

Lab: SID and Powershell

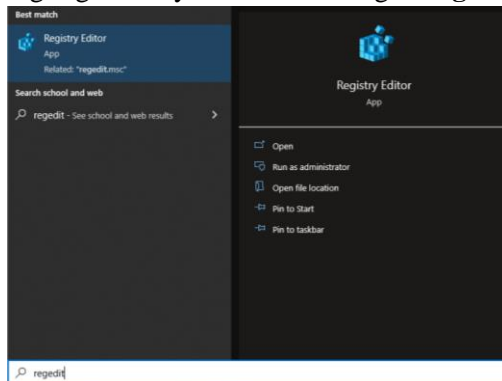
- This is worth 10 points and due tonight.
- Follow the usual naming convention (**initials of first and last**). Place your answers on the separate tasks file and submit it. DO NOT use this file for submission.
- Please **zoom in** your screenshots.

Preparation

- Read the following articles to understand how security identifiers (SIDs) work.
 - <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/understand-security-identifiers>
- Use the Proxmox server or your local VM. Logon to the server and download this file from there so that you can copy the code below.
- A tutorial about WMIC: <http://net-informations.com/q/mis/wmic.html>.
Windows Management Interface Command (WMIC) displays information about your system.

Task 1: Getting SID, SAT on Windows

- Use the following website for this task.
 - <https://www.lifewire.com/how-to-find-a-users-security-identifier-sid-in-windows-2625149>
- Logon to Windows Server 2016 VM.
 - Obtain the SID of the current login with **WMIC** command. Attach a screenshot for the SID and highlight it in yellow.
 - Obtain the SID of the current login in the Registry. Attach a screenshot for the SID and highlight it in yellow. You can get **Registry Editor** by typing **regedit** on **Search**.



Task 2: Getting SID on SQL Server

- Login to SQL Server using Windows Authentication.
- Run the following.

```
SELECT *  
FROM sys.server_principals
```

- Get the SID of the account you used for SQL Server login.
- A. SID: _____.

- Execute the following script to convert the SID of the account into the **string SID**.
- First, create the function.

```
CREATE FUNCTION fn_SIDToString
(
    @BinSID AS VARBINARY(100)
)
RETURNS VARCHAR(100)
AS BEGIN

    IF LEN(@BinSID) % 4 <> 0 RETURN(NULL)

    DECLARE @StringSID VARCHAR(100)
    DECLARE @i AS INT
    DECLARE @j AS INT

    SELECT @StringSID = 'S-'
        + CONVERT(VARCHAR, CONVERT(INT, CONVERT(VARBINARY, SUBSTRING(@BinSID, 1, 1))))
    SELECT @StringSID = @StringSID + '-'
        + CONVERT(VARCHAR, CONVERT(INT, CONVERT(VARBINARY, SUBSTRING(@BinSID, 3, 6))))

    SET @j = 9
    SET @i = LEN(@BinSID)

    WHILE @j < @i
    BEGIN
        DECLARE @val BINARY(4)
        SELECT @val = SUBSTRING(@BinSID, @j, 4)
        SELECT @StringSID = @StringSID + '-'
            + CONVERT(VARCHAR, CONVERT(BIGINT, CONVERT(VARBINARY, REVERSE(CONVERT(VARBINARY,
@val)))))
        SET @j = @j + 4
    END
    RETURN ( @StringSID )
END
```

- B. What does the function “fn_SIDToString” do?

- Next, run the function to get the SIDs.

```
SELECT SUSER_NAME(), SUSER_SID(), dbo.fn_SIDToString(SUSER_SID())
```

- C. Compare the SID from SQL Server for the administrator login with that from Windows Server for the administrator. Show the two screenshots. Use the SIDs in a string format (that is, in the S- format, not in Hex). Are they the same?
 - a. The SID of the administrator login from SQL Server (show the S-format)
 - b. The SID of the administrator login from Windows Server (show the S-format)

- Let us create logins. Run the following script.

```
CREATE LOGIN SIDTest WITH PASSWORD = 'Pa$$w0rd'
GO
```

- Get the SID of this login.

```
SELECT sid
FROM sys.server_principals
WHERE name = 'SIDTest'
GO
```

- D. SID: _____.

- Drop this login.

```
DROP LOGIN SIDTest
GO
```

- Recreate this login and identify its SID.

```
CREATE LOGIN SIDTest WITH PASSWORD = ' Pa$$w0rd'
GO
```

```
SELECT sid
FROM sys.server_principals
WHERE name = 'SIDTest'
GO
```

- E. SID: _____.

- F. Are the SIDs of login `SIDTest` the same? Describe the reason why they are (not) the same?

