Warum XP_CRYPT & SQL Shield?

Aus der Sicht eines Projektmanagers.

TEIL I: DEN BEDARF DEFINIEREN. Wo fehlt Schutz beim SQL Server?
Schutz auf Datenfeldebene
Schutz von Stored Procedures und Skripten
TEIL II: CODE-LÖSUNG – Wie man erfolgreich SQL-Code schützt.
Schutz von Skripten…
TEL III: DATEN-LÖSUNG – Wie man den Schutz Ihrer SQL-Daten erhöht
TEIL IV: DEN GESCHÄFTSFALL DURCHGEHEN – Kauf gegenüber Eigenentwicklung

TEIL I: DEN BEDARF DEFINIEREN. Wo fehlt Schutz beim SQL Server?

Schutz auf Datenfeldebene

Überraschenderweise, verschlüsselt SQL Server nicht auf Datenfeldebene. Der Zugriff auf Daten wird dadurch gewährt, indem man sich bei der Datenbank anmeldet. Tatsache ist jedoch, dass jeder, der Zugriff auf das Dateisystem hat, lediglich die Dateien der Datenbank kopieren muss, sie in ein Datenbanksystem einfügt, für das man Administratorrechte besitzt und hiermit kompletten Zugriff auf Ihre Daten erhält.

Damit ist die Wahrheit in Hinsicht auf Sicherheit, dass SQL Server sehr sicher ist, solange niemand Zugriff auf Ihr Dateisystem hat. Angesicht der vielen Exploits und Hacker im Umlauf, ist dies jedoch ein großes Vertrauen, das man als Verantwortlicher in den Schutz von teilweise vertraulichen Daten (wie Kreditkartennummern, Krankenakten, etc.) haben muss.

Schutz von Stored Procedures und Skripten

SQL Server erlaubt es Entwicklern, Logik in die Datenbank zu programmieren. Diese Logik wird abgelegt als Stored Procedures, Triggern und benutzerdefinierten Funktionen. Es gibt eine Reihe von Gründen, warum Sie diese Logik verschlüsseln möchten:

Erstens gibt es Bedenken hinsichtlich des geistigen Eigentums. Wenn jemand in der Lage ist, Einblick in die Skriptlogik zu bekommen, ist dies gleichbedeutend mit dem Einblick in Ihren Quelltext. Das bedeutet, man kann die „geheimen“ inneren Arbeitsabläufe Ihres Projekts nachvollziehen, was Reverse Engineering erheblich einfacher macht.

Zweitens, wenn jemand Einblick in Ihre Stored Procedures hat, erlaubt ihm dies auch, diese zu bearbeiten. D.h. man kann Ihre Stored Procedures überschreiben und spezielle Logik einbauen, die Ihre Datenbank beeinflusst. Was das bedeuten kann? Möglich sind u.a. das Löschen von Daten über das Zerstören Ihrer Datenbank bis hin zu schädlichen Handlungen, wie Diebstahl, d.h. geheime medizinische Daten werden geschrieben oder gelesen, wenn ein spezieller Begriff „übertragen“ wurde, oder beispielsweise bei einer E-Commerce-Anwendung, bei der statt einer Belastung eines Kontos einer bestimmten Person eine Gutschrift erfolgt, sobald diese etwas kauft.

Klingt furchterregend, nicht wahr? Nun, glücklicherweise gibt es eine Lösung für beide dieser Probleme.

TEIL II: CODE-LÖSUNG – Wie man erfolgreich SQL-Code schützt.

Schutz von Skripten…

Öffnet man die Hilfeseiten von SQL Server, findet man schnell heraus, dass SQL Server durchaus Verschlüsselung für Stored Procedures und Skripte anbietet. Bevor Sie einen Seufzer der Erleichterung ausstoßen, hier ist was die Hilfeseite Ihnen nicht sagt, nämlich dass man nach fünf Minuten Suchen im Web, kostenlos eins von vielen Programmen herunterladen kann, die Ihre „Microsoft verschlüsselten“ Stored Procedures in kürzester Zeit entschlüsseln kann. D.h. jeder Hacker mit ein bisschen Ahnung hat sogar dann die Möglichkeit, Zugriff auf Ihren SQL-Code zu bekommen und zu machen, was immer er damit will, sogar dann, wenn Sie es unter Verwendung von SQL Servers „herkömmlicher“ Verschlüsselung verschlüsselt haben.

