Avecto

Defendpoint for Mac Administration Guide

Software Version: 4.5.1.0 GA

Document Version: 1.0

Document Date: 30 May 2017

Avecto

Table of Contents

Chapter 1 - Introduction

Defendpoint is the revolutionary endpoint security software that unites IT and end users. With traditional security solutions such as antivirus only effective half of the time, Defendpoint puts you a step ahead. By combining Privilege Management and Application Control, Defendpoint protects your business from advanced targeted attacks.

With Defendpoint for Mac, users are able to run admin tasks and privileged applications without the need for an admin account. You regain control of apps with pragmatic whitelisting, ensuring that only known good applications are able to run, while users have the freedom and flexibility to perform everyday tasks.

1.1 - Features

Achieve least privilege on Mac

There are many functions that require an admin account to run. While most Mac users typically use an admin account to gain the flexibility they need, this represents a large security risk in the enterprise.

Defendpoint for Mac allows users to log on with standard user accounts without compromising productivity or performance, by allowing the execution of approved tasks, applications and installations as required, according to the rules of your policy.

Empower users and gain control

Allow and block the use and installation of specific applications, binaries, packages and bundles. By taking a simple and pragmatic approach to whitelisting, you can gain greater control of applications in use across the business. This immediately improves security by preventing untrusted applications from executing.

Unlock privileged activity

Even privileged applications and tasks that usually require admin rights are able to run under a standard user account. With Defendpoint for Mac, you can unlock approved system preferences such as date and time, printers, network settings and power management without needing admin credentials.

Take a pragmatic approach with broad rules

Broad catch-all rules provide a solid foundation, with exception handling options to handle unknown activity. Define the application and set its identification options such as filename, hash, publisher or URI. Then assign the application to the users who require enhanced rights and set up any additional options such as end user messaging and auditing.

Achieve compliance

You will have the knowledge to discover, monitor and manage user activity from the entire enterprise, drawing upon actionable intelligence to make informed decisions. Graphical dashboards with real-time data will provide a broad range of reports to aid troubleshooting and provide the information you need to proactively manage your policy on an ongoing basis.

Apply corporate branding

You can add your own branding to messages and prompts, with reusable messaging templates that make it easy to improve the end user experience. You have control over text configuration.

Customizable messaging

Working seamlessly with OS X and macOS, Defendpoint for Mac can suppress standard, restrictive messages and allows you to create your own customized authorization prompts to handle exceptions and enable users to request access. Set up access request reasons, challenge/response codes or password protection to add additional security layers, or simply improve prompts to reduce helpdesk enquiries.

Simple, familiar policy design

Firewall-style rules based on application groups make set up and management simple. Using the same Defendpoint interface and client as for Windows, you can create flexible 'Workstyles' based on the requirements of individuals and groups of users.

1.2 - Supported platforms

- OS X 10.10 Yosemite
- OS X 10.11 El Capitan
- macOS 10.12 Sierra

Ensure all operating system updates are applied.

Chapter 2 - Installation and Deployment

2.1 - Defining User Roles

Defendpoint is an easy solution to deploy but you will want to spend some time preparing suitable workstyles for your users. Implementing least privilege may require workstyles to be tailored to users' roles.

The table below shows three typical user roles, but we recommend that you create roles that are tailored to your environment.

Role	Requirement for Admin Rights
Standard Corporate User	Problem applications that require admin rights to function, and simple admin tasks.
Laptop User	Flexibility to perform ad hoc admin tasks and install software when away from the corporate network.
Technical User	Complex applications and diagnostic tools, advanced admin tasks and software installations.

Defendpoint can cater for all types of users, including the most demanding technical users such as system administrators and developers.

You should also educate users on what they should expect from a least privilege experience, before transferring them to standard user accounts. This ensures that they will report any problems they encounter during the process of moving to least privilege.

2.2 - Defendpoint Software Installation

The Defendpoint installation is performed in two stages; the Defendpoint Management Console is installed on to a Windows computer and the Defendpoint Client is installed on to Mac computers.

2.2.1 - Installing the Defendpoint Management Console

The Defendpoint Management Console is used to create and edit Defendpoint Settings that are applied to Mac computers. The Defendpoint Management Console is an MMC extension snap-in.

Using an administrator account, log on to the Windows computer you would like to manage Defendpoint from.

To install Defendpoint, run the appropriate installation package:

- For 32-bit (x86) systems run *DefendpointManagementConsoles_x86.exe*
- For 64-bit (x64) systems run *DefendpointManagementConsoles_x64.exe*

1. The installation will detect if any prerequisites are needed. Click **Install** to install any missing prerequisites. This may take a few minutes.
2. Once the prerequisites have been installed, the **Welcome** dialog appears. Click **Next** to continue.
3. After reading the license agreement, select **I accept the terms in the license agreement** and click **Next**.
4. Enter your name and the name of your organization and click **Next** .

5. If you want to change the default installation directory then click the **Change** button and select a different installation directory. Click **Next**
6. Leave the **McAfee ePolicy Orchestrator Integration** check box cleared and click **Next**.
7. Click **Install** to start installing the Defendpoint Console.
8. Once installed, click **Finish**. The Defendpoint Console has now been successfully installed.

In order to use the Event Import Wizard, you will need to install the Microsoft SQL Server 2008 R2 Native Client. For installation instructions and to download this component, visit https://www.microsoft.com/en-gb/download/details.aspx?id=16978

2.3 - Installing the Defendpoint Client on the Mac

If you are installing on to OS X 10.10 Yosemite or OS X 10.11 El Capitan and you want Defendpoint to control commands run with sudo, you need to make sure that sudo 1.8 is installed before installing the Defendpoint Client.

The Defendpoint for Mac Client allows Defendpoint Settings to be applied to the Mac computer.

1. Log on to the Mac using a local administrator account and copy the client installer package *DefendpointMacClient.pkg* onto your desktop.
2. Double-click the installation package.

The padlock icon in the top right-hand corner of the installer screen means that the package is signed by Avecto. You can view the certificate by clicking on the padlock. You should not install Defendpoint for Mac Client if the certificate is missing from the installer package.

3. Click **Continue** to start the installer.
4. Click **Continue** to accept the **End User License Agreement**. When the verification prompt appears, click **Agree** to continue.
5. Leave the default installation destination and click **Install** to being the installation.

We recommend that you install Defendpoint for Mac Client to the default location.

6. Once complete, the **Summary** dialog appears. Click **Close** to complete the installation of the Defendpoint for Mac Client.

Chapter 3 - Navigation and Licensing

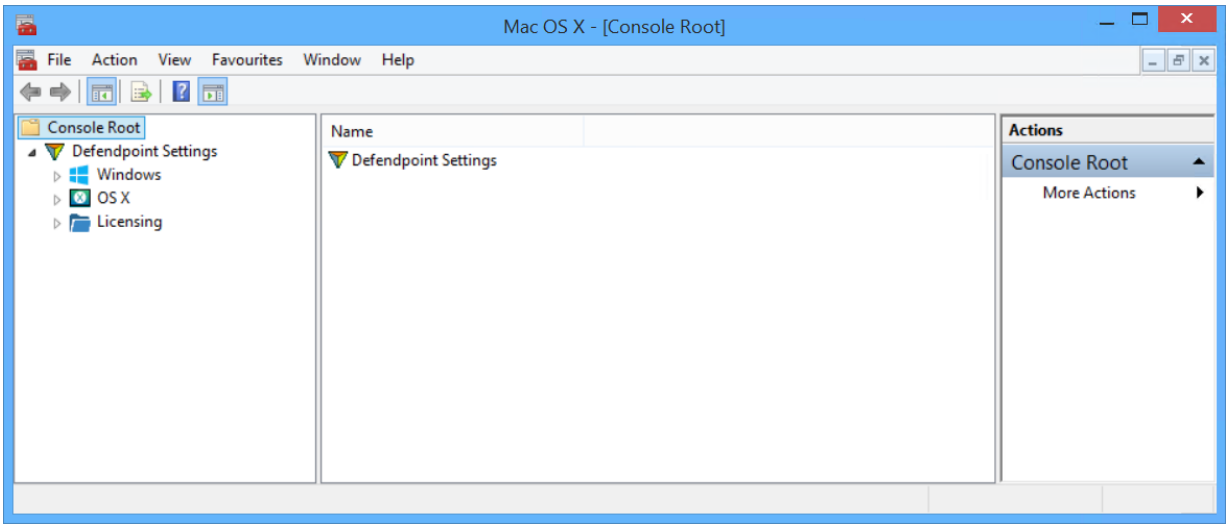
3.1 - Launching the Defendpoint Management Console

The Defendpoint Management Console is accessed as a snap-in to the Microsoft Management Console.

