# Offline Assessment for SQL Server

## Prerequisites

### How to prepare for your Offline Assessment for SQL Server

The tools machine is used to connect to each of the SQL servers hosts and SQL Server instances in your environment and retrieves  information from them, communicating over Remote Procedure Call (RPC), Server Message Block (SMB), NET Framework Data Provider, and Distributed Component Object Model (DCOM). Once the data is collected and the operational interview is completed, the Offline Assessment tool will analyze the data locally.  A checklist of prerequisite actions follows. Each item links to any additional software required  for the tools machine, and detailed steps included later in this document.

### Checklist

Please ensure the following items have been completed before starting your engagement.

*All data collection and analysis is done locally on the tools machine.*

*No data is transported outside your SQL Server environment to help protect your data. Your data is analyzed using our RAP expert system that is part of the Offline Assessment client.*

**1. General Use**

□ A Microsoft Account is required to activate and sign in to the portal to download the toolset. If you don't have one already, you can create one at http://login.live.com
To learn more about Microsoft Accounts, see: http://windows.microsoft.com/en-US/windows-live/sign-in-what-is-microsoft-account

□ Ensure access to https://services.premier.microsoft.com

**2. Data Collection**

a. Tools machine hardware and Operating System:

□ Server-class or high-end workstation machine running Windows 7, Windows 8, Windows 10, or Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016

***Note:*** *Windows Server 2003 is not supported as a tools machine. To successfully gather Performance data, please ensure the data collection machine's OS matches, or is a higher version of the highest versioned OS target machine used within the environment.*

□ Minimum: 8GB RAM, 2Ghz dual-core processor, 5 GB of free disk space plus up to 7 GB for every 100,000 objects in the assessed environment during data collection.

□ Joined to one of the domains of the environment to be assessed. When target instance is a standalone instance in a server with different domain or without a domain you can use the workaround mentioned in page 3, section 2.

*This document was last updated  March 28, 2018 To ensure you have the latest version of this document , check here:* http://download.microsoft.com/download/1/C/1/1C15BA51-840E-498D-86C6-4BD35D33C79E/Prerequisites_Offline_SQL.pdf

b. Software for Tools machine:

- □ Microsoft .NET Framework 4.6 installed
- □ Windows PowerShell 2.0 or later installed

c. Account Rights:

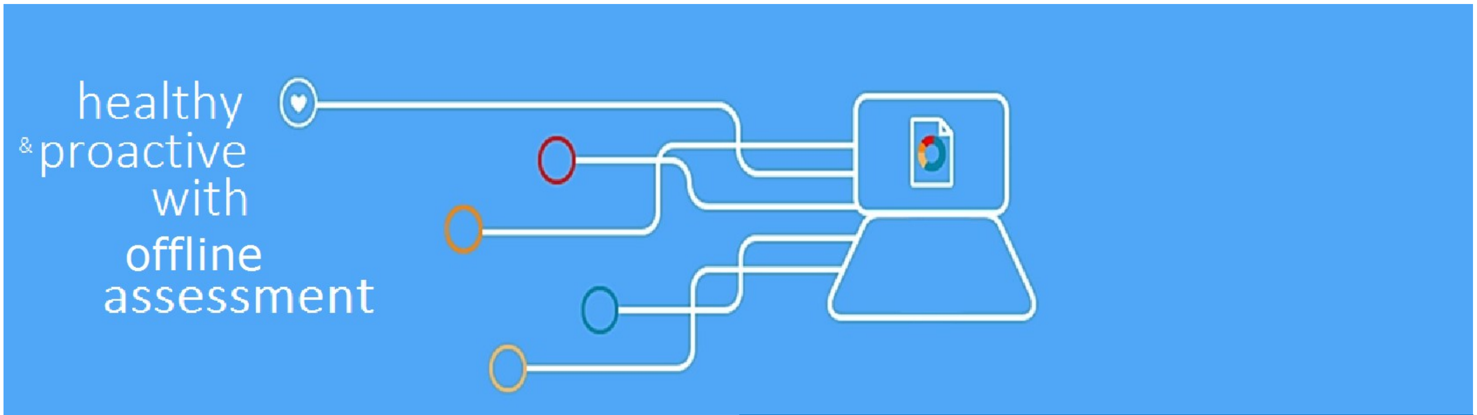
- □ Administrator permissions to all SQL server instances. If this is not possible review alternative in Appendix D. Setup procedure and limitations when not using sysadmin permissions for the toolset account
- □ Administrative access to the SQL Server hosts

d. Additional Requirements for Windows Server 2008 Servers or later:

- □ Configure the servers' firewall as described in Appendix C.
- □ Enable PSRemoting on the target servers. See Appendix E for more details.
- □ Enable PowerShell script execution on the target servers. For this, open a PowerShell window as Administrator and execute: *Set-ExecutionPolicy remotesigned*

The Appendix Data Collection Methods details the methods used to collect data.

The rest of this document contains detailed information on the steps discussed above.
Once you have completed these prerequisites, you are ready to start the Offline Assessment.



# Machine Requirements and Account Rights

## 1. Hardware and Software

Server-class or high-end workstation computer equipped with the following:

- Minimum single 2Ghz processor — Recommended dual-core/multi-core 2Ghz or higher processors.
- Minimum 4 GB RAM—Recommended 8 GB RAM.
- Minimum 5 GB of free disk space.
- Windows 10, Windows 8, Windows 7, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2 or Windows Server 2008. Windows Server 2003 is not supported as a data collection machine.

  ***Note**: To successfully gather Performance data, please ensure the data collection machine's OS matches, or is a higher version of the highest versioned OS target machine used within the environment.*

- Can be 32-bit or 64-bit operating system.
- At least a 1024x768 screen resolution (higher preferred).
- A member of the same domain as the SQL Server hosts or a member of a trusted domain.
- Microsoft® .NET Framework 4.6.— https://www.microsoft.com/en-gb/download/details.aspx?id=48130
- Windows PowerShell 2.0 or higher
  - Windows PowerShell 2.0 is part of the Windows Management Framework — http://support.microsoft.com/kb/968929
- Networked "Documents" or redirected "Documents" folders are not supported. Local "Documents" folder on the data collection machine is required.

