Untersuchung der Anwendung homomorpher Kryptosysteme unter Berücksichtigung der unautorisierten Verformbarkeit von Chiffretexten

Matthias Ulbrich
matt@uni-bonn.de
2743974

21. Juni 2017

Übersicht

Einleitung

- Homomorphe Kryptosysteme sind eine Erweiterung klassischer Kryptosysteme
- Sie unterscheiden sich durch: Sicherheitseigenschaften gegenüber Angreifermodellen, möglichen homomorphe Rechenoperationen, Chiffretextexpansion, Performance (Anzahl der Rechenschritte)

Ziel: Wir möchten eine Empfehlung für folgende Frage abgeben können.
Frage: Bob möchte ein Wasserzeichen in einem verschlüsselten Bild einbetten. Welches homomorphe Kryptosystem sollte er verwenden?

Kryptosystem

Ein Kryptosystem besteht aus fünf Mengen mit drei auf ihnen anwendbaren Algorithmen:

Kryptosystem K

- K ist Quintupel der Mengen: $K = (\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$
- Für alle Schlüssel $k \in \mathcal{K}$ gibt es Funktionen $\mathcal{E} \ni e_k : \mathcal{P} \to \mathcal{C}$ und $\mathcal{D} \ni d_k : \mathcal{C} \to \mathcal{D}$, so dass für alle Klartexte $x \in \mathcal{P}$ folgende Identität gilt:

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

Homomorphes Kryptosystem

Erweiternde Eigenschaften:

- **asymmetrisch**: Durchführung von Rechenoperationen zusätzlich zu einer einfachen Verschlüsselung durch Kenntnis des öffentlichen Schlüssels.
- **probabilistisch**: Die Anwendung der Verschlüsselungsfunktion wird durch Zufallszahlen nicht-deterministisch:

$$e_{k_p}(x, r) \neq e_{k_p}(x, r').$$

- (semihomomorph) Verschlüsselungsfunktion ist ein **Gruppenhomomorphismus** der Gruppen $(\mathcal{P}, \oplus), (\mathcal{C}, \odot)$. Für Chiffretexte $e_{k_p}(x_1) = c_1$, $e_{k_p}(x_2) = c_2$ gilt:

$$d_{k_s}(c_1 \odot c_2) = x_1 \oplus x_2.$$

Beispiel: Okamoto-Uchiyama Kryptosystem

öffentlicher Schlüssel (n, g, h), privater Schlüssel (p, q)

Verschlüsselung:

$$e(m, r) = g^m h^r \text{ mod } n \text{ mit gleichverteiltem } r \in (\mathbb{Z}/n\mathbb{Z})$$

Homomorphe Addition:

$$\begin{aligned} e(m_1, r_1)e(m_2, r_2) \text{ mod } n &= g^{m_1} h^{r_1} g^{m_2} h^{r_2} \text{ mod } n \\ &= g^{m_1+m_2} h^{r_3} \text{ mod } n \\ &= e(m_1 + m_2, r_3) \text{ mod } n \end{aligned}$$

Homomorphe Multiplikation:

$$e(m, r)^k = (g^m h^r)^k \text{ mod } n = g^{mk} h^{r'} \text{ mod } n = e(mk, r')\text{mod } n$$

Sicherheitsziele

- **Ununterscheidbarkeit von Chiffretexten (IND)**: Geben ein Chiffretext c_b von *einem* der Klartexte x_1, x_2. Der Angreifer soll $b =?$ nicht besser als ein zufällig ratender Angreifer ermitteln können.

- **Keine unautorisierte Verformbarkeit (NM)**: Der Angreifer will einen Chiffretext c von Klartext x zu einem Chiffretext c' gezielt verformen. Dabei soll c' in einer gewünschten Beziehung zu x stehen (formal: $\mathcal{R}(x, x')$). Der Angreifer soll c' nicht besser ermitteln können als wäre es zufällig erzeugt.