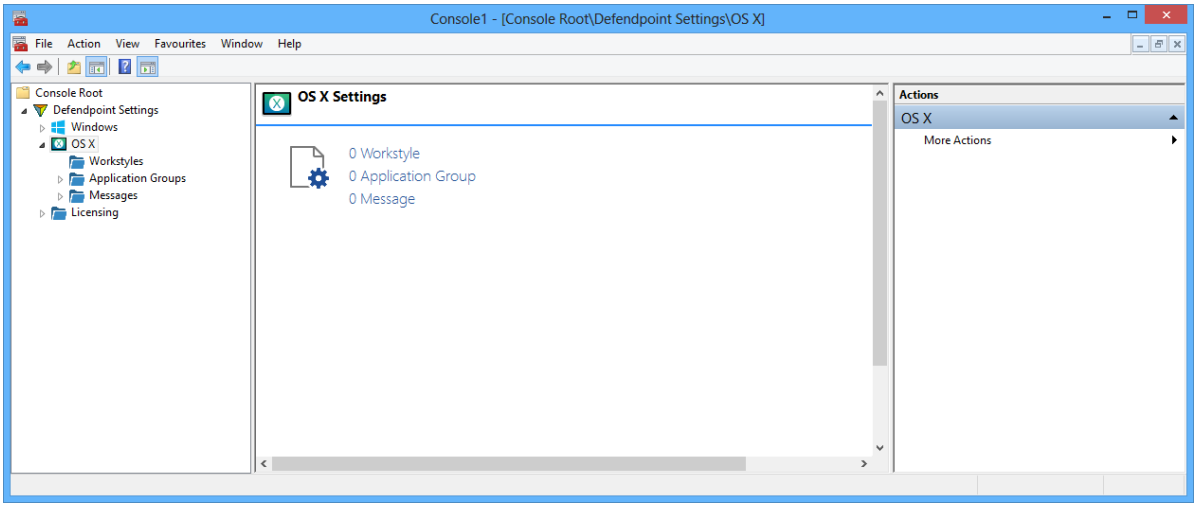
From your administrator account launch the Microsoft Management Console (MMC.exe). Type 'MMC' into the **Search Box** from the **Start Menu** and press the **Enter** key.

We will now add Defendpoint as a snap-in to the console.

1. Select **File** from the menu bar and select **Add/Remove Snap-in**.
2. Scroll down the list and select the **Defendpoint Settings** snap-in. Click **Add** and then click **OK**.
3. Optionally select **File > Save as** and save a shortcut for the snap-in to the desktop as Defendpoint.



4. Expand the **Defendpoint Settings** node in the left-hand pane and select the **OS X** node to display the main screen in the details pane.



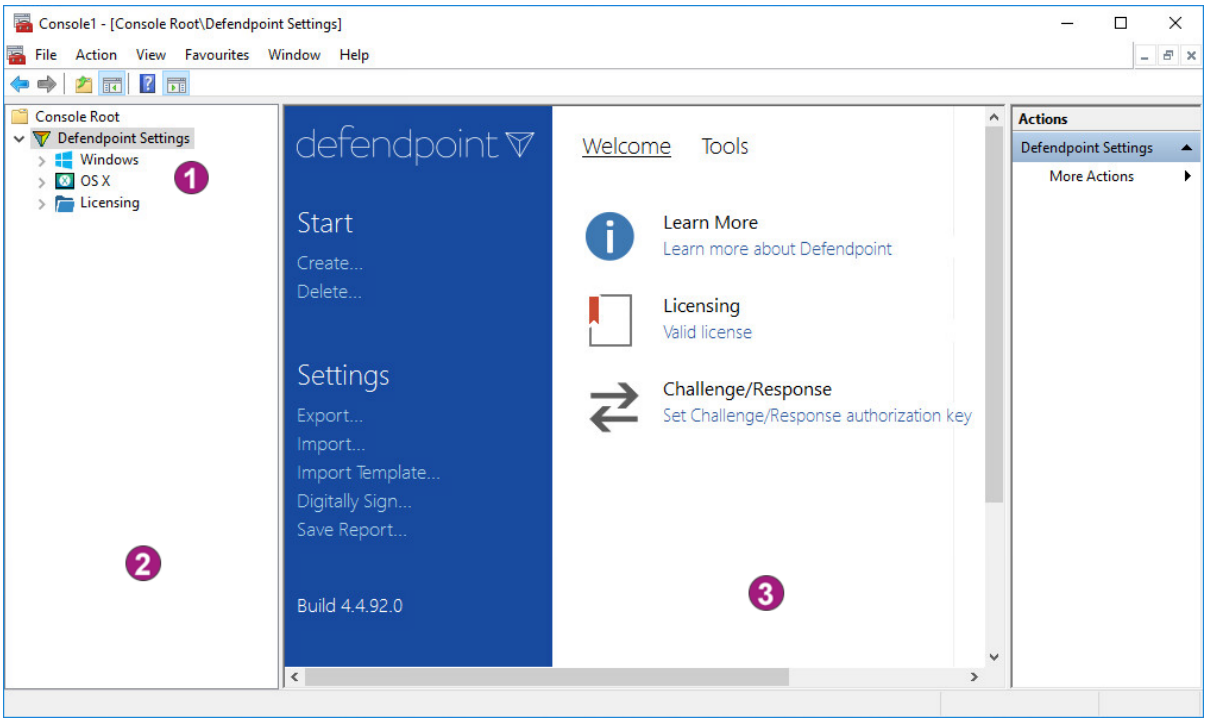
3.2 - Navigating the Policy Editor

3.2.1 - Defendpoint Naming Conventions

The left hand pane containing the Defendpoint Settings is referred to as the **Tree pane**.

The folders beneath Defendpoint Settings in the Tree pane are referred to as **Nodes**.

The middle pane, which displays content relevant to the selected Node, is referred to as the **Details pane**.

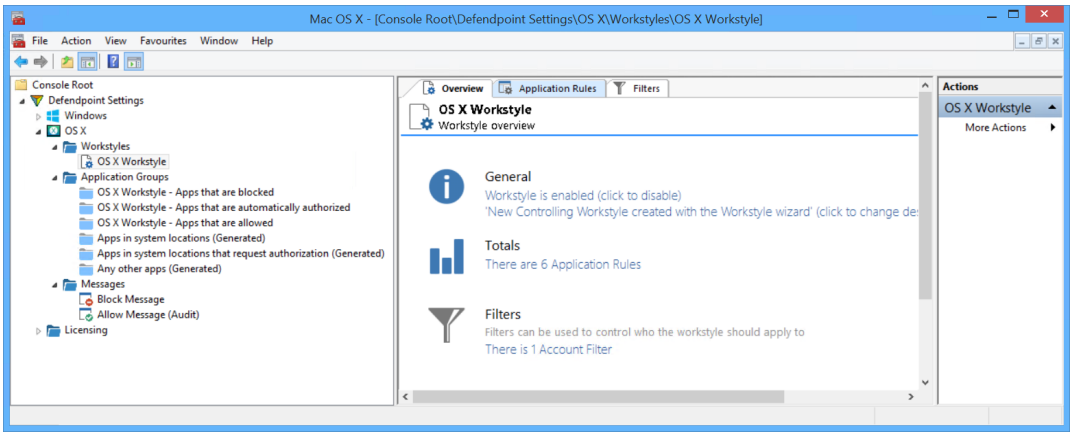


Callout Number	Description
1	Nodes
2	Tree Pane
3	Details Pane

If you expand the **OS X** node you will see three nodes:

1. **Workstyles** - assign privileges to applications.
2. **Application Groups** - define logical groupings of applications.
3. **Messages** - define end user messages.