## 2. Accounts Rights

- A domain account with the following:
  - Local administrator permissions to tools machine and all SQL server hosts to be assessed.
  - Administrative access to the SQL server instances (member of SysAdmin Role). If this is not possible review alternative procedure in Appendix D. Setup procedure and limitations when not using sysadmin permissions for the toolset account.
- If the client and target server are NOT the same domain:
  - Configure Pass through authentication between the client and the target server. Basically the local user name and password used should be the same on the client and the target server.
  - The Pass through authentication account should have the following:
    - Local administrator permissions to tools machine and all SQL server hosts to be assessed.
    - Administrative access to the SQL server instances (member of SysAdmin Role). If this is not possible review alternative in Appendix D. Setup procedure and limitations when not using sysadmin permissions for the toolset account.

**WARNING**: Do not use the "Run As" feature to start the client toolset as the discovery process and collectors might fail. The account starting the client toolset must logon to the local machine.

### 3. Network and Remote Access

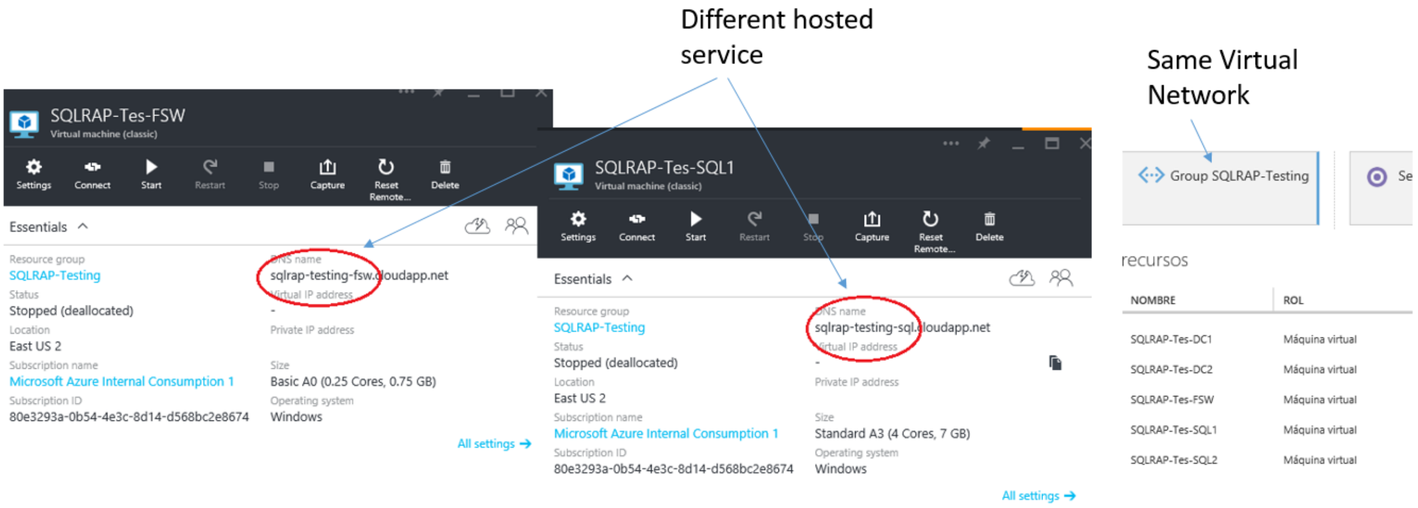
- Unrestricted network access to every server in the environment
    - This means access through any firewalls, and router ACLs that might be limiting traffic to any server. This includes remote access to DCOM, Remote Registry service, Windows Management Instrumentation (WMI) services, and default administrative shares (C$, D$, IPC$).
    - The following services must be started on the target SQL Server hosts:
        - WMI
        - Remote Registry service
        - Server service
        - Workstation service
        - File and Printer Sharing service
        - Automatic Updates service
        - Performance Logs and Alerts service
    - For Availability Groups in Azure follow the requirements in **Appendix A. Special Requirement for Availability Group Cluster in Azure**
    - If a Firewall exist (Windows or Hardware) between the tools machine and the target server follow steps and recommendations on Appendix C: Firewall Requirements.
    - PSRemoting needs to be enabled on target servers. In most environments it is already enabled. See Appendix E for details on how to enable.

.

## Appendix A: Special Requirements for Availability Group Cluster in Azure

Virtual networks in Azure put some connectivity restrictions to clusters and availability groups that affect the toolset. In summary:

1. You cannot use the listener name for discovery. You can use the cluster name or a node name.
2. If you are using cloud services (discontinued in Azure Resource Manager for IaaS) and you are using an External load balancer (internal load balancer does not have this limitation), then you need to have the tools machine in a different cloud service as shown in this image.



Different hosted service

Same Virtual Network

SQLRAP-Tes-FSW
Virtual machine (classic)

Resource group
SQLRAP-Testing
Status
Stopped (deallocated)
Location
East US 2
Subscription name
Microsoft Azure Internal Consumption 1
Subscription ID
80e3293a-0b54-4e3c-8d14-d568bc2e8674

DNS name
sqlrap-testing-fsw.cloudapp.net
Virtual IP address
-
Private IP address
Size
Basic A0 (0.25 Cores, 0.75 GB)
Operating system
Windows

SQLRAP-Tes-SQL1
Virtual machine (classic)

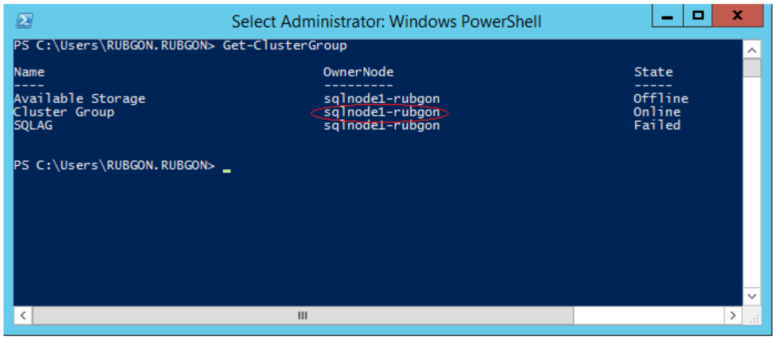
Resource group
SQLRAP-Testing
Status
Stopped (deallocated)
Location
East US 2
Subscription name
Microsoft Azure Internal Consumption 1
Subscription ID
80e3293a-0b54-4e3c-8d14-d568bc2e8674

DNS name
sqlrap-testing-sql.cloudapp.net
Virtual IP address
-
Private IP address
Size
Standard A3 (4 Cores, 7 GB)
Operating system
Windows

recursos

| NOMBRE | ROL |
|---|---|
| SQLRAP-Tes-DC1 | Máquina virtual |
| SQLRAP-Tes-DC2 | Máquina virtual |
| SQLRAP-Tes-FSW | Máquina virtual |
| SQLRAP-Tes-SQL1 | Máquina virtual |
| SQLRAP-Tes-SQL2 | Máquina virtual |

3. You need to modify the hosts file to make the Cluster Collectors work. You need to identify the IP of the active node and modify the hosts files as shown below. The hosts file is located at “C:\Windows\System32\drivers\etc”.

