**Remote Management Tools for Windows Server 2019**

*Article by – Evan Sutherland, Student, IT Systems & Security, NSCC, Halifax NS*

*02-06-2021*

**Introduction**

Microsoft Server 2019 provides a suite of Remote Management tools that Administrators can use to facilitate day to day operations. Remote Management is enabled by default in Windows Server 2019, you just simply need to add the server you are looking to manage to the server pool in your Server Manager application.

**Useful Tools**

* **Remote Desktop Services**
* **Remote Management (Server)**
* **RSAT Tools**
* **PowerShell Session**

**Remote Desktop Services –** RDS allows the end user to remotely access the server to utilize its resources in performing tasks on the client machine. This is particularly useful when working with a thin client computer trying to perform a task that is very CPU or GPU heavy that their hardware can not support. As we will be using a File Server running Windows Datacenter this would be a tool we would want to implement as Datacenter uses a GUI environment, and this would be the server where any work from the client machines would want to be stored.

**Remote Management (Server) –** This tool manages servers both remotely and locally and can control and diagnose server hardware. It contains a scripting API as well as several command line tools such as winrm.cmd, that allows an administrator to gather data and manage server resources. It also contains WS-Management Protocol which allows systems to locate and exchange management information, useful in an enterprise setting when computers are running different operating systems. Being mostly command line tools, we would use this tool with our Primary Domain Controller, as it is running a core installation without a GUI. That being said, it is still a useful tool for managing our DataCenter installation aswell.

**RSAT Tools –** RSAT Tools are a suite of tools including but not limited to Active Directory Domain Services and Lightweight Directory Services, Group Policy Management Tools, Remote Desktop Management tools, and Server Manager. A useful tool within RSAT for our Primary Domain Controller would be AD DS. AD DS comes equipped with a command line tool and a PowerShell module useful when working with our Core install of Windows Server, and will provide services such as Rights Management, Certificate Services, Domain Services, and others.

A full list of RSAT Tools can be found at <https://docs.microsoft.com/en-us/troubleshoot/windows-server/system-management-components/remote-server-administration-tools>

**PowerShell Session –** Establishing a PowerShell Session allows a user to execute PowerShell cmdlets and scripts remotely. A PowerShell Session is the environment in which PowerShell runs, and a persistent connection is established saving any values, variable, etc. for that session. You can also use the ComputerName parameter along with a cmdlet such as Enter-PSSession to create a temporary session to run a single command without saving any data from that session on the local machine. PowerShell is a powerful tool for server administration and would be useful on both the Core and Datacenter install, however as the Core install does not come with a GUI, you would get the most use out of a PowerShell Session connecting to it.

**Example of a PowerShell Session (PSSession)**

In order to run PowerShell cmdlets and scripts on a remote machine the user must first establish a connection with that machine and begin a PSSession.

When the user wants to establish a persistent connection to a remote machine, they would use the cmdlet New-PSSession followed by the -ComputerName parameter. This will allow the user to run a series of commands and have any variables/value/etc. saved in that session

A picture containing text

Description automatically generated

The user is then given a session id. The user could also store the Session ID in a variable.



This allows the user to easily invoke that session using the *$ps1* variable.

Now to run commands in the *$ps1* session the user would run the Enter-PSSession cmdlet with the -Session parameter.

A picture containing background pattern

Description automatically generated

Notice that the name of the machine you are remoting in to on the left. This tells you that you are now running PowerShell on that specific machine, and any commands you run will execute on that machine. In this example I have run the Get-NetIpAddress cmdlet and it has returned information from the remote machine not the local machine.

To end the session, use the Exit-PsSession cmdlet. This will end the connection but any variables, values, aliases etc. will be saved in the *$ps1*(or whatever variable you saved your session as) for future use.



Notice on the left the name of the remote machine is gone. Now any commands run with be for the local machine. If the user wishes to delete the session and all its related data, they can run the Remove-PsSession cmdlet.



More information about PSSessions can be found at <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_pssessions?view=powershell-7.1>

**Related Reading & Sources**

[Hands on Microsoft Windows Server 2016](https://www.cengage.ca/c/hands-on-microsoft-windows-server-2016-44-2nd-edition-2e-palmer/9781305078628/)

[Windows Remote Management](https://docs.microsoft.com/en-us/windows/win32/winrm/about-windows-remote-management)

[Remote Server Administration Tools](https://docs.microsoft.com/en-us/troubleshoot/windows-server/system-management-components/remote-server-administration-tools)

[Active Directory Domain Services Command Reference](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/cc771131(v=ws.11))

[AD DS Overview and Functions](https://www.varonis.com/blog/active-directory-domain-services/)

[RSAT: These Windows 10 Tools Put You in Control](http://techgenix.com/rsat-tools-windows-10/)

[Running Remote Commands](https://docs.microsoft.com/en-us/powershell/scripting/learn/remoting/running-remote-commands?view=powershell-7.1)

[About PSSessions](https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_pssessions?view=powershell-7.1)

- Images provided by the author with identifying data removed -