

Here are the needed steps to configure correctly the PowerShell discovery:

How to Configure PowerShell Remoting

This task describes how to enable PowerShell remote access.

This task includes the following steps:

1. Launch the PowerShell configuration

In the PowerShell command prompt run the winrm quickconfig.

Note: From the moment that the PowerShell configuration is launched, you must differ between the server side configuration and client side configuration.

±

2. Configure the server-side machine

On the server, depending on the authentication method that will be used, perform the following steps:

- a. Run `cd WSMAN:\localhost\Service\Auth`
- b. Run `dir` and verify that the required authentication type is enabled, that is, the `State = True`. If the required authentication type is disabled, run `Set-Item <AuthTypeName> True`. By default, Kerberos and Negotiate are enabled.
- c. Run `cd WSMAN:\localhost\Service` and verify that `IPv4Filter` or `IPv6Filter` are set to either `"*"` or to any other valid value for your environment.
- d. Run `cd WSMAN:\localhost\Listener`, and then `dir`. Verify that the listener actually listens to the required IPs. By default, the listener listens to all IPs if the value `"*"` is used.
- e. If you made any changes, restart the winrm service by running the `restart-service winrm` command

3. Configure the client-side machine

On the client machine, perform the following steps:

- a. Run `cd WSMAN:\localhost\Client\Auth`
- b. Run `dir` and verify that the required authentication type is enabled, that is, the `State = True`. If the required authentication type is disabled, run `Set-Item <AuthTypeName> True`.

Note: The allowed protocols must coincide with the ones configured on the server side.

- c. Run `cd WSMAN:\localhost\Client`.
 - d. Run `dir` and check value of `TrustedHosts`. By default, the value is empty so that no connection outside is possible. `TrustedHosts` is an ACL field where the allowed values are a domain name or a list of domain names and an IP address or a list of IP addresses. The value may have a special symbol `"*"`, meaning that any destination or any symbol can appear in any part of the specified destinations list. If the only value is `"*"`, then the client is allowed to connect to any host. This is the recommended value.
- To change the value for `TrustedHosts`, use `Set-Item TrustedHosts <Value>`.

```

PS WSMan:\localhost\Client> Set-Item .\TrustedHosts *
WinRM Security Configuration.
This command modifies the TrustedHosts list for the WinRM client. The computers in the TrustedHosts list might not be
authenticated. The client might send credential information to these computers. Are you sure that you want to modify
this list?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y")> y
PS WSMan:\localhost\Client> dir

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Client
Name Value Type
-----
NetworkDelays 5000 System.String
URLPrefix usman System.String
AllowUnencrypted false System.String
Auth Container
DefaultPorts Container
TrustedHosts * System.String
PS WSMan:\localhost\Client> _

```

Note: No translation from FQDN to IP is done while validating the ACL. This means that if the connection is performed by IP and only an FQDN is listed in the TrustedHosts field (or vice versa), the connection will not be allowed.

e. If you made any changes, restart the winrm service by running the restart-service winrm command.

HC by powershell

2. Prerequisite - Configure PowerShell

Before starting the discovery, ensure that PowerShell v2.0 is installed and configured on the Data Flow Probe machine. To access the installation files, see <http://support.microsoft.com/kb/968929>).

a. Enable PowerShell remoting:

- o Launch PowerShell v 2.0 as an administrator.
- o Run the **Enable-PSRemoting** cmdlet. This starts the WinRM service and sets the startup type to Automatic, enables a firewall exception for WS-Management communications, and creates a listener to accept requests on any IP address.

Note: To enable PowerShell remoting on all computers in your domain, in Domain Group Policy: Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Management (WinRM) > \WinRM Service, select **Allow automatic configuration of listeners**.

b. To trust all hosts, run the following from the command line:

```
Set-Item WSMan:\localhost\Client\TrustedHosts *
```

To trust only restricted IP addresses, specify the addresses in place of the asterisk (*).

c. Restart WinRM by running the following from the command line:

```
restart-Service winrm
```

Note: By default, WinRM uses Kerberos for authentication. To configure WinRM for https, see <http://support.microsoft.com/kb/2019527>.

Permissions:

User need to be admin or at least write and read permissions.

