Verschlüsselung in mobilen Datenbanksystemen: Klassifikation und Implementierungsaspekte

Thomas Fanghänel

Friedrich-Schiller-Universität Jena, Fakultät für Mathematik und Informatik
Ernst-Abbe-Platz 2–4, 07743 Jena
thf@informatik.uni-jena.de

Zusammenfassung: Das vorliegende Papier behandelt die Möglichkeiten der Integration von Datenverschlüsselung in das Funktionsmodell eines datenunabhängigen DBMS. Dabei werden grundlegende Probleme diskutiert und Lösungsmöglichkeiten aufgezeigt. Eine Einordnung und Bewertung aktueller Produkte gibt einen Überblick über den derzeitigen Stand der Entwicklung.

1 Einführung

Persistente Datenspeicherung in einem DBMS bedeutet Schreiben einer systemspezifischen Darstellung der Daten, der *internen Repräsentation*, auf einen geeigneten Datenträger. Demgegenüber steht die *externe Repräsentation* der Nutzerdaten, so wie sie an der Schnittstelle zwischen Anwendung und DBMS vorliegen.

Meist ist der Transformationsprozeß zwischen interner und externer Repräsentation sehr einfach, so daß auch unter Umgehung des DBMS Inhalte einer Datenbank, und hierbei besonders Textdaten, ermittelt und interpretiert werden können. Um dem vorzubeugen, stellt die Integration von Verschlüsselung in den Transformationsprozeß ein adäquates Mittel dar. In mobilen Datenbanken mit vertraulichen Datenbeständen erlangt die Verschlüsselung der Daten oftmals eine besondere Signifikanz: Viele mobile Endgeräte und deren Betriebssysteme erlauben weder sichere Nutzerauthentifizierung, noch bieten sie ein rechtebasiertes Konzept für den Dateizugriff – beides sind wesentliche Aspekte im Zusammenhang mit Datensicherheit in hostbasierten DBMSen.

2 Datenverschlüsselung im Schichtenmodell

Als Modell für den Transformationsprozeß zwischen externer und interner Datendarstellung soll das Schichtenmodell eines datenunabhängigen DBMS herangezogen werden [HR01]. Der folgende Abschnitt diskutiert verschiedene Möglichkeiten, welche Schichten sich für die Integration eines Verschlüsselungsschrittes eignen.

2.1 Grundlegende Probleme

Integration von Datenverschlüsselung in ein DBMS verursacht drei Klassen von Problemen: Funktionale sowie operationale Einschränkungen und Leistungseinbußen. Funktionale Probleme ergeben sich aus nicht mehr erfüllten Modellannahmen. Für diese Art von

Problemen gibt es Lösungsmöglichkeiten, welche aber wiederum indirekt Leistungseinbußen nach sich ziehen. Operationale Probleme, z. B. die Angabe von falschen Schlüsseln oder Passwörtern oder fehlende Rechte für das Ausführen von Ver- und Entschlüsselungsoperationen, treten beim Betrieb des DBMS mit verschlüsselten Daten auf und führen zu nicht umgehbaren Laufzeitfehlern.

Leistungseinbußen resultieren direkt oder, wie oben schon erwähnt, indirekt aus dem Vorhandensein einer Verschlüsselungskomponente. Besteht eine hohe Anfragelast auf verschlüsselten Datenbeständen, so wirkt sich schon die häufige Ausführung von rechenintensiven Entschlüsselungsoperationen direkt auf die Leistung des DBMS aus, besonders wenn keine Pufferung der entschlüsselten Darstellungen durchgeführt wird. Hier ist es wünschenswert, durch ein möglichst kleines Verschlüsselungsgranulat, wie z. B. einzelne Tabellenspalten oder Attribute, und ein adäquates Datenbankdesign die Anzahl der Ver- und Entschlüsselungsoperationen zu minimieren.

Funktionale Probleme sind durch besondere Eigenschaften sicherer Verschlüsselungsalgorithmen bedingt. Durch nicht gegebene Ordnungstreue der Abbildung zwischen externen und internen Repräsentationen [MvOV97] ist beispielsweise die semantisch äquivalente Auswertung von Ordnungsrelationen oder Ähnlichkeitskriterien auf beiden Darstellungen nicht möglich. Elementgleichheit hingegen kann auf beiden Darstellungen semantisch äquivalent geprüft werden, sofern eine bijektive Abbildung zwischen externen und internen Datendarstellungen besteht.