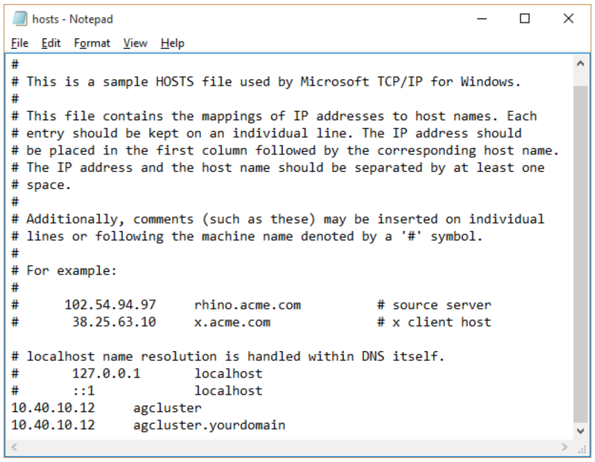


Select Administrator: Windows PowerShell

```
PS C:\Users\RUBGON.RUBGON> Get-ClusterGroup

Name                                    OwnerNode                                 State
----                                    ---------                                 -----
Available Storage                       sqlnode1-rubgon                           Offline
Cluster Group                           sqlnode1-rubgon                           Online
SQLAG                                   sqlnode1-rubgon                           Failed

PS C:\Users\RUBGON.RUBGON> _
```



hosts - Notepad

```
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#       127.0.0.1       localhost
#       ::1             localhost
10.40.10.12     agcluster
10.40.10.12     agcluster.yourdomain
```

**Note:** The hosts configuration change needs to be reviewed every time you run the toolset in case a failover has happened since the last time the toolset was executed.

## Appendix B: Data Collection Methods

Offline Assessment for SQL Server uses multiple data collection methods to collect information. This section describes the methods used to collect data from a SQL Server environment. No VB scripts are used to collect data. Data collection uses workflows and collectors. The collectors are:

1. Registry Collectors
2. FileDataCollector
3. WMI
4. TSQL Data collector
5. EventLogCollector
6. Windows PowerShell
7. SQL Error Log collection
8. Local Security Policy
9. Performance Monitor Counters data

### 1. Registry Collectors

Registry keys and values are read from the Offline Assessment for SQL Server data collection machine and all SQL Servers. They include items such as:

- Service information from HKLM\SYSTEM\CurrentControlSet\Services.

  This allows to determine where the SQL Server instances installed on given server or failover cluster and get detailed information on each service relevant to the proper function of SQL Server. We do not collect all services, only the ones relevant to SQL.

- Operating System information from HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion

  This allows to determine Operation System information such as Windows Server 2003, Windows Server 2008 or Windows Server 2012.

### 2. FileDataCollector

Enumerates files in a folder on a remote machine, and optionally retrieves those files.

### 3. Windows Management Instrumentation (WMI)

WMI is used to collect various information such as:

- WIN32_Volume

  Collects information on Volume Settings for each SQL Server. The information is used for instance to determine the system volume and drive letter which allows Offline Assessment to collect information on database files located on the system drive. And other drives.

- Win32_LogicalDisk

  Used to collect information on the logical disks. We use the information to determine the amount of free space on the disk where the database or log files are located.

### 4. TSQL Data Collector

- Queries against system tables

  Offline Assessment for SQL Server collects information from target SQL Server instance system tables using the T-SQL data collector. The information includes but not limited to database backups, Log shipping information, databases participated

**5. EventLogCollector**

Collects event logs from SQL Server hosts. We collect the last 7 days of Warnings and Errors from the Application, and System event logs.

**6. Windows PowerShell**

Collects various information, such as:

- Failover Cluster resource dependencies.

**7. SQL Error Log collection**

Collects SQL Server error log data for the last 15 days or for 6MB of size.

**8. Local Security Policy**

Local Policies, which is part of the Local Security Settings console, determine the security options for a user or service account

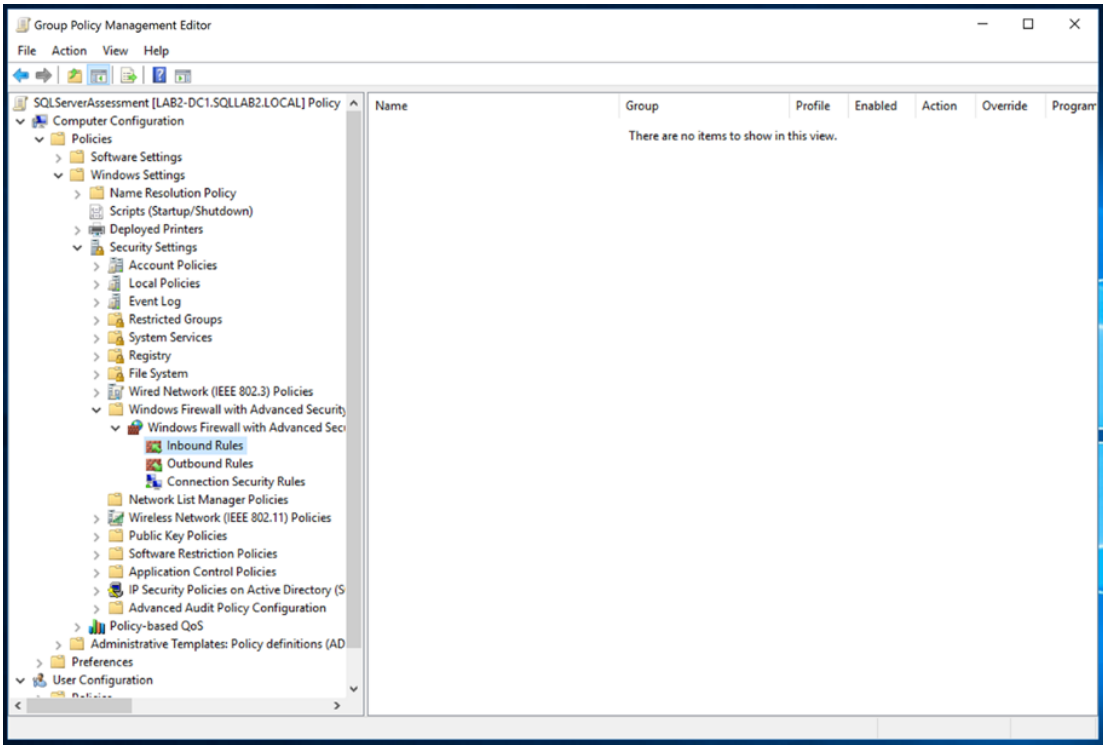
**9. Performance Monitor Counters data**

