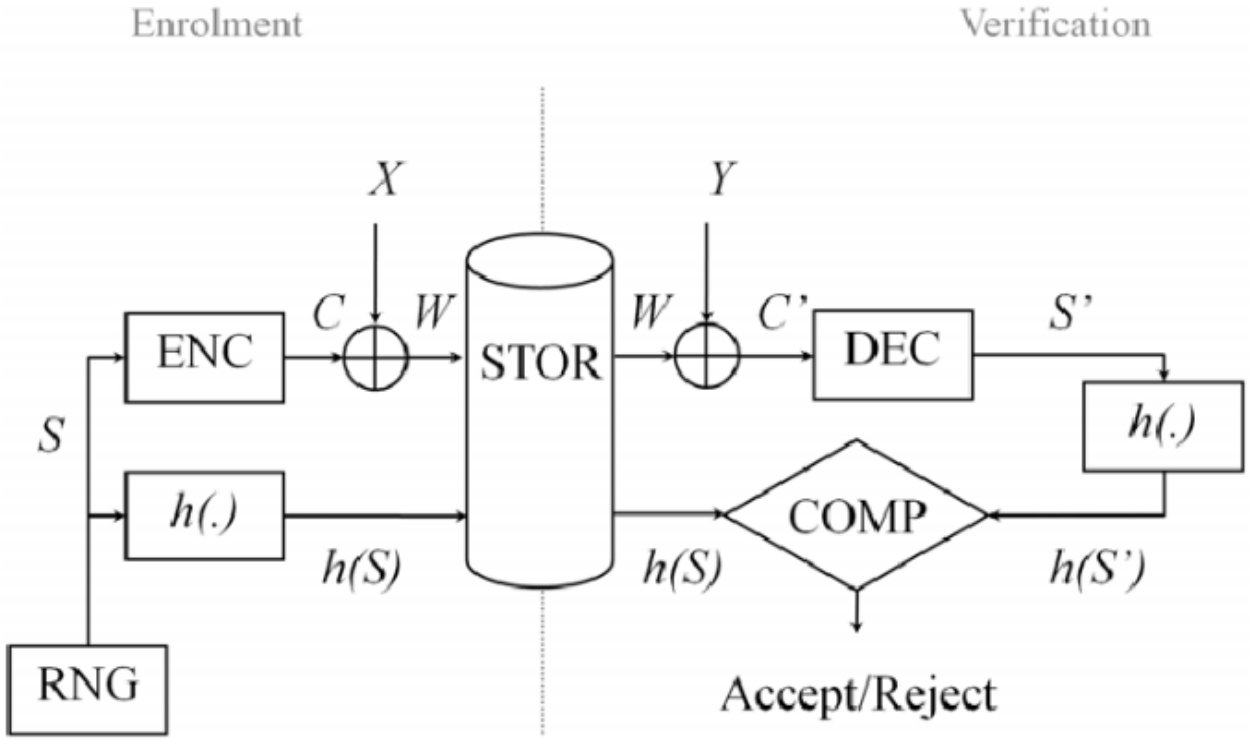
- Fingerabdruck nicht änderbar $\rightarrow$ muss geschützt werden
- Template nicht direkt gespeichert
- umgewandelt in „Protected Templates“
- reicht aus für Authentifizierung

Fuzzy Commitment

- Commitment-Schema generell
  - $G : C \times X \rightarrow W$
  - *binding*-Eigenschaft
  - *hiding*-Eigenschaft
- Fuzzy Commitment besteht aus 2 Phasen:
  - *Enrollment*-Phase: Initialisierung und Anlegen des Templates & Schlüssels
  - *Authentication*-Phase: Verfahren zur Authentifizierung

Enrollment

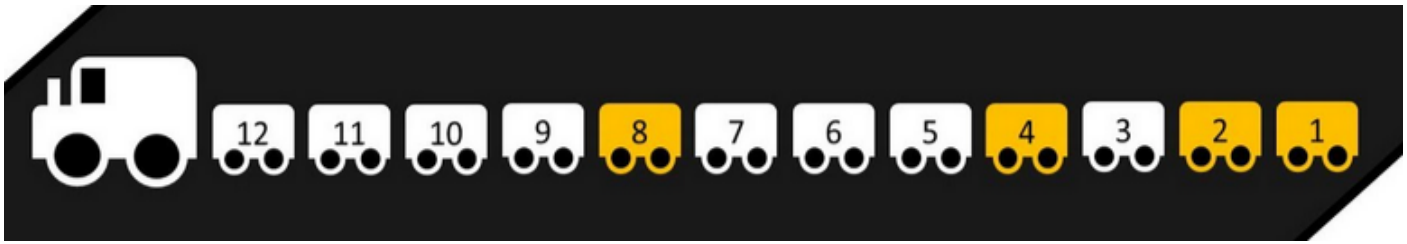
- zufälliger Wert für Schlüssel s, One-Way-Hashfunktion $h(s)$ generiert
- Schlüssel s in Hamming-Code c umgeschrieben
- Template (als Bitstring) x XOR c ergibt „Safe“ w



