
LOREM IPSUM...

MATTHIAS ULBRICH
2743974

BACHELORARBEIT

Prof. Dr. Michael Meier, Dipl.-Inf. Saffija Kasem-Madani
Institut für Informatik IV
Arbeitsgruppe für IT-Sicherheit
Rheinische-Friedrich-Wilhelms-Universität Bonn

DANKSAGUNG

Besonders bedanken möchte ich mich bei meiner Betreuerin Dipl.-Inf. Saffija Kasem-Madani für die akademische Begleitung während der Bachelorarbeit und bei Prof. Dr. Michael Meier für die Gelegenheit in seiner Arbeitsgruppe eine Bachelorarbeit zu schreiben.

Weiterer Dank gilt meinen Eltern, meiner Freundin und meiner Großtante für jegliche Unterstützung in meinem Studium bis zu diesem Punkt. Ich widme diese Arbeit euch.

» *Do what you can't!* «

Casey Neistat, 2017

SELBSTSTÄNDIGKEITSERKLÄRUNG

Hiermit versichere ich, die vorliegende Bachelorarbeit ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Bonn, 13. April 2017

Matthias Ulbrich

ZUSAMMENFASSUNG

Im Rahmen dieser Bachelorarbeit werden in [insert label] die Grundlagen eingeführt um den Umgang und die Mächtigkeit der verschiedenen Kryptosystemen greifbar zumachen. Anschließend werden in [insert label] verschiedene Anwendungsfälle von Kryptosystemen vorgestellt und Motivationen der Forscher für ihre Anwendung extrahiert. Dies passiert unter dem Hintergrund die Kryptosysteme nach ihrem Nutzen in [insert label] zu kategorisieren. Bei allen untersuchten Arbeiten wurde weiter berücksichtigt, ob sich die Forscher für die Integrität der verarbeiteten Daten interessierten, oder ob sie Verschlüsselung vorwiegend benutzen um Vertraulichkeit zu realisieren.

INHALTSVERZEICHNIS

1	EINLEITUNG	1
2	HOMOMORPHE KRYPTOSYSTEME	2
2.1	Schutzziele der Kryptografie	2
2.2	Kryptosysteme	2
2.2.1	Mathematische Grundlagen	4
3	SICHERHEITSKRITERIEN	6
3.1	Malleability	6
3.2	Privacy-Preserving	6
3.3	Sicherheitsklassen	6
3.3.1	Ununterscheidbarkeit von Geheimtexten (Ciphertext Indistinguishability) . .	6
3.3.2	Semantische Sicherheit	6
4	KLASSIFIKATION HOMOMORPHER KRYPTOSYSTEME	7
4.1	Aufteilungen	7
4.2	Autocrypt	7
4.3	Machine Learning Classification over encrypted data	8
4.4	Privacy Preserving Matrix Factorization	8
4.5	Efficient and Secure Comparison for On-Line Auctions	9
4.6	Fingerprinting Protocol for Images Based on Additive Homomorphic Property	9
4.7	Privacy Preserving Face Recognition	10
4.8	Private predictive analysis on encrypted medical data	12
4.9	Sichere Berechnung von Funktionen mit SMC oder HE	13
4.10	Bitweise Verschlüsselung vs Integerverschlüsselung	13
5	ZUM ABSCHLUSS	14
6	LITERATURVERZEICHNIS	15

1 EINLEITUNG

Ziel dieser Arbeit ist eine Erörterung der Vorteile von Anwendungen homomorpher Kryptosysteme. Homomorphe Kryptosysteme gliedern sich ihrerseits in semihomomorphe und vollhomomorphe Kryptosysteme. Während semihomomorphe Kryptosysteme eingeschränkt Operationen auf Chiffren ermöglichen - abhängig von dem gewählten Kryptosystem, ermöglichen vollhomomorphe Kryptosysteme die Berechnung jeder booleschen Funktion im Chifferraum. Da letztere jedoch aufgrund hoher Laufzeiten weniger praktikabel sind greift man oft auf semihomomorphe Kryptosysteme zurück [insert citation]. Dies führt in der Praxis zu verschiedenen, vom jeweiligen Anwendungsfall abhängigen Einsatz eines Kryptosystems.

Der Beitrag dieser Bachelorarbeit:

Es ist von vornerein nicht klar welche Gründe jemand gewählt hat ein Kryptosystem dem anderen vorzuziehen, insbesondere wenn sie den gleichen Operator im Chifferraum zur Verfügung stellen. Weiter lässt sich mit einem Kryptosystem wie Paillier der Operator des Kryptosystems von Goldwasser-Micali simulieren [insert label]. Diese Arbeit will die Gründe für den Einsatz verschiedener homomorpher Kryptosysteme untersuchen und Informationen sammeln um die einzelnen Kryptosysteme zu kategorisieren als Referenz für den zukünftigen Einsatz homomorpher Kryptographie in Forschungsarbeiten.

2 HOMOMORPHE KRYPTOSYSTEME

2.1 SCHUTZZIELE DER KRYPTOGRAPHIE

Die Kryptografie möchte mehrere Schutzziele für die Speicherung, Vervielfältigung und Übertragung von Informationen umsetzen und dazu verschiedene Verfahren bereitstellen. Grundlegende Schutzziele beim Übertragen von Informationen zwischen mehreren Personen mit Nachrichten sind dann [DKK02, p.2]:

1. **Vertraulichkeit:** Keine unauthorisierte Kenntnisnahme. Nur dazu berechnigte Personen sollen eine Information lesen können oder Zugang zur einer Information bekommen.
2. **Integrität:** Keine unauthorisierte unbemerkte Datenmanipulation. Dies schützt insbesondere vor dem Hinterlegen von Falschdaten in einer Nachricht oder einer oder dem Fehlen von Teilen einer Nachricht.
3. **Authentizität:** Der Empfänger kann den Verfasser einer Nachricht verifizieren.
4. **Nichtabstreitbarkeit:** Der Sender kann dem Versand nicht mehr abstreiten der Verfasser einer Nachricht gewesen zu sein.