Hier verfügt der Angreifer über den öffentlichen Schlüssel für seinen Angriff. $\rightarrow$ Grundsätzlich möglich: CPA1-Angriff.

Sicherheit homomorpher Kryptosyteme

Relevante kryptografische Angreifer

1. Asymmetrie
   $\rightarrow$ mindestens nötig: Sicherheit vor IND-CPA1-Angreifern.
2. Homomorphieeigenschaft impliziert unautorisierte Verformbarkeit
   $\rightarrow$ höchstens möglich: Sicherheit vor IND-CCA1-Angreifern.

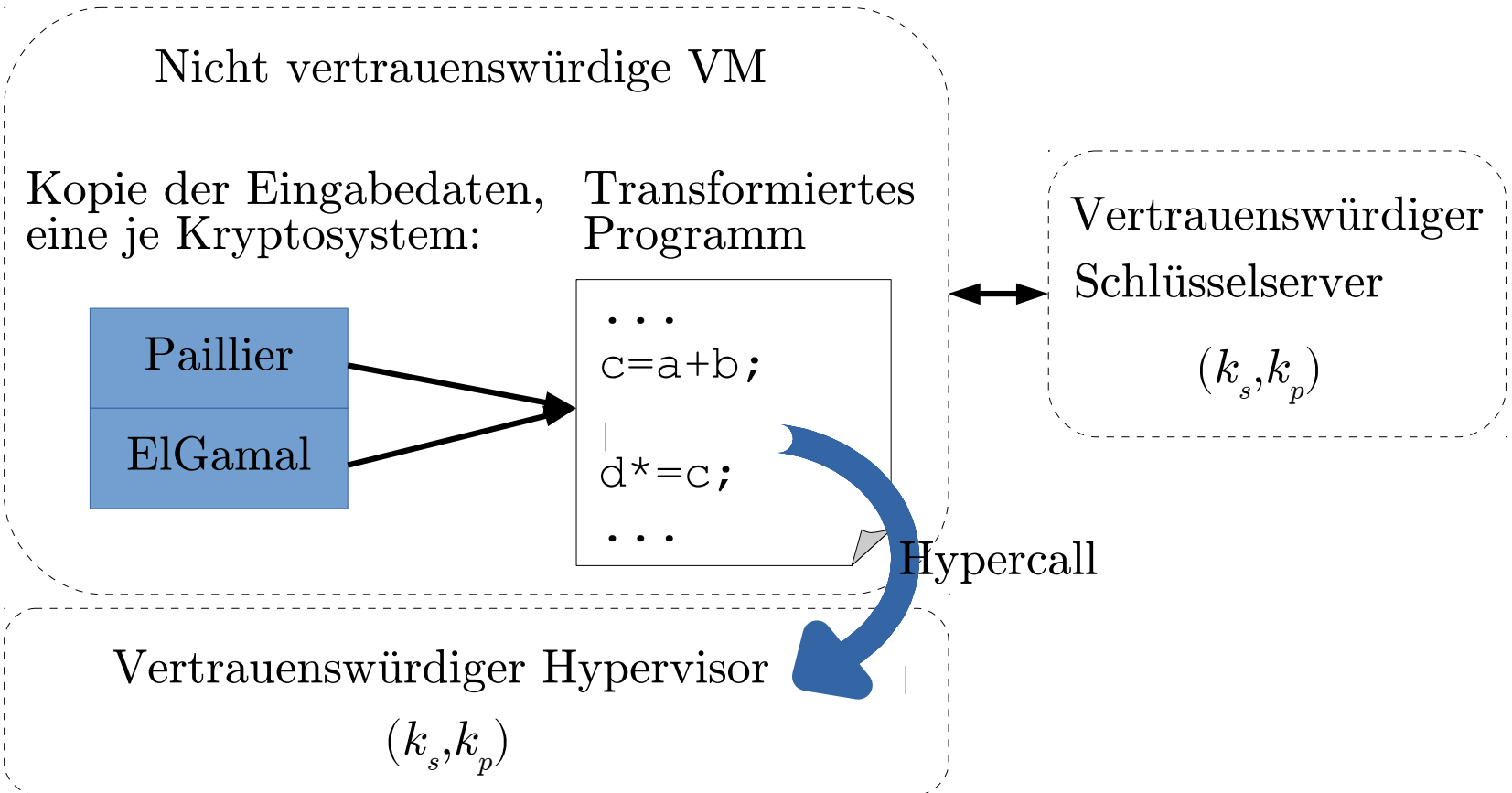
Wie wird eine Integrität von Rechenoperationen beim Einsatz homomorpher Kryptosysteme gewährleistet?