Quelle: *https* : *//www.cosy.sbg.ac.at/ uhl/biometrics_slides.pdf*

Hamming-Code I

- an jede 2^x-te Position kommt Paritätsbit
- alle **Stellen** mit Wert 1 werden verxort
- Ergebnis stellt Wert für Paritätsbits dar



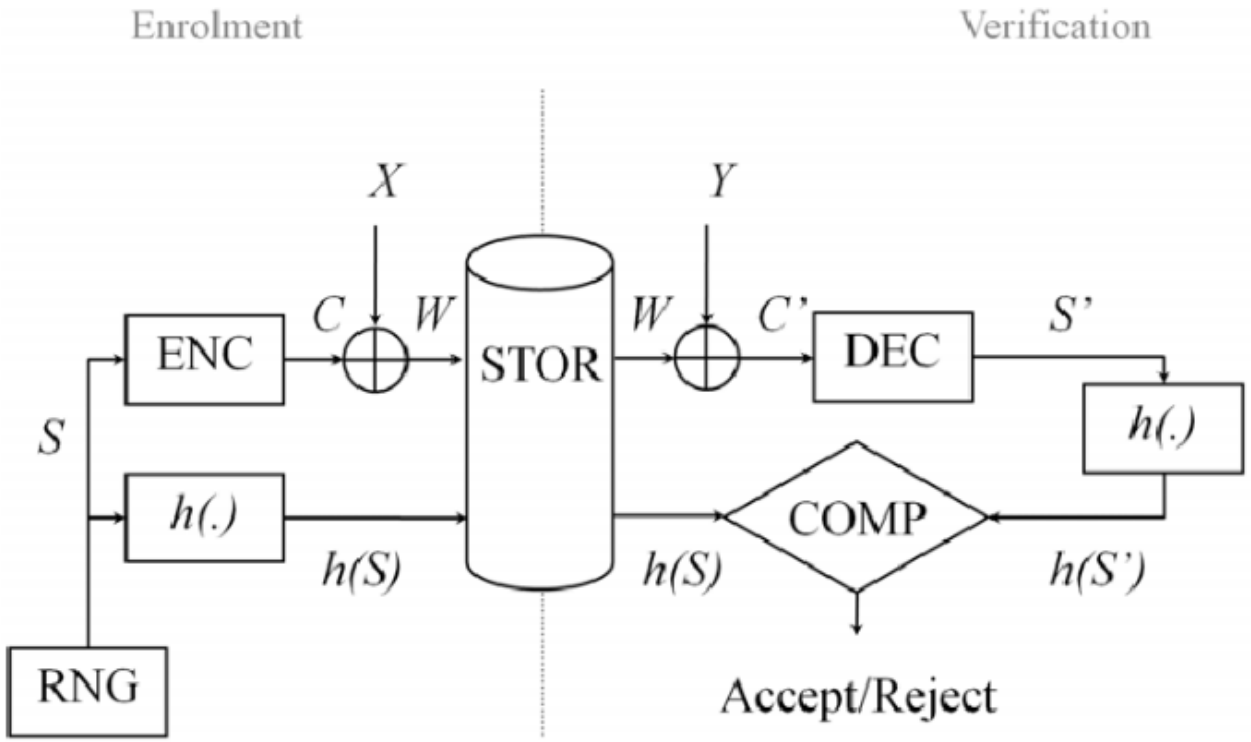
Quelle: *https : //www.cybersicherheit.guru/der — hamming — code/*

Hamming-Code II - Beispiel

- zu codieren: 1101
- füge Paritätsbit an Positionen 1, 2 und 4 ein $\rightarrow$ 110x1xx
- Stellen, an denen Wert 1 ist miteinander verXORen $\rightarrow$ Stellen 7, 6, 3
- 7 == 111
- 6 == 110
- 5 == 101
- xor == 100 $\rightarrow$ Paritätsbits haben Werte 1, 0, 0
- Codewort: 1101100