System and SQL Server instance related performance counters.

## Appendix C: Firewall Requirements

If you have a firewall between the tools machine and target servers and/or you are using Windows Firewall you need to open traffic between Tools Machine and Target Servers. The following steps apply to Windows Firewall but the same traffic needs to be open for any other firewall that is in the middle of Tools Machines and Target Servers.

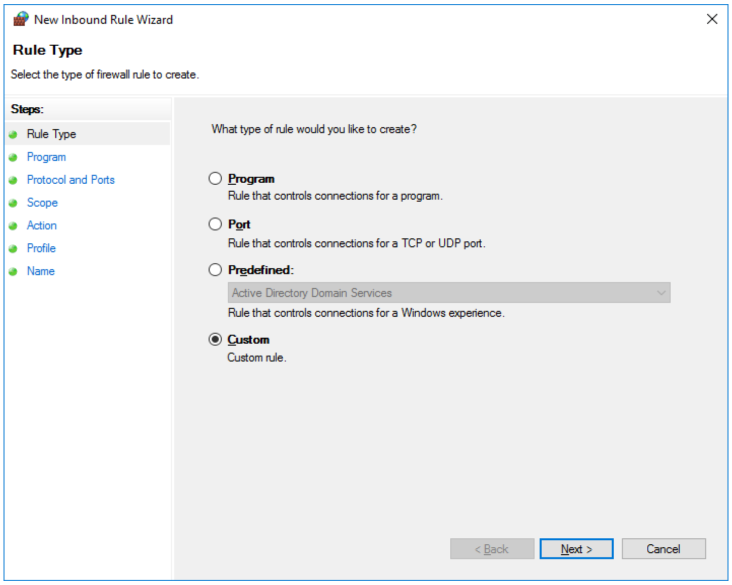
1. Create an advanced Inbound Firewall Rule to allow all network traffic from the tools machine to the SQL Servers. This can also be the applied through GPO.


Group Policy Management Editor
File
Action
View
Help
SQLServerAssessment [LAB2-DC1.SQLLAB2.LOCAL] Policy
Computer Configuration
Policies
Software Settings
Windows Settings
Name Resolution Policy
Scripts (Startup/Shutdown)
Deployed Printers
Security Settings
Account Policies
Local Policies
Event Log
Restricted Groups
System Services
Registry
File System
Wired Network (IEEE 802.3) Policies
Windows Firewall with Advanced Security
Windows Firewall with Advanced Secu
Inbound Rules
Outbound Rules
Connection Security Rules
Network List Manager Policies
Wireless Network (IEEE 802.11) Policies
Public Key Policies
Software Restriction Policies
Application Control Policies
IP Security Policies on Active Directory (S
Advanced Audit Policy Configuration
Policy-based QoS
Administrative Templates: Policy definitions (AD
Preferences
User Configuration
Policies
Name
Group
Profile
Enabled
Action
Override
Program

There are no items to show in this view.

4. To create the new rule, Right Click on “Inbound Rules” and select “New”

5. Create a custom rule and choose “Next”



**Rule Type**

Select the type of firewall rule to create.

What type of rule would you like to create?

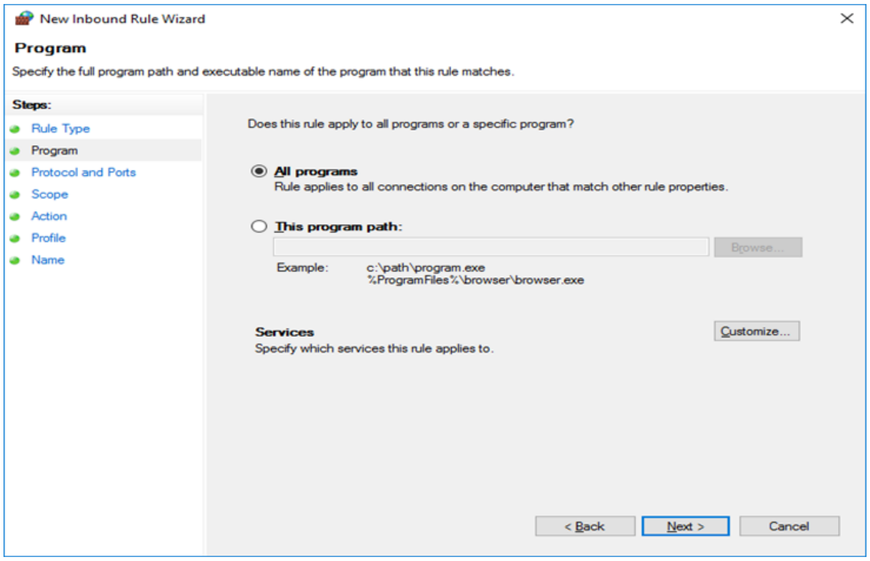
○ **Program**
Rule that controls connections for a program.

○ **Port**
Rule that controls connections for a TCP or UDP port.

○ **Predefined:**
Active Directory Domain Services
Rule that controls connections for a Windows experience.

● **Custom**
Custom rule.

6. Allow “All programs” from the tools machine and click “Next”.



New Inbound Rule Wizard

**Program**

Specify the full program path and executable name of the program that this rule matches.

Does this rule apply to all programs or a specific program?