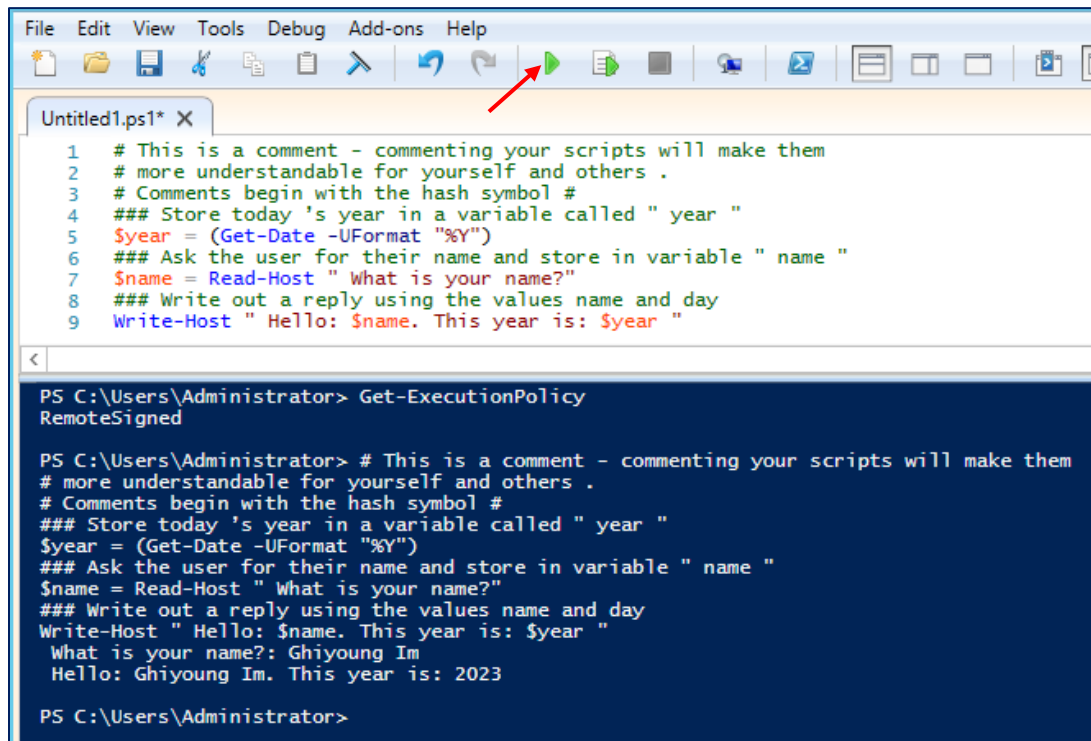
_____.

- After this, drop the login you have created.

Task 3: Learn PowerShell Scripting

- On **Windows Server**, search for “**PowerShell**.” And select **Windows PowerShell ISE**.
- Check out the execution policy by typing: **Get-ExecutionPolicy**. Remember this setting.
- You can change the execution policy by typing the following commands:
Set-ExecutionPolicy unrestricted, or **Set-ExecutionPolicy restricted**.

- First, go over the entire manual “**Getting Started with Microsoft PowerShell.pdf**” quickly. Next, complete **9.1 Exercise**. Run your script and report the outcome with a screenshot.
- You can run the script using the arrows on the menu bar.



The screenshot shows a PowerShell script editor window titled "Untitled1.ps1* X". The script contains the following code:

```
1 # This is a comment - commenting your scripts will make them
2 # more understandable for yourself and others .
3 # Comments begin with the hash symbol #
4 ### Store today 's year in a variable called " year "
5 $year = (Get-Date -UFormat "%Y")
6 ### Ask the user for their name and store in variable " name "
7 $name = Read-Host " What is your name?"
8 ### Write out a reply using the values name and day
9 Write-Host " Hello: $name. This year is: $year "
```

A red arrow points to the green play button icon in the menu bar. Below the script editor is a console window showing the execution of the script:

```
PS C:\Users\Administrator> Get-ExecutionPolicy
RemoteSigned

PS C:\Users\Administrator> # This is a comment - commenting your scripts will make them
# more understandable for yourself and others .
# Comments begin with the hash symbol #
### Store today 's year in a variable called " year "
$year = (Get-Date -UFormat "%Y")
### Ask the user for their name and store in variable " name "
$name = Read-Host " What is your name?"
### Write out a reply using the values name and day
Write-Host " Hello: $name. This year is: $year "
What is your name?: Ghiyoung Im
Hello: Ghiyoung Im. This year is: 2023

PS C:\Users\Administrator>
```