

AxoNet Plug-Ins for InstallAware

Quick reference guide

Edition 3.5.0

Contents

1. About	2
2. Upgrade Code – Plug-In.....	3
2.1. Usage.....	3
3. Volume Info – Plug-In	4
3.1. Usage.....	4
3.2. Note	5
4. Windows Firewall – Plug-In	6
4.1. Usage.....	7
4.1.1. Adding a program	7
4.1.2. Adding a port and protocol	8
4.1.3. Deleting a program	9
4.1.4. Deleting a port	10
4.2. Requirements.....	11
4.3. Release notes	11
5. Windows Advanced Firewall – Plug-In (New in 3.0)	12
5.1. Usage.....	13
5.2. Adding rules	13
5.2.1. Adding a program	13
5.2.2. Adding a port and protocol	14
5.2.3. Blocking outbound traffic on port 80.....	15
5.2.4. Block inbound traffic from all WINS servers	16
5.2.5. Remote IP magic names	16
5.3. Deleting rules	17
5.3.1. Deleting a program	17
5.3.2. Deleting a port	18
5.4. Requirements.....	20
5.5. Release notes	20
6. Disk Technology – Plug-In.....	21
6.1. Usage.....	21
6.2. Requirements.....	21
7. Service Config 2 – Plug-In.....	22
7.1. Usage.....	22
7.2. Start type.....	22
7.3. Service Recovery	23
7.4. Example	23
7.5. Requirements.....	23
8. Revision history	24

1. About

AxoNet Plug-Ins for InstallAware is a Freeware Add-On package for InstallAware 6.0 – X 5. This software is provided as is, use on your own risk. Liability for any damage due to faulty software or data is impossible.

The plug-ins have been developed with CodeGear RAD Studio 2007 (Delphi) and tested with InstallAware 18 and X4.

Copyright © 2006-2017, AxoNet Software GmbH, Martin Rothschink

2. Upgrade Code – Plug-In

This plug-in detects a product based on the upgrade code. It returns the product code if the upgrade code is found. This is useful if you upgraded from InstallShield Express to InstallAware. InstallShield Express requires a change of the product code for every new version of your product and keeps the upgrade code always the same.

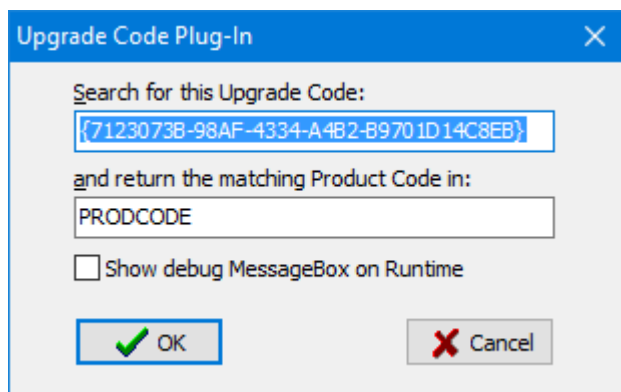
Example

Product 1.0 Product Code A, Upgrade Code U
Product 1.1 Product Code B, Upgrade Code U
Product 1.2 Product Code C, Upgrade Code U

InstallAware never changes the product code; instead the revision code is changed. If a new version is installed, a previous version is detected by the same product code and removed. If you have previously used InstallShield Express for your product, you do not know which product code is present on the target machine. Use this plug-in to detect and uninstall a previous version.

2.1. Usage

Go to the “Check Application Pre-Requisites” section. Define a variable which will store the product code. Add this plug-in and specify your InstallShield Express upgrade code (you may also use a variable which holds the GUID):



Check if a product code is returned and uninstall the old product:

```
[DEFINE REGION: Check Application Pre-Requisites]
...
Set Variable PRODCODE to
Get Product Code from Upgrade Code {7123073B-98AF-4334-A4B2-B9701D14C8EB} into PRODCODE
if Variable PRODCODE not Equals
    Install/Remove MSI Package $PRODCODE$[REMOVE=ALL]
end
...
```

3. Volume Info – Plug-In

This plug-in retrieves information about a logical drive. You can specify a single drive letter, a root directory, a full path name or an UNC share name.

Examples

C
C:\
C:\path\file
\\server\share

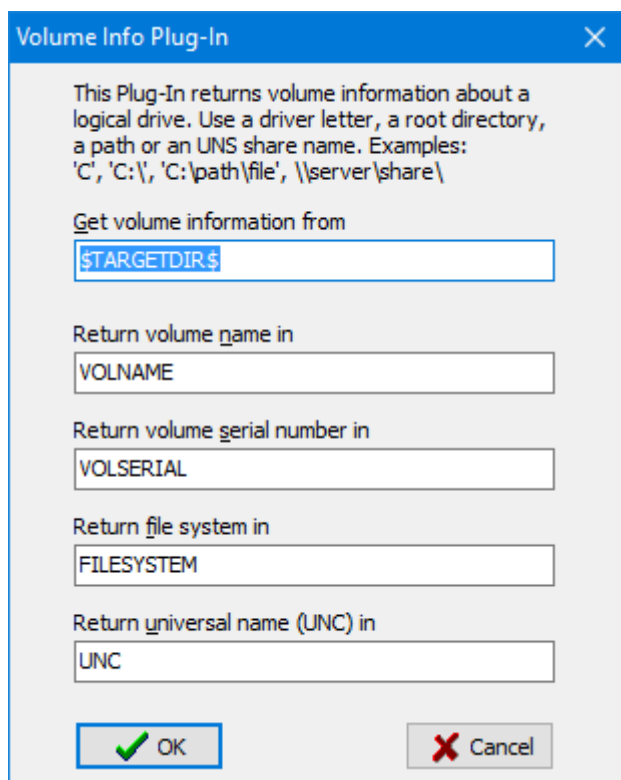
The plug-in returns the volume name, serial number, file system type and the UNC path name.

The file system type is one of

FAT
FAT32
NTFS
CDFS

3.1. Usage

Add the plug-in to your script where you need to retrieve volume information. Define variables to hold the retrieved information:



The screenshot shows a dialog box titled "Volume Info Plug-In" with a close button (X) in the top right corner. The dialog contains the following text and input fields:

This Plug-In returns volume information about a logical drive. Use a driver letter, a root directory, a path or an UNS share name. Examples: 'C', 'C:\', 'C:\path\file', \\server\share\

Get volume information from

Return volume name in

Return volume serial number in

Return file system in

Return universal name (UNC) in

At the bottom, there are two buttons: "OK" with a green checkmark icon and "Cancel" with a red X icon.

AxoNet Plug-Ins for InstallAware

Quick reference guide

```
Set Variable VOLNAME to  
Set Variable VOLSERIAL to  
Set Variable FILESYSTEM to  
Set Variable UNC to  
Get Volume Information from '$TARGETDIR$'
```

3.2. Note

If the volume is not a network share, UNC is always empty.

4. Windows Firewall – Plug-In

This plug-in configures the Windows Firewall. You can add or delete a firewall exception based on an executable program or a port number and protocol.

Example MSI code for adding exception rules

To successfully add a program to the firewall list, the program must exist! Therefore you should always place the “Configure Windows Firewall” plug-in after the Apply Install command in your MSI code:

```
if Variable ADVERTISE Equals TRUE
  Apply Advertised (get result into variable SUCCESS)
else
  Apply Install (get result into variable SUCCESS)
end
```

```
Configure Windows Firewall - add allowed program $TARGETDIR$\NOTEPAD.EXE
Configure Windows Firewall - add port opening 1234 Both
```

```
[compiler end]
```

```
Set Variable PROGRESS to 100
```

Example MSI code for deleting exception rules

This is typically done on uninstall. Place the “Configure Windows Firewall” plug-in after the TO-DO comment.

```
Comment: Modify Target System
```

```
[DEFINE REGION: Perform Uninstallation]
```

```
if Variable REMOVE Equals TRUE
```

```
  Comment: Uninstall product
```

```
  Comment: TO-DO: Insert any additional uninstall commands here
```

```
Configure Windows Firewall - delete allowed program $TARGETDIR$\NOTEPAD.EXE
```

```
Configure Windows Firewall - delete port opening 1234 Both
```