2.2 KRYPTOSYSTEME

Ein Kryptosystem ist ein Sammlung von Algorithmen um das Schutzziel der Vertraulichkeit beim Übertragen von Informationen in Nachrichten umzusetzen.

Damit ermöglicht ein Kryptosystem zwei Parteien Alice und Bob über einen ungeschützten Kanal in dem die Nachricht übertragen wird zu kommunizieren, ohne das eine dritte Partei welche mithört Zugang zu dem Inhalt der Nachricht bekommt.

Die zugrundelegenden Algorithmen und resultierende Eigenschaften über die Beziehung von Klartexten zu Chiffretexten führen zu ein verschiedenen Klassen von Kryptosystemen. Diese Kryptosysteme führen wir in diesem Abschnitt ein. In der Literatur ist es üblich bei „einfacheren“ Kryptosystem die deterministisch oder symmetrisch sind, diese Bezeichnungen wegzulassen. Zur besseren Abgrenzung werden in diesem Abschnitt Kryptosysteme immer mit ihren Eigenschaften genannt (z.B. deterministisches symmetrisches Kryptosystem).

Formal definieren wir [Stio6, p.1]:

Definition 2.2.1 (Kryptosystem). *Ein Kryptosystem ist ein Quintupel $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ welches folgenden Eigenschaften genügt:*

1. \mathcal{P} ist eine endliche Menge von Klartexten, der Klartextrraum.
2. \mathcal{C} ist eine endliche Menge von Chiffretexten, der Chiffretraum.

3. \mathcal{K} ist eine endliche Menge möglicher Schlüssel, der Schlüsselraum.
4. Für alle Schlüssel $k \in \mathcal{K}$ gibt es eine Verschlüsselungsfunktion $\mathcal{E} \ni e_k : \mathcal{P} \rightarrow \mathcal{C}$ und zugehörige Entschlüsselungsfunktion $\mathcal{D} \ni d_k : \mathcal{C} \rightarrow \mathcal{P}$, so dass für alle Klartexte $x \in \mathcal{P}$ folgende Identität gilt:

$$d_k(e_k(x)) = x$$

Grundsätzlich gibt es also drei Algorithmen in einem Kryptosystem: Einen für die Erzeugung des Schlüssels, einen für die Verschlüsselung und einen für die Entschlüsselung [Cry].

Definition 2.2.2 (Symmetrisches Kryptosystem). Sei $K = (\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ ein Kryptosystem. Dann nennen wir K symmetrisch, wenn sowohl die Verschlüsselungsfunktion e_k als auch die Entschlüsselungsfunktion d_k vollständig von demselben Schlüssel $k \in \mathcal{K}$ abhängen. Vollständig bedeutet, dass diese Funktionen insbesondere nicht nur von einem Teilschlüssel von k abhängen, wenn der Schlüssel k aus mehreren Parametern zusammengesetzt ist. Letzteres ist in 2.2.1 nicht vorausgesetzt.

Ein Nachteil von Symmetrischen Kryptosystemen liegt auf der Hand: Da sowohl Alice als auch Bob den gleichen geheimen Schlüssel benötigen, muss dieser über einen sicheren Kanal übertragen werden. Daher sind symmetrische Kryptosysteme auch bekannt als private-key Kryptosysteme.

Im Gegensatz dazu gibt es Kryptosysteme bei denen sich K aus einem privaten und öffentlichen Teilschlüssel zusammensetzt von Alice den öffentlichen Teilschlüssel bekannt geben kann um dritten zu ermöglichen ihr Informationen vertraulich zukommen zu lassen. Öffentlicher und privater Teilschlüssel stehen in einem Zusammenhang, der jedoch für Angreifer mit begrenzter Rechenkapazität nicht möglich ist zu erschließen.

Definition 2.2.3 (Asymmetrisches Kryptosystem). Sei $K = (\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ ein Kryptosystem. Dann nennen wir K asymmetrisch wenn sich der Schlüssel $k \in \mathcal{K}$ zusammensetzt aus $k = (k_s, k_p)$. Die Verschlüsselungsfunktion ist dann $\mathcal{E} \ni e_{k_p} : \mathcal{P} \rightarrow \mathcal{C}$, während die Entschlüsselungsfunktion $\mathcal{D} \ni d_{k_s} : \mathcal{C} \rightarrow \mathcal{P}$ ist. Während e_k von beliebigen Parteien ausgeführt werden kann, kann d_k nur vom Besitzer des privaten Teilschlüssel k_s ausgeführt werden. k_s muss geheim gehalten werden.

Diese drei Definitionen genügen noch nicht um zu beschreiben, in welcher Beziehung Klartexte x zu ihren Chiffreten c stehen wenn sie mit dem gleichen Schlüssel k in verschiedenen Ausführungen von e_k erzeugt werden. Dies ist von Bedeutung für mögliche Angriffe der Kryptoanalyse welche in 3 vorgestellt werden.

Definition 2.2.4 (Deterministisches Kryptosystem). ¹ Sei $K = (\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ ein Kryptosystem. Dann nennen wir K deterministisch wenn gilt: Für einen beliebigen festen Schlüssel $k \in \mathcal{K}$ ist e_k injektiv.

Seien nun $c_1, c_2 \in \mathcal{C}, c_1 = c_2$ zwei Chiffreten unter e_k , dann folgt daraus für ihre Klartexte, dass $x_1 = x_2$. Also führt der gleiche Klartext unter Verwendung desselben Schlüssel bei verschiedenen Ausführungen von der Verschlüsselungsfunktion d_k zu einem identischen Chiffret!

Jetzt können wir in Abgrenzung zu dieser Definition das Probabilistische Kryptosystem einführen. Ein Probabilistisches Kryptosystem erzeugt für gleiche Klartexte bei demselben Schlüssel mit jeder Ausführung der Verschlüsselungsfunktion ein anderes Chiffret.

