Microsoft

Offline Assessment for Windows Server Security

Prerequisites

Internet connectivity is needed to

- *access the RAP as a Service portal*
- *activate your account*
- *download the toolset*
- *download the latest wsusscn2.cab file*

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

No data is transported outside your environment without your agreement, in order keep your data protected. Data is analyzed locally using our RAP expert system that is part of the Offline Assessment client.

How to prepare for your Offline Assessment for Windows Server Security

The tools machine is used to connect to each of the servers in your environment and retrieves information from them, communicating over Remote Procedure Call (RPC), Server Message Block (SMB), WMI, and Powershell Remoting. 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.

At a high level, your steps to success are:
1. Install prerequisites on your tools machine and configure your environment.
2. Collect data from your Devices.
3. Complete the operational survey.

The Offline Assessment for Windows Server Security is available for Windows servers running Windows Server® 2008/R2. Windows Server® 2012/R2, Windows Server® 2016. Although there are no limitations to the number of target machines, an engineer can efficiently cover up to 150 targets during the engagement.

Checklist

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

1. General Use

☐ A Microsoft Account is required to activate and sign in to the RaaS portal.
If you don't have one already, you can create one at http://login.live.com

- Learn more about Microsoft Accounts

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

This document was last updated May 15, 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_WSS.pdf

- □ Ensure the Internet browser on the data collection machine has JavaScript enabled. Follow the steps listed at How to enable scripting in your browser. Internet Explorer 11 and Microsoft Edge are the supported browsers for this offering. Most other modern HTML5 based browsers will also work.

- □ The site https://ppas.uservoice.com provides access to the Support Forum and Knowledge Base Articles for RAP as a Service (RaaS) and Offline Client.

2. Data Collection

a. Tools machine hardware and Operating System:

- □ Server-class or high-end workstation machine running (Windows7/Windows 8/Windows 10), or (Windows Server (Server 2008 R2/Server 2012/Server 2012 R2/Server 2016).
- □ Minimum: 8 GB RAM (Recommended requirements 16GB or 32GB based on environment size), 2Ghz dual-core processor, 5 GB of free disk space.
- □ Joined to one of the domains of the forest to be assessed.

b. Software for Tools machine:

- □ Microsoft .NET Framework 4.0 installed.
- □ Windows PowerShell 5.0 or later installed.
  - □ Windows 10 and Windows Server 2016 come with Powershell V5 by default.
  - □ All supported operating systems prior to Windows 10 and Windows Server 2016 will require Powershell V5 to be installed. (PowerShell V5 comes as part of Windows Management Framework 5.1 and is available from https://www.microsoft.com/en-us/download/details.aspx?id=54616
  - □ Windows Update offline scan file (Wsusscn2.cab)

c. Account Rights:

- □ Domain user with Local Administrator permissions on all destination servers.
- □ Unrestricted network access to every Server in the environment to be scanned.