Authentication I

- neues Template y wird eingelesen (User hält Finger an Sensor)
- $w \oplus y = c'$ (Kandidaten-Codewort)



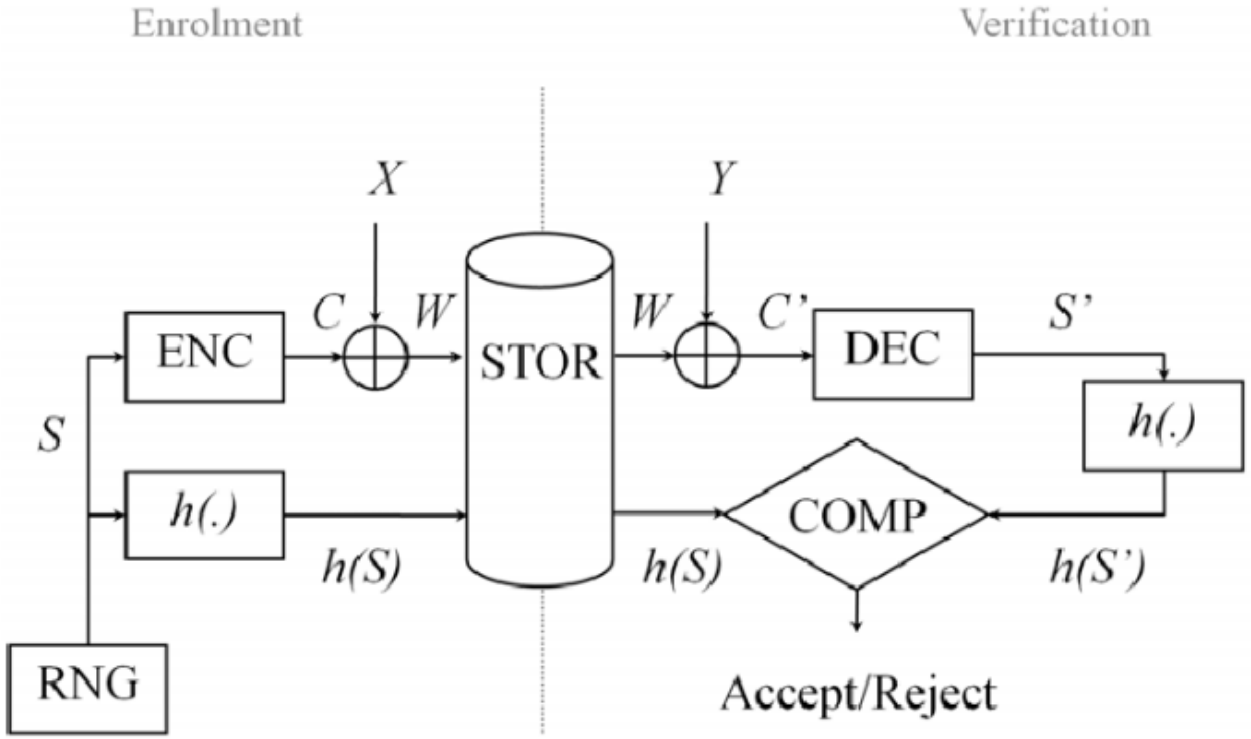
Quelle: *https* : *//www.cosy.sbg.ac.at/ uhl/biometrics_slides.pdf*

Hamming-(De)code

- c' muss nun decodiert werden $\rightarrow$ es werden erneut die Stellen aller 1en verXORt
- Beispiel: $c' = 0010010$
- $2 == 010$
- $5 == 101$
- xor $== 111 \rightarrow$ bedeutet an der Stelle 3 ist ein Bitfehler aufgetreten, er kann korrigiert werden
- wenn Ergebnis $== 0 \rightarrow$ fehlerfreie Übertragung
- nur 1-Bit-Fehler kann korrigiert werden

Authentication II

- Paritätsbits werden wieder entfernt $\rightarrow$ Kandidatenschlüssel s'
- Einsetzen von s' in $h(.)$, wenn $h(s) == h(s') \rightarrow$ Authentifizierung erfolgreich