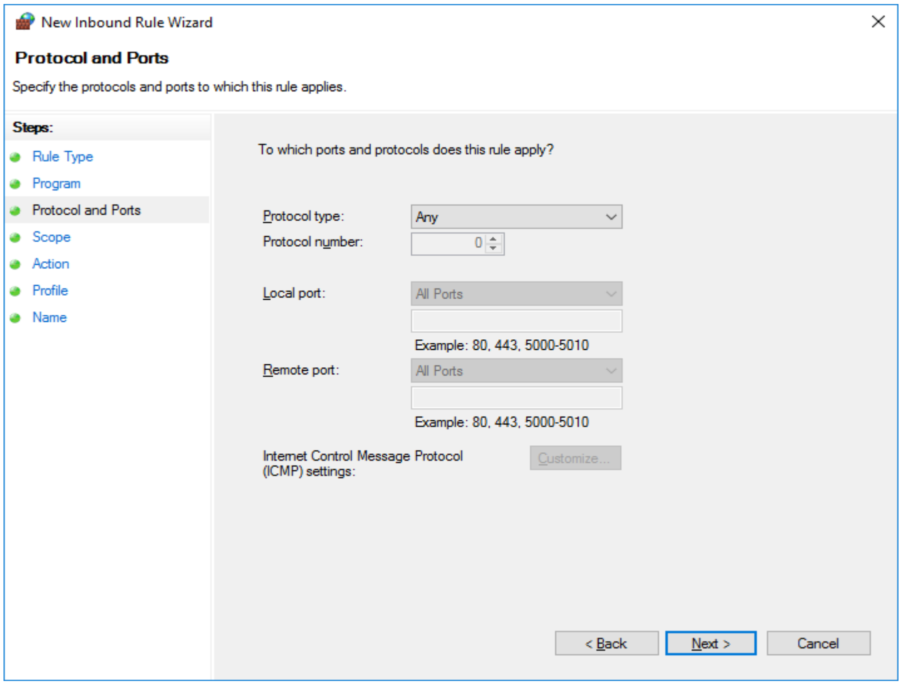
● **All programs**
Rule applies to all connections on the computer that match other rule properties.

○ **This program path:**

Example: c:\path\program.exe
%ProgramFiles%\browser\browser.exe

**Services**
Specify which services this rule applies to.

7. Allow all protocols and ports, then click “Next”.



New Inbound Rule Wizard

**Protocol and Ports**

Specify the protocols and ports to which this rule applies.

To which ports and protocols does this rule apply?

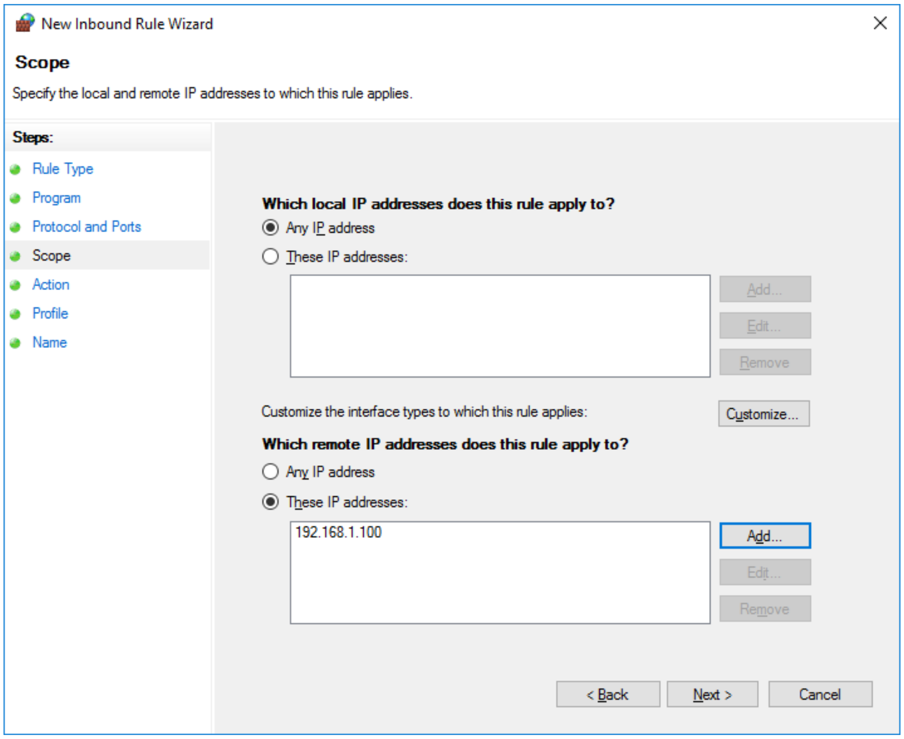
Protocol type: Any
Protocol number: 0

Local port: All Ports
Example: 80, 443, 5000-5010

Remote port: All Ports
Example: 80, 443, 5000-5010

Internet Control Message Protocol (ICMP) settings:

8. Specify the IP address of the tools machine and click “Next”.



New Inbound Rule Wizard

Scope

Specify the local and remote IP addresses to which this rule applies.

Steps:

- Rule Type
- Program
- Protocol and Ports
- Scope
- Action
- Profile
- Name

Which local IP addresses does this rule apply to?

● Any IP address

○ These IP addresses:

Customize the interface types to which this rule applies:

Which remote IP addresses does this rule apply to?

○ Any IP address

● These IP addresses:

192.168.1.100

9. Choose to “Allow the connection” and click Next

10. Choose to select network profile “Domain” and click “Next”

11. Choose a name for the rule (Example: SQLAssessmentToolsMachine)

## Appendix D: Setup procedure and limitations when not using sysadmin permissions for the toolset account

If granting sysadmin privileges is not possible, you can use the script provided below to grant only necessary privileges to the account running the assessment. Consider that a few rules can only collect the necessary data when running with sysadmin privilege. As a consequence, these rules will be skipped.

In the case of read-only databases (other than availability group databases), when using the script below, the necessary user won't be created and several rules will be skipped as a result. To include these read-only databases in the analysis, you need to temporarily change the database mode to READ_WRITE mode, then run the script provided below to create the necessary user and permissions. After this, you can set the database mode back to READ_ONLY and run the toolset. The alternative way to include the read-only databases in the analysis is to run the toolset as a sysadmin user.

The permission script grants permissions that usually are already granted to public role by default, to address the case where the default permissions have been revoked from public role as a result of SQL Server hardening.