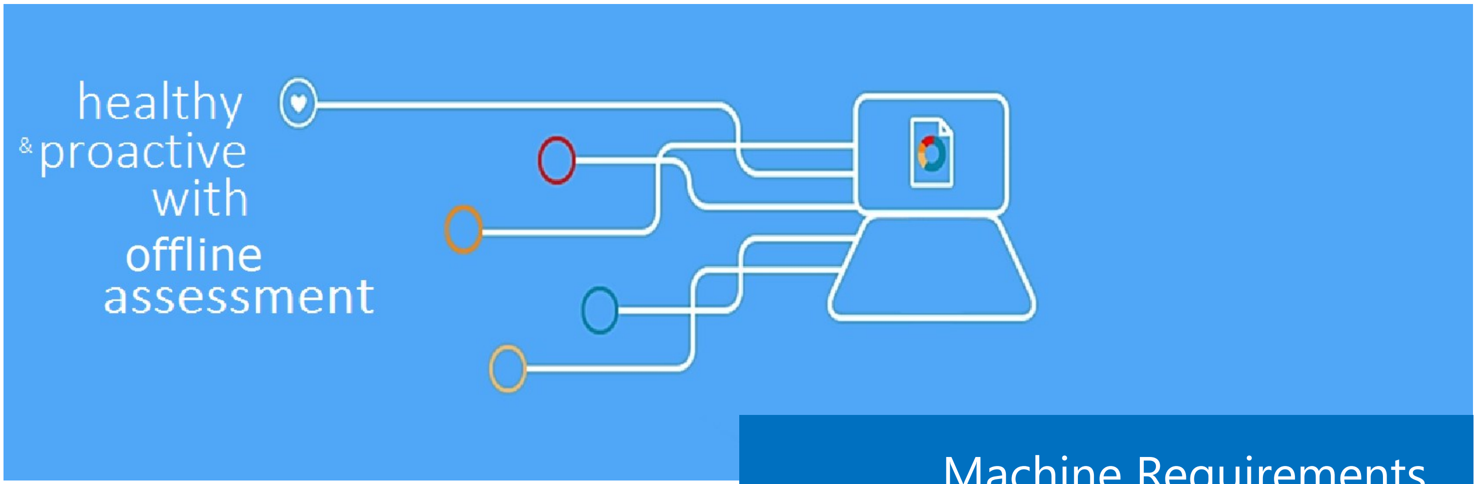
d. Additional Requirements

- □ Configure servers' firewall for Powershell remoting

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 Dual 2Ghz processor — Recommended multi-core 2Ghz or higher processors.
- Minimum 8 GB RAM. Recommended requirements 16GB or 32GB based on environment size.
- Minimum 5 GB of free disk space.
- Windows 7, Windows 8, Windows 10, Windows Server 2008 R2, Windows Server 2012/Windows Server 2012 R2/ Windows Server 2016.
- Requires 64-bit operating system.
- At least a 1024x768 screen resolution (higher preferred).
- A member of the same forest as target hosts.
- Microsoft .NET Framework 4.0 — http://www.microsoft.com/en-us/download/details.aspx?id=17851
- Windows PowerShell 5.0 or higher— https://www.microsoft.com/en-us/download/details.aspx?id=54616
- A networked or redirected "Documents" folder on the tools machine is not supported.
- Office 2013 or higher.

2. Scanning Security Updates and collecting auditing policy configuration with Windows PowerShell V5

PowerShell V5 on the tools machine is used to scan the servers for installed and missing security patches as well as collecting audit policy configuration.

- Scanning for security updates: Download the Windows Update offline scan file (Wsusscn2.cab). The latest cab file can be downloaded from the following link: http://go.microsoft.com/fwlink/?LinkId=76054. The file should be transferred to the collection machine and placed in the root of the OS drive, **C:\wsusscn2.cab,** folder.
- Windows Update Agent must be running on all in scope machines.
- PowerShell version 2 or greater is required on target machines 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

3. Accounts Rights

- ♦ A domain account with the following:
  - ∗ Local administrator permissions to all target servers to be assessed.

WARNING: Do not use the Run As feature to start OfflineAssessmentClient.exe. Some collectors might fail. The account starting the offline client must logon to the local machine.

4. Network and Remote Access

- ♦ Short name resolution must work from the Tools machine. This typically means making sure DNS suffixes for all domains in the forest are added on the Tools machine.
- ♦ Unrestricted network access to every scoped server.
  - ∗ 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$).
  - ∗ Ensure that the machine you use to collect data has complete TCP/UDP access, including RPC access to all servers. For a complete list of protocols, services and ports required by Active Directory, see http://support.microsoft.com/kb/179442 .
  - ∗ PowerShell may be unable to scan servers with the Windows Firewall enabled in its default configuration. Windows servers have the firewall enabled by default and will reject remote scans without special steps taken. The Windows Firewall on each target domain server must have a configured inbound rule in place for PowerShell remoting to each server. Refer to the next section for more details.

5. Additional requirements for Windows Server 2008-2012 R2 (or later if defaults modified) Target Machines: The following three items must be configured to support data collection: PowerShell Remoting, WinRM service and Listener, and Inbound Allow Firewall Rules.

Note1: *Windows Server 2012 R2 and Windows Server 2016 have WinRM and PowerShell remoting enabled by default. The following settings will only need to be modified if the default configuration for target machines has been altered.*

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

- ∗ **PowerShell Remoting / WinRM Service and Listener** : Follow these steps to configure and enforce PowerShell Remoting:
  - ∗ Execute **Enable-PSRemoting** on each target 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 Inbound allow Firewall Rules:** This can be done individually or add a single rule on the target servers which allows all inbound ports from the tools machine.

Two steps are involved:

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

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

5a. Log into the chosen data collection machine to identify its current IP address using IPConfig.exe from the command prompt.