Untersuchte Kriterien für die Klassifizierung

Um Bob eine Empfehlung geben zu können sollen Auswahlkriterien eines Kryptosystems identifiziert werden:

- besondere Eigenschaften des Kryptosystems
- realisierte Berechnungen im Chiffreraum
- Umgang mit unautorisierter Verformbarkeit der Chiffretexte: Wie wird eine Integrität der Chiffretexte gewährleistet?

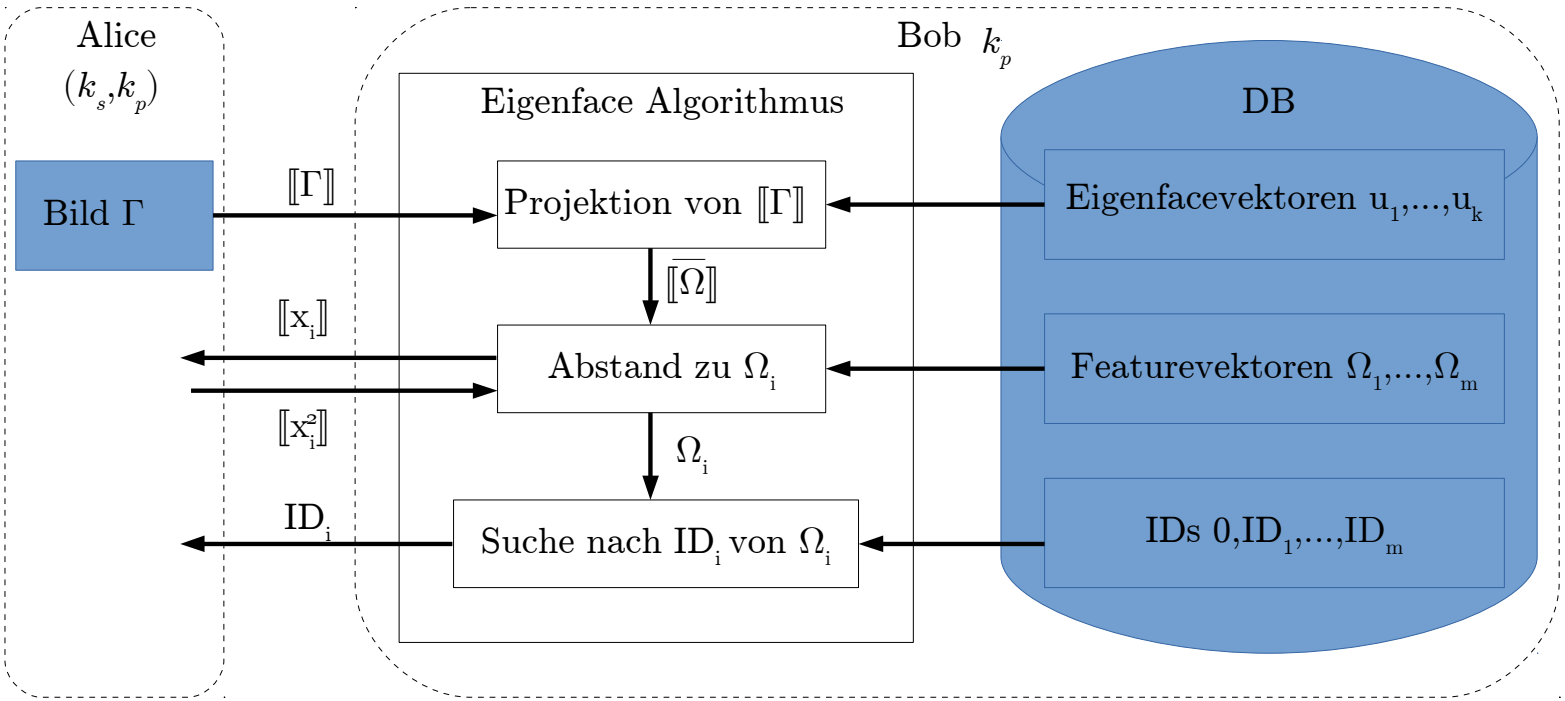
Untersuchung von Autocrypt [TSCS13]