Quelle: *https* : *//www.cosy.sbg.ac.at/ uhl/biometrics_slides.pdf*

Fuzzy Commitment

- Vorteile
  - entstehende Unschärfe kann ausgeglichen werden
  - Template selbst wird nicht gespeichert $\rightarrow$ gut geschützt
- Nachteile
  - Template muss als Bitstring dargestellt werden (möglichst kurz, da nur 1-Bit-Fehler erkannt werden)
  - Länge des Template-Bitstrings x muss mit jener des Keys s übereinstimmen, wegen XOR

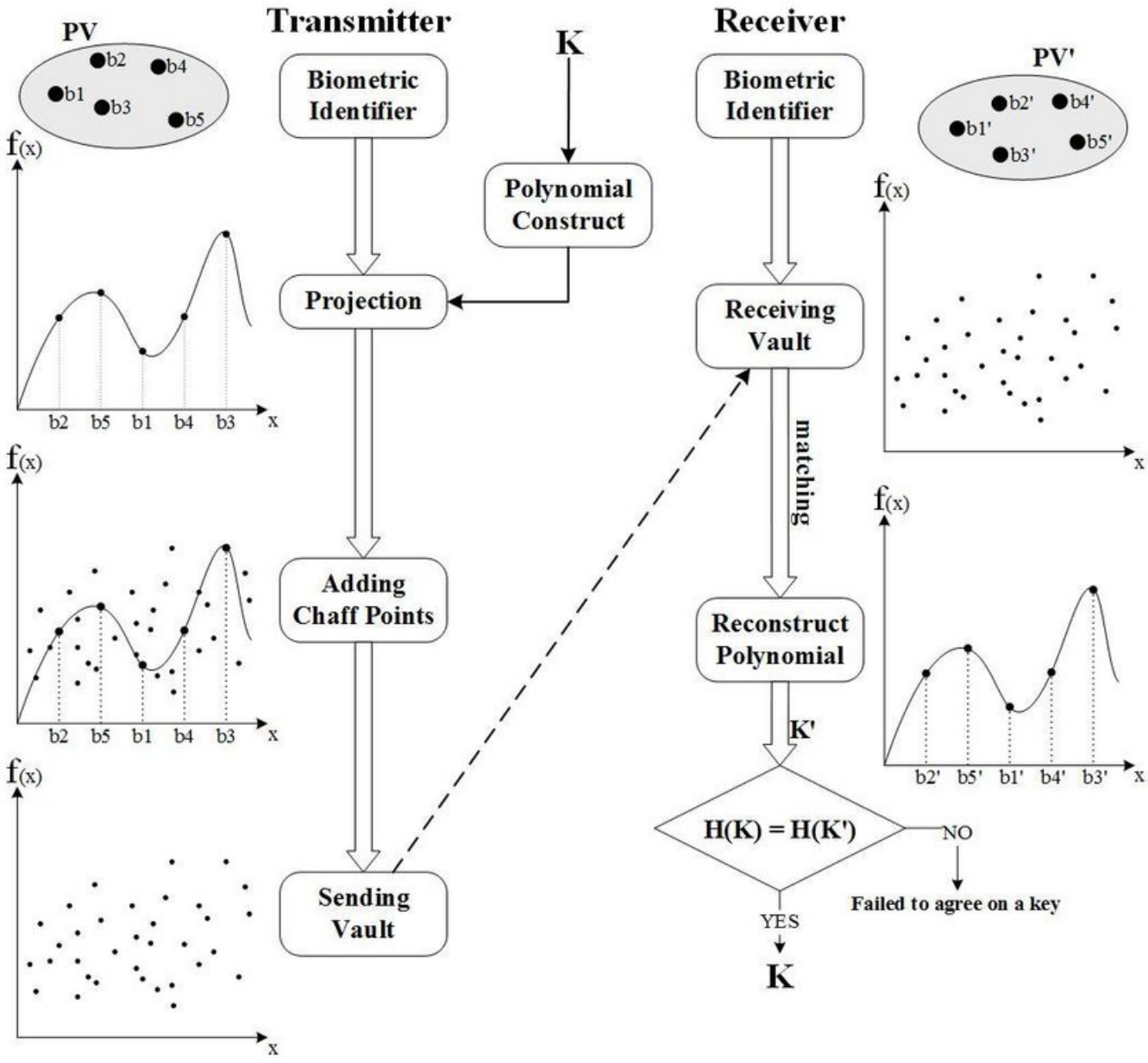
Fuzzy Vault

- Fingerabdrücke sind nicht zu 100% reproduzierbar
- Konzept das Fehler tolleriert?

