

# Defendpoint PowerShell API

Administration Guide

February 2015

### **Copyright Notice**

The information contained in this document (“the Material”) is believed to be accurate at the time of printing, but no representation or warranty is given (express or implied) as to its accuracy, completeness or correctness. Avecto Ltd, its associated companies and the publisher accept no liability whatsoever for any direct, indirect or consequential loss or damage arising in any way from any use of or reliance placed on this Material for any purpose.

Copyright in the whole and every part of this document belongs to Avecto Ltd (“the Owner”) and may not be used, sold, transferred, copied or reproduced in whole or in part in any manner or form or in or on any media to any person other than in accordance with the terms of the Owner’s Agreement or otherwise without the prior written consent of the Owner.



# Contents

<b>1. Introduction</b>	<b>4</b>
<b>2. Get-DefendpointSettings</b>	<b>5</b>
2.1. Syntax	5
2.2. Detailed Description	5
2.3. Parameters	5
2.4. Input and Return Types	6
2.5. Examples	6
2.5.1. Example 1	6
2.5.2. Example 2	6
2.5.3. Example 3	6
<b>3. Set-DefendpointSettings</b>	<b>7</b>
3.1. Syntax	7
3.2. Detailed Description	7
3.3. Parameters	7
3.4. Input and Return Types	8
3.5. Examples	8
3.5.1. Example 1	8
3.5.2. Example 2	9
3.5.3. Example 3	9
<b>4. Get-DefendpointFileInformation</b>	<b>10</b>
4.1. Syntax	10
4.2. Detailed Description	10
4.3. Parameters	10
4.4. Input and Return Types	11
4.5. Examples	11
4.5.1. Example 1	11
4.5.2. Example 2	11
4.5.3. Example 3	11



## Introduction

The Avecto Defendpoint PowerShell API enables administrators to configure Defendpoint using PowerShell scripts. This enables integrations with external systems, and provides an alternative to using the Avecto Management Consoles.

There are three cmdlets to provide this functionality:

Get-DefendpointSettings	Gets the local, or domain Defendpoint settings.
Set-DefendpointSettings	Saves Defendpoint settings to either Local Group Policy, a local file, or a specified Group Policy object (GPO).
Get-DefendpointFileInformation	Gets information from one or more files that can be used to construct Defendpoint settings.

In addition to this document, there is an accompanying help file which documents the classes which represent a Defendpoint configuration (PowerShell API.chm). That file is in the same directory as this one.



# Get-DefendpointSettings

Gets the local or domain Defendpoint settings.

## 1.1. Syntax

```
Get-DefendpointSettings -LocalPolicy <Boolean> [-UserPolicy <Boolean>] [-XML <Boolean>]  
[<CommonParameters>]
```

```
Get-DefendpointSettings -LocalFile <Boolean> [-FileLocation <String>] [-XML <Boolean>]  
[<CommonParameters>]
```

```
Get-DefendpointSettings -Domain <Boolean> -LDAP <String> [-UserPolicy <Boolean>] [-XML  
<Boolean>] [<CommonParameters>]
```

## 1.2. Detailed Description

The Get-DefendpointSettings cmdlet gets the Defendpoint settings either from the Local Group Policy, from a specified Group Policy Object (GPO), or from a file. The output is a DefendpointSettings object or an XML-formatted string.

## 1.3. Parameters

### **-LocalPolicy <Boolean>**

Gets the Defendpoint settings from the Local Group Policy.

### **-UserPolicy <Boolean>**

If set, the user policy is retrieved. Otherwise computer policy is retrieved.

### **-LocalFile <Boolean>**

Gets the Defendpoint settings from a local Defendpoint settings file that is specified by the FileLocation parameter. If FileLocation is not set, it defaults to the standard location of the local Defendpoint settings file (%PROGRAMDATA%\Avecto\Privilege Guard\ PrivilegeGuardConfig.xml).

### **-FileLocation <String>**

Specifies the location of the Defendpoint settings file. If not set, this defaults to the standard location of the local Defendpoint settings file (%PROGRAMDATA%\Avecto\Privilege Guard\PrivilegeGuardConfig.xml).

### **-Domain <Boolean>**

Gets the Defendpoint settings from the Group Policy Object (GPO) that is specified by the path in the LDAP parameter.