Klassifizierungskriterien:

1. Paillier, ElGamal sind IND-CPA sicher $\rightarrow$ Transformiertes Programme ist IND-CPA sicher.
   Zwei semihomomorphe Kryptosysteme in Kombination, da dies effizienter ist als die Verwendung *eines* vollhomomorphen Kryptosystems.
2. Das Paillier Kryptosystem wird eingesetzt, um homomorphe Bitoperationen zu ermöglichen.
3. Eine Integrität der Rechenoperationen wird ausdrücklich nicht berücksichtigt.

Untersuchung von Privacy Preserving Face Recognition [EFG+09]



Klassifizierungskriterien:

1. Einsatz mehrerer Kryptosysteme zur Optimierung der Performance: Paillier (schnelle Schlüsselgenerierung, Vorberechnungen), DGK für schnelle Vergleiche.
2. Projektion (Skalarprodukt), quadratischer Abstand, jedoch nur möglich unter besonderen Vorraussetzungen des Anwendungsfalls.
3. unautorisierte Verformbarkeit nicht ausnutzbar: ein verformter Chiffretext wird im Protokoll nicht mehr zurück erhalten und der Angreifer kontrolliert höchstens eine Partei

Anwendungsfälle homomorpher Kryptosysteme 1/2

Veröffentlichung	Kryptosystem (Implementierung)	Angreifer	Integritätsprüfung der Chiffretexte
[TSCS13]	ElGamal (libcrypt [lib]), Paillier (CryptDB [PRZB11])	Bösartig	Nein, ausdrücklich nicht berücksichtigt
[BPTG15]	Goldwasser-Micali, Paillier (jeweils eigene Implementierung [RB] in C++ mit GMP [gmp] und NTL [nlt]), BGV (HElib [HEl][HS13])	HBC	Nein
[NIW+13]	Hash-ElGamal (eigene Implementierung)	Bösartig	**Signierung mit MAC-Code**
[DGK07]	DGK (eigene Implementierung in Java 1.5 unter Verwendung der Klasse BigInteger)	HBC	Nein
[KT05]	Okamoto-Uchiyama (eigene Implementierung)	HBC	unautorisierte Verformbarkeit nicht ausnutzbar: ein verformter Chiffretext wird im Protokoll nicht mehr zurück erhalten und der Angreifer kontrolliert höchstens eine Partei
[EFG+09]	Paillier, DGK (jeweils eigene Implementierung mit kryptografischen Optimierungen aus [DJ01] in C++ mit GMP [gmp])	HBC	Nicht ausnutzbar im HBC Modell, da nicht vom Protokoll abgewichen wird.
[BLN14]	BLLN (eigene Implementierung mit kryptografischen Optimierungen aus [Bra])	HBC	unautorisierte Verformbarkeit nicht ausnutzbar: ein verformter Chiffretext wird im Protokoll nicht mehr zurück erhalten und der Angreifer kontrolliert höchstens eine Partei