Die genannten Probleme wirken sich bei der Anfragebearbeitung vor allem auf die Nutzbarkeit von Zugriffspfaden aus. Werden etwa Indexbäume über verschlüsselten Darstellungen aufgebaut, so können sie nicht mehr für die Unterstützung von Bereichsanfragen oder unscharfen Anfragen herangezogen werden. Punktanfragen hingegen können durch Indexe unterstützt werden, sofern das Verschlüsselungsgranulat ein Spaltenwert ist.

2.2 Klassifikation

Es sind verschiedene Zugänge denkbar, eine Verschlüsselungskomponente nachträglich in ein schon bestehendes DBMS einzubinden.

Anwendungsintegrierte Verschlüsselung. Ver- und Entschlüsselungsoperatoren sind in die Anwendung eingebettet. Das DBMS selbst enthält keine Logik für den Umgang mit verschlüsselten Daten. Dadurch können lediglich Nutzerdaten verschlüsselt werden, und als Verschlüsselungsgranulat sind nur Spaltenwerte möglich.

Da verschlüsselte Daten an das DBMS übergeben werden, ist die Funktionalität des DBMS stark eingeschränkt. Bereichsanfragen, Constraints zur Bereichsüberwachung von Datentypen und Vergleichsprädikate zwischen verschlüsselten und unverschlüsselten Spaltenwerten sind als Bestandteil von Anfragen nicht möglich. Referentielle Integrität zwischen verschlüsselten und unverschlüsselten Spalten kann nicht durch das DBMS sichergestellt werden.

Datenbanksystembasierte Verschlüsselung. DBMS und Anwendung teilen sich die Verantwortung für die Datenverschlüsselung, wobei auf Seiten des DBMS Gebrauch von Erweiterungsmechanismen, wie z. B. UDFs oder Triggern, gemacht wird. Die Anwendung

muß durch adäquat formulierte Anfragen diese Erweiterungsmechanismen benutzen. Das eigentliche DBMS weist keine spezielle Anpassung für den Umgang mit verschlüsselten Daten auf, dadurch ist eine datenbanksystembasierte Lösung, analog zur anwendungsintegrierten Verschlüsselung, auf Spaltenwerte als Granulat festgelegt. Analog können auch nur Nutzerdaten, aber keine Systemdaten mittels datenbanksystembasierter Verschlüsselung behandelt werden.

Viele Nachteile des anwendungsintegrierten Ansatzes können durch einen datenbanksystembasierten Ansatz vermieden werden, da Ver- und Entschlüsselungsoperationen auf SQL-Ebene kontrolliert und angestoßen werden. So lassen sich prinzipiell Bereichsanfragen formulieren, das Problem der eingeschränkten Nutzbarkeit von Indexen besteht aber weiterhin.

Datenbanksystemintegrierte Verschlüsselung. Durch Einbettung der Verschlüsselung in die Schichtenarchitektur des DBMS kann prinzipiell allen in Abschnitt 2.1 genannten funktionalen Problemen begegnet werden. Allgemein können folgende Varianten unterschieden werden:

– *Verschlüsselung im Datensystem:* Die Verschlüsselung geschieht während der Abbildung von Tupeln auf die externe Satzdarstellung oder bei Transformation der externen in die interne Satzdarstellung. Diese Methode besitzt ähnliche Eigenschaften wie ein datenbanksystembasierter Zugang, und sie eignet sich nur zur Verschlüsselung von Nutzerdaten.

  Als Granulate können Spaltenwerte oder Sätze benutzt werden. Die Nutzung von Zugriffspfaden ist wiederum stark eingeschränkt, da diese in jedem Falle auf der verschlüsselten Darstellung von Sätzen beruhen. Die Gültigkeit von Fremdschlüsselbeziehungen und Wertevergleichen zwischen verschlüsselten und unverschlüsselten Datenwerten kann hingegen durch das DBMS verifiziert werden.

– *Verschlüsselung im Zugriffssystem:* Verschlüsselungsoperationen werden während der Abbildung von internen Sätzen in Seiten vorgenommen. Damit ist es möglich, sowohl Nutzer- als auch Systemtabellen zu verschlüsseln.