The script to  create the login and grant permissions is provided below. It needs to be run in every target SQL Server instance. The script spans several pages.

```
DECLARE @UserName nvarchar(255) = 'DomainName\RaaSUser', --replace with your domain and username, the user needs to exist in the domain
                   @Command nvarchar(max)

SET @Command = 'USE master;
          CREATE LOGIN [' + @UserName + '] FROM WINDOWS WITH DEFAULT_DATABASE=[master], DEFAULT_LANGUAGE=[us_english];'
EXEC sp_executesql @Command

--Create user on each database
SET @Command = 'USE [?];
IF EXISTS (SELECT 1
                   FROM sys.databases d LEFT JOIn sys.dm_hadr_database_replica_states r ON d.database_id = r.database_id
                   WHERE d.is_read_only = 0
                   AND (r.is_primary_replica = 1 OR r.is_primary_replica IS NULL)
                   AND d.name = DB_NAME()
)
          CREATE USER [' + @UserName + '] FOR LOGIN [' + @UserName + '];
'
EXECUTE master.sys.sp_MSforeachdb @Command
--master permissions
SET @Command = '
          USE master;
          GRANT VIEW SERVER STATE TO [' + @UserName + ']
          GRANT VIEW ANY DEFINITION TO [' + @UserName + ']
          GRANT SELECT ON sys.master_files TO [' + @UserName + ']
          GRANT SELECT ON sys.databases TO [' + @UserName + ']
          GRANT SELECT ON sys.configurations TO [' + @UserName + ']
          GRANT SELECT ON sys.sql_logins TO [' + @UserName + ']
          GRANT SELECT ON sys.server_principals TO [' + @UserName + ']
          GRANT SELECT ON sys.server_role_members TO [' + @UserName + ']
          GRANT SELECT ON sys.endpoints TO [' + @UserName + ']
          GRANT SELECT ON sys.database_mirroring_endpoints TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_os_loaded_modules TO [' + @UserName + ']
          GRANT SELECT ON sys.servers TO [' + @UserName + ']
          GRANT SELECT ON sys.server_audits TO [' + @UserName + ']
          GRANT SELECT ON sys.server_event_sessions TO [' + @UserName + ']
          GRANT SELECT ON sys.tcp_endpoints TO [' + @UserName + ']
          GRANT SELECT ON sys.database_mirroring TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_db_index_usage_stats TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_os_performance_counters TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_os_sys_info TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_os_nodes TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_os_schedulers TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_db_partition_stats TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_db_persisted_sku_features TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_db_missing_index_details TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_db_missing_index_groups TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_db_missing_index_group_stats TO [' + @UserName + ']
          GRANT SELECT ON sys.dm_xe_sessions TO [' + @UserName + ']
```

```
        GRANT SELECT ON sys.dm_exec_query_stats TO [' + @UserName + ']
        GRANT SELECT ON sys.dm_exec_text_query_plan TO [' + @UserName + ']
        GRANT SELECT ON sys.dm_exec_sql_text TO [' + @UserName + ']
        GRANT SELECT ON sys.dm_os_wait_stats TO [' + @UserName + ']
        GRANT SELECT ON sys.dm_exec_connections TO [' + @UserName + ']
'
EXECUTE master.sys.sp_MSforeachdb @Command

SET @Command = 'GRANT EXEC ON sys.xp_enumerrorlogs TO [' + @UserName + ']
        GRANT EXEC ON sys.sp_executesql TO [' + @UserName + ']
        GRANT EXEC ON sys.sp_validatelogins TO [' + @UserName + ']

        --For SQL Server 2012 or later
        IF CONVERT(int,SUBSTRING(CONVERT(varchar,SERVERPROPERTY(''ProductVersion'')), 1, 2)) >= 11
        BEGIN
                GRANT SELECT ON sys.availability_groups TO [' + @UserName + ']
                GRANT SELECT ON sys.availability_replicas TO [' + @UserName + ']
                GRANT SELECT ON sys.availability_group_listener_ip_addresses TO [' + @UserName + ']
                GRANT SELECT ON sys.availability_group_listeners TO [' + @UserName + ']
                GRANT SELECT ON sys.dm_hadr_availability_replica_states TO [' + @UserName + ']
                GRANT SELECT ON sys.dm_db_stats_properties TO [' + @UserName + ']
                GRANT SELECT ON sys.dm_hadr_availability_group_states TO [' + @UserName + ']
                GRANT SELECT ON sys.dm_hadr_database_replica_states TO [' + @UserName + ']
        END

        --For SQL 2017 or later
        IF CONVERT(int,SUBSTRING(CONVERT(varchar,SERVERPROPERTY(''ProductVersion'')), 1, 2)) >= 14
        BEGIN
                GRANT SELECT ON sys.dm_db_log_info TO [' + @UserName + ']
        END'
EXECUTE master.sys.sp_MSforeachdb @Command

--msdb permissions
SET @Command = '
        USE msdb
        GRANT SELECT ON dbo.backupmediafamily TO [' + @UserName + ']
        GRANT SELECT ON dbo.backupset TO [' + @UserName + ']
        GRANT SELECT ON dbo.backupfile TO [' + @UserName + ']
        GRANT SELECT ON dbo.backupmediaset TO [' + @UserName + ']
        GRANT SELECT ON dbo.restorefile TO [' + @UserName + ']
        GRANT SELECT ON dbo.restorefilegroup TO [' + @UserName + ']
        GRANT SELECT ON dbo.restorehistory TO [' + @UserName + ']
        GRANT SELECT ON dbo.sysdbmaintplans TO [' + @UserName + ']
        GRANT SELECT ON dbo.log_shipping_monitor_secondary TO [' + @UserName + ']
        GRANT SELECT ON dbo.log_shipping_secondary_databases TO [' + @UserName + ']
        GRANT SELECT ON dbo.log_shipping_secondary TO [' + @UserName + ']
        GRANT SELECT ON dbo.log_shipping_monitor_primary TO [' + @UserName + ']
        GRANT SELECT ON dbo.log_shipping_primary_databases TO [' + @UserName + ']
        GRANT SELECT ON dbo.sysjobs TO [' + @UserName + ']
        GRANT SELECT ON dbo.sysjobhistory TO [' + @UserName + ']
        GRANT SELECT ON dbo.suspect_pages TO [' + @UserName + ']

        IF EXISTS(SELECT 1 FROM sys.objects WHERE name = ''MSdistributiondbs'')
                GRANT SELECT ON dbo.MSdistributiondbs TO [' + @UserName + ']
'
EXECUTE master.sys.sp_MSforeachdb @Command

--user databases permissions
SET @Command = 'USE [?];
        IF EXISTS (SELECT 1
                        FROM sys.databases d LEFT JOIn sys.dm_hadr_database_replica_states r ON d.database_id = r.database_id
                        WHERE d.is_read_only = 0
                        AND (r.is_primary_replica = 1 OR r.is_primary_replica IS NULL)
                        AND d.name = DB_NAME()
        )
        BEGIN
                GRANT SELECT ON sys.foreign_keys TO [' + @UserName + ']
                GRANT SELECT ON sys.database_files TO [' + @UserName + ']
                GRANT SELECT ON sys.allocation_units TO [' + @UserName + ']
                GRANT SELECT ON sys.extended_properties TO [' + @UserName + ']
                GRANT SELECT ON sys.objects TO [' + @UserName + ']
                GRANT SELECT ON sys.partitions TO [' + @UserName + ']
```

```
                   GRANT SELECT ON sys.schemas TO [' + @UserName + ']
                   GRANT SELECT ON sys.indexes TO [' + @UserName + ']
                   GRANT SELECT ON sys.internal_tables TO [' + @UserName + ']
                   GRANT SELECT ON sys.database_principals TO [' + @UserName + ']
                   GRANT SELECT ON sys.all_objects TO [' + @UserName + ']
                   GRANT SELECT ON sys.database_permissions TO [' + @UserName + ']
                   GRANT SELECT ON sys.database_role_members TO [' + @UserName + ']
                   GRANT SELECT ON sys.symmetric_keys TO [' + @UserName + ']
                   GRANT SELECT ON sys.asymmetric_keys TO [' + @UserName + ']
                   GRANT SELECT ON sys.assembly_modules TO [' + @UserName + ']
                   GRANT SELECT ON sys.assemblies TO [' + @UserName + ']
                   GRANT SELECT ON sys.assembly_types TO [' + @UserName + ']
                   GRANT SELECT ON sys.xml_indexes TO [' + @UserName + ']
                   GRANT SELECT ON sys.columns TO [' + @UserName + ']
                   GRANT SELECT ON sys.index_columns TO [' + @UserName + ']
                   GRANT SELECT ON sys.foreign_key_columns TO [' + @UserName + ']
                   GRANT SELECT ON sys.tables TO [' + @UserName + ']
                   GRANT SELECT ON sys.numbered_procedures TO [' + @UserName + ']
                   GRANT SELECT ON sys.database_audit_specifications TO [' + @UserName + ']
                   GRANT SELECT ON sys.filegroups TO [' + @UserName + ']
                   GRANT SELECT ON sys.stats TO [' + @UserName + ']
                   GRANT SELECT ON sys.sysindexes TO [' + @UserName + ']
                   GRANT SELECT ON sys.check_constraints TO [' + @UserName + ']
          END
'
EXECUTE master.sys.sp_MSforeachdb @Command
```