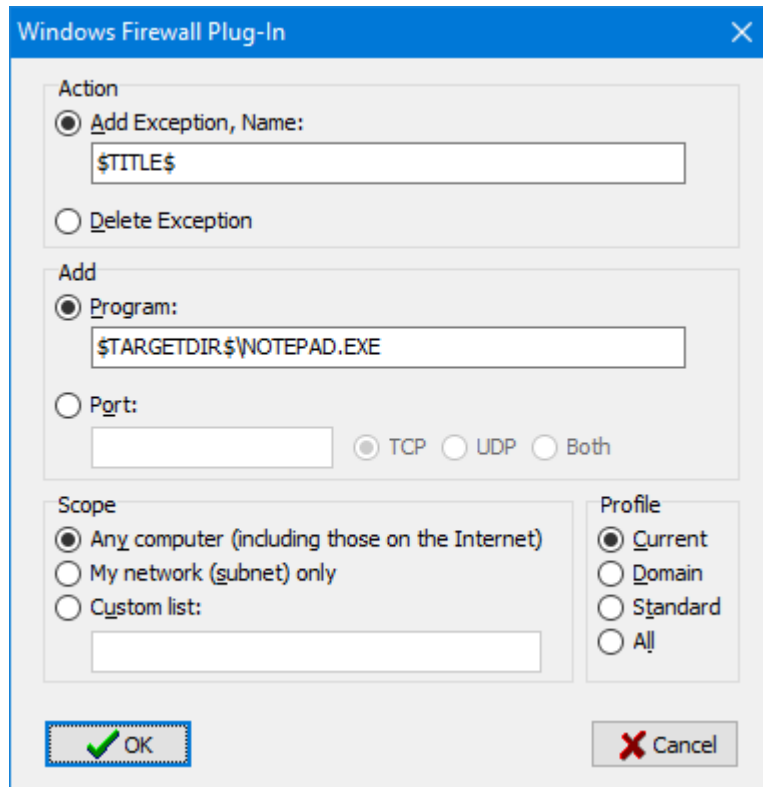
```
Apply Uninstall (get result into variable SUCCESS)
```

```
Set Variable PROGRESS to 100
```

4.1. Usage

4.1.1. Adding a program

To add a program, enter an exception name and the full program path. You can use variables for both fields. The exception name is displayed in the Windows Firewall applet after installation. Optionally you may modify the scope and the profile:



The screenshot shows the 'Windows Firewall Plug-In' dialog box. It has a blue title bar with a close button. The dialog is divided into several sections:

- Action:** Two radio buttons are present. The first, 'Add Exception, Name:', is selected. Below it is a text input field containing '\$TITLE\$'. The second radio button is 'Delete Exception'.
- Add:** Two radio buttons are present. The first, 'Program:', is selected. Below it is a text input field containing '\$TARGETDIR\$\NOTEPAD.EXE'. The second radio button is 'Port:'. Below it is a text input field and three radio buttons: 'TCP' (selected), 'UDP', and 'Both'.
- Scope:** Three radio buttons are present. The first, 'Any computer (including those on the Internet)', is selected. Below it is a text input field. The other two radio buttons are 'My network (subnet) only' and 'Custom list:'.
- Profile:** Four radio buttons are present. The first, 'Current', is selected. The other three are 'Domain', 'Standard', and 'All'.

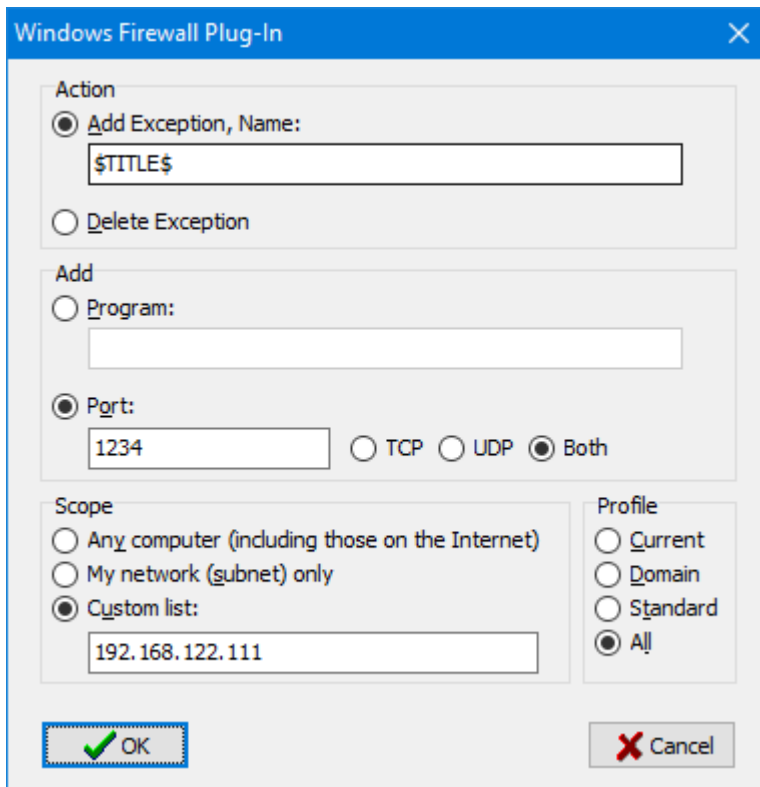
At the bottom of the dialog, there are two buttons: 'OK' (with a green checkmark icon) and 'Cancel' (with a red X icon).

AxoNet Plug-Ins for InstallAware

Quick reference guide

4.1.2. Adding a port and protocol

To add a port number and protocol you have to enter an exception name, the port number and the protocol. If you select "Both", two entries are created, one for UDP and one for TCP:



The screenshot shows the "Windows Firewall Plug-In" dialog box. It has a blue title bar with a close button. The main area is divided into several sections:

- Action:** Radio buttons for "Add Exception, Name:" (selected) and "Delete Exception". A text box below "Add Exception, Name:" contains "\$TITLE\$".
- Add:** Radio buttons for "Program:" and "Port:". The "Port:" section has a text box containing "1234" and radio buttons for "TCP", "UDP", and "Both" (selected).
- Scope:** Radio buttons for "Any computer (including those on the Internet)", "My network (subnet) only", and "Custom list:". The "Custom list:" section has a text box containing "192.168.122.111".
- Profile:** Radio buttons for "Current", "Domain", "Standard", and "All" (selected).

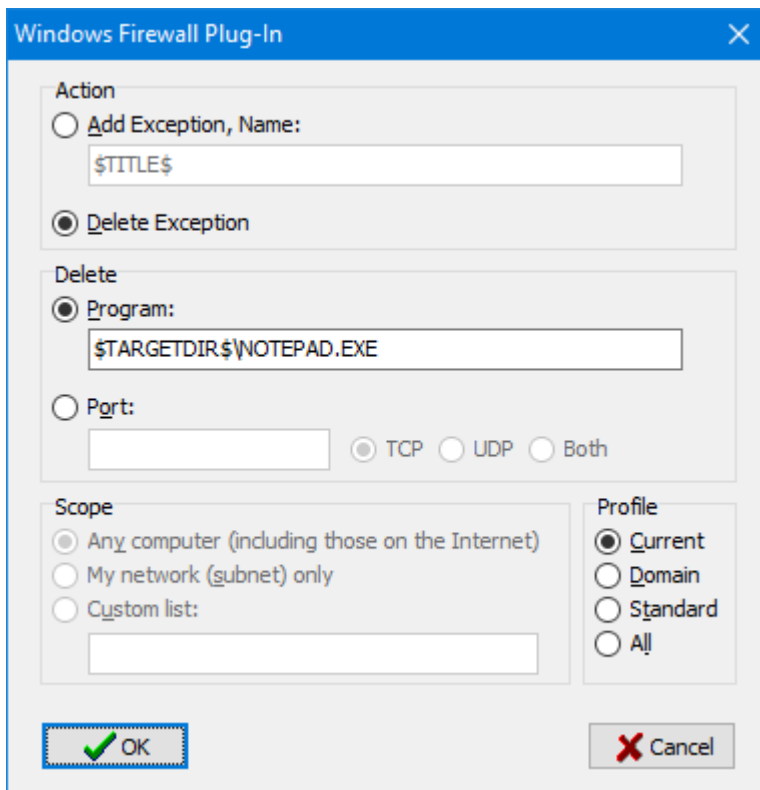
At the bottom, there are two buttons: "OK" (with a green checkmark icon) and "Cancel" (with a red X icon).