Anwendungsfälle homomorpher Kryptosysteme 2/2

Veröffentlichung	Chiffretextwechsel	Grund für Chiffretextwechsel
[TSCS13]	ElGamal $\leftrightarrow$ Paillier	Durchführung von Addition und Multiplikation mit gleicher Variable
[BPTG15]	Goldwasser-Micali $\rightarrow$ BGV Goldwasser-Micali $\rightarrow$ Paillier	Evaluierung von Polynomen, Weiterrechnen unter Paillier
[NIW+13]	-	-
[DGK07]	-	-
[KT05]	-	-
[EFG+09]	Paillier $\rightarrow$ DGK	schnellere Vergleiche
[BLN14]	-	-

Auswahlkriterien homomorpher Kryptosysteme

Kryptosystem	Auswahlkriterien	Typ	OP	Sicherheit
BGV [BGV12]	Evaluation von Polynomen	LVHE	LVHE	IND-CPA
BLLN [BLLN13]	Evaluation von Polynomen, keine Chiffretextexpansion bei Multiplikation, jedoch langsamer als BGV	LVHE	LVHE	IND-CPA
DGK [DGK07]	Speziell entwickelt für schnelle Vergleiche, kleiner Klartextraum	SH	+	IND-CPA
ElGamal [YPB14]	Homomorphe Multiplikation	SH	*	IND-CCA1
Goldwasser-Micali[GM82]	Vergleiche nach Protokoll [Veu11]	SH	XOR	IND-CPA
Hash-ElGamal [NIW+13]	Effizienter als additive SH Kryptosysteme um Chiffretexte zu maskieren	SH	XOR	IND-CPA
Okamoto-Uchiyama [OU]	Einfachere Implementierung als bei Paillier, da weniger Rechenschritte	SH	+, konst-*	IND-CPA
Paillier [Pai99]	Großer Klartextraum ermöglicht Verschlüsselung von Fließkommazahlen durch Multiplikation mit großen Exponenten, XOR durch Verschlüsselung der Integerbits, privates Skalarprodukt, Vergleiche nach Protokoll [Veu11]	SH	+, XOR, $\langle \cdot, \cdot \rangle$, konst-*	IND-CPA

Klassen nach Einsatzszenarien

1. **Evaluation von Polynomen** durch leveled vollhomomorphe Kryptosysteme: BGB, BLLN.
2. **Vergleiche** durch spezialisierte Kryptosysteme: DGK.
3. **Flexibel einsetzbare** Kryptosysteme: Paillier, Okamoto-Uchiyama.

Weiter gibt es...

4. **Vorausberechenbare** Kryptosysteme: Paillier, Okamoto-Uchiyama, DGK.

$$\text{Paillier: } k_p = (n, g) \qquad c = g^x r^n \bmod n^2$$

Zusammenfassung

1. Unautorisierte Verformbarkeit von Chiffretexten wird oft nicht untersucht (nicht Gegenstand) oder ist nicht ausnutzbar.
2. Integrität der Chiffretexte wird in einem Fall gewährleistet durch Signierung mit MAC-Codes.
3. Es folgende drei Klassen für die homomorphe Kryptosysteme identifiziert: 1) KS zur Polynomevaluation, 2) Spezialisierte KS (Vergleiche), 3) Flexible KS

Frage: Bob möchte ein Wasserzeichen in einem verschlüsselten Bild einbetten. Welches homomorphe Kryptosystem sollte er verwenden?
Antwort: viele Additionen von zwei Werten $\rightarrow$ beschleunigbare additiv-homomorphe Kryptosysteme: Paillier, Okamoto-Uchiyama, DGK

Ausblick

Recap: Homomorphieeigenschaft $\rightarrow$ unautorisierte Verformbarkeit der Chiffretexte

Non-Interactive Verifiable Computing: Client lagert die Berechnung der Funktion F von privaten Eingaben $x_1, \ldots, x_k$ an Arbeiter aus. Die Arbeiter...