User need to have permissions to connect to any domain:

For details of this special case, see "HOW TO ENABLE REMOTING FOR ADMINISTRATORS IN OTHER DOMAINS" at <http://technet.microsoft.com/en-us/library/dd347642.aspx>

Screenshots of PowerShell configuration on CLIENT and SERVER:

Configuration on the remote server that we want to discover (main settings)

```
Administrator: Windows PowerShell
PS WSMan:\localhost> ls

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost

Name                Value                Type
-----                -
MaxEnvelopeSizeKb    150                  System.String
MaxTimeouts          60000                System.String
MaxBatchItems        32000                System.String
MaxProviderRequests  4294967295           System.String
Client               Container
Service              Container
Shell                Container
Listener             Container
Plugin               Container
ClientCertificate    Container

PS WSMan:\localhost> _
```

Services configuration:(here we enable http/https compatibility)

```
Administrator: Windows PowerShell
PS WSMan:\localhost\Service> ls

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Service

Name                Value                Type
-----                -
RootSDDL            O:NEG:BAD:P(A;;;GA;;;BR)S:P(AU;FA;GA;;;WD)(AU;SA;GMCK;;;UD)
MaxConcurrentOperations 4294967295           System.String
MaxConcurrentOperation... 15                  System.String
EnumerationTimeouts    60000               System.String
MaxConnections        25                  System.String
MaxPacketRetrievalTime... 120                 System.String
AllowUnencrypted       false               System.String
Auth                   Container
DefaultPorts          *                   Container
IPv4Filter             *                   System.String
IPv6Filter             *                   System.String
EnableCompatibilityHtt... true                System.String
EnableCompatibilityHtt... false               System.String
CertificateThumbprint   *                   System.String

PS WSMan:\localhost\Service> _
```

Client configuration:(trustedhosts value should be * to allow any connection)

```
PS WSMan:\localhost\Client> ls

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Client

Name                Value                Type
-----                -
NetworkDelays        5000                 System.String
URLPrefix             wsman                System.String
AllowUnencrypted       false               System.String
Auth                   Container
DefaultPorts          *                   Container
TrustedHosts          *                   System.String

PS WSMan:\localhost\Client> _
```

Listener configuration:

```
PS WSMan:\localhost> cd listener
PS WSMan:\localhost\Listener> ls

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Listener

Name                Type                Keys
-----                -
Listener_641507880  Container           <Address=*, Transport=HTTP>
Listener_1369396199 Container           <Address=*, Transport=HTTP>

PS WSMan:\localhost\Listener> _
```

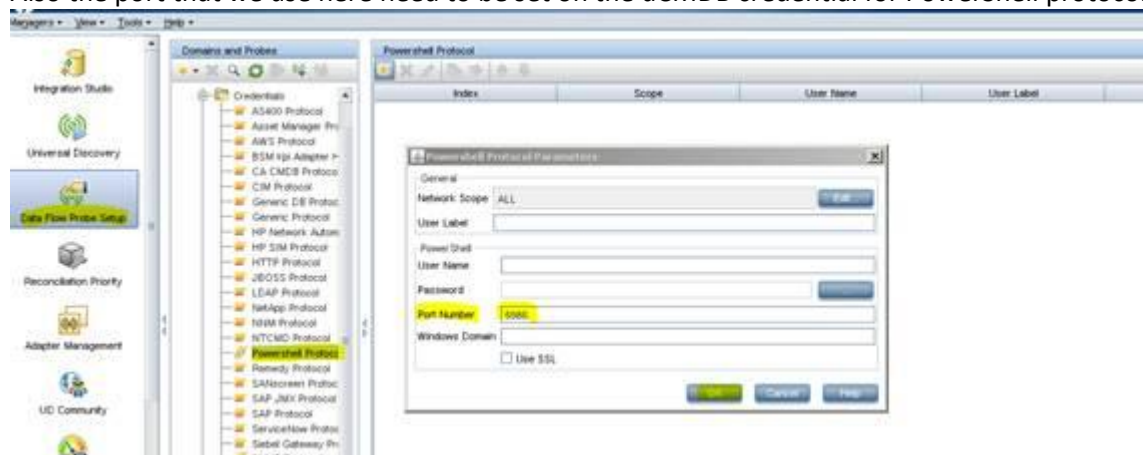
Client default ports: (here you can change the ports in case you don't want to use the default ones)

```
PS WSMan:\localhost\Client\DefaultPorts> ls

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Client\DefaultPorts

Name                Value                Type
-----                -
HTTP                5985                System.String
HTTPS               5986                System.String
```

Also the port that we use here need to be set on the uCMDB credential for PowerShell protocol:



Client auth: (here we can enable/disable permission)

```
PS WSMan:\localhost\Client\Auth> ls

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Client\Auth

Name                Value                Type
-----                -
Basic               true                System.String
Digest              true                System.String
Kerberos             true                System.String
Negotiate            true                System.String
Certificate          true                System.String
CredSSP              false               System.String
```

Service Auth: (here we can enable/disable permission)

```
PS WSMan:\localhost\Service\Auth> ls

WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Service\Auth

Name                Value                Type
-----                -
Basic               false               System.String
Kerberos             true                System.String
Negotiate            true                System.String
Certificate          false               System.String
CredSSP              false               System.String
CbtHardeningLevel    Relaxed              System.String
```

***Note:**

Command to change any setting: Set-Item

Example:

Set-Item WSMan:\localhost\Service\EnableCompatibilityHttpListener -Value true