AxoNet Plug-Ins for InstallAware

Quick reference guide

4.1.3. Deleting a program

To delete a program from the exception list you only need to specify the full program path:

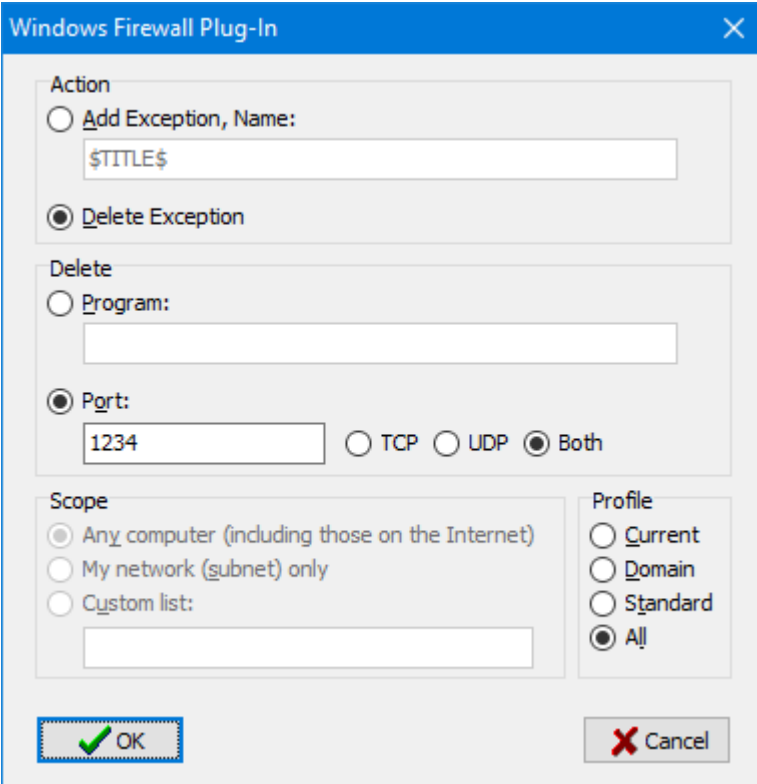


AxoNet Plug-Ins for InstallAware

Quick reference guide

4.1.4. Deleting a port

To delete a port from the exception list, specify the port number and protocol:



The screenshot shows the 'Windows Firewall Plug-In' dialog box. The 'Action' section has 'Delete Exception' selected. The 'Delete' section has 'Port:' selected with '1234' in the text box and 'Both' selected for the protocol. The 'Scope' section has 'Any computer (including those on the Internet)' selected. The 'Profile' section has 'All' selected. The 'OK' button is highlighted with a red dashed border.

Windows Firewall Plug-In

Action

Add Exception, Name:
\$TITLE\$

Delete Exception

Delete

Program:
[Empty text box]

Port:
1234 TCP UDP Both

Scope

Any computer (including those on the Internet)
 My network (subnet) only
 Custom list:
[Empty text box]

Profile

Current
 Domain
 Standard
 All

OK Cancel

4.2. Requirements

This plug-in requires Windows XP SP2 or later.

4.3. Release notes

On Windows Vista and Windows 7, opening a port exception may not work. Server 2008 and 2008 R2 and Windows 8/10 are not affected.

We recommend to use the new Windows Advanced Firewall Plug-In for Windows Vista and later!

5. Windows Advanced Firewall – Plug-In (New in 3.0)

This plug-in configures the Windows Advanced Firewall in Windows Vista/Server 2008 and later. You can add or delete a firewall exception based on an executable program, a port number or a protocol.

Example MSI code for adding exception rules

To successfully add a program to the firewall list, the program must exist! Therefore, you should always place the “Configure Windows Advanced Firewall” plug-in after the Apply Install command in your MSI code:

```
if Variable ADVERTISE Equals TRUE
  Apply Advertised (get result into variable SUCCESS)
else
  Apply Install (get result into variable SUCCESS)
end
```

```
Configure Windows Advanced Firewall - add rule name="Block Outbound Port 80" direction=out
action=block localport=80 protocol=tcp
Configure Windows Advanced Firewall - add rule name="Block WINS" direction=in action=block
remoteip=wins
Configure Windows Advanced Firewall - add rule name="Allow Messenger" direction=in
action=allow program="c:\program files\messenger\msmsgs.exe" remoteip=localsubnet [compiler
end]
```

```
Set Variable PROGRESS to 100
```

Example MSI code for deleting exception rules

This is typically done on uninstall. Place the “Configure Windows Firewall” plug-in after the TO-DO comment.

```
Comment: Modify Target System
[DEFINE REGION: Perform Uninstallation]
if Variable REMOVE Equals TRUE
  Comment: Uninstall product
  Comment: TO-DO: Insert any additional uninstall commands here
```

```
Configure Windows Advanced Firewall - delete rule name="Block Outbound Port 80"
direction=out localport=80 protocol=tcp
Configure Windows Advanced Firewall - delete rule name="Block WINS" direction=in
remoteip=wins
Configure Windows Advanced Firewall - delete rule name="Allow Messenger" direction=in
program="c:\program files\messenger\msmsgs.exe" remoteip=localsubnet
```

```
Apply Uninstall (get result into variable SUCCESS)
```

```
Set Variable PROGRESS to 100
```

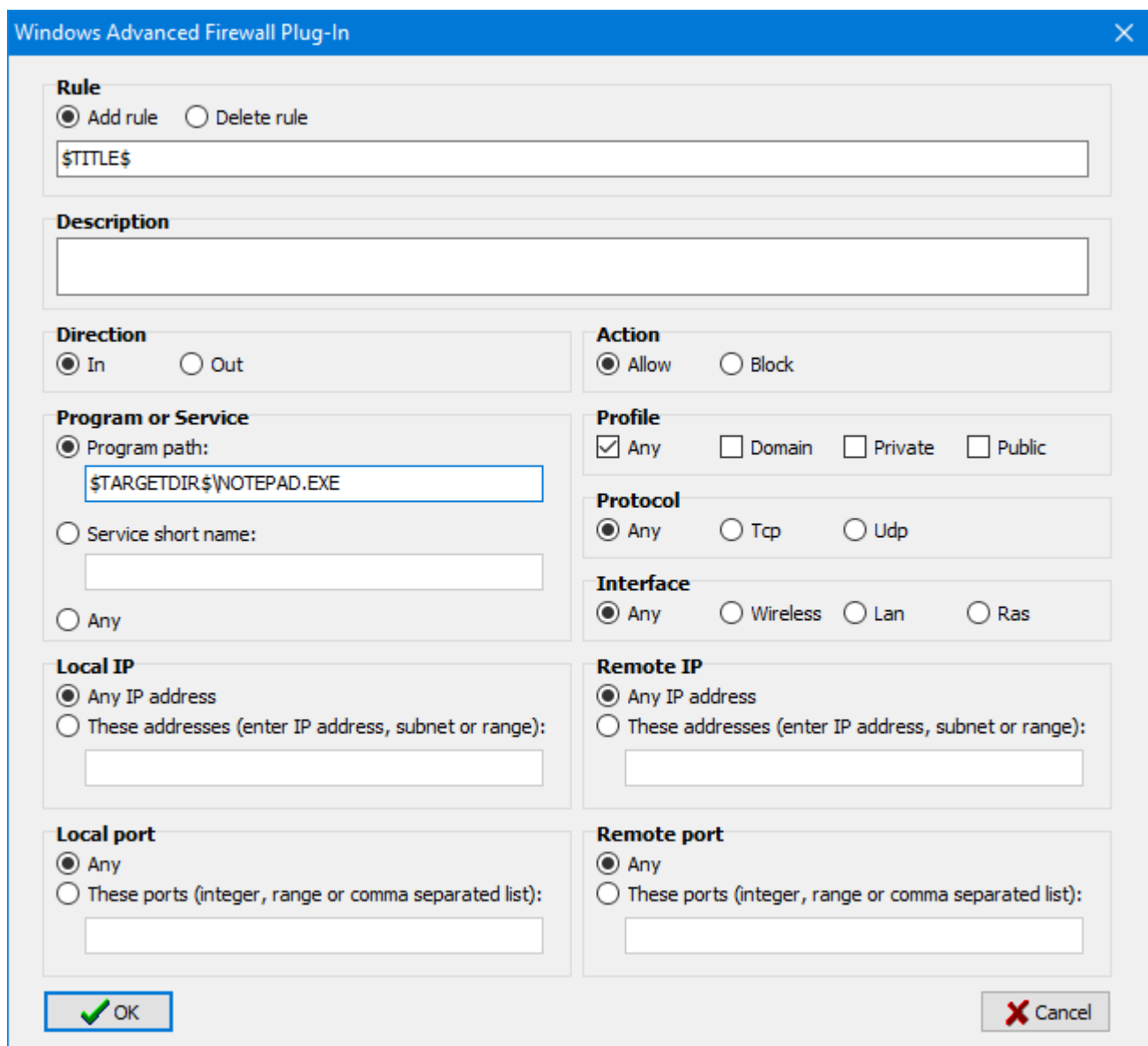
5.1. Usage

5.2. Adding rules

Every rule requires a name. The name should be unique, and **must not** be "all" if you add a new rule.