You may want to remove the permissions granted after the assessment is run, in that case you can use the script provided below:

```
--Clean procedure. First log off any session using the RaaSUser
DECLARE @UserName nvarchar(255) = 'Domain\RaaSUser',
                @Command nvarchar(max)
SET @Command = 'USE [?];
        IF EXISTS(SELECT 1 FROM sys.database_principals WHERE name = '''+ @UserName +''')
                DROP USER [' + @UserName + '];'
EXECUTE master.sys.sp_MSforeachdb @Command
SET @Command = 'USE master;
        DROP LOGIN [' + @UserName + ']'
EXEC sp_executesql @Command
```

## Appendix E: PowerShell Remoting Setup

To complete the assessment with the accurate results, you will need to configure all in-scope target machines for Powershell remoting.

PowerShell on the tools machine is used to collect some information that cannot be collected remotely in any other way.

PowerShell version 2 or greater is required on target SQL Servers and comes installed by default starting with Windows Server 2008 R2. For Windows Server 2008 SP2, PowerShell version 2 is not installed by default. It is available for download here https://aka.ms/wmf3download

**Additional requirements for Windows Server 2008-2012 R2 (or later if defaults modified) Target Machines:**

The following three items must be configured on target SQL Servers to support data collection: PowerShell Remoting, WinRM service and Listener, and Inbound Allow Firewall Rules.

**Note 1**: *Windows Server 2012 R2 and Windows Server 2016 have WinRM and PowerShell remoting enabled by default. The following configuration steps detailed below will only need to be implemented if the default configuration for target machines has been altered.*

**Note 2**: *Windows Server 2008—Windows Server 2012 have WinRM disabled by default. The following settings will need to be configured to support PowerShell Remoting:*

- Execute **Enable-PSRemoting** Powershell cmdlet on each target machine within the scope of the assessment. This one command will configure PS-Remoting, WinRM service and listener, and enable required Inbound FW rules. A detailed description of everything Enable-PSRemoting does is documented here.

OR

- Configure **WinRM / PowerShell remoting** via Group Policy (Computer Configuration\Policies\Administrative Templates\Windows Components\Windows Remote Management (WinRM)\WinRM Service)
  - In 2008 R2 it's "**Allow automatic configuration of listeners**".
  - In 2012 R2 (and later) it's "**Allow remote server management through WinRM**".
- Configure **WinRM service for automatic start** via Group Policy (Computer Configuration\Policies\Windows Settings\Security Settings\SystemServices)

  Define **Windows Remote Management** (WS-Management) service for **Automatic startup mode**

  Configure **Inbound allow Firewall Rules:** This can be done individually in the local firewall policy of every in-scope target SQL Servers or via a group policy which allow communication from the tools machine.

Two steps are involved to configure a group policy to enable both WinRM listener and the required inbound allow firewall rules:

A) Identify the IP address of the source computer where data collection will occur from.