Once a workstyle has been created and selected in the tree pane, the workstyle tabs will be displayed in the details pane.



1. **Overview** – Provides a general overview of the workstyle contents
2. **Application Rules** – Allows you to insert, edit or remove Application rules
3. **Filters** – Allows you to add or delete Filters

Tabs that contain active settings cannot be toggled off.

3.2.2 - Automatic Saving

By default the Defendpoint Settings editor will automatically save any changes back to the appropriate GPO (or local XML file if you are using the standalone console).

Automatic saving can be disabled, by deselecting the **Auto Commit Settings** menu option on the **Defendpoint Settings** node, but this is not recommended unless you are having performance issues. If you deselect the **Auto Commit Settings** option then you must select the **Commit Settings** menu option to manually save any changes back to the GPO. The **Auto Commit Settings** option is persisted to your user profile, so it will be set for all future editing of Defendpoint Settings.

3.3 - Licensing

The Defendpoint Client will not function unless it receives a valid license code, which needs to be added in the Defendpoint Console.

3.3.1 - Inserting licenses

To insert a license:

1. Expand the **Defendpoint Settings** node.
2. Select the **Licensing** node.
3. Type the license code into the edit box at the top of the licensing page. The edit box will turn from red to green, once you have entered a valid license code. A description of the license code will be displayed and the **Add** button will be enabled.
4. Click **Add** to add the license to the list of current licenses.

Chapter 4 - Workstyles

Defendpoint makes use of workstyles, which are used to assign application rules for a specific user, or group of users. Workstyles are automatically created for you when you use the Workstyle Wizard, and can also be created manually. The Workstyle Wizard will also create auto-generated application rules depending on the type of workstyle you choose to create.

4.1 - Workstyle Wizard

The Workstyle Wizard guides you through the process of creating a Defendpoint workstyle. The options you select determine the function of the workstyle.

Workstyle type

The first choice to make is the type of workstyle you want to create. There are two types of workstyle that can be created in Defendpoint:

- **Controlling workstyle** – allows you to apply rules for access to binaries, bundles, packages, sudo commands and system preference panes.
- **Blank workstyle** – allows you to create an empty workstyle without any predefined elements.

Filtering

The next choice to make is which users the workstyle will be applied to:

- Standard users only
- Everyone, including administrators

The default choice is **Standard users only**. Additional **Account Filters** can be added to the workstyle after it has been created. For more information on Filtering, see **Filtering Workstyles** detailed on page 14.

Workstyle settings

Defendpoint allows you to configure both Privilege Management and Application Control rules, and the Workstyle Wizard lets you configure the rules for each.

4.2 - Creating Workstyles

To create a workstyle:

1. Expand the **Defendpoint Settings** node.
2. Expand the **OS X** node.
3. Right-click the **Workstyles** node and then click **Create Workstyle**. The workstyle wizard will be displayed.
4. Select a workstyle *Type*:
   - **Controlling** - allows you to apply controls for access to applications and privileges.
   - **Blank** - allows you to create an empty workstyle without any predefined elements.
5. Click **Next**.
6. Select a filter for the new workstyle. The default choice is **Standard users only**. If you want to apply the new workstyle to all users (including administrators), select **Everyone, including Administrators**.
7. If you are creating a Controlling workstyle, select one or both Defendpoint Modules and click **Next**.
8. The workstyle wizard will display pages appropriate to the Defendpoint module(s) you selected in Step **7**. Complete the pages relevant to the workstyle type and any modules you have selected.
9. On the final page of the workstyle wizard provide a **Name** and a **Description** for the workstyle. If the workstyle has been configured to use a **Challenge - Response** message you will be asked to enter an authentication key. See **Challenge / Response Authorization** detailed on page 29.
10. Select whether you would like to activate the workstyle now.
11. Click **Finish** to create the workstyle and exit the wizard.

Depending on the type of workstyle you created and any modules that have been included, Defendpoint will auto-generate certain groups and rules, messages and filters. These auto-generated elements are appropriate to the options that are selected in the workstyle wizard.

4.2.1 - Disabling / Enabling Workstyles

When a workstyle is disabled the settings will remain, but the workstyle will be ignored.

To disable a workstyle:

1. Select the workstyle (in the tree pane or details pane).
2. Right-click the workstyle and then click **Disable Workstyle**.

To enable a workstyle (that is currently disabled):

1. Select the workstyle (in the tree pane or details pane).
2. Right-click the workstyle and then click **Enable Workstyle**.

4.2.2 - Workstyle Precedence

If you create multiple workstyles, then those that are higher in the list will have a higher precedence. Once an application matches a workstyle, no further workstyles will be processed for that application, so it is important that you order your workstyles correctly if an application could match more than one workstyle.

To give a workstyle a higher precedence:

1. Right-click the workstyle and then select **Move Up**.
2. Repeat step 2 until you have the workstyle positioned appropriately.

To give a workstyle a lower precedence, follow the procedure above, but click **Move Down**. You may also click **Move Top** or **Move Bottom** to move a rule to the top or bottom of the list.

4.3 - Filtering Workstyles

The **Filters** tab of a workstyle can be used to further refine when a workstyle will actually be applied.

By default, a workstyle will apply to all users/computers who receive it. However, you can add one or more filters that will restrict the application of the workstyle:

- **Account Filter** – this filter restricts the workstyle to specific users or groups of users.

If you want the workstyle to apply only if *all* filters match, select the option **ALL filters must match** from the drop-down list. If you want the workstyle to apply when *any* filter matches, select the option **ANY filter can match** from the drop-down list.

Filters can also be configured to apply if there are no matches. This is referred to as an '*exclude*' filter. To set an exclude filter, right-click the filter and check the option **Apply this filter if it does NOT match**.

4.3.1 - Account Filters

Account filters specify the users and groups the workstyle will be applied to.

When a new workstyle is created, a default Account filter will be added to target either **Standard users only**, or **Everyone (including administrators)**, depending on your selection in the workstyle wizard.

To restrict a workstyle to specific groups or users:

1. On the **Filter** tab click **Add a filter...**.
2. Click **Add an Account Filter**.
3. Select either **Add a new local OS X account** or **Add a new domain account**.
4. The **Add Account** dialog will appear.
5. Enter the relevant User or Group details and click **OK**.

By default, an Account filter will apply if any of the User or Group accounts in the list match the user. If you have specified multiple User and Group accounts within one Account filter, and want to apply the workstyle only if ALL entries in the Account filter match, then check the option **All items below should match**.

You may add more than one Account filter if you want the user to be a member of more than one group of accounts for the workstyle to be applied.

If an Account filter is added, but no User or Group accounts are specified, a warning will be displayed advising **No accounts added**, and the Account filter will be ignored.

If **All items below should match** is enabled, and you have more than one User account listed, the workstyle will never apply as the user cannot match two different User accounts.

Chapter 5 - Managing Applications

Application groups are used to define logical groupings of applications.

Application groups are assigned to workstyles, so you must define Application groups for all of the applications you want to assign to a workstyle.

5.1 - Creating Application Groups

To create an application group:

1. Expand the **OS X** node.
2. Right-click the **Application Groups** node and then click **New Application Group**.
3. A new application group will be created (**Application Group 1**). You can rename the group by double-clicking on the group name. You can now add applications to the application group.

Application Group Description

You may set a description for an application group by accessing the application group properties:

1. Right-click the **Application Group** and then click **Properties**.
2. Set the **Description** in the **Properties** dialog.
3. Click **OK**.

5.2 - Inserting Binaries

To insert a binary:

1. Select the relevant application group.
2. Right-click in the details pane to access the context menu.
3. Select **Insert Application** and then select **Binary...** from the sub-menu.
4. After selecting an application type to insert, the **Insert Application** wizard is launched.
5. Enter a **File or Folder Name** for the application or click the **Template...** button. For more information about Application Templates see **Inserting Applications from Templates** detailed on page 21. Click **Next**.
6. Enter a description for the application (the description will automatically be extracted from the file you entered, if it has a description). Click **Next**.
7. Configure the **Application Definitions** for the application. For information about application definitions see below.
8. Click **Next** and **Finish**.