  Die Granularität der Verschlüsselung ist auf Satzebene beschränkt, einzelne Spaltenwerte können hingegen nicht mehr aufgelöst werden. Indexe auf verschlüsselten Datenbankinhalten können ohne Einschränkung benutzt werden, allerdings ist eine persistente Speicherung der Zugriffspfaddarstellungen ohne zusätzliche Vorkehrungen nur unverschlüsselt möglich.

– *Verschlüsselung im Speichersystem:* Verschlüsselung auf Ebene des Speichersystems arbeitet auf Basis von Seiten oder Blöcken, und damit auf physischen statt logischen Datenbankgranulaten. Daher ist keine sehr feine Auflösung bei der Verschlüsselung möglich, weil potentiell viele Sätze pro Operation verschlüsselt werden. Das impliziert Tabellen als kleinste Einheit für die Verschlüsselung.

  Andererseits besteht auf dieser Ebene aber kein Unterschied mehr zwischen Nutzer-, System-, Protokoll- oder Zugriffspfaddaten. Somit lassen sich alle Arten von persistent gespeicherten Daten mit einem ins Speichersystem integrierten Ansatz verschlüsseln. Funktionale Einschränkungen sind nicht zu erwarten, da die Entschlüsselung vor dem Aufbau von Daten- oder Zugriffspfadstrukturen vorgenommen wird.

Betriebssystemintegrierte und hardwaregestützte Verschlüsselung. Eine vollständige Transparenz der Verschlüsselungsoperationen für das DBMS wird erreicht, wenn diese direkt ins Betriebssystem eingebettet oder in Hardware realisiert sind. Beispiele sind das Windows Encrypting File System (EFS) [Mic98, Mic00, Rus99] bzw. diverse Mechanismen bei austauschbaren Datenträgern für mobile Endgeräte [Gle02, Pal01, vW01].

Betriebssystemintegrierte Verschlüsselung dient dazu, die auf einem Datenträger gespeicherten Daten *systemabhängig* zu machen, erhöht also die Datenträgersicherheit. Ein Lesen der Daten unter Umgehung des Betriebssystems oder nach Austausch des Datenträgers ist nicht möglich. Hardwareintegrierte Verschlüsselung realisiert analog eine *Geräteabhängigkeit* der Daten und ist vor allem im Bereich des Digital Rights Management zu finden. Beide Ansätze sind nicht ausreichend, zur Erhöhung der Datensicherheit bei mobilen Datenbanken beizutragen, da sie keine *Nutzerabhängigkeit* der Daten, wie z. B. durch einen Passwortschutz, erreichen.

3 Produktübersicht

Dieser Abschnitt soll einen kurzen Überblick über aktuelle Implementierungen von Verschlüsselungslösungen in mobilen Datenbanksystemen geben und diese miteinander vergleichen.

Oracle Lite. Oracle 9*i* Lite unterstützt Verschlüsselung auf Datenbankebene. Es müssen externe Dienstprogramme für Ver- und Entschlüsselung von Datenbanken benutzt werden, ein Nutzerinterface auf SQL-Ebene existiert nicht. Die Verschlüsselung ist, bis auf die obligatorische Angabe des korrekten Passworts beim Verbindungsaufbau, für Anwendungen transparent. Die in Oracle Lite implementierte Verschlüsselungskomponente folgt dem datenbanksystemintegrierten Ansatz, und vermutlich ist sie ins Speichersystem integriert, da keinerlei funktionale Einschränkungen existieren.

iAnywhere Adaptive Server Anywhere und UltraLite. Sowohl Adaptive Server Anywhere als auch UltraLite in den Versionen 8.0 erlauben Datenverschlüsselung. Es werden jeweils alle zu einer Datenbank gehörenden Daten verschlüsselt, wobei Verschlüsselung und Entschlüsselung von Datenbanken über produktspezifische SQL-Erweiterungen gesteuert werden. Bei jedem Verbindungsaufbau muß ein Datenbankpasswort angegeben werden. Beide Produkte implementieren eine ins Datenbanksystem integrierte Verschlüsselungskomponente. Da alle Daten, Indexe und Protokolldateien eingeschlossen, verschlüsselt werden, ist zu vermuten, daß die Verschlüsselung auf physischen Datenbankgranulaten basiert und ebenfalls ins Speichersystem integriert ist.