5.2.1. Adding a program

To add a program, enter an exception name and the full program path. You can use variables for both fields. The exception name is displayed in the Windows Firewall applet after installation. Optionally you may add a description and modify local and remote ip or port, profile, protocol and interface:



Windows Advanced Firewall Plug-In

Rule
 Add rule Delete rule
\$TITLE\$

Description
[Empty text field]

Direction
 In Out

Action
 Allow Block

Program or Service
 Program path:
\$TARGETDIR\$\\NOTEPAD.EXE
 Service short name:
[Empty text field]
 Any

Profile
 Any Domain Private Public

Protocol
 Any Tcp Udp

Interface
 Any Wireless Lan Ras

Local IP
 Any IP address
 These addresses (enter IP address, subnet or range):
[Empty text field]

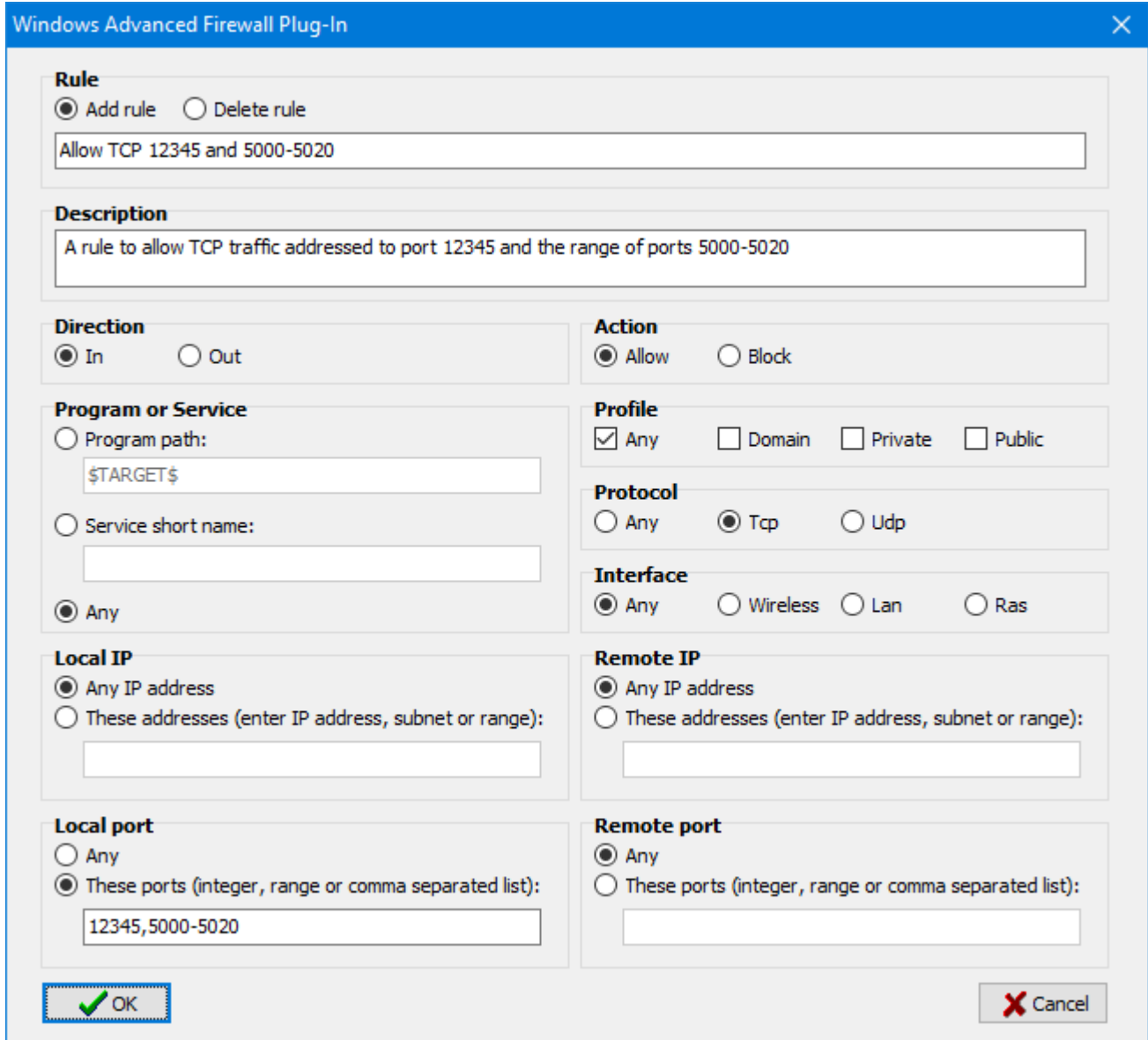
Remote IP
 Any IP address
 These addresses (enter IP address, subnet or range):
[Empty text field]

Local port
 Any
 These ports (integer, range or comma separated list):
[Empty text field]

Remote port
 Any
 These ports (integer, range or comma separated list):
[Empty text field]

5.2.2. Adding a port and protocol

To add a port number and protocol you have to enter an exception name, the port number and the protocol. The following command creates a rule to allow TCP traffic addressed to port 12345 and the range of ports 5000-5020:

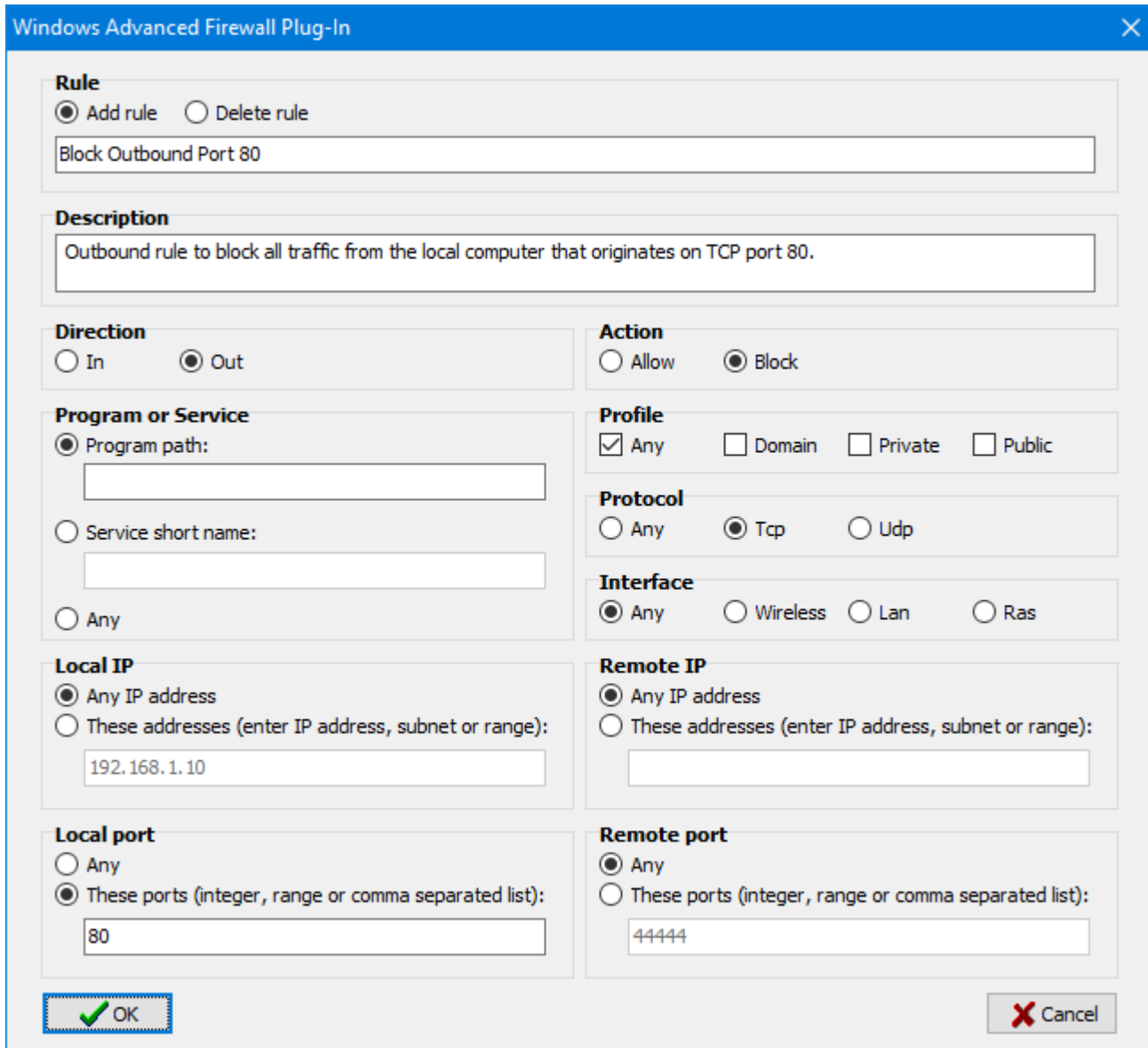


The screenshot shows the 'Windows Advanced Firewall Plug-In' dialog box. The 'Rule' section has 'Add rule' selected and the name 'Allow TCP 12345 and 5000-5020'. The 'Description' is 'A rule to allow TCP traffic addressed to port 12345 and the range of ports 5000-5020'. Under 'Direction', 'In' is selected. Under 'Action', 'Allow' is selected. In 'Program or Service', 'Any' is selected. Under 'Profile', 'Any' is checked. Under 'Protocol', 'Tcp' is selected. Under 'Interface', 'Any' is selected. Under 'Local IP', 'Any IP address' is selected. Under 'Remote IP', 'Any IP address' is selected. Under 'Local port', 'These ports (integer, range or comma separated list):' is selected with '12345,5000-5020' entered. Under 'Remote port', 'Any' is selected. The 'OK' button is highlighted with a green checkmark, and the 'Cancel' button is visible.

Note: a local or remote port requires protocol TCP or UDP, you can't use any here!

5.2.3. Blocking outbound traffic on port 80

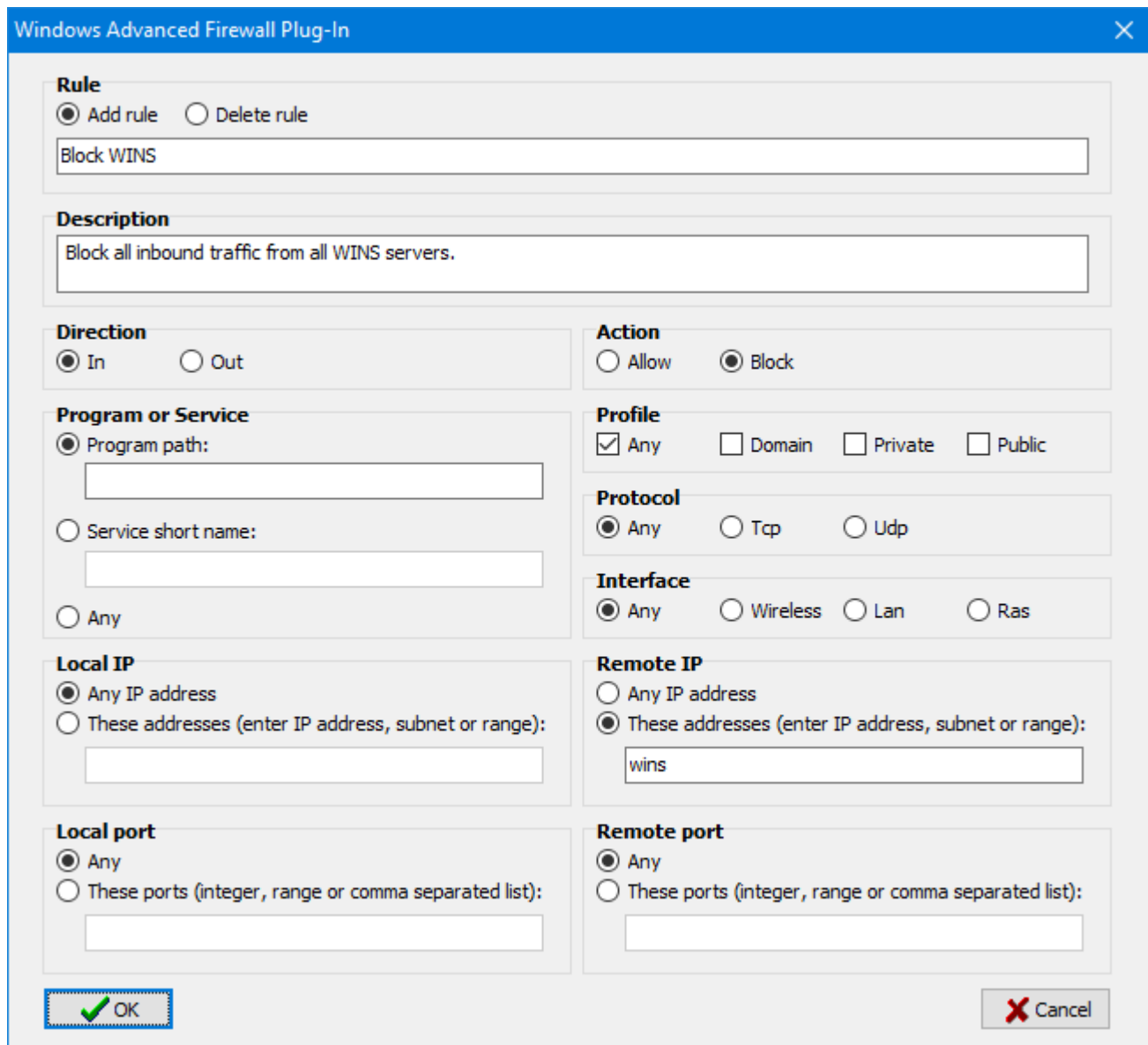
The following command creates an outbound rule to block all traffic from the local computer that originates on TCP port 80:



The screenshot shows the 'Windows Advanced Firewall Plug-In' dialog box. The 'Rule' section has 'Add rule' selected and the name 'Block Outbound Port 80'. The 'Description' field contains 'Outbound rule to block all traffic from the local computer that originates on TCP port 80.' The 'Direction' is set to 'Out' and the 'Action' is set to 'Block'. Under 'Program or Service', 'Program path:' is selected. The 'Profile' section has 'Any' checked. The 'Protocol' is set to 'Tcp'. The 'Interface' is set to 'Any'. The 'Local IP' is set to 'Any IP address' with the value '192.168.1.10'. The 'Remote IP' is set to 'Any IP address'. The 'Local port' is set to 'These ports (integer, range or comma separated list):' with the value '80'. The 'Remote port' is set to 'Any' with the value '44444'. At the bottom, there are 'OK' and 'Cancel' buttons.

5.2.4. Block inbound traffic from all WINS servers

The following command creates a rule that blocks all inbound traffic from all WINS servers:



The screenshot shows the 'Windows Advanced Firewall Plug-In' dialog box. The 'Rule' section has 'Add rule' selected and the name 'Block WINS'. The 'Description' is 'Block all inbound traffic from all WINS servers.' The 'Direction' is 'In' and the 'Action' is 'Block'. The 'Program or Service' section has 'Program path' selected. The 'Profile' section has 'Any' selected. The 'Protocol' section has 'Any' selected. The 'Interface' section has 'Any' selected. The 'Local IP' section has 'Any IP address' selected. The 'Remote IP' section has 'These addresses (enter IP address, subnet or range):' selected with 'wins' entered. The 'Local port' section has 'Any' selected. The 'Remote port' section has 'Any' selected. There are 'OK' and 'Cancel' buttons at the bottom.

