

PowerShell query remote SQL Server with different Windows Credentials

Asked 4 years, 2 months ago Active 4 years, 2 months ago Viewed 2k times



0



I am trying to query a SQL Server using a script block and passing in a different set of windows credentials. The script works if i run it on the same machine as the SQL Server but fails when i try and run it remotely. I know i can access the SQL server remotely because i can access the SQL server with SQL Server Managment Studio and run the query manually. Below is the code i am trying to use. I specify the server, database, and query prior to the point in the script.

```
$credentials = Get-Credentials
#Script block
$sqlscript = {
    #SQL variables
    $sqlDataSource = $sqlServer
    $sqlDatabase = $database

    $connectionString = "Server="+$sqlDataSource+";Database="+$sqlDatabase+";Integrated
Security=SSPI;"

    $connection = New-Object System.Data.SqlClient.SqlConnection
    $connection.ConnectionString = $connectionString

    $connection.Open()
    $command = New-Object System.Data.SqlClient.SqlCommand($queryString,$connection)
    $command.CommandTimeout=$queryTimeout

    # use dataset to get sql results
    $dataset=New-Object system.Data.DataSet
    $dataAdapter=New-Object system.Data.SqlClient.SqlDataAdapter($command)
    [void]$dataAdapter.fill($dataset)
    $connection.close()

    # dataset has separate results tables
    #$versionData = $dataset.Tables[0] #Version query results #not needed
    $hardwareData = $dataset.Tables[1] #Hardware query results
    $softwareData = $dataset.Tables[2] #Software query results

}
start-job -ScriptBlock $sqlscript -Credential $credentials
# output to separate CSV files
#$versionData | Export-CSV $outputPath -notype #not needed
$hardwareData | Export-CSV $outputPathHardware -notype
$softwareData | Export-CSV $outputPathSoftware -notype
```

This is the status of the job:

Id	Name	PSJobTypeName	State	HasMoreData	Location	Command
7	Job7	BackgroundJob	Failed	False	localhost	...

[sql-server](#) [powershell](#) [authentication](#) [windows-authentication](#) [scriptblock](#)

Join Stack Overflow to learn, share knowledge, and build your career.

Sign up



Follow



T-Heron

4,754 7 22 44



seanmt13

242 4 15

2 Answers

Active Oldest Votes



1

Ahh the joys of second-hop remoting. Basically what's going wrong here is that after creating a remote powershell session you cannot access any resources on servers other than the one you are connected to since your credentials are not actually passed to the second server when you log in. The only real way around this is to use [CredSSP authentication](#), which you can do using [Enable-WSManCredSSP](#). Make sure to read over the links as there are some specific requirements and potential issues introduced when using CredSSP compared with standard auth.

Share Improve this answer Follow

answered Nov 28 '16 at 20:13



Mike Garuccio

2,258 1 7 18

Maybe i am missing something. But i am not trying to start a remote session. I am creating a new session on the same machine as the PowerShell script. I am passing a different set of Windows Credentials than what i am logged in with to that session. If i remove the script block and just log into the machine with windows credentials that have access to the SQL server it works just fine. Even if i run the script from different machine than the SQL Server. – [seanmt13](#) Nov 28 '16 at 21:51

It doesn't matter if the session is remote or local. The standard authentication for a session which was provided credentials does not pass along the username and password, so they cannot be used to connect to another system. – [Mike Garuccio](#) Nov 28 '16 at 22:09

Actually credentials do get passed when you invoke Start-Job. That is the whole point of using the script-block. I can call that with a different credentials than the current session. Here are the links that pointed me in the right direction. [MSDN Start-Job SQL Connection with Start-Job](#) – [seanmt13](#) Nov 28 '16 at 22:57



1

So i figured out what the issue was. As i stated earlier in my comment I was correct the new session when invoking Start-Job does pass the parameters. If you look at SQL connection properties and no user/password is passed it will attempt to authenticate via windows authentication with the credentials you passed to the new session. The issue was my own fault in get-credentials i supplied the username and password as follows.

User: user@domain.local



Password: mypassword

The issue is that it needs to be like this:

User: DOMAIN\user

Password: mypassword

Join Stack Overflow to learn, share knowledge, and build your career.

[Sign up](#)

Share Improve this answer Follow

answered Nov 28 '16 at 22:52



seanmt13
242 4 15