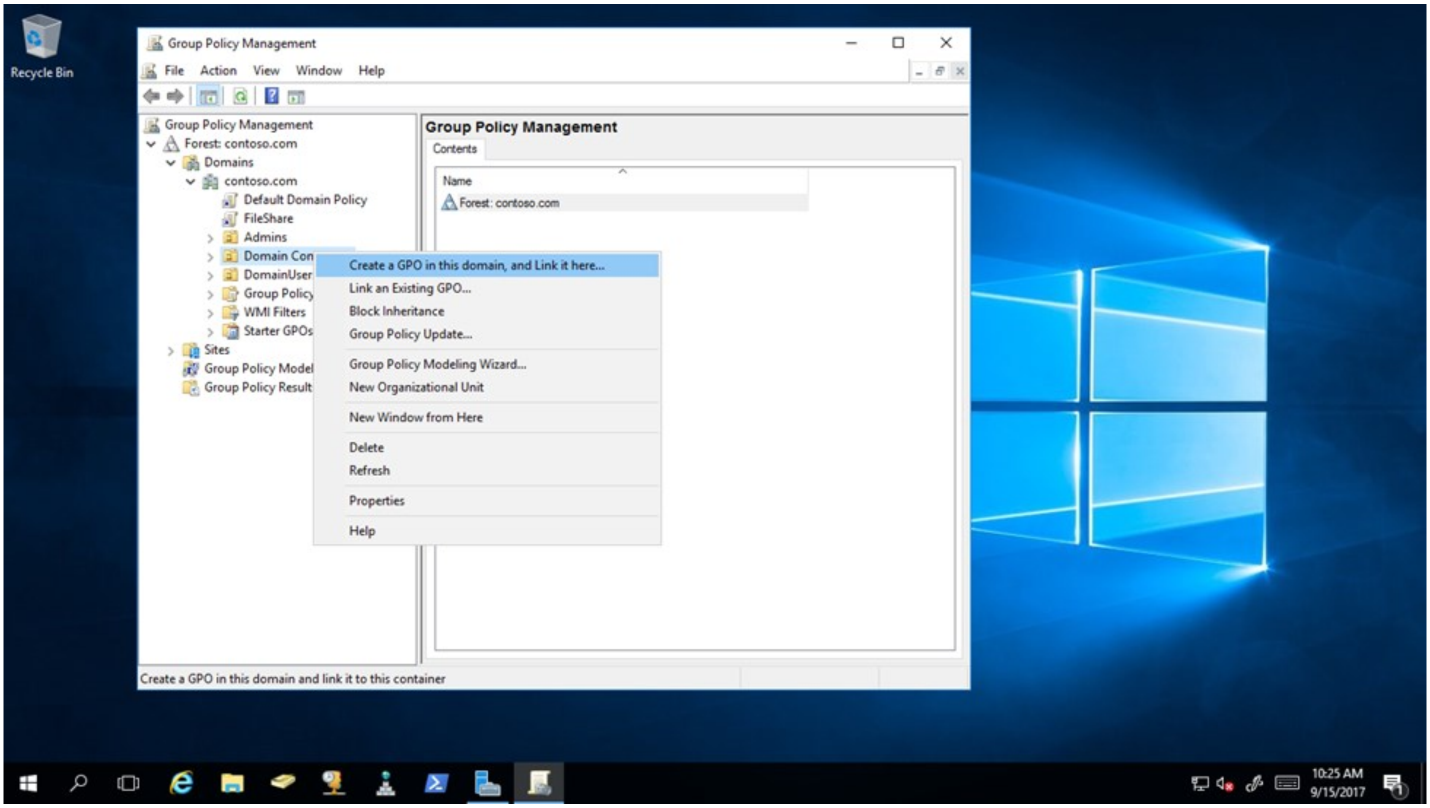
An example output is as follows

```
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 in scope machines.

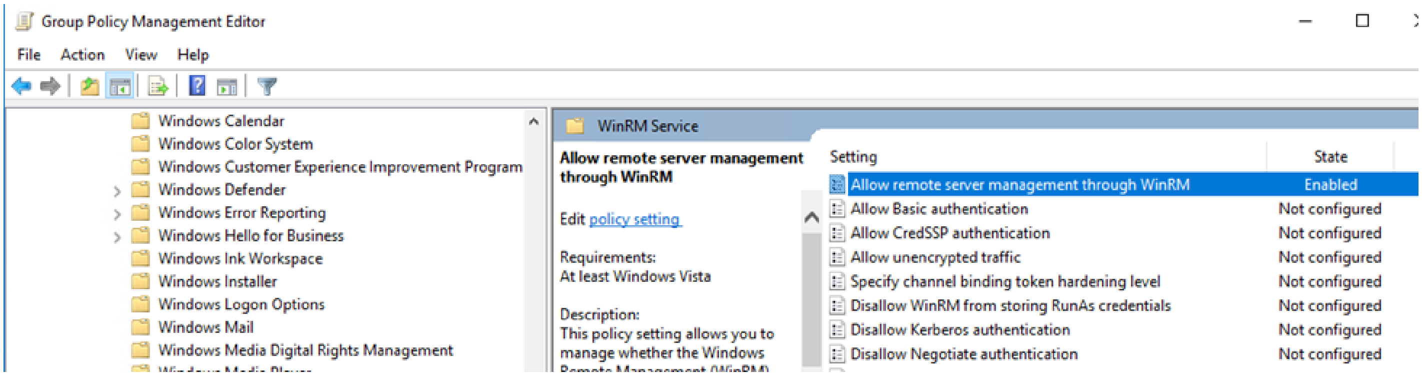
5b. Create, configure, and link a group policy object to the in scope server OUs in each domain in the forest.

1. Create a new GPO. Make sure the GPO applies to the in scope server organizational units. Give the new group policy a name based on your group policy naming convention or something that identifies its purpose similar to “Server Security Assessment”



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

2. Within the GPO open: (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. You will need to specify IPv4 and IPv6 filters. ("*" will allow all inbound servers access, but specifying the IP address of the tools machine is preferred)



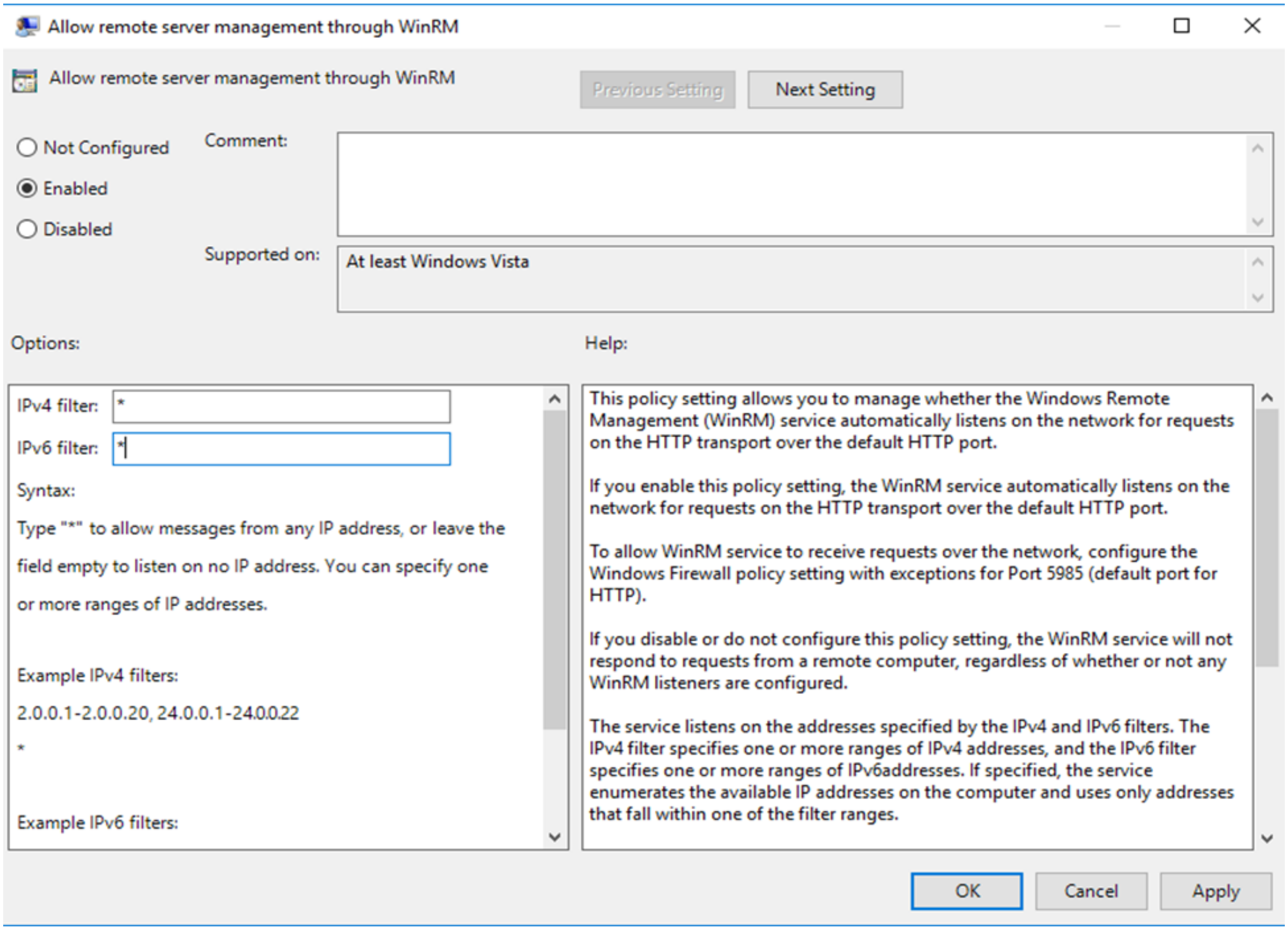
Allow remote server management through WinRM

Edit policy setting

Requirements:
At least Windows Vista

Description:
This policy setting allows you to manage whether the Windows Remote Management (WinRM)

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


Allow remote server management through WinRM
Allow remote server management through WinRM