5.2.5. Remote IP magic names

You can use some magic names for remote IP:

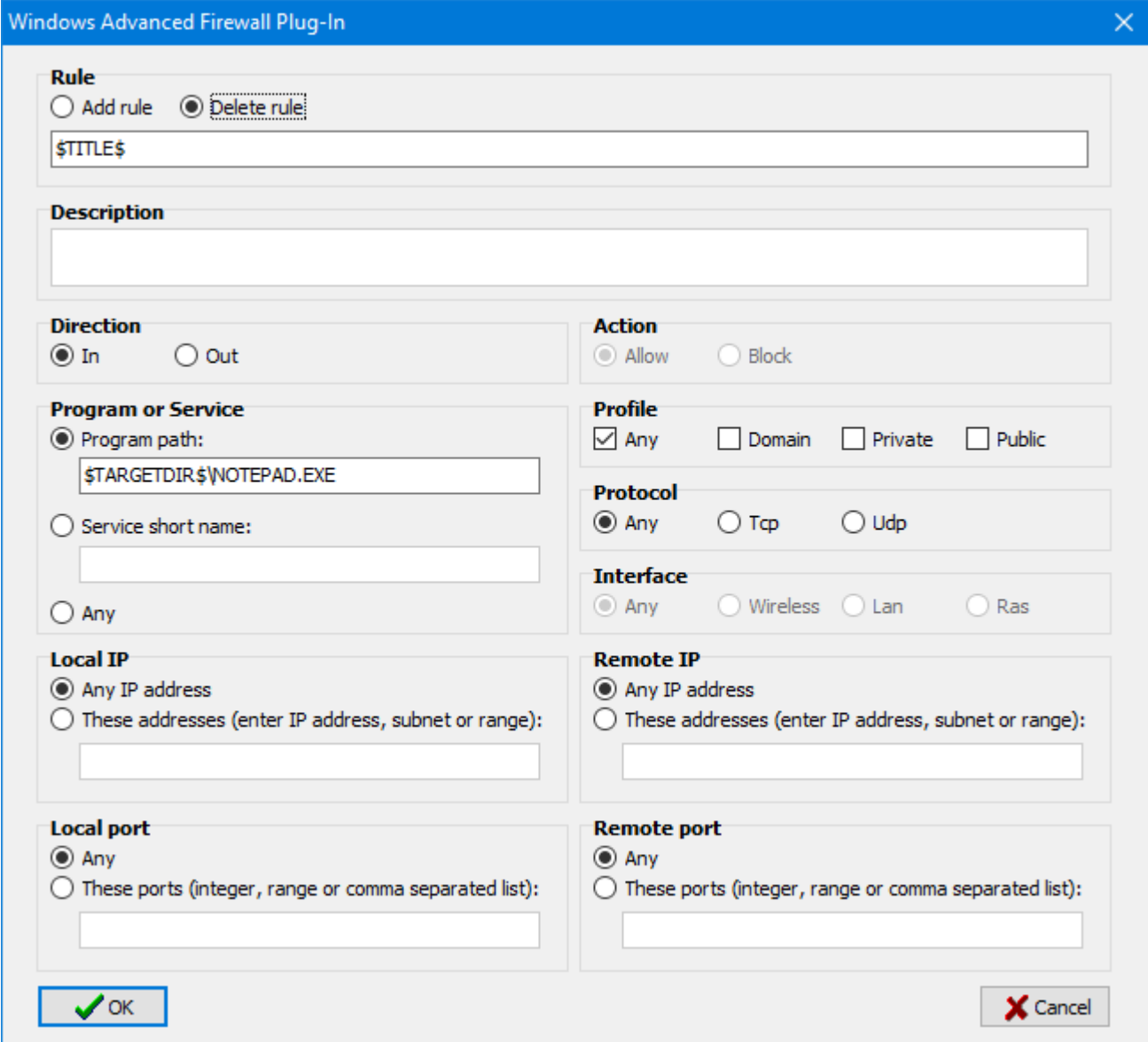
- **any**: Matches any IP address.
- **localsubnet**: Matches any IP address that is on the same IP subnet as the local computer.
- **dns|dhcp|wins|defaultgateway**: Matches the IP address of any computer that is configured as the identified server type on the local computer.

5.3. Deleting rules

If you delete a rule, you can use the name "all" to specifies that all rules matching the criteria in the other parameters are deleted. If no other parameters are included, **then all connection security rules are deleted!**

5.3.1. Deleting a program

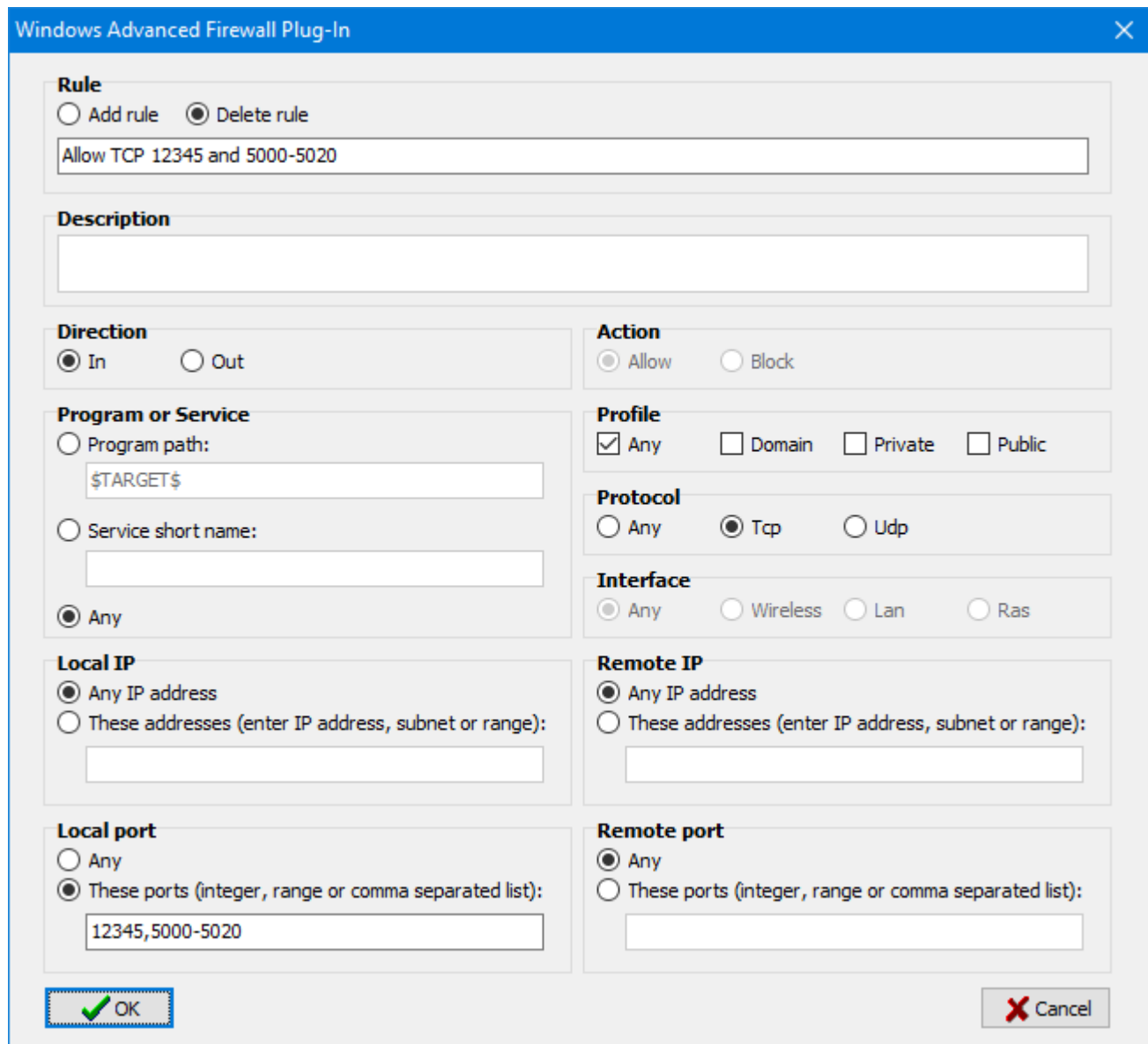
To delete a program from the exception list, copy the rule and change 'add' to 'delete'. To remove the program from our example, use these settings:



The screenshot shows the 'Windows Advanced Firewall Plug-In' dialog box. The 'Rule' section has 'Delete rule' selected. The 'Rule' name field contains '\$TITLE\$'. The 'Description' field is empty. The 'Direction' section has 'In' selected. The 'Action' section has 'Allow' selected. The 'Program or Service' section has 'Program path:' selected, with '\$TARGETDIR\$\NOTEPAD.EXE' entered. The 'Profile' section has 'Any' selected. The 'Protocol' section has 'Any' selected. The 'Interface' section has 'Any' selected. The 'Local IP' section has 'Any IP address' selected. The 'Remote IP' section has 'Any IP address' selected. The 'Local port' section has 'Any' selected. The 'Remote port' section has 'Any' selected. At the bottom, there are 'OK' and 'Cancel' buttons.

