combit GmbH
Untere Laube 30
78462 Konstanz

Whitepaper

Produkt: combit Relationship Manager

HowTo: Microsoft SQL Server SSL-Verschlüsselung aktivieren

Inhalt

SQL Server SSL-Verschlüsselung aktivieren

In Microsoft SQL Server kann SSL (Secure Sockets Layer) zum Verschlüsseln der Daten verwendet werden, die über ein Netzwerk zwischen einer SQL Server-Instanz und einer Clientanwendung übertragen werden und erhöht so die Sicherheit. Durch das Aktivieren der Verschlüsselung kommt es jedoch zu Leistungseinbußen. Wenn der gesamte Datenverkehr zwischen SQL Server und einer Clientanwendung mithilfe von SSL verschlüsselt wird, sind die folgenden zusätzlichen Verarbeitungsschritte erforderlich:

- Ein zusätzlicher Netzwerkroundtrip ist zum Zeitpunkt des Verbindungsaufbaus erforderlich.
- Die von der Anwendung an die SQL Server-Instanz gesendeten Pakete müssen von der Client-Netzwerkbibliothek verschlüsselt und von der Server-Netzwerkbibliothek entschlüsselt werden.
- Die von der SQL Server-Instanz an die Anwendung gesendeten Pakete müssen von der Server-Netzwerkbibliothek verschlüsselt und von der Client-Netzwerkbibliothek entschlüsselt werden.

Vorgehensweise

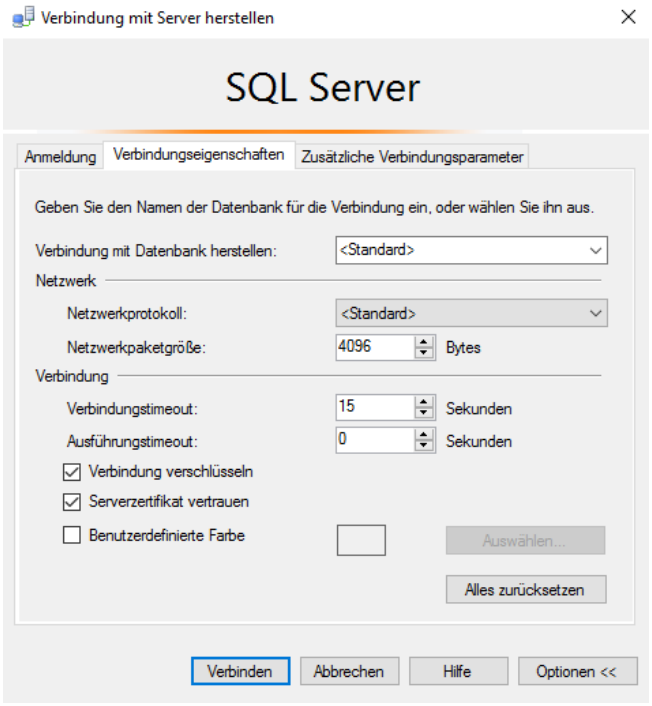
1. SSL-Zertifikat erstellen: (Steht bereits ein Zertifikat zur Verfügung, so muss kein neues erstellt werden)
    a. Rufen Sie *Systemsteuerung > Administrationstools > IIS Manager > Serverzertifikate > Selbst signiertes Zertifikat erstellen* (Control Panel > Administrative Tools > IIS Manager > Server Certificates > Create Self-Signed Certificate) auf.
    b. Benennen Sie das Zertifikat, klicken Sie auf OK und exportieren Sie das Zertifikat.
    c. Schließen Sie den IIS Manager.
2. Starten Sie die Microsoft Management Console: *Start > Ausführen > mmc* (Start > Run > mmc)
    a. Rufen Sie *Datei > Snap-in hinzufügen/entfernen > Zertifikate > Hinzufügen > Computeraccount* (File > Add/Remove Snap-in > Certificates > Add > Computer account) auf.
    b. Wählen Sie den Computer aus, der vom Snap-in verwaltet werden soll und klicken Sie auf *Fertigstellen > OK* (Finish > OK).
    c. Erweitern Sie Zertifikate (Certificates), klicken Sie mit der rechten Maustaste auf den Ordner Persönlich (Personal) und rufen Sie *Alle Tasks > Importieren* (All Tasks > Import) auf.
    d. Befolgen Sie die Anweisungen des Assistenten und importieren Sie entweder das zuvor erstellte oder bereits vorhandene Zertifikat.
    e. Schließen Sie die Microsoft Management Console und starten Sie den SQL Server-Dienst neu. ***Wichtig:*** Stellen Sie sicher, dass der Service-Account Zugriff auf Zertifikate hat. Möglicherweise muss er als lokaler Account ausgeführt werden.
3. Öffnen Sie den SQL Server Configuration Manager:
    a. Erweitern Sie die SQL Server-Netzwerkkonfiguration, klicken Sie mit der rechten Maustaste auf die Option für Protokolle des SQL Servers und wählen Sie dann Eigenschaften.
    b. Wählen Sie auf der Registerkarte für Flags im Feld *Verschlüsselung erzwingen* (Force encryption) die Option *Ja* (Yes) aus und klicken Sie dann auf OK.
    c. Wenn die Option *ForceEncryption* für das Datenbankmodul auf *Ja* festgelegt ist, werden sämtliche Client/Server-Kommunikationen verschlüsselt, und Clients, die die Verschlüsselung nicht unterstützen, erhalten keinen Zugriff.
    d. Ist die Option *ForceEncryption* für das Datenbankmodul auf *Nein* festgelegt, kann die Verschlüsselung von der Clientanwendung angefordert werden, ist jedoch nicht erforderlich.
    e. Wählen Sie aus der Zertifikatschaltfläche das Zertifikat aus und klicken Sie auf OK, um das Dialogfenster zu schließen.
    f. Starten Sie den SQL Server-Dienst neu.
    g. Die SSL-Verschlüsselung ist jetzt auf dem SQL Server aktiv