Previous Setting
Next Setting
OK
Cancel
Apply

○ Not Configured
● Enabled
○ Disabled

Comment:

Supported on: At least Windows Vista

Options:

IPv4 filter: *

IPv6 filter: *

Syntax:

Type "*" to allow messages from any IP address, or leave the field empty to listen on no IP address. You can specify one or more ranges of IP addresses.

Example IPv4 filters:

2.0.0.1-2.0.0.20, 24.0.0.1-24.0.0.22

*

Example IPv6 filters:

Help:

This policy setting allows you to manage whether the Windows Remote Management (WinRM) service automatically listens on the network for requests on the HTTP transport over the default HTTP port.

If you enable this policy setting, the WinRM service automatically listens on the network for requests on the HTTP transport over the default HTTP port.

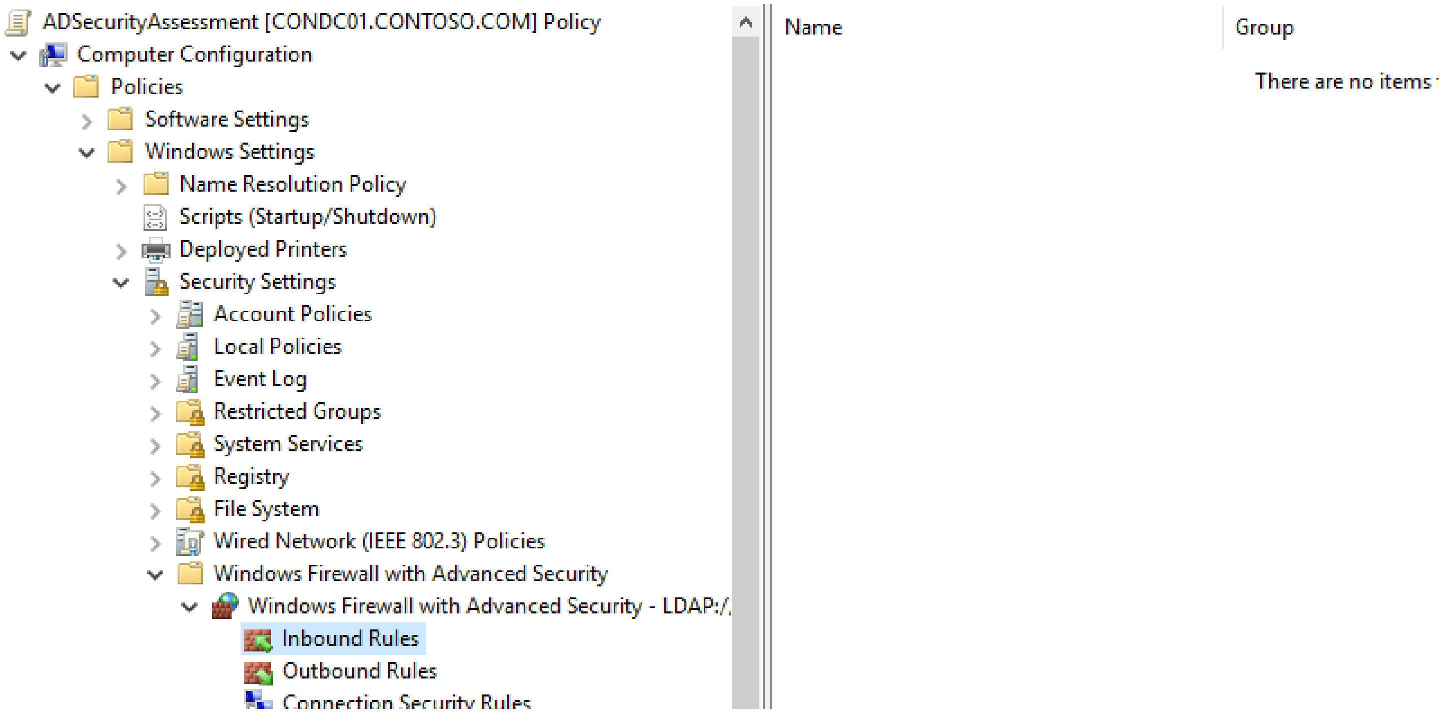
To allow WinRM service to receive requests over the network, configure the Windows Firewall policy setting with exceptions for Port 5985 (default port for HTTP).

If you disable or do not configure this policy setting, the WinRM service will not respond to requests from a remote computer, regardless of whether or not any WinRM listeners are configured.

