

## Re: Tracing user & permissions (triggers and stored p)

---

*Source:* <http://www.derkeiler.com/Newsgroups/microsoft.public.sqlserver.security/2006-01/msg00146.html>

---

- *From:* "Guillem Codina" <[GuillemCodina@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:GuillemCodina@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Wed, 25 Jan 2006 00:50:06 -0800
- 

Dear Dan,

Thanks for your help. You just guided me on the right direction; now I can correctly execute the xp\_cmdshell command in a stored procedure wich was called from a trigger. To solve my problem I chosed to configure the SQL Agent Proxy Account for non-sysadmin users.

I did not choose the alternative of inserting the data into staging tables and scheduling a periodic SQL Agent job to transfer the file to the communications daemon because I use some of the stored procedure's variables to generate the file name (the Autonomous Port of Barcelona uses a quite complex flat file naming standard) which needs to be different for each vessel. This file also needs to be generated as soon as the vessel operations are finished. Nevertheless, staging tables + sql agent job seems at first a much more organized and less stressing way to perform these type of tasks.

Again, THANKS A LOT FOR YOUR HELP.

"Dan Guzman" escribió:

- > Please specify the version of SQL Server you are using. Assuming 2000,
- > there should be no difference in the Windows account used regardless of how
- > xp\_cmdshell is run (proc vs. trigger executes proc). The process launched
- > from xp\_cmdshell executes under the security context of the SQL Server
- > service account when run by a sysadmin role member. xp\_cmdshell runs under
- > the configurable SQL Agent Proxy account for non-sysadmin users. The proxy
- > account can be configured from Enterprise Manager under Management-->SQL
- > Server Agent-->Properties-->Job System.
- >
- > There are issues when running BCP from a trigger. A trigger always executes
- > in the context of a SQL Server transaction. Consequently, modified data
- > will be locked and cannot be accessed by the external BCP utility unless
- > NOLOCK is specified. Also, long-running transactions are bad for both
- > concurrency and performance.

Re: Tracing user & permissions (triggers and stored p)

>  
> I suggest you consider an alternative approach for your requirements. A  
> common practice is to insert the needed data into staging tables and  
> schedule a periodic SQL Agent job to generate the files.  
>  
> --  
> Hope this helps.  
>  
> Dan Guzman  
> SQL Server MVP  
>  
> "Guillem Codina" <Guillem.Codina@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message  
> [news:9156ECAF-E52B-4044-859F-A693F08C6109@xxxxxxxxxxxxxxxxxxxx](mailto:news:9156ECAF-E52B-4044-859F-A693F08C6109@xxxxxxxxxxxxxxxxxxxx)  
> > To whom might be able to help me:  
> >  
> > STORY:  
> > I work in a stevedor terminal in the Autonomous port of Barcelona.  
> > Every time operations on a vessel are closed (instruction from software)  
> > an  
> > sql Trigger executes a Stored Procedure passing it some variables.  
> > The Stored Procedure ends generating flat files which will then be  
> > captured  
> > by a Daemon who sends them by FTP to a client for later processing.  
> >  
> > The flat files are generated using the following command:  
> >  
> > Exec master.dbo.xp\_cmdshell @cmd, no\_output  
> >  
> > @cmd is a variable which will store a value like this:  
> >  
> > BCP "Select \* From terminal.Temp\_BCP" queryout  
> > "P:\CONTXBUQ.A59562454.ESQ0817002I.20060120111821.X.EDIPLANO.N.msg.txt" -Sxxxxxxx  
> > -Usa -Pxxxxxxxxxx -c  
> >  
> > PROBLEM:  
> > The exec comand ends on error:  
> >  
> > SQLState = 08001, NativeError = 17  
> > Error = [Microsoft][ODBC SQL Server Driver][Shared Memory]No existe el  
> > servidor SQL Server o se ha denegado el acceso al mismo.  
> > SQLState = 01000, NativeError = 2  
> > Warning = [Microsoft][ODBC SQL Server Driver][Shared Memory]ConnectionOpen  
> > (Connect()).  
> >  
> > I am certain that parameters -S -U and -P are correct as the whole exec  
> > command.  
> >  
> > I guess that the key to the problem is "permissions". When I execute  
> > from  
> > QueryAnalyzer the permissions are the ones from the user who has logged  
> > into

Re: Tracing user & permissions (triggers and stored p)

>> SQL or WINDOWS (depending on the security system used).  
>> Stored Procedure's permissions must work different, specially when its  
>> execution comes from a Trigger. I would like to know how to check for the  
>> chain of permissions to see what user is really executing the xp\_cmdshell.  
>> SQL Server xp's must have very restrictive policies since many of them can  
>> access server resources. Do you believe that the user executing the  
>> cmdshell  
>> is by default LocalSystem? If so, how can I change this user? are there  
>> any  
>> parameters for this purpose? what user could perform with no problems  
>> this  
>> cmdshell?  
>>  
>>  
>> Thanks a lot in advance.  
>>  
>  
>  
>  
.

---

• **Follow-Ups:**

- ◆ **Re: Tracing user & permissions (triggers and stored p)**  
◇ From: Dan Guzman

• **References:**

- ◆ **Re: Tracing user & permissions (triggers and stored p)**  
◇ From: Dan Guzman

- Prev by Date: **Re: How to stop Login attempts from cerain IP's**
- Next by Date: **Is there a possibility to be hacked**
- Previous by thread: **Re: Tracing user & permissions (triggers and stored p)**
- Next by thread: **Re: Tracing user & permissions (triggers and stored p)**
- Index(es):
  - ◆ **Date**
  - ◆ **Thread**