It's important to select a file for the application type you have chosen, otherwise it will fail to match when the Defendpoint Client processes the application group.

The **Insert Application** wizard provides various application definitions. The Defendpoint Client must match every definition you configure before it triggers a match (the rules are combined with a logical AND). The following definitions are available:

- File or Folder Name
- File Hash (SHA-1 Fingerprint)
- Command Line Arguments
- Publisher
- Parent Process
- Application requests authorization

5.3 - Inserting Application Bundles

To insert an application bundle:

1. Select the relevant application group.
2. Right-click in the details pane to access the context menu.
3. Select **Insert Application** and then select **Bundle…** from the sub-menu.
4. After selecting an application type to insert, the **Insert Application** wizard is launched.
5. Enter a **File or Folder Name** for the application or click the **Template…** button. For more information about Application Templates see **Inserting Applications from Templates** detailed on page 21. Click **Next**.
6. Enter a description for the application (the description will automatically be extracted from the file you entered, if it has a description). Click **Next**.
7. Configure the **Application Definitions** for the application. For information about application definitions see below.
8. Click **Next** and **Finish**.

It's important to select a file for the application type you have chosen, otherwise it will fail to match when the Defendpoint Client processes the application group.

The **Insert Application** wizard provides various application definitions. The Defendpoint Client must match every definition you configure before it triggers a match (the rules are combined with a logical AND). The following definitions are available:

- File or Folder Name
- File Hash (SHA-1 Fingerprint)
- Publisher
- Parent Process
- Source
- File Version matches
- URI
- Application requests authorization

5.4 - Inserting System Preference Panes

To insert a system preference pane:

1. Select the relevant application group.
2. Right-click in the details pane to access the context menu.
3. Select **Insert Application** and then select **System Preference Pane…** from the sub-menu.
4. After selecting an application type to insert, the **Insert Application** wizard is launched.
5. Enter a **URI** for the application or click the **Template…** button. For more information about Application Templates see **Inserting Applications from Templates** detailed on page 21. Click **Next**.
6. Enter a description for the application (the description will automatically be extracted from the file you entered, if it has a description). Click **Next**.
7. Configure the **Application Definitions** for the application. For information about application definitions see below.
8. Click **Next** and **Finish**.

The **Insert Application** wizard provides various application definitions. The Defendpoint Client must match every definition you configure before it triggers a match (the rules are combined with a logical AND). The following definitions are available:

- File or Folder Name
- File Hash (SHA-1Fingerprint)
- Publisher
- Source
- File Version matches
- Application requests authorization

Some third-party preference panes may behave as a Bundle application type when their padlocks are clicked by a user. If you are unable to match the preference pane as a Preference Pane type, check the audit log and use a Bundle type to match the preference pane, using the 'Path' property that has been audited by Defendpoint. For more information please see **Advanced Settings** detailed on page 45.

5.5 - Inserting Packages

To insert a package:

1. Select the relevant application group.
2. Right-click in the details pane to access the context menu.
3. Select **Insert Application** and then select **Package…** from the sub-menu.
4. After selecting an application type to insert, the **Insert Application** wizard is launched.
5. Enter a **File or Folder Name** for the application. Click **Next**.
6. Enter a description for the application (the description will automatically be extracted from the file you entered, if it has a description). Click **Next**.
7. Configure the **Application Definitions** for the application. For information about application definitions see below.
8. Click **Next** and **Finish**.

It's important to select a file for the application type you have chosen, otherwise it will fail to match when the Defendpoint Client processes the application group.

The **Insert Application** wizard provides various application definitions. The Defendpoint Client must match every definition you configure before it triggers a match (the rules are combined with a logical AND). The following definitions are available:

- File or Folder Name
- File Hash (SHA-1Fingerprint)
- Publisher
- Application requests authorization

Only one package may be opened by the Installer at any one time. Any attempt to open and install a second package whilst the first package is open will result in the termination of the Installer and neither package will be installed; any package that was in the process of installing will be interrupted and the installation may be incomplete.

5.6 - Inserting Sudo Commands

If sudo command rules are not configured in Defendpoint, then the Sudoers file is used to determine the user's ability to run sudo commands.

To insert a sudo command:

1. Select the relevant application group.
2. Right-click in the details pane to access the context menu.
3. Select **Insert Application** and then select **Sudo Command...** from the sub-menu. The **Insert Application** wizard is launched.
4. Enter a **File or Folder Name** for the application. Click **Next**.
5. Enter a description for the application (the description will automatically be extracted from the file you entered, if it has a description). Click **Next**.
6. Configure the **Application Definitions** for the application. For information about application definitions see below.
7. Click **Next** and **Finish**.

It's important to select a file for the application type you have chosen, otherwise it will fail to match when the Defendpoint Client processes the application group.

The **Insert Application** wizard provides various application definitions. The Defendpoint Client must match every definition you configure before it triggers a match (the rules are combined with a logical AND). The following definitions are available:

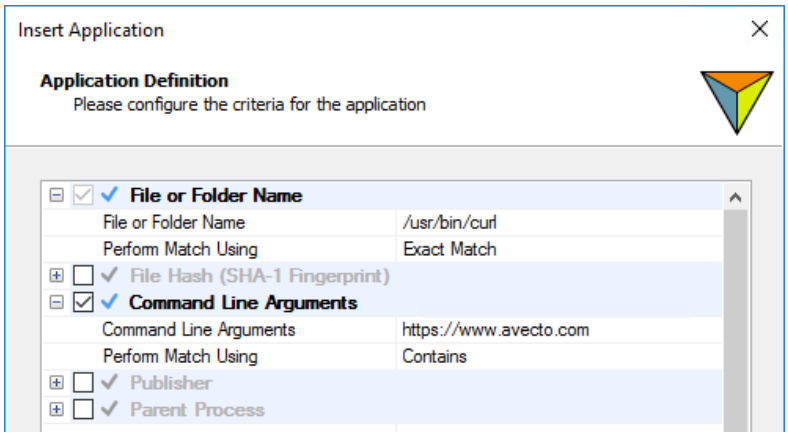
- File or Folder Name
- File Hash (SHA-1 Fingerprint)
- Command Line Arguments
- Publisher
- Parent Process

When configuring sudo command rules, use the File or Folder Name application definition to define the command you want to control when it's run with sudo.

For example, to control the `curl` command you need to set up a sudo command application rule so that:

- The **File or Folder Name** definition is set to '/usr/bin/curl' with the **Perform Match Using** set to 'Exact Match'
- The **Command Line Arguments** definition is set to the server or website that you want to control using this rule

The following screenshot shows the application definition for controlling `curl` commands that transfer data to or from the Avecto website:



We recommend using explicit paths in your application definitions, for example using `/usr/bin/curl` instead of `curl` and avoid using ~ to refer to the user's home directory.

5.6.1 - Sudo Switches

Defendpoint supports running sudo commands with the following switches:

- **-b**, **--background**
- **-e**, **--edit** – this switch needs configuring in Defendpoint for it to be supported, see **Edit -e Switch** detailed on the next page
- **-i**, **--login**
- **-S**, **--stdin**
- **-s**, **--shell**
- **-V**, **--version**

When a sudo command is run, Defendpoint ignores any switches that have been used and will match the rest of the command against the application definition. If Defendpoint matches against a rule that allows execution, the sudo command runs with any supported switches that were used. Any switches that are not supported by Defendpoint are ignored.

If Defendpoint matches on a passive rule or doesn't match any rules, then the sudo command runs with any supported or unsupported switches that have been used.