The service listens on the addresses specified by the IPv4 and IPv6 filters. The IPv4 filter specifies one or more ranges of IPv4 addresses, and the IPv6 filter specifies one or more ranges of IPv6addresses. If specified, the service enumerates the available IP addresses on the computer and uses only addresses that fall within one of the filter ranges.

3. Create an advanced Inbound Firewall Rule to allow all network traffic from the tools machine to the target servers. This can be the applied to the same GPO that was used in step 1 above. (Computer Configuration\Windows Settings\Security Settings\Windows Firewall with Advanced Security\Windows Firewall with Advanced Security –LDAP:/xxx\Inbound Rules)

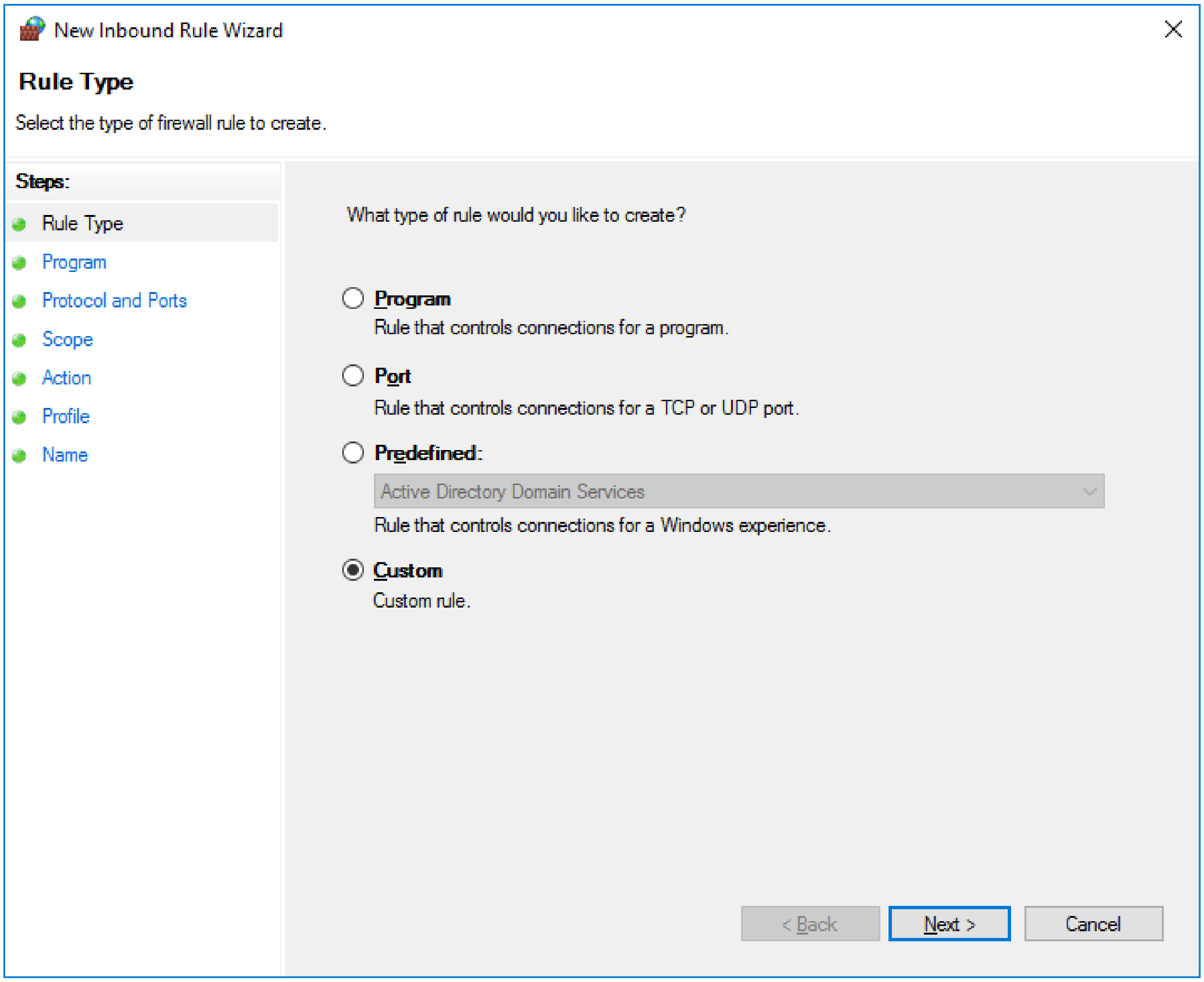
4. To create the new rule, Right Click on “Inbound Rules” and select “New”



ADSecurityAssessment [CONDC01.CONTOSO.COM] 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 Security - LDAP:/,
            - Inbound Rules
            - Outbound Rules
            - Connection Security Rules

There are no items

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?

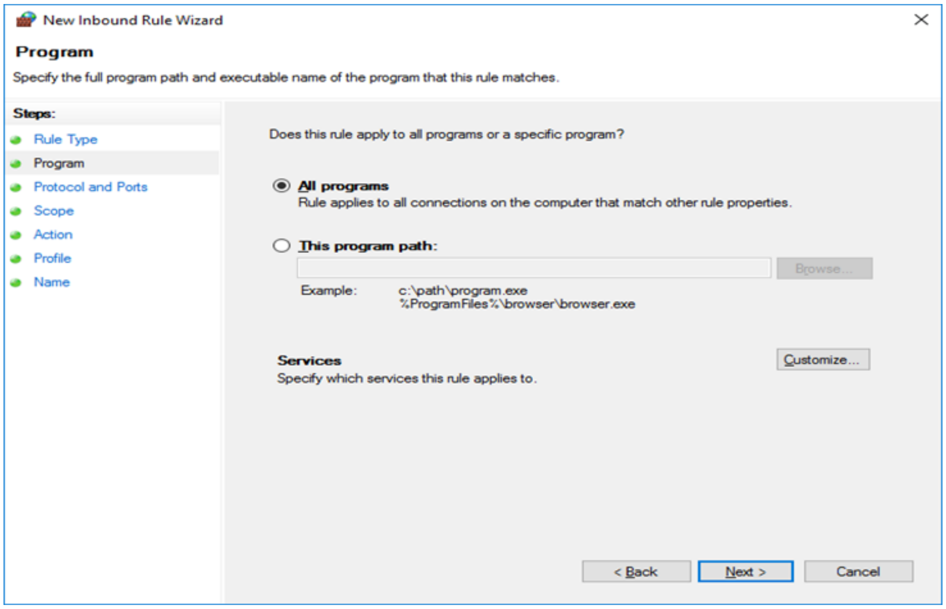
○ **Program**
Rule that controls connections for a program.

○ **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.

Steps:

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