Microsoft SQL Server CE Edition. Microsofts SQL Server 2000 Windows CE Edition unterstützt in Version 1.1 Verschlüsselung von Datenbanken, wobei Funktionen des Betriebssystems Windows CE Version 3.0 oder höher vorausgesetzt und benutzt werden. Auch SQLServer CE gestattet lediglich die Verschlüsselung kompletter Datenbanken, wobei auch hier das Nutzerinterface als Erweiterung des SQL-Befehlsumfangs realisiert ist. Der Hauptunterschied zu den Lösungen von Oracle und iAnywhere besteht in der engen Bindung an Betriebssystemfunktionalität. So ist z. B. die Schlüsselverwaltung kein Teil des DBMS, sondern wird ebenfalls vom Windows CryptoAPI übernommen. Es liegt also eine Mischform zwischen datenbank- und betriebssystemintegrierter Verschlüsselung vor.

4 Zusammenfassung und Ausblick

Um beim Umgang mit verschlüsselten Datenbankinhalten Anwendungsunabhängigkeit zu erreichen, muß ein datenbanksystemintegrierter Ansatz gewählt werden. Dabei gilt: Je feiner das Verschlüsselungsgranulat sein soll, um so höher muß die Verschlüsselungskomponente im Schichtenmodell angesiedelt sein, und desto mehr funktionale Probleme müssen geeignet behandelt werden. Als vernünftigste Lösung erscheint eine Integration ins Speichersystem. Da hierbei keine funktionalen Probleme auftreten, bewegen sich die am DBMS erforderlichen Anpassungen in den engen Grenzen des Speichersystems.

Die in kommerziellen mobilen DBMSen zum Einsatz kommenden Lösungen weisen eine gewisse Homogenität auf. Es sind keine datenbanksystembasierten oder betriebssystemintegrierten Ansätze in diesen Produkten zu finden, ausschließlich datenbanksystemintegrierte Lösungen kommen zum Einsatz. Aber gerade im Hinblick auf die Optimierbarkeit von Anwendungen und die Möglichkeit anwendungsspezifischer Anpassung der Datenbank ist eine datenbanksystembasierte Lösung, wie sie IBMs DB2 UDB [IBM01] oder Oracle 9*i* [Ora01] bieten, den in Abschnitt 3 beschriebenen integrierten Lösungen überlegen, da durch geeigneten DB-Entwurf und gutes Anfragedesign eine erhebliche Leistungssteigerung erreicht werden kann.

Die untersuchten Produkte erstaunen auch dahingehend, daß sie ohne Ausnahme nur Verschlüsselung auf Datenbankebene gestatten. Gerade im Hinblick auf die eingeschränkte Leistungsfähigkeit mobiler Endgeräte der jeweiligen Zielplattformen wäre ein feineres Verschlüsselungsgranulat für viele Anwendungsszenarien wünschenswert.

Literaturverzeichnis

[Gle02] Clemens Gleich. Das Gigabyte im Geldbeutel: Mini-Massenspeicher in Flash-Technologie. *c't Magazin für Computertechnik*, (8):164–166, 2002.

[HR01] Theo Härder und Erhard Rahm. *Datenbanksysteme: Konzepte und Techniken der Implementierung*. Springer Verlag Berlin Heidelberg, 2. Auflage, 2001.

[IBM01] IBM DB2 Universal Database Version 7.2/Version 7.1 FixPak 3 – Release Notes. IBM Corp., 2001.

[Mic98] Microsoft Corporation. Encrypting File System for Windows 2000. A Microsoft Windows 2000 Server White Paper, 1998.

[Mic00] Microsoft SQL Server 2000 Books Online. Microsoft Corporation, 2000.

[MvOV97] Alfred J. Menezes, Paul C. van Oorschot und Scott A. Vanstone. *Handbook of Applied Cryptography*. The CRC Press Series on Discrete Mathematics and its Applications. CRC Press LLC, 1997.

[Ora01] Oracle Corporation. Database Encryption in Oracle 9*i*™. An Oracle Technical White Paper, Februar 2001.

[Pal01] Palm, Inc. Palm's New Dual Expansion Architecture. White Paper, 2001.

[Rus99] Mark Russinovich. Inside Encrypting File System, parts 1 & 2. *Windows & .NET Magazine*, Juni – Juli 1999.

[vW01] William van Winkle. Shuffle the deck—Here's the deal concerning PC Cards. *Smart Computing*, 6(3):60–65, August 2001.