Das Konzept eines Probabilistischen Kryptosystems wurde ursprünglich von Goldwasser und Micali eingeführt in [GM84]. Wir definieren in Anlehnung an [DKK02, p.345]:

¹Diese Definition ist eine Formalisierung von saloppen Beschreibungen in der Literatur.

Definition 2.2.5 (Probabilistisches Kryptosystem). Ein Probabilistisches Kryptosystem ist ein Sextupel $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{R})$. Wie schon in 2.2.1 ist \mathcal{P} ist der Klartextraum, \mathcal{C} der Chifferraum und \mathcal{K} der Schlüsselraum. Neu sind:

- \mathcal{R} ist eine endliche Menge von Zufallszahlen
- Für alle Schlüssel $k \in \mathcal{K}$ gibt es eine Verschlüsselungsfunktion $\mathcal{E} \ni e_k : \mathcal{P} \times \mathcal{R} \rightarrow \mathcal{C}$ und zugehörige Entschlüsselungsfunktion $\mathcal{D} \ni d_k : \mathcal{C} \times \mathcal{R} \rightarrow \mathcal{P}$, so dass für alle Klartexte $x \in \mathcal{P}$ und alle Zufallszahlen $r \in \mathcal{R}$ folgende Identität gilt: $d_k(e_k(x, r)) = x$
- Für eine Zufallszahl $r \in \mathcal{R}$ und verschiedene Klartexte $x_1, x_2 \in \mathcal{P}, x_1 \neq x_2$ gilt: $e_k(x_1, r) \neq e_k(x_2, r)$.

Ein Probabilistisches Kryptosystem benutzt also Zufall in der Verschlüsselungsfunktion, so dass der gleiche Klartext verschieden verschlüsselt wird. Mit Probabilistischen Kryptosystemen werden meistens Asymmetrische Verschlüsselungsverfahren gemeint, es ist jedoch auch möglich mit Symmetrischen Verschlüsselungsverfahren diese Eigenschaft zu erreichen, z.B. bei Verwendung von Blockchiffren im Cipher Block Chaining Mode. Die Menge der Zufallszahlen \mathcal{R} entspricht dann der Menge möglicher Initialisierungsvektoren.

HOMOMORPHES KRYPTOSYSTEM

STUFENFIXES HOMOMORPHES KRYPTOSYSTEM

2.2.1 MATHEMATISCHE GRUNDLAGEN

Wir werden später Homomorphe Kryptosystem im Detail einführen. Um ihre Anwendung zu verstehen, ist es jedoch nötig folgende Algebraische Strukturen nach [Fig11] einzuführen um ein Verständnis dafür zu schaffen wie Homomorphe Kryptosysteme verwendet werden können.

Definition 2.2.6 (Gruppe). Eine Gruppe ist ein Tupel $(G, +)$ bestehend aus der Menge G und einer Verknüpfung $+$ auf G mit folgenden Eigenschaften:

- $+$ ist assoziativ
- Es existiert bzgl. $+$ ein neutrales Element e in G .
- Jedes g in G ist invertierbar.

Ist die Verknüpfung einer Gruppe zusätzlich kommutativ, so nennt man sie abelsch.

Definition 2.2.7 (Ring). Ein Ring ist ein Tripel $(R, +, \cdot)$ bestehend aus der Menge R und zwei Verknüpfungen $+$ und \cdot auf R mit folgenden Eigenschaften:

- R ist bzgl. $+$ eine abelsche Gruppe.
- \cdot ist assoziativ
- Es gelten die Distributivgesetze: $\forall a, b, c \in R : a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ und $(a + b) \cdot c = (a \cdot c) + (b \cdot c)$

Hat R bzgl. \cdot ein neutrales Element, so nennen wir R einen „Ring mit Eins“. Ist R bzgl. \cdot kommutativ, so nennen wir R einen „kommutativen Ring“.

Definition 2.2.8 (Körper). Sei K ein kommutativer Ring mit Eins wie in 2.2.7, so heißt K Körper, wenn die neutralen Elemente bzgl. der Verknüpfungen $+$ und \cdot verschieden sind und alle Elemente bzgl. \cdot invertierbar sind.

Um den Homomorphismus Einführen zu können benötigen wir eine noch mächtigere algebraische Struktur.

Definition 2.2.9 (K-Vektorraum). Sei $(K, +, \cdot)$ ein Körper und V eine Menge. Zusätzlich existieren zwei Verknüpfungen $\oplus : V \times V \rightarrow V$ und $\otimes : K \times V \rightarrow V$, so dass gilt:²

- \oplus ist assoziativ, kommutativ, hat ein neutrales Element bzgl. V und für alle Elemente $v \in V$ Inverse in V
- \otimes ist assoziativ, hat ein neutrales Element bzgl. V
- Elemente aus K und V sind distributiv

Dann nennen wir V einen K -Vektorraum.

Definition 2.2.10 (Homomorphismus). Seien V, W zwei K -Vektorräume und $f : V \rightarrow W$ eine Abbildung. Dann nennen wir f einen Homomorphismus (oder linear) von V nach W wenn gilt:

- $\forall x, y \in V : f(x) + f(y) = f(x + y)$
- $\forall x \in V, \lambda \in K : f(\lambda \cdot x) = \lambda \cdot f(x)$

Theorem 2.2.1 (Jede boolsche Funktion ist mit NANDs konstruierbar). ...

Homomorphes Kryptosystem: Eine Verknüpfung im Chifferraum führt zur einer Verknüpfung im Klartextraum. Erstere ist in der Regel die Multiplikation und letztere Addition, Multiplikation oder XOR.

²Evtl. ausführlicher formulieren

3 SICHERHEITSKRITERIEN

3.1 MALLEABILITY

Malleability beschreibt die Möglichkeit, dass ein Angreifer einen Chiffretext c von Klartext k gezielt verformen kann um einen daraus abgeleiteten Chiffretext $f(c) = c'$ zu erzeugen welcher in einer ihm bekannten Beziehung f zu c steht. Existiert nun zwischen den Klartexten k und k' eine Beziehung die der Angreifer umkehren kann, kann er zu k' den ursprünglichen Klartext k bestimmen. [Smao3, p. 292]

Eigenschaften: Ein malleable Kryptosystem ist angreifbar mit chosen ciphertext Angriffen. (CCA2)

Kommentar: Das zwischen den Chiffretexten und den Klartexten eine ähnliche Beziehung steht die der Angreifer kennt ist eine Eigenschaft die genau Isomorphismen ermöglichen! Daher sind homomorphe Kryptosysteme per Design anfällig für malleability.