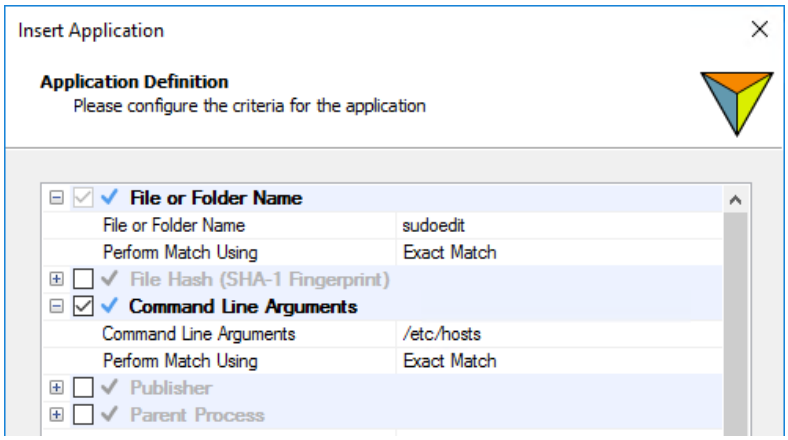
The -l --list switch, which lists the commands that the user is allowed to run, does not take into account the commands that are restricted by Defendpoint.

Edit -e Switch

The -e --edit switch, also known as sudoedit, allows the user to edit one or more files using their preferred text editor. The text editor is defined by setting the SUDO_EDIT, VISUAL or EDITOR environment variable in their Terminal session. Otherwise, the default editor, Vim, is used. To configure your policy to support the -e switch, you need to set up a sudo command application rule so that:

- The **File or Folder Name** definition is set to 'sudoedit' with the **Perform Match Using** set to 'Exact Match'
- The **Command Line Arguments** definition is set to the path of the file(s) that you want to control using this rule

For example, the application definition shown in the following screenshot supports the sudo command `sudo -e /etc/hosts`.



5.7 - Inserting Applications from Templates

Application Templates provide a simple way to pick from a list of known applications. A standard set of templates are provided that cover binaries, bundles and system preference panes.

Each category has a list of applications. Picking an application will cause the Application dialogs to be pre-populated with the appropriate information.

To insert an Application Template:

1. Select the relevant Application group.
2. Right-click the applications list in the Details pane to access the context menu.
3. Select **Insert Application** and then select **Application Template** from the sub-menu.
4. The **Application Template** dialog will appear.
5. Use the search box to locate a specific Application Template or scroll through the available templates.
6. Select one or more Application Templates (using the **Ctrl** key).
7. Click **OK** to add the selected Application Templates to the Application group.

Application templates can also be added from within the **Insert Application** wizard, by clicking **Template...**. When you launch an Application Template from the **Insert Application** wizard, the template browser only shows the templates that are for the type of application you are inserting. For more information, see **Application Templates** detailed on page 44.

5.8 - Inserting Applications from Events

Defendpoint for Mac can be configured to send application events to iC3. The **Event Import** wizard can then be used to import events into Application Groups and create application definitions based on the properties collected by an audit event. The wizard provides a simple and convenient way to find specific applications based on any or all of the following search criteria:

- **Event Type** – The type of event you are interested in. Choose either: *Any application*, or choose from one of the following:
  - Applications that requested authorization
  - Applications that were blocked
  - Sudo commands and their command line arguments
  - Binaries and their command line arguments
- **Timeframe** – the period of time to search for applications. Choose from one of the following:
  - **From** – Pick a range starting from a predefined time period. From here you may also choose Anytime, to include all events.
  - **Specific period**... – Pick an optional **From** and **To** date to include events collected during that period of time.

Once the search criteria has been entered, the wizard will return a list of unique applications that were audited, matching the criteria you specified. From here you can browse the list, or to find a particular application you can type into the Search field to instantly filter the list based on the text you enter.

Once you have found an application or applications, select (or multi-select by holding down the **Control** or **Shift** key while selecting) and then click **OK** to create new application definitions from your selection.

Once the definitions have been created, you can edit the definition and modify the matching criteria. All matching criteria will be pre-populated with values collected from the application.

5.9 - Application Rules

Application Rules are applied to applications that are launched either directly by the user or by a running process. They can be created and edited from the workstyle **Application Rules** tab. If you have a blank workstyle you can create rules from the workstyle **Overview** tab.

The **Application Rules** tab can be used to enforce rules for whitelisting, monitoring and application discovery, and handling OS X and macOS Authorization Requests.

Each rule has a number of elements:

Rule

- **Target Application Group** – the Application Group that the rule is associated with.
- **Action** – The action that the rule dictates once a match has been made.
- **End User Message** – Any message that may be displayed to the user.

Auditing

- **Raise an Event** – An event will be logged to the client machine's syslog file on OS X, or the Unified Logging database on macOS.

- **Raise an Event and send to iC3** – An event will be sent to iC3 via the iC3 Agent Adapter, if the adapter is installed.

5.9.1 - Inserting an Application Rule

To insert an application rule:

1. Select the relevant workstyle in the tree pane.
2. Select the **Application Rules** tab in the details pane.
3. Right-click in the **Application Rules** tab and click **Insert Application Rule…**. The **Create Application Rule** dialog appears.
4. Select the relevant application group from **Target Application Group > Click to select…** drop-down menu.
5. Select the desired **Action**: **Passive (No Change)**, **Allow Execution**, or **Block Execution**.
6. If you want to prompt the user before the application is executed or blocked, then select a message or notification from **Show End User Message**. The list will show **Allow** or **Block** messages depending on your choice in the previous step. This option is disabled if you chose **Passive (No Change)** in the previous step. For more information see **End User Messaging** detailed on page 24.
7. If you want to audit the Application rule being matched, then select **On** for **Raise an Event**. This will log an event to the syslog for each application or activity that matches the rule.

5.9.2 - Application Rule Precedence

If you add more than one Application rule to a workstyle, then entries that are higher in the list will have a higher precedence. Once an application matches an Application rule, no further rules or workstyles will be processed. If an Application could match more than one workstyle or rule, then it is important that you order both your workstyles and rules correctly.

To give a rule a higher precedence within a workstyle:

1. Right-click the rule and then select **Move Up**.
2. Repeat step 1 until you have the **Rule** positioned appropriately.

To give a rule a lower precedence, follow the procedure above, but click **Move Down**. You may also click **Move Top** or **Move Bottom** to move a rule to the top or bottom of the list.

The **Summary View** and **Detail View** can be used to show information about your rules in either graphical form or in table form.

Chapter 6 - End User Messaging

You can define any number of end user messages and notifications. Messages can optionally be displayed when a user's action matches a rule. Messages can be displayed for both allow and block actions.

Messages provide an effective way of alerting the user before an action is performed. For example, advising that an application launch has been blocked or allowed, or that it needs to be authorized before it will launch.

Messages give the user information about the application and the action taken, and can be used to request information from the user. Messages also allow authorization and authentication controls to be enforced before access to an application is granted.

Messages are customizable with visual styles, corporate branding and display text, so you are offered a familiar and contextual experience. When running sudo commands on a Mac, text-only versions of any configured messages appear in the Terminal. Messages are assigned to Application Rules.

Once defined, a message may be assigned to an individual rule in the tab by editing the rule. Depending on the type of workstyle you've created, Defendpoint may auto-generate certain messages for you to use.

6.1 - Creating Messages

To create a message:

1. Select the **Messages** node.
2. Right-click the **Messages** node and select **New Message…**.
3. The **New Message** wizard will appear.
4. Select a message template from the **Use a Message Box template** drop-down list.
5. Click **Next**.
6. Customize the message (more advanced message configuration can be performed after the message has been created).
7. Click **Finish**.

A new message will be created under the **Messages** node. You can rename the message by double-clicking on the message name.

You may now further refine the message by selecting it and editing the properties which are displayed in the right hand pane under the **Message Design** and the **Message Text** tabs.

6.2 - Message Name and Description

You can set a description for a message by accessing the properties for a message:

1. Right-click the **Message** in the tree pane and select **Properties**.
2. Set the **Description** in the **Properties** dialog.
3. Click **OK**.

6.3 - Message Design

Messages have a wide array of configuration options, which are detailed below.

As you change the various message options the preview message will automatically be updated. To test the message box, use the preview facility (program and content information will contain appropriate placeholders). You cannot currently preview the Terminal version of a message.

Although the preview shows the message box in a Windows-style format, the messages will appear in a Mac-style format on the endpoints.

Once you have configured the message options you should configure the **Message Text** for the message.

6.3.1 - Miscellaneous Settings

- **Show message on secure desktop** – this option is not configurable for OS X or macOS messages.