5.3.2. Deleting a port

To delete a port from the exception list, specify the port number and protocol:



The screenshot shows the 'Windows Advanced Firewall Plug-In' dialog box. The 'Rule' section has 'Delete rule' selected. The rule name is 'Allow TCP 12345 and 5000-5020'. The 'Direction' is 'In'. The 'Action' is 'Allow'. The 'Program or Service' is 'Any'. The 'Profile' is 'Any'. The 'Protocol' is 'Tcp'. The 'Interface' is 'Any'. The 'Local IP' is 'Any IP address'. The 'Remote IP' is 'Any IP address'. The 'Local port' is '12345,5000-5020'. The 'Remote port' is 'Any'. The 'OK' button is highlighted with a green checkmark.

Rule
 Add rule Delete rule
Allow TCP 12345 and 5000-5020

Description
[Empty text box]

Direction
 In Out

Action
 Allow Block

Program or Service
 Program path:
\$TARGET\$
 Service short name:
[Empty text box]
 Any

Profile
 Any Domain Private Public

Protocol
 Any Tcp Udp

Interface
 Any Wireless Lan Ras

Local IP
 Any IP address
 These addresses (enter IP address, subnet or range):
[Empty text box]

Remote IP
 Any IP address
 These addresses (enter IP address, subnet or range):
[Empty text box]

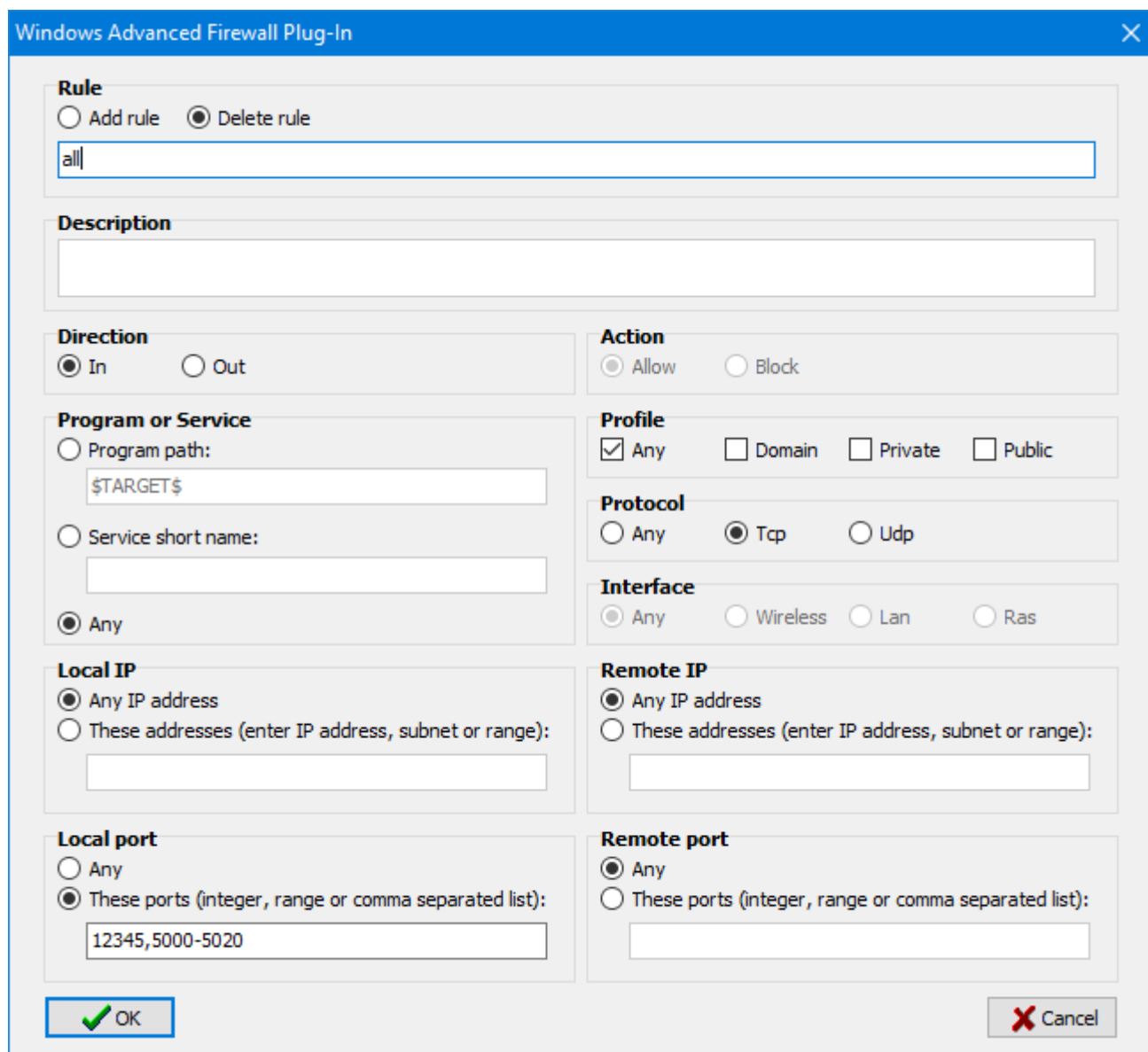
Local port
 Any
 These ports (integer, range or comma separated list):
12345,5000-5020

Remote port
 Any
 These ports (integer, range or comma separated list):
[Empty text box]

AxoNet Plug-Ins for InstallAware

Quick reference guide

On delete, you can use the special rule name "all" to remove all rules matching the criteria in the other parameters:



The screenshot shows the 'Windows Advanced Firewall Plug-In' dialog box. The 'Rule' section has 'Delete rule' selected and the name 'all' entered. The 'Direction' is 'In', 'Action' is 'Allow', 'Profile' is 'Any', 'Protocol' is 'Tcp', and 'Interface' is 'Any'. The 'Local IP' and 'Remote IP' are both set to 'Any IP address'. The 'Local port' is '12345,5000-5020' and 'Remote port' is 'Any'. The 'OK' button is highlighted with a green checkmark.

Section	Option	Value
Rule	Operation	Delete rule
Rule	Name	all
Description	Description	
Direction	Direction	In
Action	Action	Allow
Program or Service	Program path	\$TARGET\$
Program or Service	Service short name	
Program or Service	Any	Selected
Profile	Profile	Any
Protocol	Protocol	Tcp
Interface	Interface	Any
Local IP	Local IP	Any IP address
Remote IP	Remote IP	Any IP address
Local port	Local port	12345,5000-5020
Remote port	Remote port	Any

5.4. Requirements

This plug-in requires Windows Vista or Server 2008 and later. You should wrap the “Configure Windows Advanced Firewall” plug-in command in an “if ... end” section if you create setups for older operating systems.

5.5. Release notes

The add command is cumulative. You can add multiple rules with the same name! Any existing rule is not updated, instead an additional rule is created. This may create unwanted results if you uninstall a previous version but do not remove the existing rules.

To prevent rule duplication, place a delete rule command in the uninstall section

```
Comment: Modify Target System
[DEFINE REGION: Perform Uninstallation]
if Variable REMOVE Equals TRUE
  Comment: Uninstall product
  Comment: TO-DO: Insert any additional uninstall commands here
  Configure Windows Advanced Firewall - delete rule name="Block Outbound Port 80"
  direction=out localport=80 protocol=tcp
  Configure Windows Advanced Firewall - delete rule name="Block WINS" direction=in
  remoteip=wins
  Configure Windows Advanced Firewall - delete rule name="Allow Messenger" direction=in
  program="c:\program files\messenger\mmsgs.exe" remoteip=localsubnet
  Apply Uninstall (get result into variable SUCCESS)
  Set Variable PROGRESS to 100
```

and in the Pre-Requisites section before Install/Remove of your old package:

```
[compiler if Variable BUILDMODE not Equals PATCH]
if Variable NEEDSUPGRADE Equals TRUE
  Set Variable REMOVEOLD to
  Set Variable ERROROLD to
  Configure Windows Advanced Firewall - delete rule name="Block Outbound Port 80"
  direction=out localport=80 protocol=tcp
  Configure Windows Advanced Firewall - delete rule name="Block WINS" direction=in
  remoteip=wins
  Configure Windows Advanced Firewall - delete rule name="Allow Messenger" direction=in
  program="c:\program files\messenger\mmsgs.exe" remoteip=localsubnet
  Install/Remove MSI Package $PRODUCTCODE$[REMOVE=ALL] (get result into variable REMOVEOLD)
```