### **-LDAP <String>**

The LDAP path of the Group Policy Object (GPO). For example:

```
"LDAP://DC13.Contoso.com/CN={31B2F340-016D-11D2-945F-  
00C04FB984F9},CN=Policies,CN=System,DC=Contoso,DC=com"
```



## -XML <Boolean>

Specifies that the Defendpoint settings should be output as an XML-formatted string.

## -CommonParameter

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [About Common Parameter](#).

## 1.4. Input and Return Types

The input type is the type of the objects that you can pipe to the cmdlet. The return type is the type of the objects that the cmdlet emits.

Input Type	Parameters listed above
Return Type	By default, Get-DefendpointSettings returns a DefendpointSettings object. If you use the XML parameter, it will return the Defendpoint policy as an XML-formatted string.

## 1.5. Examples

### 1.5.1. Example 1

```
C:\PS>Get-DefendpointSettings -Domain -LDAP "LDAP://DC13.Contoso.com/CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=Contoso,DC=com"
```

Gets the Defendpoint settings of the Group Policy Object specified by the LDAP path.

### 1.5.2. Example 2

```
C:\PS>Get-DefendpointSettings -LocalFile
```

Gets the local Defendpoint settings as a DefendpointSettings object.

### 1.5.3. Example 3

```
C:\PS>Get-DefendpointSettings -LocalPolicy -XML > C:\temp\local.xml
```

Gets the Defendpoint settings on the computer from the Local Group Policy, and outputs this in XML format to the specified file.



# Set-DefendpointSettings

Saves Defendpoint settings to either Local Group Policy, a local file, or a specified Group Policy object (GPO).

## 1.6. Syntax

```
Set-DefendpointSettings -XMLSettings <String> -LocalPolicy [-UserPolicy <Boolean>] [-Merge <Boolean>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-DefendpointSettings -XMLSettings <String> -LocalFile [-FileLocation <String>] [-Merge <Boolean>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-DefendpointSettings -XMLSettings <String> -Domain <Boolean> -LDAP <String> [-UserPolicy <Boolean>] [-Merge <Boolean>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-DefendpointSettings -SettingsObject <DefendpointSettings> -LocalPolicy [-UserPolicy <Boolean>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-DefendpointSettings -SettingsObject <DefendpointSettings> -LocalFile [<String>] [-FileLocation <String>] [-Merge <Boolean>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-DefendpointSettings -SettingsObject <DefendpointSettings> -Domain <Boolean> -LDAP <String> [-UserPolicy <Boolean>] [-Merge <Boolean>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

## 1.7. Detailed Description

The Set-DefendpointSettings cmdlet takes an XML file or Defendpoint settings object as input, and saves it to either a local file, Local Group Policy, or Group Policy Object (GPO). If the Merge parameter is not set, the existing Defendpoint settings in the target location will be overwritten by the new settings.

## 1.8. Parameters

### **-XMLSettings <String>**

The path of a Defendpoint settings XML file that should be used as input.

### **-SettingsObject <DefendpointSettings>**

The DefendpointSettings object that should be used as input. It can be obtained from Get-DefendpointSettings or by creating a new configuration using New-Object.

### **-LocalPolicy <Boolean>**

Saves the Defendpoint settings to the Local Group Policy.

### **-UserPolicy**

If set, the user policy is updated. If not, computer policy is updated.

### **-Merge <Boolean>**

If set, the input settings are merged with the target file. If not, the target file is overwritten.



### **-LocalFile <Boolean>**

Saves the Defendpoint settings to a local file defined by the FileLocation parameter.

If FileLocation is not set, the settings are saved to the local Defendpoint settings file (%PROGRAMDATA%\Avecto\Privilege Guard\PrivilegeGuardConfig.xml).

### **-FileLocation <String>**

The file to save the settings to. If not set, the file is saved to the local Defendpoint settings file (%PROGRAMDATA%\Avecto\Privilege Guard\ PrivilegeGuardConfig.xml).

### **-Domain <Boolean>**

Saves to a Group Policy Object (GPO). Used with the LDAP parameter.

### **-LDAP <String>**

The LDAP path of the Group Policy Object (GPO). For example:

```
"LDAP://DC13.Contoso.com/CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=Contoso,DC=com"
```

### **-Confirm <SwitchParameter>**

Prompts you for confirmation before running the cmdlet.

### **-WhatIf <SwitchParameter>**

Shows what would happen if the cmdlet runs. The cmdlet is not run.

### **-CommonParameter**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [About Common Parameters](#).

## **1.9. Input and Return Types**

The input type is the type of the objects that you can pipe to the cmdlet. The return type is the type of the objects that the cmdlet emits.

Input Type	This cmdlet can take the Defendpoint settings as a DefendpointSettings object or as an XML file containing Defendpoint settings.
Return Type	None

## **1.10. Examples**

### **1.10.1. Example 1**