Was können Sie tun? Nun, glücklicherweise bietet SQL Shield eine Verschlüsselung Ihrer Stored Procedures an, die kein bekanntes Knackprogramm entschlüsseln kann. D.h. ein Hacker, der sieht, dass Ihre Stored Procedures verschlüsselt wurden, wird nicht in der Lage sein – unabhängig wie oft er es versucht diese Knackprogramme zu verwenden – Ihre SQL-Skripte zu entschlüsseln, was Sie schützt.

Bei diesem Stand der Dinge, wird es noch wichtiger Ihre Daten zu schützen.

TEL III: DATEN-LÖSUNG – Wie man den Schutz Ihrer SQL-Daten erhöht

Es gibt verschiedene Algorithmen, die Sie für die Verschlüsselung Ihrer Daten verwenden können. XP_CRYPT unterstützt u.a. RSA (asymmetrisches Verschlüsselungsverfahren), AES, Triple DES, DESX und RC4 (Symmetrische Verschlüsselungsverfahren). Sie können den Algorithmus in Abhängigkeit von Ihrem Bedarf wählen.

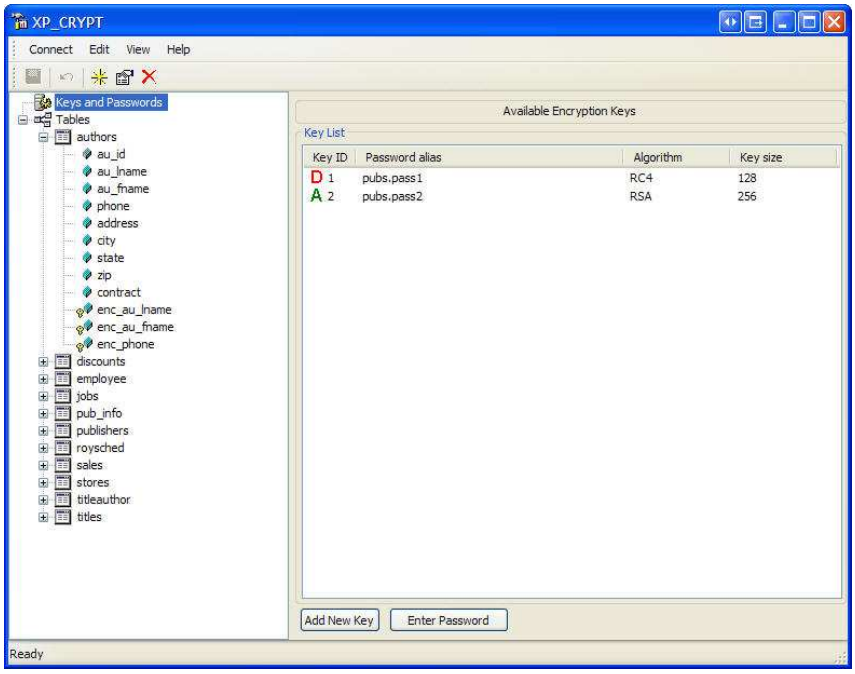
Jedoch ist zu beachten, dass asymmetrische Verschlüsselungsverfahren relativ langsam sind im Vergleich zu symmetrischen Verschlüsselungsverfahren.

Mit der Verwendung der XP_CRYPT GUI, die im Grunde genommen ein Programm darstellt, das auf einfache Weise Programmroutinen in Ihre Datenbank einfügt, ist das Verschlüsseln von Datenfeldern mit XP_CRYPT ein Klacks.
Die XP_CRYPT GUI automatisiert einen beträchtlichen Teil der Arbeit. Sie fügt all die Schnittstellen und Hilfsprogramme hinzu und wendet diese auf Ihre Datenbank an. Sie können mehrere Algorithmen auf einfache Weise hinzufügen, jeden mit seinem eigenen Schlüssel. Aufgrund des größeren Aufwands einiger der stärkeren Algorithmen, ist es durchaus sinnvoll verschiedene Arten der Verschlüsselung, je nach Länge und Typ des Felds und der Höhe der benötigten Sicherheit anzubieten.