Fuzzy Vault

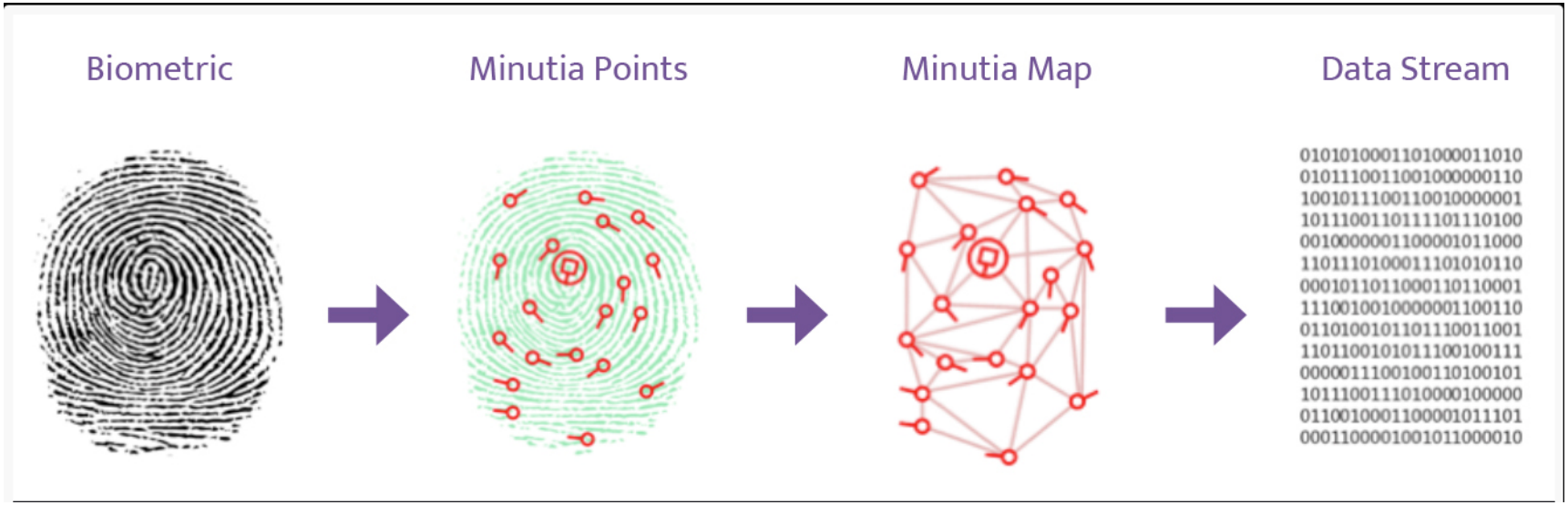
- Alice möchte wissen ob jemand ihre Interessen teilt
- Alice sichert Interessen in einem Save
- Bob hat ähnliche Interessen
- Bob kann den Save entsperren

Prinzip



Quelle: *https : //www.researchgate.net/figure/Key − distribution − solution − based − on − fuzzy − vault − scheme − 112_fig1_321080531/*

Biometric Template



Quelle: *http* : *//www.tellen.co.nz/biometric — doorkeeper/prettyPhoto*[*gallery*6320]/0