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

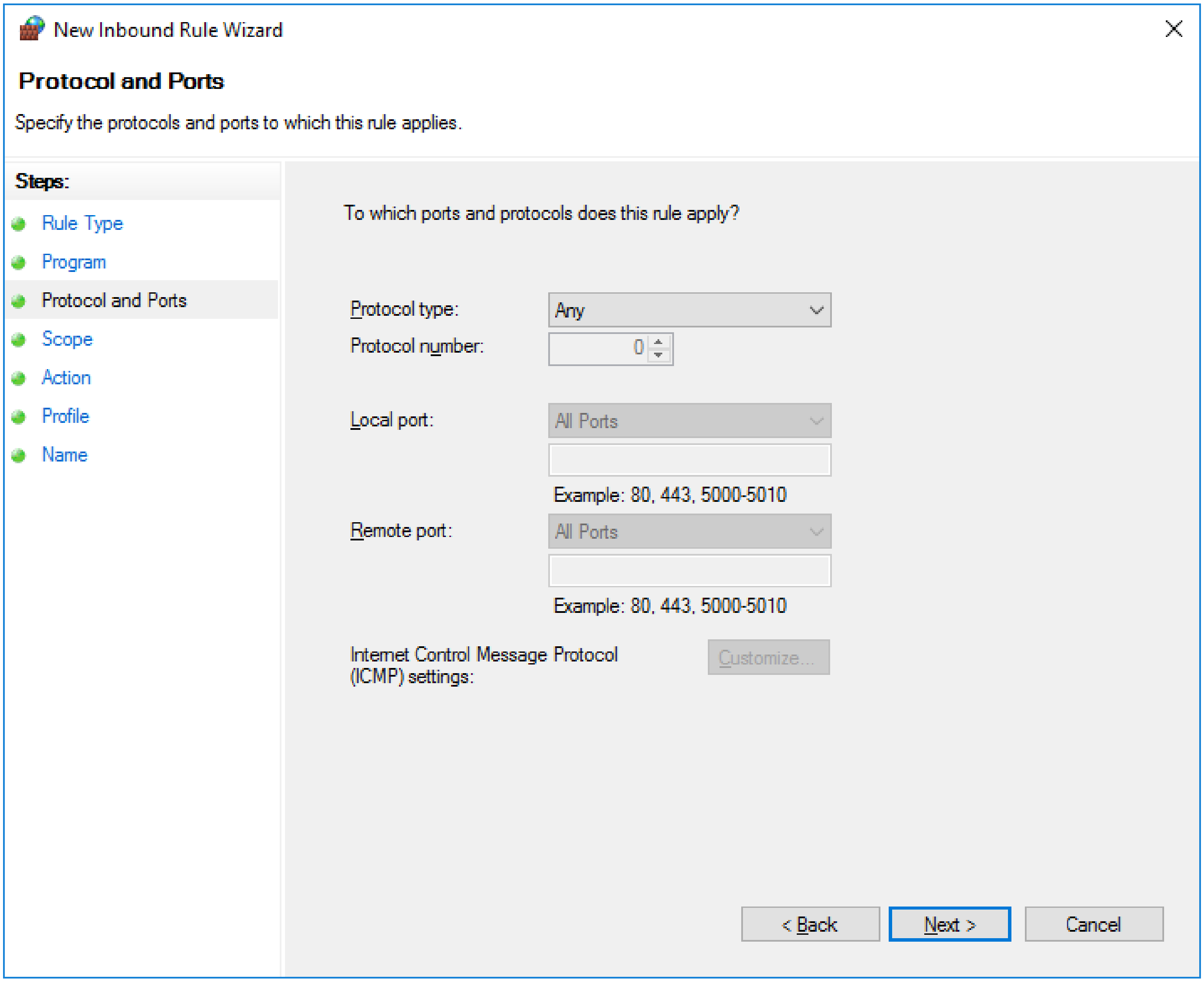
● All programs
Rule applies to all connections on the computer that match other rule properties.

○ 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.

Steps:

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

To which ports and protocols does this rule apply?

Protocol type: Any

Protocol number: 0

Local port: All Ports

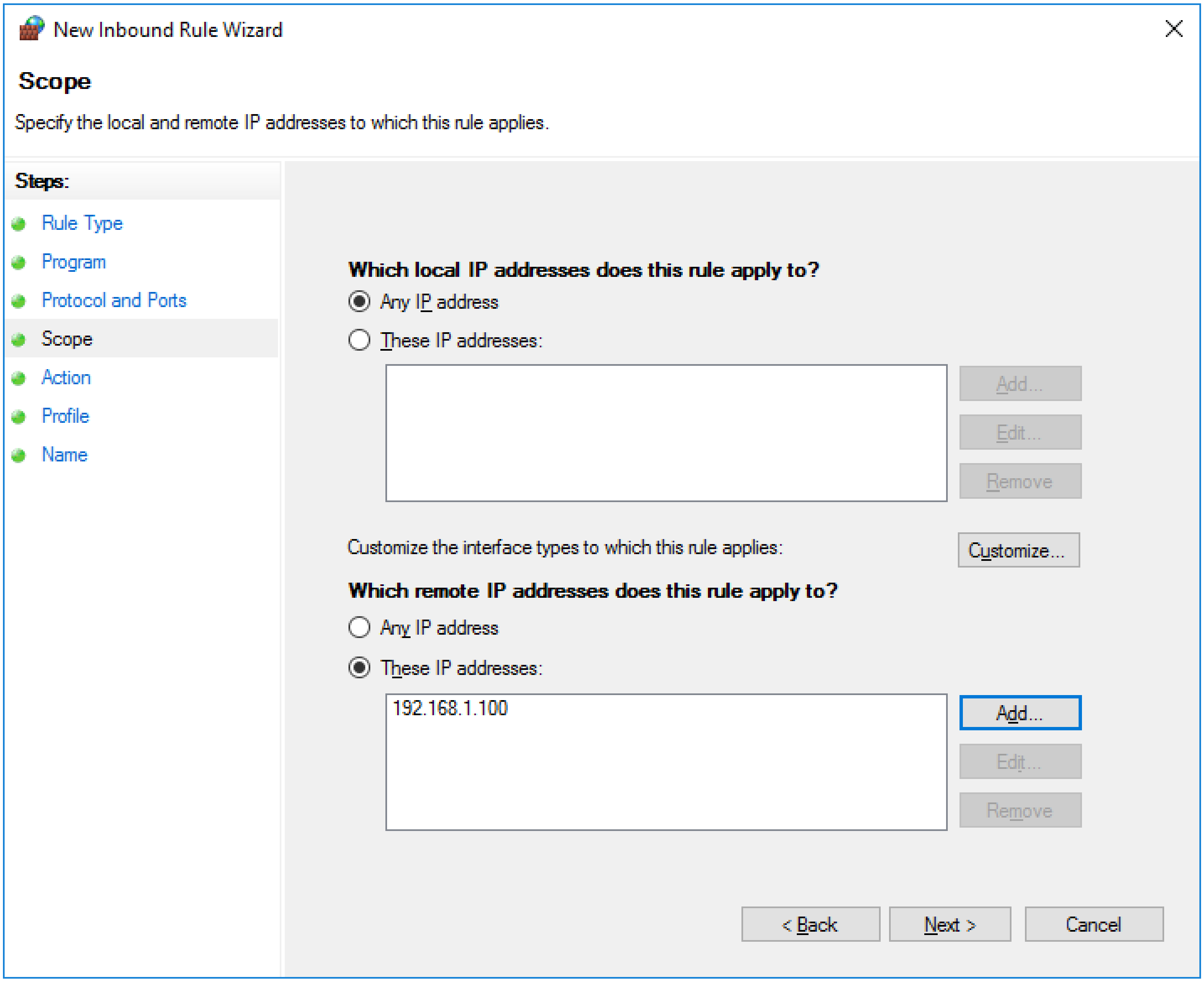
Example: 80, 443, 5000-5010

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: ServerSecurityAssessmentToolsMachine)

Appendix: Data Collection Methods

Offline Assessment for Windows Server Security uses multiple data collection methods to collect information. This section describes the methods used to collect data from Windows Server environment.
Data collection uses workflows and collectors. The collectors are:

1. Registry Collectors.
2. LDAP Collectors.
3. Windows PowerShell.
4. FileDataCollector.
5. Windows Management Instrumentation (WMI).
6. Custom C# Code.
7. Validation.

1. Registry Collectors

Registry keys and values are read from the Windows Servers. They include items such as:

- Service information from HKLM\SYSTEM\CurrentControlSet\Services.