1. ...bestimmen das Ergebnis der Evaluation von F auf $x_1, \ldots, x_k$
2. ...liefern ein Beweis, dass das Ergebnis korrekt evaluiert wurde.

Übertragen auf diese Untersuchung: 1) unverformtes x_i geht in die Berechnung ein, 2) F benutzt homomorphe Operatoren.

Literaturverzeichnis

Brakerski, Zvika ; Gentry, Craig ; Vaikuntanathan, Vinod:
(Leveled) fully homomorphic encryption without bootstrapping.
In: *ITCS*, 2012

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

Bos, Joppe W. ; Lauter, Kristin ; Naehrig, Michael:
Private predictive analysis on encrypted medical data.
In: *Journal of biomedical informatics* 50 (2014), S. 234–243

Bost, Raphael ; Popa, Raluca A. ; Tu, Stephen ; Goldwasser, Shafi:
Machine Learning Classification over Encrypted Data.
In: *NDSS*, 2015

Brakerski, Zvika:
Fully homomorphic encryption without modulus switching from classical GapSVP.
In: *Advances in Cryptology–CRYPTO 2012*. Springer, S. 868–886

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

Damgård, Ivan ; Jurik, Mads:
A generalisation, a simpli. cation and some applications of paillier's probabilistic public-key system.
In: *International Workshop on Public Key Cryptography* Springer, 2001, S. 119–136

Erkin, Zekeriya ; Franz, Martin ; Guajardo, Jorge ; Katzenbeisser, Stefan ; Lagendijk, Inald ; Toft, Tomas:
Privacy-preserving face recognition.
In: *International Symposium on Privacy Enhancing Technologies Symposium* Springer, 2009, S. 235–253

Goldwasser, Shafi ; Micali, Silvio:
Probabilistic encryption & how to play mental poker keeping secret all partial information.
In: *Proceedings of the fourteenth annual ACM symposium on Theory of computing* ACM, 1982, S. 365–377

GNU Multiple Precision Library (GMP).
`https://gmplib.org/`, . –
(Aufgerufen am 21. Mai 2017)

HELib.
`https://github.com/shaih/HElib`, . –

(Aufgerufen am 21. Mai 2017)

Halevi, Shai ; Shoup, Victor:

Design and implementation of a homomorphic-encryption library.
In: *IBM Research (Manuscript)* 6 (2013), S. 12–15

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

HELib.

`https://gnupg.org/related_software/libgcrypt/`, . –
(Aufgerufen am 21. Mai 2017)

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

NTL: A Library for doing Number Theory.

`http://www.shoup.net/ntl/`, . –
(Aufgerufen am 21. Mai 2017)

Okamoto, Tatsuaki ; Uchiyama, Shigenori:

A new public-key cryptosystem as secure as factoring.
In: *Advances in Cryptology—EUROCRYPT'98*

Paillier, Pascal:

Public-key cryptosystems based on composite degree residuosity classes.
In: *International Conference on the Theory and Applications of Cryptographic Techniques* Springer, 1999, S. 223–238

Popa, Raluca A. ; Redfield, Catherine ; Zeldovich, Nickolai ; Balakrishnan, Hari:
CryptDB: protecting confidentiality with encrypted query processing.
In: *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles* ACM, 2011, S. 85–100

Raphael Bost, Stephen T. Raluca Ada Popa P. Raluca Ada Popa:
Implementation of Machine Learning Classification over Encrypted Data.
`https://github.com/rbost/ciphermed/tree/master/src/crypto/`, . –
(Aufgerufen am 21. Mai 2017)

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

Veugen, Thijs:

Comparing encrypted data.
In: *Multimedia Signal Processing Group, Delft University of Technology, The Netherlands, and TNO Information*

and Communication Technology, Delft, The Netherlands, Tech. Rep (2011)

Yi, Xun ; Paulet, Russell ; Bertino, Elisa:
Homomorphic encryption and applications. Bd. 3. Springer, 2014