6.3.2 - Message Header Settings

- **Header Style** – select the type of header, which can be No header, Defendpoint, Warning, Question or Error.
- **Show Title Text** – determines whether to show the title text.
- **Text Color** – select the color for the title text (the automatic color is based on the Header Style).
- **Background Type** – set the background of the header, which can be Solid background, Gradient background or Custom image. (The default **Background Type** is *Custom Image* making the **Color 1** and **Color 2** options initially unavailable).
- **Color 1** – select the color for a **Solid background** or the first color for a **Gradient background** (the automatic color is based on the **Header Style**).
- **Color 2** – select the second color for a **Gradient background** (the automatic color is based on the selected **Header Style**).
- **Custom Image** – select the image for a **Custom image** background. This option is only enabled if you have selected **Custom Image** for the **Background Type**. Click the "**...**" button to import, export, modify or delete images using the **Image Manager**.

6.3.3 - Message Body Settings

The **Message Body Settings** display specific information about the program or content. These options are not configurable for OS X or macOS messages.

6.3.4 - User Reason Settings

This option determines whether to prompt the end user to enter a reason before an application launches (**Allow Execution** message type) or to request a blocked application (**Block Execution** message type).

- **User Reason Type** – set this option to **Text box** to allow users to write a reason or request.
- **Remember User Reasons (per-application)** – this option is not configurable for OS X or macOS messages.

6.3.5 - User Authorization

- **Authorization Type** – set this option to **User must authorize** to force the user to re-authenticate before proceeding. If you wish to use this option for over the shoulder administration, then set this option to **Designated user must authorize**.
- **Authentication Method** – if the **Authorization type** has been set to **User must authorize** or **Designated user must authorize** then this option will be set to **Password only**.
- **Designated Users** – if the **Authorization Type** has been set to **Designated user must authorize** then click the "**…**" button to add one more user accounts or groups of users that will be allowed to authorize the message.

6.3.6 - Sudo User Authorization

- **Don't ask for password if already entered** – this text option determines the time period during which a user will not be asked to re-enter their password when running sudo commands. This text option is only enabled if the Authorization Type has been set to **User must authorize** or **Designated user must authorize**, see **User Authorization** detailed above. The available options are:
  - **Ask every time**
  - **Less than 1 minute ago**
  - **Less than 5 minutes ago**
  - **Less than 15 minutes ago**
  - **Only ask once per session**

6.3.7 - Challenge / Response Authorization

- **Enabled** – set this option to **Yes** to present the user with a challenge code. In order for the user to proceed, they must enter a matching Response code. Note that when this option is enabled for the first time, you will be requested to enter a Shared Key. For more information, see **Challenge / Response Authorization** detailed on page 29.
- **Authorization Period (per-application)** - this option is not configurable for OS X or macOS messages.
- **Suppress messages once authorized** - this option is not configurable for OS X or macOS messages.
- **Show Information tip** – this option is not configurable for OS X or macOS messages.
- **Maximum Attempts** – this option is not configurable for OS X or macOS messages.

6.3.8 - Authorization Settings

- **Designated User and Challenge both required** – this option is not configurable for OS X or macOS messages.

6.3.9 - Email Settings

- **Allow user to email an application request** – this option is not configurable for OS X or macOS messages.
- **Mail To** – this option is not configurable for OS X or macOS messages.
- **Subject** – this option is not configurable for OS X or macOS messages.

6.3.10 - Message Design Options for Terminal Messages

The following message design options can be applied to messages displayed in the Terminal by Defendpoint:

- **Message Header Settings** detailed on page 25
  - **Show Title Text**
- **User Reason Settings** detailed on page 25
  - **User Reason Type**
- **User Authorization** detailed on the previous page
  - **Authorization Type**
  - **Authorization Method**
  - **Designated Users**
- **Sudo User Authorization** detailed on the previous page
  - **Don't ask for password if already entered**
- **Challenge / Response Authorization** detailed on the previous page
  - **Enabled**

6.4 - Message Text

Some of the text in the message can be configured in the **Message Text** section. For more information on the text options that can be configured for OS X and macOS messages see **OS X and macOS Message Text Options** detailed on the next page.

As you change the message text the preview message will automatically be updated, based on the selected language. To test the message box, click the preview message (any program or content information will contain placeholders). You cannot currently preview Terminal messages.

6.4.1 - Setting the Message Text

New messages include default strings for each setting, in the English language. If you want to customize your message text, we recommend editing the message text configuration.

To set the message text strings, edit the text values in the text property grid.

6.4.2 - OS X and macOS Message Text Options

The following list contains all of the message text options that are configurable for OS X and macOS messages:

- **General**
  - **Header Message**
  - **Body Message**
- **Publisher**
  - **Verification Failure**
- **User Reason**
  - **Reason Error Message** – this option can only be configured for Terminal messages.
- **User Authorization**
  - **Unauthorized Credentials** – this option can only be configured for Terminal messages.
- **Challenge / Response Authorization**
  - **Hint Text** – this option can only be configured for Terminal messages.
  - **Error Message Text** – this option can only be configured for Terminal messages.

6.4.3 - Message Text Options for Terminal Messages

The following message text options can be applied to messages displayed in the Terminal by Defendpoint:

- **General**
  - **Header Message**
  - **Body Message**
- **User Reason**
  - **Reason**
  - **Reason Error Message**
- **User Authentication**
  - **User name**
  - **Password**
  - **Unauthorized Credentials**
- **Challenge / Response Authorization**
  - **Hint Text**
  - **Information Tip Text**
  - **Error Message Text**
- **Buttons**
  - **OK Button**
  - **Cancel Button**

6.5 - Managing Languages

Currently, you can only configure one language for Defendpoint for Mac.

6.6 - Image Manager

The Image Manager associated with message creation allows you to **Add**, **Modify**, **Export** and **Delete** images that are referenced in message headers.

All images are stored inside the workstyles as compressed and encoded images.

It is strongly recommended that you delete any unused images to minimize the size of the policies, as Defendpoint does not automatically delete unreferenced images.

The Image Manager is only accessible when the **Background Type** field on the **Message Design** tab is set to **Custom Image**. The **Custom Image** field is then enabled. Click the '**...**' button to the right. The **Manage Images** dialog will be displayed.

To add an image to a message:

1. Click **Add**.
2. The **Image Properties** dialog will appear.
3. Click **Import**.
4. Browse for an image and click **Open**.
5. Set a description for the image.
6. Click **OK**.

To modify an image:

1. Select the image in the list and click **Modify**.
2. The **Image Properties** dialog will appear.
3. Click **Import**.
4. Alter the description and click **OK**.

To export an image:

1. Select the image in the list and click **Export**.
2. Browse to a folder and click **Save**.

To delete an image:

1. Select the image in the list and click **Delete**.
2. When prompted, click **Yes** to delete the image.

If an image is referenced by any messages then you will not be allowed to delete it.

6.7 - Challenge / Response Authorization

Challenge / Response Authorization provides an additional level of control for access to applications and privileges, by presenting users with a 'challenge' code in an End User Message. In order for the user to progress, they must enter a corresponding 'response' code into the message.

Challenge / Response Authorization is configured as part of an End User Message, and can be used in combination with any other authorization and authentication features of Defendpoint messaging.

Users will be presented with a different, unique challenge code each time a Challenge / Response message is displayed.

Challenge and response codes are presented as an 8 digit number, which is ideal for verbal communication with a telephone helpdesk, and minimizes the chance of incorrect or accidental entry.

If a user cancels the message, the challenge code is invalidated and a new code will be presented if the user runs the same application.

For more information on configuring Challenge / Response Authorization enabled End User Messages, see **Message Design** detailed on page 25.

6.7.1 - Shared Key

The first time you create a Defendpoint End User Message with a Challenge you are asked to create a Shared Key. The Shared Key is used by the Defendpoint Client to generate Challenge codes at the end point. The Shared Key is also required to generate the response code to match a challenge code created with the same key.

Once you have entered an Shared Key, it will be applied to all End User Messages in the same Defendpoint Settings, for all messages that have Challenge / Response Authorization enabled.