3.2 PRIVACY-PRESERVING

3.3 SICHERHEITSKLASSEN

3.3.1 UNUNTERSCHIEDBARKEIT VON GEHEIMTEXTEN (CIPHERTEXT INDISTINGUISHABILITY)

3.3.2 SEMANTISCHE SICHERHEIT

Ein deterministisches Kryptosystem wie in ?? kann nie semantisch sicher sein!

4 KLASSIFIKATION HOMOMORPHER KRYPTOSYSTEME

4.1 AUFTEILUNGEN

4.2 AUTOCRYPT

[TSCS₁₃] Problemstellung: Server sind ständig durch Angriffe bedroht die bis hin zu ihrer kompletten Übernahme geraten können. Um Datendiebstahl und Vertraulichkeitsverletzungen vorzubeugen ist es ratsam nur mit verschlüsselten Datenbeständen zu arbeiten. Ein IT-System oder Programm so anzupassen, dass es auf verschlüsselten Daten korrekt arbeitet wollen die Wissenschaftler automatisieren indem sie die Arbeit der Programmtransformation mit einem Compiler abwickeln: Autocrypt.

Der Server läuft als virtuelle Maschine und Inhalte werden außerhalb der unvertrauten VM auf einem keyserver verschlüsselt. Autocrypt bestimmt automatisch benötigte Verschlüsselungsdattentypen für die Variablen und konvertiert zwischen diesen im Programmablauf her durch einfügen von hypercalls. Die Verschlüsselungsdattentypen werden gewählt nach der Verknüpfung die sie zu Verfügung stellen. Wenn also im Ursprungscode Additionen von zwei Variablen durchgeführt werden, dann werden zunächst Paillerverschlüsselungsdattentypen erzeugt. Wird das Ergebnis allerdings später multipliziert, dann muss der Dattentyp konvertiert werden zu einem Elgamalverschlüsselungsdattentyp.

Bei der Entwicklung von Autocrypt sollen alle Rechenoperationen privacy-preserving sein. Als Transformationstool ist eine Integrität der Daten auf denen gerechnet wird daher nicht berücksichtigt worden.

Kategorisierungskriterien: Pailler wurde wegen seiner Homomorphie und Additionsverknüpfung verwendet (\rightarrow additiv-homomorph). Analoges Argument für Elgamal. Zwischen diesen beiden Verfahren wird hin und her konvertiert, da dies schneller ist als *ein* vollhomomorphes Verfahren (\rightarrow Klasse schneller Verfahren). Weiter ist Pailler flexibel einsetzbar für die Addition von Zahlen byteweise oder bitweise. Letzteres ermöglicht die Konstruktion eines homomorphen XOR Operators. (\rightarrow homomorph XOR)

Malleability: Die Autotoren haben als Zielsetzung die unerlaubte Kenntnisnahme von Daten auf dem Server zu unterbinden. Eine Überprüfung der Integrität von Rechenoperationen der von Autocrypt konvertierten Programmbestandteile ist daher kein Fokus der Arbeit [p. 4].

4.3 MACHINE LEARNING CLASSIFICATION OVER ENCRYPTED DATA

[BPTG15] Problemstellung: Es soll ein privacy-preserving Maschinenlernenverfahren erstellt werden, bei dem sowohl die zu klassifizierenden Daten als auch die Klassifizierergesamtheiten vertraulich bleiben. Es wird eine Bibliothek konstruiert, aus der Modular beliebige privacy-preserving Klassifizierer erstellt werden können.

In einem ersten Ansatz wurde überlegt privacy-preserving mit Secure Multiparty Computation umzusetzen, welches sich jedoch als zu langsam herausgestellt hat. Aus dem gleichen Grund wird auch auf den Einsatz von vollhomomorpher Verschlüsselung verzichtet. Es ist schneller mit für Klassifizierungsverfahren spezialisierten Protokollen zu arbeiten.

Es wird wie in [TSCS13] XOR mit Pailler simuliert. Zusätzlich wird ein privates Skalarprodukt auf Basis von Pailler berechnet. Gegeben seien die Vektoren $x = (x_1, \dots, x_n)$ und $y = (y_1, \dots, y_n)$ wobei alle Einträge Klartexte sind. Das mit pub Paillierschlüsselte Skalarprodukt ist dann:

$$Enc_{pub}(\langle x, y \rangle) = \prod_i Enc_{pub}(y_i)^{x_i} \bmod N^2 \quad (4.1)$$

noch auszuführen

Eine weitere Tatsache die im Paillierkryptosystem ausgenutzt wird ist der Klartextraum ungefähr 2^{1024} bit ist. Anstelle von lediglich Integern können Floatzahlen mit Pailler verschlüsselt werden, wenn man die IEEE 754 floating point Darstellung verwendet welche große Exponenten benötigt.

In dieser Arbeit wurde auch eine leveled vollhomomorphe Verschlüsselung (HELib) verwendet, jedoch der Umfang und die Gründe dafür bleiben ohne nähere Erläuterung [p. 4].

Kategorisierungskriterien: Es wurden die Kryptosysteme von Paillier und Goldwasser-Micali verwendet. Beide aufgrund ihrer schnelleren Performance und der mathematischen Verknüpfung sie anbieten. (\rightarrow additiv-homomorph) (\rightarrow xor-homomorph). Analog zur unverschlüsselten Konstruktion von Gleitkommazahlen aus ganzen Zahlen kann mit Pailler ein Operator für die homomorphe Addition von Gleitkommazahlen konstruiert werden. (\rightarrow floatingpoint-additiv-homomorph)

Malleability: Es werden die homomorphen Kryptosysteme lediglich zum Rechnen im Chifferraum verwendet. Ein Angriff der Malleability ausnutzt wird nicht betrachtet. Dies ist nachvollziehbar, da hier ein deterministischer Algorithmus abgearbeitet wird.