B) Create a new GPO linked to the SQL Server organizational unit, and define an inbound rule for the tools machine

**A.) Log into the chosen data collection machine to identify its current IP address using IPConfig.exe from the command prompt.**

An example output is as follows

```
C:\>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:
```

Connection-specific DNS Suffix  . :

Link-local IPv6 Address . . . . . : fe80::X:X:X:X%13

IPv4 Address. . . . . . . . . . . : **X.X.X.X**

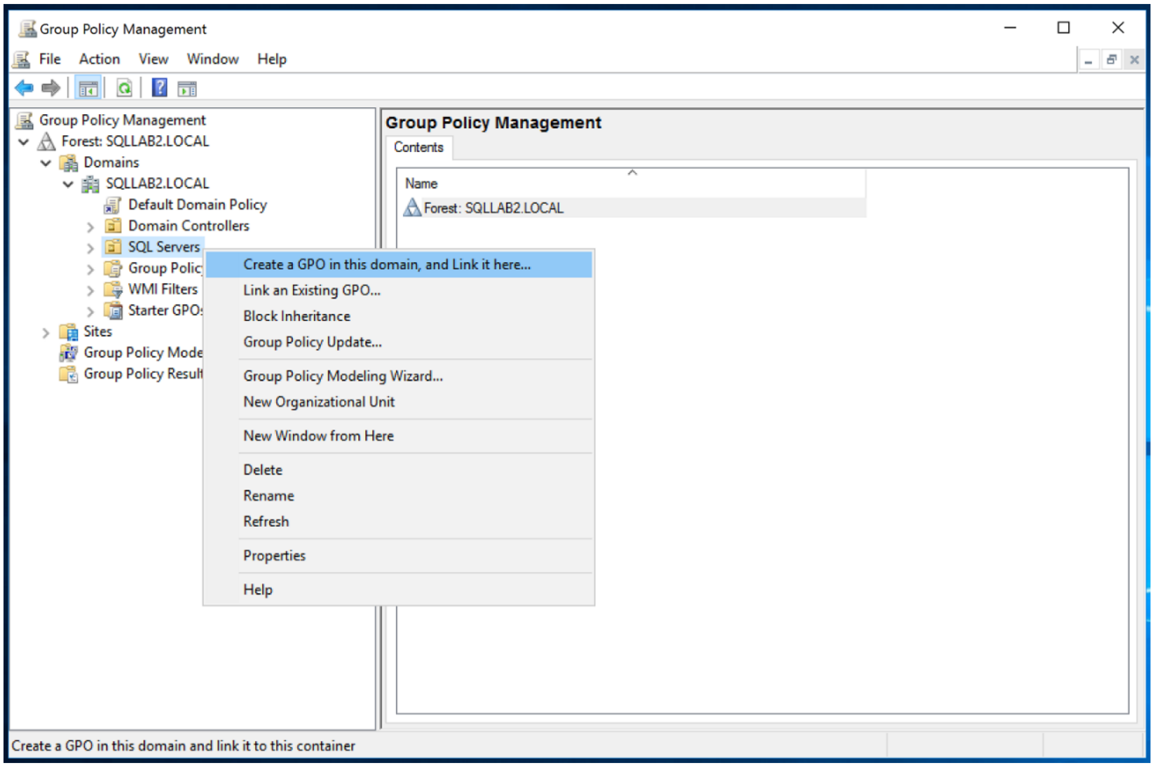
Subnet Mask . . . . . . . . . . . : X.X.X.X

Default Gateway . . . . . . . . . : X.X.X.X

Make a note of the IPv4 address of your machine.  The final step in the configuration will use this address to ensure only the data collection machine can communicate with the Windows Update Agent on the SQL Servers.

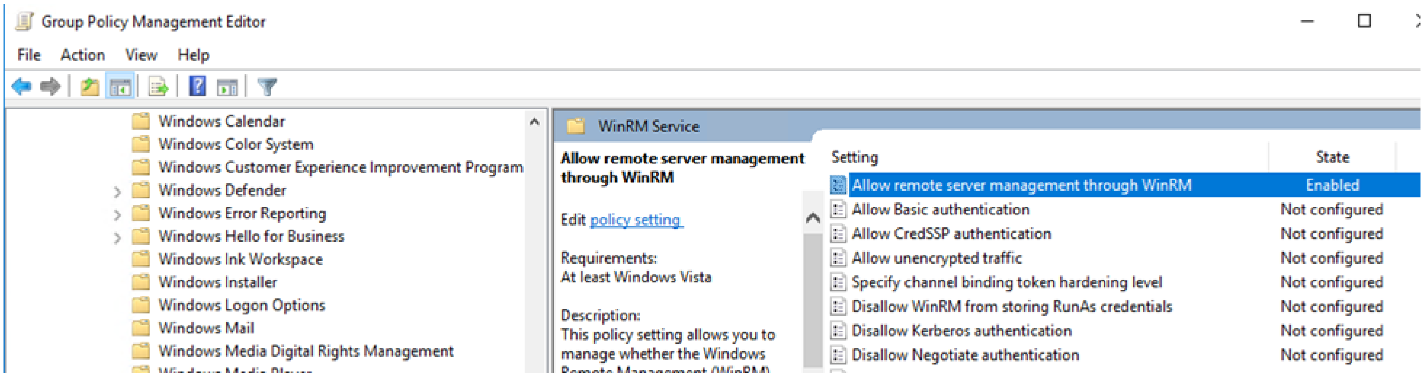
**B.) Create, configure, and link a group policy object to the SQL Servers OU which has target SQL Servers.**

1. Create a new GPO. Make sure the GPO applies to the SQL Servers organizational unit. Give the new group policy a name based on your group policy naming convention or something that identifies its purpose similar to "SQLServerAssessment"



Create a GPO in this domain and link it to this container

2. Within the GPO open, right click on the new GPO and select Edit and go to "Computer Configuration\Policies\Administrative Templates\Windows Components\Windows Remote Management (WinRM)\WinRM Service". Enable "**Allow remote server management through WinRM**" or "**Allow automatic configuration of listeners**" depending on your OS.



**Allow remote server management through WinRM**

Requirements:
At least Windows Vista

Description:
This policy setting allows you to manage whether the Windows

| Setting | State |
|---|---|
| Allow remote server management through WinRM | Enabled |
| Allow Basic authentication | Not configured |
| Allow CredSSP authentication | Not configured |
| Allow unencrypted traffic | Not configured |
| Specify channel binding token hardening level | Not configured |
| Disallow WinRM from storing RunAs credentials | Not configured |
| Disallow Kerberos authentication | Not configured |
| Disallow Negotiate authentication | Not configured |