**Xp\_cmdshell and permissions**

What do you need to do in order to use xp\_cmdshell?

Note that there are obvious security implications of doing this. (I'm not recommending usage of xp\_cmdshell in general, this is a technical blog post!) We first need to think about what happens here, from an architectural level:

Somebody who has logged in to SQL Server executes xp\_cmdshell. For this, SQL Server need to spawn a process in Windows. A process in Windows need to execute in a Windows user account.

So, what Windows account is used? If the SQL Server login who is executing xp\_cmdshell is sysadmin, then SQL Server will use the service account (it will not "pretend to be somebody else"). But if the login isn't sysadmin, then we need to configure what Windows account to be used (using sp\_xp\_cmdshell\_proxy\_account). Note that this configuration is the same for all non-sysadmins!

But there's a little bit more to it. Below is an outline of what need to be done. Step 2 and 3 are only needed if the one who is to execute xp\_cmdshell isn't sysadmin. Note that the steps don't have to be performed in the order listed below.

We need to allow usage of xp\_cmdshell in general (on 2005). Use "Surface Area Configuration" or sp\_configure for this.

We need to have a user in the master database which has execute permission on xp\_cmdshell. If you are uncertain about the difference between logins and users, you should read up about it in BOL.

We need to say what Windows account should be used when a non-sysadmin user is executing xp\_cmdshell.

So, here's the TSQL script that does all above:

--1, -- To allow advanced options to be changed.

EXEC sp\_configure 'show advanced options', 1;

GO

-- To update the currently configured value for advanced options.

RECONFIGURE;

GO

-- To enable the feature.

EXEC sp\_configure 'xp\_cmdshell', 1;

GO

-- To update the currently configured value for this feature.

RECONFIGURE;

GO

--2, grant permission to xp\_cmdshell

USE master

CREATE LOGIN JohnDoe WITH PASSWORD = 'jlkw#.6('

--Note, we are in the master database!!!

CREATE USER JohnDoe FROM LOGIN JohnDoe

--Run as login x

EXECUTE AS login = 'JohnDoe'

--Below fails, no execute permission on xp\_cmdshell

EXEC xp\_cmdshell 'DIR C:\\*.\*'

REVERT

GO

--Note, we are in the master database!!!

GRANT EXECUTE ON xp\_cmdshell TO JohnDoe

--Try again

EXECUTE AS login = 'JohnDoe'

--Execution of xp\_cmdshell is allowed.

--But I haven't configured the proxy account...

EXEC xp\_cmdshell 'DIR C:\\*.\*'

REVERT

GO

--3, specify the proxy account for non-syadmins

--Replace obvious parts!

EXEC sp\_xp\_cmdshell\_proxy\_account 'Domain\WinAccount','pwd'

EXECUTE AS login = 'JohnDoe'

--Execution of xp\_cmdshell is allowed.

--And executes successfully!!!

EXEC xp\_cmdshell 'DIR C:\\*.\*'

REVERT

--Cleanup

EXEC sp\_xp\_cmdshell\_proxy\_account null

DROP USER JohnDoe

DROP LOGIN JohnDoe

EXEC sp\_configure 'xp\_cmdshell', 0

RECONFIGURE