4.4 PRIVACY PRESERVING MATRIX FACTORIZATION

[NIW⁺13] Bei der Generierung von userspezifischen Empfehlungen anhand vorheriger Wahlen eines Users ist Matrizenfaktorisierung ein weit verbreitetes Verfahren. Um dieses privacy-preserving zu machen soll ein System designt werden, welches Empfehlung geben kann ohne die Userbewertungen zu lernen.

Bei dem Design wird aus Performancegründen hash-ElGamal verwendet um verschlüsselte Bewertungen zu maskieren für die Einheit, welche im Besitz des privaten Schlüssels ist. In dem Design bekommt das Recommendersystem (RecSys) vom User ein mit dem öffentlichen Schlüssel von Cryptoserviceprovider (CSP) verschlüsseltes Rating c . Damit der CSP dies Rating nicht aufdecken kann, addiert RecSys einen zufälligen Wert μ auf das Rating. CSP erhält $c' = c + \mu$.

Kategorisierungskriterien: Es wurde hash-ElGamal verwendet wegen seiner schnelleren Performance gegenüber Paillier und seiner Additivität (\rightarrow additiv-homomorph)

Malleability: Bei dem Design wird von einem honest-but-curious Angreifer ausgegangen. Also könnte RecSys aus Neugierde $\mu = 0$ addieren und so CSP ermöglichen alle Userratings zu lernen. Im HBC-Modell dürfen RecSys und CSP jedoch nicht vom Protokoll abweichen und könnten daher nicht kooperativ diese Information abschöpfen, denn CSP weiß nicht, dass RecSys eine Nulladdition durchführt welches die Maskierung aufhebt.

4.5 EFFICIENT AND SECURE COMPARISON FOR ON-LINE AUCTIONS

Ivan Damgard et al. stellen in [DGK07] ein neues additives Kryptosystem DGK vor um schnelle vergleiche einer öffentlich bekannten Zahl x und einer bitweise (vgl. 4.8) verschlüsselten Zahl m durchzuführen die auf einem Server und einem Hilfsserver verteilt ist. Bei der online Versteigerung steht x für das momentane Höchstgebot, während m für das private mögliche Höchstgebot steht. Sie haben dieses Kryptosystem für ihren Anwendungsfall designt, da sie einen möglichst kleinen Klartextraum haben wollen. Die Verwendung eines kleinen Klartextraums hat zum Vorteil, dass mit kleineren Exponenten gerechnet wodurch ihr Verfahren an Effizienz im Vergleich gegenüber anderen Ansätzen gewonnen hat.

In ihrem Vergleich $m \leq x$ ist letztere Zahl öffentlich, jedoch ist das Verfahren erweiterbar, so das beide Eingabeparameter privat sind. Ein auf ihrem Verfahren basierender Vergleich von zwei privater Zahlen wird in 4.7 vorgestellt.

Kategorisierungskriterien: additiv-homomorph, kleiner Klartextraum

Malleability: Das vorgestellten Kryptosystems wurde nicht auf Malleability untersucht obwohl bösartige Anfragen möglich wären, da von einem honest-but-curious Angreifermodell ausgegangen wird. Ein Teilnehmer kann somit falsche Höchstgebote abgeben, jedoch gewinnt man dadurch keinen Vorteil. Entweder wird man früher aus dem Gebotsverfahren geschmissen oder er kann nach dem Gebotsverfahren feststellen ob der Betreiber die Vergleiche inkorrekt durchgeführt hat.

4.6 FINGERPRINTING PROTOCOL FOR IMAGES BASED ON ADDITIVE HOMOMORPHIC PROPERTY

[KT05] Das vorgestellte Protocol wurde unter dem Hintergrund eingeführt, da bisher bekannte Lösungen entweder eine zu langsame Verschlüsselungsrate hatten oder das Einsetzen des Wasserzeichens zu aufwendig ist. Es wird ein homomorphes public-key Kryptosystem eingesetzt um asymmetrische Fingerprints zu erzeugen. Dies ist notwendig, da bei symmetrischen Fingerprinting der Verkäufer ein benutzerspezifisches Wasserzeichen erzeugen kann und somit auch einen rechtmäßigen Käufer eine Raubkopie unterstellen könnte.

Obwohl das Protocol mit Pailler umgesetzt werden könnte, hat man sich für das Kryptosystem von Okamoto-Uchiyama [OU98] entschieden, da weniger Rechenoperationen durchgeführt werden müssen [p.2132].

Die Asymmetrie und Homomorphie vom Okamoto-Uchiyama Kryptosystem wird wie folgt beim Fingerprinting ausgenutzt:

1. Der Käufer erzeugt einen Fingerabdruck, verschlüsselt ihn mit seinem öffentlichen Schlüssel, und sendet ihn an den Verkäufer. Mit einem Zero-Knowledge-Proof wird dem Verkäufer nachgewiesen, dass das Chiffre tatsächlich einen nutzerspezifischen Fingerabdruck enthält.
2. Nun verschlüsselt der Verkäufer sein digitales Bild unter dem gleichen öffentlichen Schlüssel wie der Käufer und bettet den Fingerabdruck durch homomorphe Verknüpfung ein.
3. Der Käufer entschlüsselt das Bild mit dem Wasserzeichen ohne jedoch in der Lage zu sein das Wasserzeichen zu entfernen, da ihm nicht bekannt ist in welchen Positionen oder Frequenzbereichen (bei Anwendung des Wasserzeichens im Frequenzraum unter diskreter Kosinustransformation) das Wasserzeichen das ursprüngliche Bild verändert hat.

Weiter gewährleistet die semantische Sicherheit von Okamoto-Uchiyama anonymes Kaufen [p.2134], d.h. der Verkäufer ist nicht in der Lage die Identität des Käufers aufzudecken.

