

Re: Decryption within an application

Source: <http://www.derkeiler.com/Newsgroups/microsoft.public.sqlserver.security/2007-05/msg00095.html>

- *From:* "Mike C#" <xyz@xxxxxxx>
 - *Date:* Thu, 17 May 2007 22:49:19 -0400
-

"Chuck Reif" <creif@xxxxxxxxxxxxxxxxxxxx> wrote in message <news:uvmlanAmHHA.4852@xxxxxxxxxxxxxxxxxxxxxxxx>

I need to encrypt one column of data in a single table and I pretty much have all the operations figured out, including maintaining both the encrypted data and a one way hash for searches. I have a view which decrypts the data properly when the symmetric key has been opened (and obviously returns null when the key is not open).

I want the view to return the decrypted data only when the user is accessing the database from a single application. This application maintains a single database connection per session. My thought was to open the key when the database connection is established by the application and close it when the application exits, thereby granting access only through the application. Is that an acceptable practice?

If I do that, should I protect the key with a password that is then compiled in the application so that I can open the key? This means that every installation will have a key protected by the same password. Or is there a better way to do that?

Thanks for any help.

Well, when the key is opened it's specific to a session. So you could have several sessions opening up the same key simultaneously and I wouldn't think you'd encounter any problems. Of course you will probably want to do some thorough testing to be sure, and also make sure you don't take a performance hit there. I wouldn't recommend storing the key hard-coded in your application. How about using the Automatic Key Management feature of SQL 2005? The only real downside to it is that all sysadmins can then decrypt your data (if that's a concern for you – it is for some folks).