

Re: Limiting a user to only executing a stored only that access an

Re: Limiting a user to only executing a stored only that access an

Source: <http://www.derkeiler.com/Newsgroups/microsoft.public.sqlserver.security/2006-07/msg00083.html>

- *From:* Michael @ SGMS <MichaelSGMS@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Fri, 14 Jul 2006 09:47:01 -0700
-

This is not the case at all.

I have a stored procedure on my test database that executes a statement to retrieve data from Northwind. I am trying to limit my users ability to see any database internals related to Northwing and only let him see the data. (No object definition items)

So I create an empty database. Give the user a login and access to that database. Then I create a proc that says Select * from Northwind..Customers. It this the ownership issue your referring too.

Man this should not be so painful.

"Sue Hoegemeier" wrote:

I'm not following your example – it sounds like you may have a stored procedure in NorthwindTest that executes some SQL statement that references objects in another database named Northwind using three part name qualifier. If that's the case and you are on SP 3 or higher for SQL Server 2000, you may be hitting issues on how you have cross database ownership configured. Check books online and the topics for Cross-Database Ownership Chaining
The topic: Using Ownership Chains
explains things pretty well.

–Sue

On Fri, 14 Jul 2006 09:11:01 -0700, Michael @ SGMS
<MichaelSGMS@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote:

Thank you for your response. Your correct I am using SQL Server 2000.

My approach fails however. I cannot find a way to give the user in db1 access to the data in db2 without giving him/her a login to that database. Once they have a login, they can see all sorts of objects I would rather that

Re: Limiting a user to only executing a stored only that access an
they don't.

Am I missing something simple here.

Take the Northwind database. I created a NorthwindTest database. I took a stored proc from northwind, modified it appropriately so that its fully qualified (Northwind..blah). I cannot get the proc in NorthwindTest to execute properly without the user (who I only want to have a login to NorthwindTest) login to Northwind.

"Arnie Rowland" wrote:

I assume that you are using SQL 2000.

Your suggestion of creating a db that only contains sprocs (no data) and accesses a second db could serve your purpose. Be sure that the 'general login' does not exists in the db containing data. Any activity that requires admin privileges, such as using xp_cmdshell won't work unless the 'general login' has admin privileges in the data db –therefore defeating your 'scheme'.

They will be able to access the sproc definitions so this will not allow you to obfuscate object names –if that is of any importance. Encryption would help with that.

It sounds like you are providing them access through your firewall. Have you taken precautions such as a firewall rule that restricts access to the IP ranges from the vendor? (If not, the world will be testing their cracking tools on your server.)

—
Arnie Rowland*
"To be successful, your heart must accompany your knowledge."

"Michael @ SGMS"
<MichaelSGMS@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message

Re: Limiting a user to only executing a stored only that access an

news:560B14F2-3874-441C-AAC9-F30A35EC23B4@xxxxxxxxxxxxxxxxxxx

We deal with third party vendors that we are required to grant access to data in our database. They have Enterprise Manager, so when a general login in is created they can see iyr views tables, stored procs, and any other object.
(I know about encryption, but I want to go further than that)

What I would like is to create another database, that has only stored procedures in it that access data in other dbs. Can this be done? Is there an easier way to meet my objective? I don't want third parties looking at anything regardless of the tool they are using.

In the near future we will make our data that is public available through a Web service, but in the meantime is there an answer?