Kategorisierungskriterien: additiv-homomorph, semantisch-sicher

Malleability: Die Autoren sind sich bewusst der Malleability des verwendeten Kryptosystems, ohne jedoch mögliche Angriffe zu untersuchen. Der Hauptfokus des Sicherheitsbegriffs liegt in der Unmöglichkeit des Käufers, das Wasserzeichen zu entfernen und für den Verkäufer, nicht in der Lage zu sein die Identität des Käufers zu aufzudecken.

4.7 PRIVACY PRESERVING FACE RECOGNITION

Das Team um Zakeriya Erkin et al. [EFG⁺09] stellt ein originelles privacy-preserving Gesichtserkennungssystem vor bei dem sowohl die Eingabebilder als auch das Ergebnis ihrer Analyse vom Server verdeckt bleiben. Der Analyse zugrunde liegt ein Eigenfacesalgorithmus, welcher auf verschlüsselten Bildern arbeitet. Eingesetzt werden die Kryptosysteme Paillier und DKG [DGK07] welches auch in 4.5 zum Einsatz kommt. Sie wählen die Basis g des öffentlichen Schlüssel als $g = n + 1$, welches die Verschlüsselung beschleunigt [p.237][DJ01]. DKG wird lediglich auch Effizienzgründen anstelle von Paillier eingesetzt. Der Klartextrraum im implementierten DKG ist kleiner. Da die Exponenten kleiner sind, ist die Verschlüsselung effizienter.

Da die Verfahren nur mit Integers arbeiten, werden Featurevektoren des Gesichtserkennungssystem diskretisiert in dem auf den nächsten Integer gerundet wird.

Rechenoperationen im Chifferraum im privaten Gesichtserkennungssystem: Wann immer die eckigen Klammern auftauchen, ist ein Element verschlüsselt unter dem öffentlichen Schlüssel von Alice, welche ein Gesicht analysieren möchte. Sie übergibt dieses Gesicht verschlüsselt Bob, der dank homomorpher Kryptographie in der Lage ist die Analyse durchzuführen ohne das Gesicht direkt zu sehen.

- *Projektion* des verschlüsselten Eingabebildes Γ auf die Basis von Eigenfacevektoren u_1, \dots, u_K . Hier zu wird mit der gleichen Technik wie in 4.3 ein Skalarprodukt durch Potenzieren berechnet. Das Ergebnis ist ein verschlüsselter Featurevektor des Eingabebildes $[[\bar{\Omega}]]$.

- *Abstand* D von Featurevektoren $\{\Omega_1, \dots, \Omega_M\}$ der Datenbank des Servers zum Featurevektor des Eingabebildes $\llbracket \bar{\Omega} \rrbracket$. Da man nur an der relativen Ordnung der Abstände interessiert ist, genügt der Vergleich der quadrierten Abstände:

$$\begin{aligned} D(\Omega, \bar{\Omega}) &= \|\Omega - \bar{\Omega}\|^2 = (\omega_1 - \bar{\omega}_1)^2 + \dots + (\omega_K - \bar{\omega}_K)^2 \\ &= \underbrace{\sum_{i=1}^K \omega_i^2}_{S_1} + \underbrace{\sum_{i=1}^K (-2\omega_i \bar{\omega}_i)}_{S_2} + \underbrace{\sum_{i=1}^K \bar{\omega}_i^2}_{S_3} \end{aligned}$$

Da Bob den Server betreibt, kennt er die Komponenten ω_i der Featurevektoren in der Datenbank und kann S_1 direkt berechnen. Da er die Komponenten $\bar{\omega}_i$ des Featurevektor vom Eingabebild nur verschlüsselt vorliegen hat, muss er S_2 analog wie beim Skalarprodukt in 4.3 durch potenzieren berechnen. Letzendlich kann Bob S_3 nur in Kooperation mit Alice berechnen, da bei beide Faktoren des Produkts ihm unbekannt sind. Dazu maskiert er die Komponenten des Featurevektors mit gleichverteilten Zufallswerten r_i :

$$\llbracket x_i \rrbracket = \llbracket \bar{\omega}_i + r_i \rrbracket$$

Diese maskierten Komponenten sendet er an Alice, welche diese mit ihrem privaten Schlüssel entschlüsselt und quadriert, das Ergebnis $S'_3 = \sum_{j=1}^K x_i^2$ wieder verschlüsselt und dann an Bob zurücksendet. Bob erhält dann S_3 durch:

$$\llbracket S_3 \rrbracket = \llbracket S'_3 \rrbracket \cdot \prod_{j=1}^K (\llbracket \bar{\omega}_i \rrbracket^{(-2r_i)} \cdot \llbracket -r_i^2 \rrbracket)$$

Was korrekt ist, da die i -te Komponente sich ergibt aus:

$$\llbracket x_i^2 \rrbracket \cdot \llbracket \bar{\omega}_i \rrbracket^{(-2r_i)} \cdot \llbracket -r_i^2 \rrbracket = \llbracket (\bar{\omega}_i + r_i)^2 - 2r_i \bar{\omega}_i - r_i^2 \rrbracket = \llbracket \bar{\omega}_i^2 \rrbracket$$

- *Vergleichen* zweier privater Zahlen im DGK Kryptosystem. Gegeben seien zwei bitweise verschlüsselte Zahlen $\llbracket d \rrbracket$ und $\llbracket r \rrbracket$. Alice generiert das Schlüsselpaar unter DGK und sendet den öffentlichen Schlüssel, sowie die ihre verschlüsselten Bits $\llbracket d_{l-1} \rrbracket, \dots, \llbracket d_0 \rrbracket$ zu Bob. Bob berechnet dann:

$$\llbracket c_i \rrbracket = \llbracket d_i - r_i - 1 + 3 \sum_{j=i+1}^{l-1} d_j \oplus r_j \rrbracket = \llbracket d_i \rrbracket \cdot \llbracket -r_i \rrbracket \cdot \llbracket 1 \rrbracket \cdot \left(\sum_{j=i+1}^{l-1} \llbracket d_j \oplus r_j \rrbracket \right)^3$$

Diese Formel hat die Eigenschaft: *alle* c_i sind ungleich Null genau dann, wenn d die größere der beiden Zahlen ist. Umgekehrt ist mindestens ein c_i null, wenn $d_i \leq r_i$. Nämlich das höchstwertigste Bit wo sich die beiden Zahlen unterscheiden. Die 1 wird addiert, damit ein c_i gerade Null annimmt, wenn d kleinergleich ist. Der Faktor 3 garantiert, dass c_i ungleich Null ist sobald sich die Zahlen in einem Bit unterscheiden.