In weniger als fünf Minuten können Sie die Felder in Ihrer Datenbank verschlüsseln. Das Programm kümmert sich hierbei um folgende Dinge:

1) Die Erzeugung von Tabellen zur Verwaltung Ihrer Passwörter.
2) Erzeugung von Feldern, die Ihre gewählten Datenfelder in verschlüsselter Form repräsentieren (Sie löschen die Klartextfelder per Hand).
3) Erzeugung eines Views, das die von Ihnen verschlüsselten Felder entschlüsselt darstellt.
4) Eine Funktion, die den Schlüssel für die Entschlüsselung wahrend einer Benutzersession aktiviert und behält.

Die folgenden Bilder stellen einige der Fenster dar, die benutzt werden, um Ihrer Datenbank Verschlüsselung auf Datenfeldbasis hinzuzufügen.



Bold 1 – Hier können mehrere Verschlüsselungsverfahren der Datenbank unter Nutzung der XP_CRYPT GUI hinzugefügt werden.

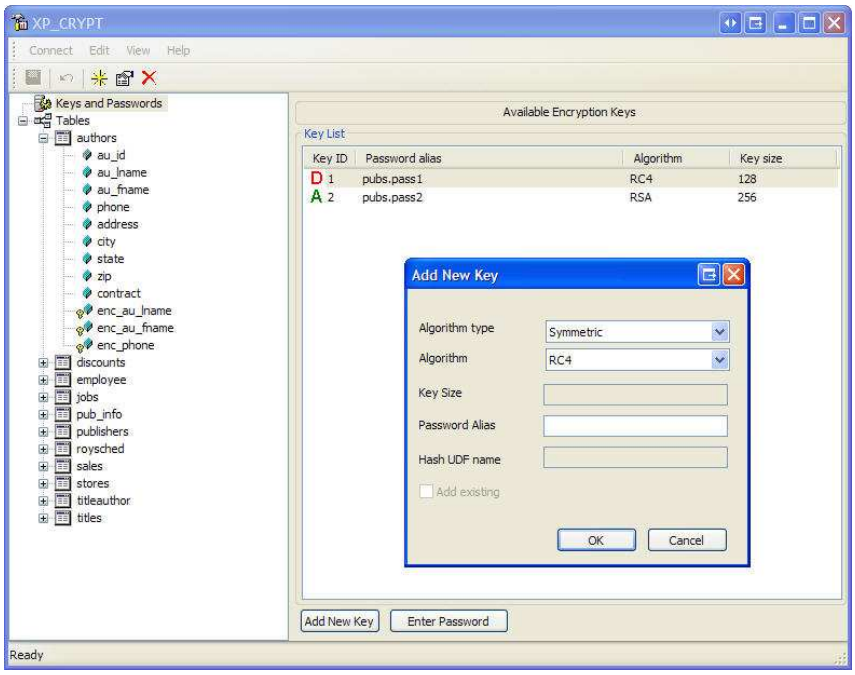


Bild 2 – Hier können Einstellungen eines bestimmten Verschlüsselungsverfahrens festgelegt werden. Verschiedene Algorithmen werden unterstützt, einschließlich symmetrischer und asymmetrischer Verfahren.