Vorsicht:

SSL-Verbindungen, die mithilfe eines selbstsignierten Zertifikats verschlüsselt werden, bieten keine hohe Sicherheit. Sie sind anfällig für "Man-in-the-Middle"-Angriffe.
In einer Produktionsumgebung oder auf Servern, die mit dem Internet verbunden sind, sollten Sie sich nicht auf SSL mit Verwendung selbstsignierter Zertifikate verlassen.

Einzelne Verbindung zwischen Microsoft SQL Server Management Studio und SQL Server Verschlüsseln

1. Bei der Verbindung zu einem Server auf Optionen klicken.
2. In der Registerkarte *Verbindungseigenschaften* unter Verbindung die Checkbox bei *Verbindung verschlüsseln* (Force Encryption) und *Serverzertifikat vertrauen* (Trust Server Certificate) aktivieren.
3. Dadurch lässt sich eine Verbindung zwischen dem Microsoft SQL Server Management Studio und einem SQL Server verschlüsseln, obwohl die Force Encryption Option auf dem SQL Server nicht aktiviert ist.



Optional: Aktivierung einer verschlüsselten Verbindung für einen bestimmten Client

1. Starten Sie auf dem Server die Microsoft Management Console: *Start > Ausführen > mmc* (Start > Run > mmc)
2. Erweitern Sie *Zertifikate* (Certificates) und rufen Sie *Alle Tasks > Exportieren* (All Tasks > Export) auf.
3. Befolgen Sie die Anweisungen des Assistenten und exportieren Sie das Zertifikat.
4. Kopieren Sie das Zertifikat auf den Client Computer.
5. Starten Sie die Microsoft Management Console auf dem Client Computer: *Start > Ausführen > mmc* (Start > Run > mmc)
6. Rufen Sie *Datei > Snap-in hinzufügen/entfernen > Zertifikate > Hinzufügen > Eigenes Benutzerkonto* (File > Add/Remove Snap-in > Certificates > Add > Personal Useraccount) auf.
7. Navigieren Sie auf dem Clientcomputer zum Ordner *Vertrauenswürdige Stammzertifizierungsstellen* (Trusted Root Certification Authorities) in der Microsoft Management Console.
8. Klicken Sie mit der rechten Maustaste auf den Ordner *Vertrauenswürdige Stammzertifizierungsstellen* (Trusted Root Certification Authorities) und wählen Sie anschließend unter *Alle Aufgaben -> Importieren* (All Tasks > Import) das zu importierende Zertifikat aus.
9. Schließen Sie die Microsoft Management Console und starten Sie den SQL Server-Dienst auf dem Server neu.
10. Öffnen Sie auf dem Client den SQL Server Configuration Manager.
11. Klicken Sie mit der rechten Maustaste auf *SQL Native Client Konfiguration* (32Bit für den cRM) und anschließend auf Eigenschaften.
12. Wählen Sie auf der Registerkarte für Flags im Feld *Protokoll Verschlüsselung erzwingen* (Force Protocol Encryption) die Option *Ja* (Yes) aus und klicken Sie dann auf OK.
13. Der Client kann jetzt die SSL-Verschlüsselung verwenden.

Zertifikatsanforderungen

1. Das Zertifikat muss sich im Zertifikatspeicher des lokalen Computers oder im Zertifikatspeicher des aktuellen Benutzers befinden.
2. Die aktuelle Systemzeit muss nach der *Gültig von* (Valid from) Eigenschaft und vor der *Gültig bis* (Valid to) Eigenschaft des Zertifikats liegen.
3. Das Zertifikat muss für die Serverauthentifizierung vorgesehen sein. Hierfür muss die *Enhanced Key Usage*-Eigenschaft des Zertifikats auf den Wert *Server Authentication (1.3.6.1.5.5.7.3.1)* festgelegt sein.
4. Das Zertifikat muss mit der *KeySpec*-Option von *AT_KEYEXCHANGE* erstellt werden. Normalerweise enthält die Schlüsselverwendungseigenschaft (*KEY_USAGE*) des Zertifikats auch die Schlüsselverschlüsselung (*CERT_KEY_ENCIPHERMENT_KEY_USAGE*).
5. Mit der Subject-Eigenschaft des Zertifikats muss angegeben werden, dass der allgemeine Name (Common Name, CN) mit dem Hostnamen oder dem vollqualifizierten Domänennamen (Fully Qualified Domain Name, FQDN) des Servercomputers übereinstimmt. Wenn SQL Server auf einem Failovercluster ausgeführt wird, muss der allgemeine Name mit dem Hostnamen oder FQDN des virtuellen Servers übereinstimmen, und die Zertifikate müssen auf allen Knoten im Failovercluster bereitgestellt werden.