Man sieht, dass alleine durch die Möglichkeit Linearkombinationen von Chiffren zu bilden ein additiv-homomorphes Kryptosystems komplexer Operatoren ausführen kann!

Kategorisierungskriterien: additiv-homomorph, Simulierte Operatoren: Projektion und Abstand (wobei teilweise Parameter unverschlüsselt vorlagen) und Vergleich (beide Zahlen verschlüsselt)

Malleability: Anstelle von malleability wird in dieser Veröffentlichung lediglich von „re-randomization“ gesprochen, was sich bezieht auf eine Neuverschlüsselung einer Zahl z.B. durch Addition von Null. Alice vertraut Bob der korrekten Durchführung des Eigenfacesalgorithmus und Bob akzeptiert, dass Alice grobe Eigenschaften über seiner Gesichtserkennungssystem lernen kann. Es wird von einem honest-but-curious Modell ausgegangen. Eine Malleability wurde nicht untersucht, jedoch ist klar, dass Bob nie Einsicht in private biometrische Daten von Alice erhält. Denn Bob operiert auf Chiffren die unter Alices öffentlichen Schlüssel verschlüsselt sind. Bob kann eine Malleability nicht ausnutzen, weil er keine entschlüsselten Informationen bei Alice anfragen kann.

Alice ist zwar in der Lage beliebig veränderte Chiffre Bob zurückzugeben, z.B. die x_i bei der Abstandsberechnung. Jedoch führt dies lediglich zu einem verfälschten Ergebnis des Eigenfacesalgorithmus welchen Bob ausführt. Damit kann Alice Malleability nicht für eigene Vorteile ausnutzen.

4.8 PRIVATE PREDICTIVE ANALYSIS ON ENCRYPTED MEDICAL DATA

In [BLN14] stellen Joppe W. Bos et. al. ein Verfahren vor zur privaten Analyse von in die Cloud ausgelagerten medizinischen Daten. Sie implementieren eine logische Regression auf homomorph verschlüsselten Daten um die Wahrscheinlichkeit einer Herzkreislauferkrankung zu prädictieren. Zum Einsatz kommt ein Stufenfixes (engl. leveled) homomorphes Kryptosystem (2.2). In dem Prädiktionsmodell mit logistischer Regression ist die Prädiktionsformel

$$P(x) = \frac{e^x}{e^x + 1}$$

wobei x eine Linearkombination von gewichteten Regressionskoeffizienten repräsentiert. Somit ist x offenbar homomorph berechenbar. Die Prädiktionsformel wird homomorph berechenbar indem man sie durch eine Taylorreihe annähert.

In Anlehnung an [BLLN13] [NLV11] stammen daher die zu verknüpfenden Variablen aus einem Polynomring $R = \mathbb{Z}/(X^n + 1)$. Der Klartextrraum und Chifferraum besteht aus Polynomen $\sum_{i=0}^{n-1} a_i X^i, a_i \in \mathbb{Z}$. Operationen im Chifferraum entsprechen der Addition und Multiplikation von Polynomen mod $X^n + 1$. Mit diesem Stufenfixen homomorphen Kryptosystem ist es dann möglich alle Operationen zur Berechnung der Prädiktionsformel durchzuführen. Trotzdem ist noch folgendes Problem vorhanden: Die Eingabe welche eine Linearkombination von prädictierenden Regressionskoeffizienten ist, sowie die letztere selbst sind in der Anwendung typischerweise rationale Zahlen.

Um das Kryptosystem verwenden zu können, müssen alle Eingaben als Polynome dargestellt werden, z.B. durch die 2-adische Darstellung: $\sum_{i=0}^{n-1} a_i 2^i, a_i \in \{0, 1\}$. Wir können mit rationalen Zahlen rechnen wenn wir eine fixe Fließkommagenauigkeit vorab festlegen (d.h. Grad des Polynoms). Dann werden rationale Zahlen zu Integern rück-/transformiert durch Division/Multiplikation mit dem zugehörigen Faktor k^{10} .

Kategorisierungskriterien: leveled-homomorphic-encryption (multiplikativ, additiv), simuliert p-adische Darstellung womit rationale Zahlen dargestellt werden können

Malleability: Eine malleability der Kryptosystems wurde nicht untersucht. Der Cloudprovider übernimmt die Rolle outsourceter Rechenkapazität die Eingaben deterministisch verarbeitet. Im

Kontext des Anwendungsfalls sind sowohl Kunde als auch Cloudprovider an komplett ehrlicher, aber vertraulicher Verarbeitung interessiert.

4.9 SICHERE BERECHNUNG VON FUNKTIONEN MIT SMC ODER HE

In den Studien [DGK07, p.420] und [SSW09, p.2] wurden zwei Techniken identifiziert um Funktionen sicher zu berechnen: Secure Multi-Party Computation und Homomorphe Verschlüsselung. Dabei kommen beide Studien unabhängig voneinander zu den gleichen Schlüssen über die Vorzüge der jeweiligen Verfahren:

- SMC Vorteil: viel Kommunikation
- SMC Nachteil: geringe Runden- und Berechnungskomplexität
- HE Vorteil: geringe Kommunikation
- HE Nachteil: hohe Runden- und Berechnungskomplexität

4.10 BITWEISE VERSCHLÜSSELUNG VS INTEGERVERSCHLÜSSELUNG

Mit bitweiser Verschlüsselung lassen sich Grundoperatoren XOR und AND erzeugen. Und damit nach Abschnitt 2.2.1 jede boolesche Funktion bestimmen. Nachteil ist, dass ein Chiffretext weniger Information erhält. Bitweise Verschlüsselung erzeugt einen separaten Chiffretext für jedes Bit und erzeugt damit einen großen Overhead, so dass nach Möglichkeit auf Integerverschlüsselung ausgewichen wird. [?]