To change the Shared Key:

1. Right-click the **Defendpoint Settings** node and choose **Set Challenge / Response Authorization Key...**.
2. In the **Challenge / Response Authorization Key** dialog, edit the **Enter Key** and **Confirm Key** with the new Authorization Key.
3. Click **OK** to complete. If the keys entered don't match, you will be presented with a warning message.

We recommend that your Shared Key is at least 15 characters and includes a combination of alphanumeric, symbolic, upper, and lowercase characters. As a best practice, the Shared Key should be changed periodically.

6.7.2 - Generating a Response Code

Response codes are generated using `PGChallengeResponseUI.exe`, which is installed as part of the Defendpoint Management Console installation, and is located in the following directory:

```
C:\Program Files\Avecto\Privilege Guard Management Consoles\
```

To generate a response code using the PGChallengeResponseUI utility:

1. Run the program `PGChallengeResponseUI.exe`.
2. In **Enter shared key**, enter the Shared Key you defined earlier, and in **Enter challenge code**, enter the challenge code presented to the user.
3. The response code will automatically be displayed once both the **Shared Key** and the 8 character challenge code have been entered.

The **Generated Response** value is then entered into the **End User Message** which presented the corresponding challenge.

PGChallengeResponseUI.exe is a standalone utility and can be distributed separately from the Defendpoint Management Console.

Generating a Response Code from the command line

Response codes can also be generated from the command line using the PGChallengeResponse.exe command line utility, which is installed as part of the Defendpoint Management Console installation, and is located in the following directory:

```
C:\Program Files\Avecto\Privilege Guard Management Consoles\
```

To generate a response code from the command line:

1. Open the Command Prompt by clicking the Start Menu and typing `cmd.exe`.
2. In the Command Prompt, type the following command, then press Enter: `cd "\program files\avecto\privilege guard management consoles"`
3. Once you have opened the Privilege Guard Management Consoles directory, type the following command (where `<challenge>` is the challenge code presented to a user): `pgchallengeresponse.exe <challenge>`
4. At the Authorization Key prompt, enter the correct Shared Key, then press **Enter**.

PGChallengeResponseUI.exe is a standalone utility and can be distributed separately from the Defendpoint Management Console.

Automating Response Code Generation

The `PGChallengeResponse.exe` utility supports full command line use, allowing it to be easily integrated into any third party workflow that supports the execution of command line executables. The command line is as follows:

```
PGChallengeResponse.exe <challenge code> <authorization key>
```

Where `<challenge code>` is the code presented to the user and `<authorization key>` is the key that was configured within the Defendpoint Settings which presented the End User Message.

The utility will return the response code as an exit code, so it can be captured from within a custom script or wrapper application. Below is an example VBScript:

```
Dim WshShell, oExec
Dim strChallenge,strKey,strExecutable
strExecutable = "C:\Program Files\Avecto\Privilege Guard Management
Consoles\PGChallengeResponse.exe"
strChallenge = InputBox("Enter Challenge Code","Challenge")
strKey = InputBox("Enter Authorization Key","Key")
Set WshShell = WScript.CreateObject("WScript.Shell")
Set oExec = WshShell.Exec(strExecutable & " " & strChallenge & " " & strKey)
Do While oExec.Status = 0
WScript.Sleep 100
Loop
msgbox "Response Code: " & oExec.ExitCode
Set WshShell = Nothing
Set oExec = Nothing
```

Chapter 7 - Defendpoint Settings Management

Defendpoint for Mac settings can be deployed to OS X and macOS endpoints using the Avecto iC3 platform, or you can use your own strategy to deploy an XML configuration file.

7.1 - Using iC3 to Manage Settings

For information on creating and editing Defendpoint settings in iC3, please refer to the section *iC3 Policies in Defendpoint Settings (iC3) MMC* in the **iC3 Getting Started Guide**.

7.2 - Manual Deployment of Settings via XML File

Defendpoint settings can be exported from the MMC as a standalone XML configuration file, which can be distributed to OS X and macOS endpoints using your own deployment strategy.

To export the Defendpoint Settings to an XML file:

1. Select the **Defendpoint Settings** node.
2. Right-click and select **Export…**.
3. Select an appropriate destination for the exported XML file, ensuring the file is named **pguard.xml**.

7.2.1 - Adding Defendpoint Settings to a Mac Client computer

Defendpoint settings are stored in the file `/etc/pguard/pguard.xml`, and can be overwritten with an exported XML file from the MMC. To prevent any invalid permissions being applied, it is recommended that this file is replaced using the following command. In this example, the source XML file is located on your Desktop:

```
sudo cp ~/Desktop/pguard.xml /etc/pguard/pguard.xml
```

The Defendpoint client will apply the new settings immediately, and does not require any restart.

Do not delete the pguard.xml file as this will interfere with the client machine's ability to enforce policy. If the pguard.xml file is deleted from a client machine, replace the file and restart the machine.

Chapter 8 - Defendpoint Events

The Defendpoint Client sends events to the local syslog on OS X or the Unified Logging database on macOS. Defendpoint events can be viewed from the Console application by navigating to `/var/log/Defendpoint/audit.log` on OS X.

The following events are logged by the Defendpoint Client:

8.1 - Process Events

Event ID	Description
100	Process has started with admin rights added.
106	Process has started with no change.
116	Process execution was blocked.
117	Process was stopped terminated.
120	Process execution was cancelled by the user
199	Process execution was blocked, the maximum number of challenge/response failures was exceeded.

Each process event contains the following information:

- FileName
- ProcessId
- ParentProcessId
- Workstyle
- ApplicationGroup
- Reason
- FileHash

8.2 - Configuration Events

Event ID	Description
10	Defendpoint not licensed for configured action
200	Successfully loaded Defendpoint configuration (information)
201	Loaded Defendpoint configuration but encountered non-critical problem (warning)
202	Failed to load Defendpoint configuration (error)
210	Successfully downloaded Defendpoint configuration

8.3 - User / Computer Events

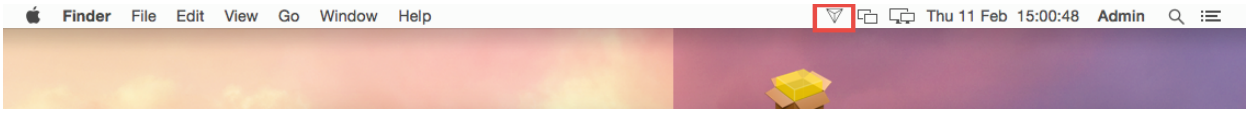
Event ID	Description
400	Defendpoint Service started (information)
401	Defendpoint Service stopping (Information)

Chapter 9 - Troubleshooting

9.1 - Check Defendpoint is installed and functioning

If you are having problems the first step is to check that you have installed the client and that the client is functioning.

The easiest way to determine that the client is installed and functioning is to check that the Defendpoint logo is present in the OS X and macOS Status menus on the menu bar as shown below.



9.2 - Check that Defendpoint is licensed

One of the most common reasons for Defendpoint not functioning is the omission of a valid license from the Defendpoint Settings. To avoid problems it is simpler to add a valid license to every set of Defendpoint Settings that you create. If Defendpoint is not licensed an event will be raised in the syslog or the Unified Logging database.

9.3 - Check Workstyle Precedence

Assuming that Defendpoint is functioning and licensed, most other problems are caused by configuration problems or workstyle precedence problems.

Once an application matches an application group entry in the **Application Rules**, then processing will not continue for that application. Therefore, it is vital that you order your entries correctly:

- If you create multiple workstyles then workstyles higher in the list have a higher precedence.
- If you have multiple Application rules in a workstyle then entries higher in the list have a higher precedence.

Appendix A - Appendices

A.1 - Application Definitions

Application definitions allow you to target applications based on specific properties. When an application is executed, Defendpoint will query the properties of the application and attempt to match them against the matching criteria in the definition. If a match is made, then the rule is applied. If any of the matching criteria do not match then neither will the definition, and Defendpoint will attempt to match against subsequent definitions in the application group.

Defendpoint will continue this process for subsequent application groups defined in Application Rules until a successful match is made and the rule is applied. If no matches are made, then no rule will be applied to the application, and it will run as normal.

The following sections describe all of the available matching criteria for each type of application definition.