- Operating System information from HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion

2. LDAP Collectors

LDAP queries are used to collect data for the Domain, DCs, Partitions, group memberships, account names, and other components from AD itself. For a complete list of ports required by AD, see: http://support.microsoft.com/kb/179442.

3. Windows PowerShell

Collects various information, such as:

- Audit policy configuration and security update collection

4. FileDataCollector

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

- Windows binary information like srv.sys.

5. **Windows Management Instrumentation (WMI)**

WMI is used to collect various information such as:

- WIN32_Volume

  Collects information on Volume Settings for each in scope machine. The information is used for instance to determine the system volume and drive letter which allows the client to collect information on files located on the system drive.
- Win32_Process

  Collect information on the processes running on each machine in the scope. The information provides insight in processes that consume a large amount of threads, memory or have a large page file usage.

6. Custom C# Code

Collects information not captured using other collectors. The primary example here is the collection of effective user rights on the Windows servers.

7. Checking Validation

Collects information not captured using other collectors. The primary example here is the collection of effective user rights on Servers.

- Check computer Registry FQDN name and WMI against every target machine

```
get-wmiobject Win32_ComputerSystem -computer localhost | fl Name,Domain


Expected  Result:


                   Name   : <ComputerName>
                   Domain : dns.name
```

- Check if administrative shares are available against every target machine

```
get-wmiobject WIN32_Share -computer "<ComputerName>" | ?{$_.Name -eq "C$"} | FL Name


                   Expected  Result: Name : C$
```

- Check Scheduled Tasks access against every target machine

```
                   Enter-PSSession -Computer <ComputerName>
                   Expected Result: [ComputerName]: PS C:\Users\UserName\Documents>
```

```
$([xml](schtasks /query /XML ONE /S "<ComputerName>")).Tasks.Task.Count


                   Expected  Result: > 0
```

- Verifying PowerShell Remoting is enabled:

```
                   Enter-PSSession -Computer <ComputerName>
                   Expected Result: [ComputerName]: PS C:\Users\UserName\Documents>
```