Parameter

- Universum F
- Secret s
- Polynomfunktion p
- Bob kann den Save entsperren

LOCK

$X, R \rightarrow \emptyset$

$s \rightarrow p$

for $i = 1$ *to* t

$(a_i, p(a_i)) \rightarrow (x_i, y_i)$

$X \bigcup \{x_i\} \rightarrow X$

$R \bigcup \{(x_i, y_i)\} \rightarrow R$

for $i = t + 1$ *to* r

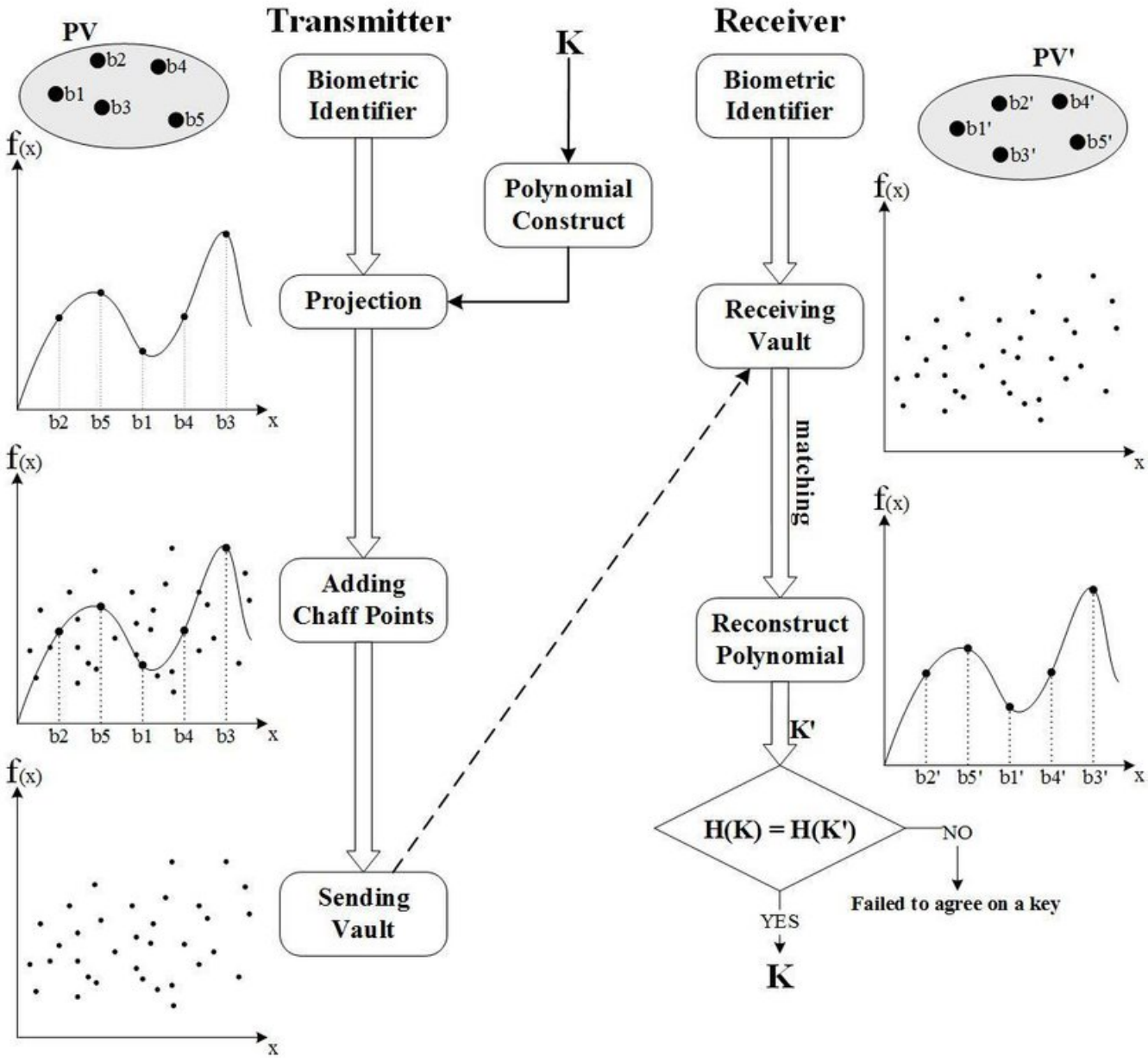
$x_i \in F - X$

$y_i \in F - \{(x_i, y_i\}$

$R \cup \{(x_i, y_i)\} \rightarrow R$

return R

Prinzip



Quelle: *https : //www.researchgate.net/figure/Key − distribution − solution − based − on − fuzzy − vault − scheme − 112_fig1_321080531/*

UNLOCK

$Q \rightarrow \emptyset$

for $i = 1$ *to* t

$R \xrightarrow{b_i, \circ} (x_i, y_i)$

$Q \bigcup (x_i, y_i) \rightarrow Q$

$RS_{DECODE}(k, Q) \rightarrow s'$

return s'

Vor- und Nachteile

- Vorteile
  - Fehler der Sensoren/Finger werden toleriert
  - Chaff-Points bestimmen die Sicherheit
- Nachteile
  - eventuell weniger Sicherheit gegenüber anderen Verschlüsselungssysteme bei wenigen Chaff-Points
  - Risiko für hohe Komplexität

Vielen Dank für Ihre Aufmerksamkeit