6. Disk Technology – Plug-In

This plug-in retrieves information about a disk drive. You can specify a drive letter or a directory or a variable like \$TARGETDIR\$

Examples

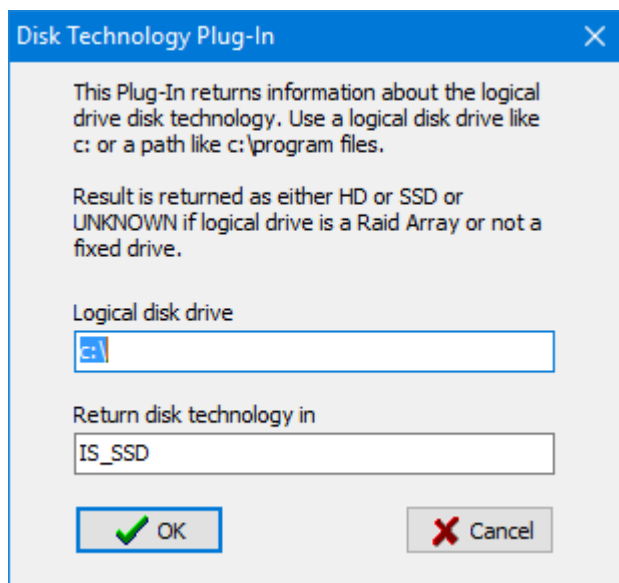
C:\
C:\path\file
\$TARGETDIR\$

The plug-in returns the disk technology. This is one of

HD	if disk is a hard disk
SSD	if disk is a solid-state disk
UNKNOWN	if disk is a virtual disk, a raid array or a storage pool, a mapped network drive, a flash drive or if detection is simply not possible

6.1. Usage

Add the plug-in to your script where you need to retrieve disk technology. Define a variable to hold the retrieved information:



```
Set Variable IS_SSD to  
Detect Disk Technology from 'c:\' (get result into variable IS_SSD)
```

6.2. Requirements

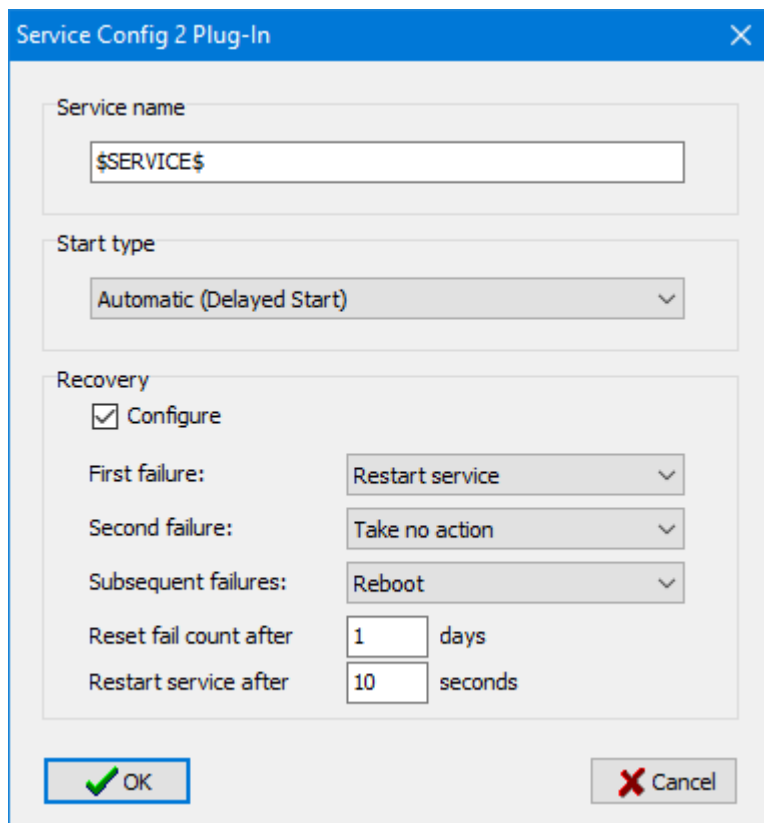
This plug-in requires elevation (admin rights) and Windows Vista and later.

7. Service Config 2 – Plug-In

This plug-in configures the start type and recovery options of an already installed Windows Service.

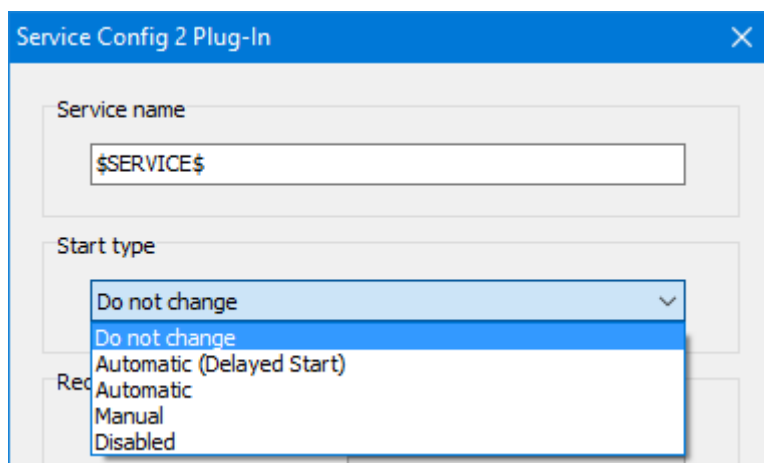
7.1. Usage

Add the plug-in to your script where you need to configure your service. The service must already exist, so a good place is after Apply Install.



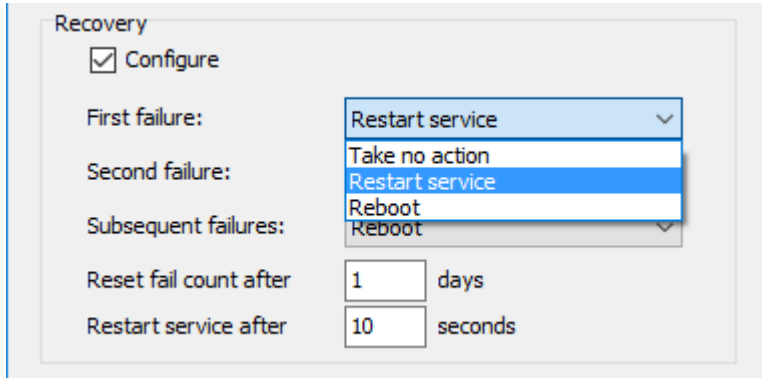
7.2. Start type

Select one of the four possible settings or keep the current setting:



7.3. Service Recovery

Enable "Configure" to set recovery actions for your service. You can set an action for the first, second and all subsequent failures:



Recovery

Configure

First failure: Restart service

Second failure: Restart service

Subsequent failures: Reboot

Reset fail count after: 1 days

Restart service after: 10 seconds

Possible actions are: No action, service restart or reboot of the machine.

7.4. Example

You can configure the start mode depending on the disk technology. On a traditional, slow hard disk, you may want to start the service with a delay:

```
if Variable IS_SSD Equals SSD
  Configure Windows Service - LightsOut2Svc, Start type = Automatic
else
  Configure Windows Service - LightsOut2Svc, Start type = Automatic (Delayed Start)
end
```

To restart a failed service, configure a restart:

```
Set Variable SERVICE to LightsOut2Svc
Configure Windows Service - $SERVICE$, First Failure = Restart service, Second Failure =
Restart service, Subsequent Failures = Restart service, Restart after 10 seconds, Reset
failures after 1 days
```

7.5. Requirements

This plug-in requires elevation (admin rights) and Windows Vista and later.

8. Revision history

Version 3.5.0:

- Added Disk Technology Plug-In
- Added Service Config 2 Plug-In

Version 3.0.1

- Allow variables in rule name for Advanced Firewall – Plug-In

Version 3.0

- Added new Windows Advanced Firewall - Plug-In
- Fixed an uninstall bug in Windows Firewall – Plug-In

Version 2.2

- Fixed UNC path problem in Firewall – Plug-In

Version 2.1

- Added UNC path to Volume Info - Plug-In
- Fixed x64 problem in Upgrade Code - Plug-In

Version 2.0

- Added Windows Firewall - Plug-In

Version 1.0

- First public release