```
C:\PS>Set-DefendpointSettings -LocalPolicy -XMLSettings C:\Settings.xml
```





### 1.10.2. Example 2

```
C:\PS>Set-DefendpointSettings -XMLSettings C:\Settings.xml -LDAP  
"LDAP://DC13.Contoso.com/CN={31B2F340-016D-11D2-945F-  
00C04FB984F9},CN=Policies,CN=System,DC=Contoso,DC=com"
```

Overwrites the Defendpoint settings in the Group Policy Object specified with those in C:\Settings.xml.

### 1.10.3. Example 3

```
C:\PS>Set-DefendpointSettings -LocalFile | Set-DefendpointSettings -LDAP  
"LDAP://DC13.Contoso.com/CN={31B2F340-016D-11D2-945F-  
00C04FB984F9},CN=Policies,CN=System,DC=Contoso,DC=com" -Merge
```

Gets the local Defendpoint settings, and then merges the policy with the existing Defendpoint settings in the Group Policy Object specified in the LDAP path. See the Merge parameter description for more details on how two settings are merged.



# Get-DefendpointFileInformation

Gets information from one or more files that can be used to construct Defendpoint settings.

## 1.11. Syntax

```
Get-DefendpointFileInformation [-Path] <String[]> [<CommonParameters>]
```

```
Get-DefendpointFileInformation [-Directory] <String> [-FileType <ApplicationType[]>] [-Recurse <Boolean>] [<CommonParameters>]
```

```
Get-DefendpointFileInformation -Com
```

## 1.12. Detailed Description

The Get-DefendpointFileInformation cmdlet retrieves information from a file, or list of files. This information can then be used to construct Defendpoint settings.

## 1.13. Parameters

### **-Path <String[]>**

The path of the file(s). Supports regular expressions.

### **-Directory <String>**

A directory to look in for files. To search all subfolders and files in the directories, include the Recurse parameter.

### **-FileType <ApplicationType[]>**

The file type(s) to search for. The file type options are:

ApplicationType.Executable	Executable
ApplicationType.ControlPanelApplet	Control Panel Applet
ApplicationType.ManagementConsoleSnapin	Management Console Snap in
ApplicationType.InstallerPackage	Installer Package
ApplicationType.WindowsScript	Windows Script
ApplicationType.PowerShellScript	PowerShell Script
ApplicationType.BatchFile	Batch File
ApplicationType.RegistrySettings	Registry Settings



ApplicationType.ActiveXControl	ActiveX
ApplicationType.COMClass	COM Class

### **-Recurse <Boolean>**

Searches all subfolders and files in the directory/directories specified by the Directory parameter.

### **-Com**

Searches the registry for all registered COM objects.

### **-CommonParameter**

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see [About Common Parameters](#).

## **1.14. Input and Return Types**

The input type is the type of the objects that you can pipe to the cmdlet. The return type is the type of the objects that the cmdlet emits.

Input Type	Parameters detailed above
Return Type	This cmdlet returns a list of Application definition objects. Please see the PowerShell API Help file (PowerShell API.chm) for more details.

## **1.15. Examples**

### **1.15.1. Example 1**

```
C:\PS>Get-DefendpointFileInformation -Path "C:\Program Files\Microsoft Office\Office15\*.exe"
```

Returns a list of Application definitions for all the executables in the specified path.

### **1.15.2. Example 2**

```
C:\PS>Get-DefendpointFileInformation -Directory "C:\Program Files\Microsoft Office\Office15"
```

Returns a list of Application definitions for all the application files in the specified directory.

### **1.15.3. Example 3**

```
C:\PS>Get-DefendpointFileInformation -Com
```

Returns a list of Application definitions for all registered COM objects.



# Avecto + You...

## UK

**Hobart House**  
Cheadle Royal Business Park  
Cheadle, Cheshire, SK8 3SR

**Phone** +44 (0)845 519 0114  
**Fax** +44 (0)845 519 0115

## Americas

**125 Cambridge Park Drive**  
Suite 301, Cambridge, MA 02140  
USA

**Phone** 978-703-4169  
**Fax** 978 910 0448



**avecto.com**  
info@avecto.com