4.11 ZUSAMMENFASSUNG...

Kryptosystem	hom. Operator	sim. Operator	modi
Pailler	+	XOR, $\langle \cdot, \cdot \rangle$	bitwise, bytewise, float
ElGamal	\cdot	-	-
hash-ElGamal	XOR	-	-
Goldwasser-Micali	XOR		
BGV (HELib)	vollhom.	any	

5 ZUM ABSCHLUSS

Jede wissenschaftliche Arbeit ist selbstverständlich den subjektiven Kriterien des Autors unterworfen, die seinen persönlichen Stil einen Sachverhalt zu präsentieren und zu charakterisieren. Dieser Stil muss nicht unbedingt mit dem des Lesers übereinstimmen. Daher sind sicher auch vereinzelte Aspekte, die in diesem Leitfaden besprochen werden, durch unsere subjektive Vorstellung motiviert, wie eine gute Ausarbeitung zu erstellen ist.

Die meisten der in diesem Leitfaden angesprochenen Punkte spiegeln jedoch Erfahrungswerte wider, die sich bei der Präsentation unserer eigenen wissenschaftlichen Texte bewährt haben. Daher lohnt es sich, diesen Leitfaden bei der Erstellung Ihrer wissenschaftlichen Arbeit im Hinterkopf zu behalten, um die größten technischen Fehler im vorhinein zu vermeiden.

6 LITERATURVERZEICHNIS

- [BLLN13] Bos, Joppe W. ; LAUTER, Kristin E. ; LOFTUS, Jake ; NAEHRIG, Michael: Improved Security for a Ring-Based Fully Homomorphic Encryption Scheme. In: *IMA Int. Conf.* Springer, 2013, S. 45–64
- [BLN14] Bos, Joppe W. ; LAUTER, Kristin ; NAEHRIG, Michael: Private predictive analysis on encrypted medical data. In: *Journal of biomedical informatics* 50 (2014), S. 234–243
- [BPTG15] BOST, Raphael ; POPA, Raluca A. ; TU, Stephen ; GOLDWASSER, Shafi: Machine Learning Classification over Encrypted Data. In: *NDSS*, 2015
- [Cry] *Cryptosystem* - Wikipedia. <https://en.wikipedia.org/wiki/Cryptosystem>, . – (Accessed on 03/28/2017)
- [DGK07] DAMGÅRD, Ivan ; GEISLER, Martin ; KRØIGAARD, Mikkel: Efficient and secure comparison for on-line auctions. In: *Australasian Conference on Information Security and Privacy* Springer, 2007, S. 416–430
- [DJ01] DAMGÅRD, Ivan ; JURIK, Mads: A generalisation, a simplification and some applications of paillier’s probabilistic public-key system. In: *International Workshop on Public Key Cryptography* Springer, 2001, S. 119–136
- [DKK02] DELFS, Hans ; KNEBL, Helmut ; KNEBL, Helmut: *Introduction to cryptography*. Bd. 2. Springer, 2002
- [EFG⁺09] ERKIN, Zekeriya ; FRANZ, Martin ; GUJARDO, Jorge ; KATZENBEISSER, Stefan ; LAGENDIJK, Inald ; TOFT, Tomas: Privacy-preserving face recognition. In: *International Symposium on Privacy Enhancing Technologies Symposium* Springer, 2009, S. 235–253
- [Fis11] FISCHER, Gerd: *Lernbuch Lineare Algebra und Analytische Geometrie*. In: *Vieweg+ Teubner, Wiesbaden* (2011)
- [GM84] GOLDWASSER, Shafi ; MICALI, Silvio: Probabilistic encryption. In: *Journal of computer and system sciences* 28 (1984), Nr. 2, S. 270–299
- [KT05] KURIBAYASHI, Minoru ; TANAKA, Hatsukazu: Fingerprinting protocol for images based on additive homomorphic property. In: *IEEE Transactions on Image Processing* 14 (2005), Nr. 12, S. 2129–2139
- [NIW⁺13] NIKOLAENKO, Valeria ; IOANNIDIS, Stratis ; WEINSBERG, Udi ; JOYE, Marc ; TAFT, Nina ; BONEH, Dan: Privacy-preserving matrix factorization. In: *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security* ACM, 2013, S. 801–812

- [NLV11] NAEHRIG, Michael ; LAUTER, Kristin ; VAIKUNTANATHAN, Vinod: Can homomorphic encryption be practical? In: *Proceedings of the 3rd ACM workshop on Cloud computing security workshop* ACM, 2011, S. 113–124
- [OU98] OKAMOTO, Tatsuaki ; UCHIYAMA, Shigenori: A new public-key cryptosystem as secure as factoring. In: *Advances in Cryptology—EUROCRYPT’98* (1998), S. 308–318
- [Sma03] SMART, Nigel P.: *Cryptography: An Introduction*. Bd. 5. McGraw-Hill New York, 2003
- [SSWo9] SADEGHI, Ahmad-Reza ; SCHNEIDER, Thomas ; WEHRENBURG, Immo: Efficient privacy-preserving face recognition. In: *International Conference on Information Security and Cryptology* Springer, 2009, S. 229–244
- [Stio6] STINSON, Douglas R.: *Cryptography: theory and practice*. CRC press, 2006
- [TSCS13] TOPLE, Shruti ; SHINDE, Shweta ; CHEN, Zhaofeng ; SAXENA, Prateek: AUTOCRYPT: enabling homomorphic computation on servers to protect sensitive web content. In: *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security* ACM, 2013, S. 1297–1310