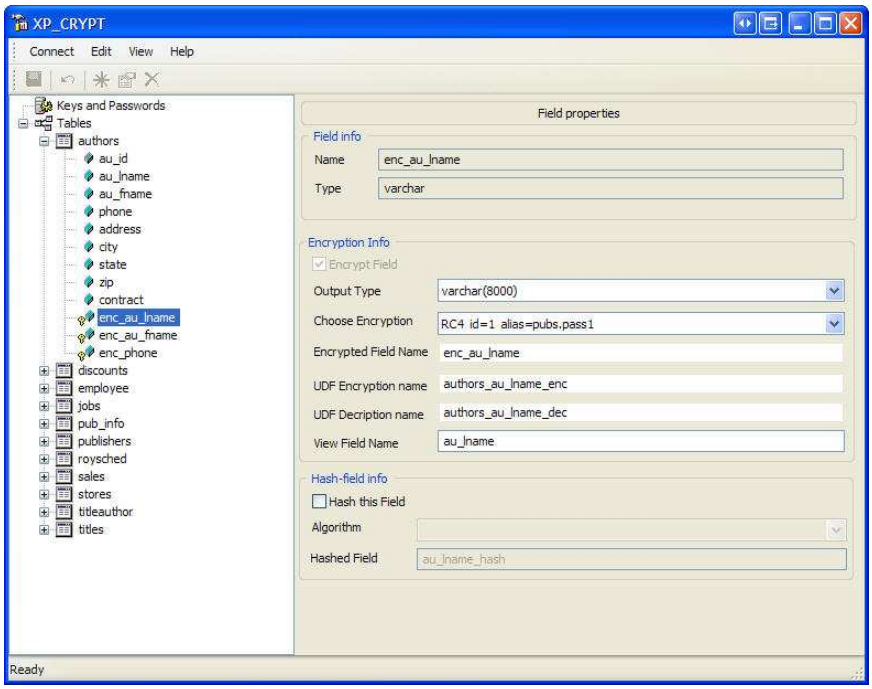


Bild 3 – Hier erfolgt die Zuordnung von Verschlüsselungsalgorithmen auf einzelne oder mehrere Felder und die Auswahl des Ausgabetyps, Art der Verschlüsselung und der Feldnamen für die Prozeduren, die von XP_CRYPT GUI erzeugt werden.

TEIL IV: DEN GESCHÄFTSFALL DURCHGEHEN – Kauf gegenüber Eigenentwicklung

Es kommt die Zeit bei der man ein Hilfsmittel ausprobiert und sich die Frage stellt, ob man es kauft oder selber entwickelt.

1) Erweiterte Stored Procedures **müssen fehlerfrei sein**. Eine schlecht entwickelte erweiterte Stored Procedure, die häufig von Ihrer Datenbank aufgerufen wird, kann Ihre Datenbank zum Stehen bringen oder zerstören. Die Tatsache, dass XP_CRYPT ein kommerzielles Produkt ist, das bereits von vielen Benutzern genutzt wird, verringert dieses Risiko.
2) Ein Vorteil, eine Anwendung selbst zu entwickeln, ist die Verfügbarkeit des Quelltextes. Glücklicherweise können Sie eine Version von XP_CRYPT inklusive Quelltext erwerben.
3) **Zeit ist Geld**. Eine Eigenentwicklung kostet sehr viel Geld, wenn Sie Unterstützung mehrerer Verschlüsselungsverfahren und ausgiebige Tests wünschen. Zusätzlich erlaubt die schöne XP_CRYPT GUI ein schnelles Vorankommen auf einfache Weise.

4) Wie viel sind Ihre Daten wert? Oder anders gefragt, was ist das Schlimmste, das im Falle eines Verlusts der Datenintegrität oder wenn Sie sie nicht mehr entschlüsseln können, passieren kann? Probleme wie diese können passieren, wenn Fehler auftreten. Das Zahlen von mehreren tausend Dollar für den Seelenfrieden durch eine erprobte Lösung ist um einiges besser als sich zu fragen, ob der Fehler in einer Stored Procedure oder dem selbstgeschriebenen Verschlüsselungsprogramm liegt. Wenn die **Daten sehr wertvoll sind**, fragen Sie sich selbst, wie hoch die Kosten sein werden, wenn die Dinge in der Datenbank nicht richtig laufen.
5) Dies ist eine **bewährte Lösung**. XP_CRYPT wird bereits von staatlichen Organisationen der USA, Finanzdienstleistern, medizinischen Einrichtungen und Universitäten in der ganzen Welt eingesetzt.
6) Lizenzierung – XP_CRYPT bietet verschiedene Möglichkeiten der Lizenzierung an. **Die Preise sind sehr angemessen**, da sie einen Bruchteil der Preise von Konkurrenten ausmachen.
   Lizenzmöglichkeiten sind u.a.:
   a. **Einzellizenzen** für einzelne Server
   b. **Firmenlizenz** kann erworben werden, welche alle Server einer Firma abdeckt
   c. **Wiederverkäufliche** Lizenz ist gedacht für Softwarehersteller, die Lösungen weiterverkaufen, die geschützt werden sollen und die Technologie in die ausgelieferte Software integrieren.

Wenn man die Vor- und Nachteile in Betracht zieht, eine eigene Lösung zu schreiben, sieht man sehr schnell, dass eine Lösung inklusive Quelltext zu einem angemessenen Preis die beste ist. Glücklicherweise ist XP_CRYPT eine sehr gute Lösung und ist zu einem sehr guten Preis verfügbar.