A.1.1 - File or Folder Name

This matching criteria allows you to target applications based on their name / path on disk. It is an effective way of automatically whitelisting applications that are located in trusted areas of the filesystem (for example `/Applications` or `/System`), and for targeting specific applications based on their full path.

This matching criteria can be used in combination with other criteria in a definition, giving you more granularity over which applications you can target based on their properties . Although you may enter relative filenames, we strongly recommended that you enter the full path to a file.

This matching criteria includes the following matching options:

- File or Folder Name (for example `/Applications/iTunes.app`)
- Perform Match Using
  - Exact Match
  - Starts With
  - Ends With
  - Contains
  - Regular Expressions (see **Regular Expressions Syntax** detailed on page 43)

Each option supports the use of wildcards; “?” will match any one character, “*” will match any string of characters.

File Name matching is case sensitive, so you must ensure that the correct case has been applied to any configuration of file name or file path matching criteria.

This matching criteria can be used with the following application types:

- Binaries
- Bundles
- Packages
- System Preferences
- Sudo Commands

A.1.2 - Command Line Arguments

The Command Line Arguments matching criteria allows you to target a binary or sudo command based on the arguments passed to the command that is being executed on the command line. Command line arguments can be executed either through the Terminal, or through a script. With this matching criteria you can apply a specific action (such as block, allow or just audit) to specific command line arguments, rather than just applying actions to the use of the binary or sudo command.

The Command Line Arguments matching criteria will match specifically the arguments passed to the binary or sudo command. The following example shows a command for listing the contents of the /Applications directory:

```
MyMac:~ standarduser$ ls -la /Applications
```

- `ls` is the binary being executed, and is targeted by using the File or Folder Name matching criteria in a Binary definition.
- `-la /Applications` are the arguments being passed to `ls`, and is targeted by using the Command Line Arguments matching criteria in a Binary definition.

Defendpoint will only match the command line arguments, which will not include the beginning binary or sudo command being executed. If you want to match both the binary / sudo command and the command line, then both the **File or Folder Name** and the **Command Line Arguments** matching criteria must be enabled and populated in the definition.

This matching criteria allows you to target all, or just parts of the command line being used. This is achieved by inserting wildcards into the **Command Line Arguments** string, defining which part of the command line you want to match, or by using a regular expression.

This matching criteria includes the following matching options:

- Command Line Arguments (for example `-la /Applications`)
- Perform Match Using
  - Exact Match
  - Starts With
  - Ends With
  - Contains
  - Regular Expressions (See **Regular Expressions Syntax** detailed on page 43)

Each option supports the use of wildcards; “?” will match any one character, “*” will match any string of characters.

Command Line Arguments matching is case sensitive, so you must ensure that the correct case has been applied to any configuration of Command Line Arguments matching criteria.

This matching criteria can be used with the following application types:

- Binaries
- Sudo Commands

A.1.3 - File Hash

A File Hash is a digital fingerprint of an application, generated from the contents of application binary or bundle. Changing the contents of an application will result in an entirely different hash. Every application, and every version of the same application, has a unique hash. Defendpoint uses hashes to compare the application being executed against a hash stored in the configuration.

File Hash matching is the most specific criteria, as it can be used to ensure that the application being run is the exact same application that was used when creating the definition, and that it has not been modified.

This matching criteria includes the following matching options:

- File Hash

This matching criteria can be used with the following application types:

- Binaries
- Bundles
- Packages
- System Preferences
- Sudo Commands

Although File Hash is the more reliable matching criteria for matching a specific application, you must ensure that definitions are kept up to date. When updates are applied to the endpoint, new versions of applications may be added, and so their SHA-1 hashes will be different. Applications on different versions of OS X or macOS will also have different SHA-1 hashes.

A.1.4 - File Version

For application types that have defined versions, you can optionally use the File Version matching criteria to target applications of a specific version or range of versions. This allows you to apply rules and actions to certain versions of an application, for example blocking an application if it's version is less than the version defined in the definition.

File Version matching can be applied either as a minimum required version, as a maximum required version, or you can use both to define a range of versions (between a minimum and a maximum).

This matching criteria includes the following matching options:

- File Min Version
- File Max Version

This matching criteria can be used with the following application types:

- Bundles
- System Preferences

A.1.5 - Publisher

Some applications are digitally signed with a certificate, giving a guarantee that the application is genuine and from a specific vendor. The certificate also ensures that the application has not been tampered with by an unauthorized source. The vendor who owns the certificate can be identified from certain properties of the certificate, which are referred to as *Authorities*. A certificate typically contains several *Authorities* linked together in a *chain of trust*.

If you want to check if an application has been digitally signed, and what the certificate Authorities are, use the following command – in this example, checking the certificate of the iTunes.app application bundle:

```
Codesign -dvvv /Applications/iTunes.app/
```

If the application has a certificate, there will be one or more Authorities listed in the output:

```
Authority=Software Signing
Authority=Apple Code Signing Certification Authority
Authority=Apple Root CA
```

In the output, the first Authority listed is the authority most specific to the application. In this example, you can see that Apple use the certificate *Authority* `Software Signing` to digitally sign `iTunes.app`.

With the Publisher matching criteria, you can target applications based on the publisher information contained in its certificate. This matching criteria can also be used in combination with other matching criteria, as a way of ensuring the application is a genuine application from the vendor.

All apps downloaded from the Apple Store will have certificates with the same authority, as Apple resigns all applications before making them available in the Apple Store.

This matching criteria includes the following matching options:

- Publisher (For example, the Publisher for Apple applications is Software Signing)
- Perform Match Using
  - Exact Match
  - Starts With
  - Ends With
  - Contains
  - Regular Expressions (See **Regular Expressions Syntax** detailed on page 43)

Each option supports the use of wildcards; “?” will match any one character, “*” will match any string of characters.

Publisher matching is case sensitive, so you must ensure that the correct case has been applied to any configuration of Publisher matching criteria.

This definition can be used with the following application types:

- Binaries
- Bundles
- Packages
- System Preferences
- Sudo Commands

A.1.6 - Parent Process

When a new application executes it is executed by another process, or 'parent' process. In most cases on OS X and macOS, the parent process will be `launchd`. However, sometimes applications like binaries and bundles are executed by other applications. For example, binaries like `curl` can be executed from the `Terminal`, and will be created as a child of the Terminal process. However, `curl` can also be used by applications.

The Parent Process matching criteria allows you to the target applications based on their parent process, so that you can apply different rules and actions depending on where the application is being executed from. In the example above, you can use Parent Process matching to allow curl to be used by an authorized application, but still block users from executing it directly, in the Terminal.

Parent Processes are defined as an application group, so that you can identify multiple parents without having to create multiple definitions. This also means that the parent process can be defined as any type of application (binary, bundle, system preference or package), using any of the relevant matching criteria for each application.

This matching criteria includes the following matching options:

- Parent Process Group (Drop down list of all Application Groups that exist in the configuration)

This definition can be used with the following application types:

- Binaries
- Bundles
- Sudo Commands

A.1.7 - URI

Every macOS application bundle has a defined Uniform Resource Identifier (URI), a property that uniquely identifies the application to the system. URI's follow a specific structure, typically referencing the vendor and application. For example, the URI for Apple iTunes is `com.apple.iTunes`.

The URI matching criteria provides an effective way of targeting applications where the filename or file path may not always be known. It is also an effective way of targeting applications from a specific vendor.

This matching criteria can also be used in combination with other matching criteria, as a way of ensuring the application is a genuine application from the vendor.

This matching criteria includes the following matching options:

- URI (For example, `com.apple.iTunes`)
- Perform Match Using
  - Exact Match
  - Starts With
  - Ends With
  - Contains
  - Regular Expressions (See **Regular Expressions Syntax** detailed on the next page)

Each option supports the use of wildcards; “?” will any match one character, “*” will match any string of characters.

URI matching is case sensitive, so you must ensure that the correct case has been applied to any configuration of URI matching criteria.

This definition can be used with the following application types:

- Bundles

A.1.8 - Application requests authorization

When an application triggers an authorization request, the application will use a unique **Auth Request URI**. This URI will be different to the URI of the application itself. This matching criteria allows you to target any authorization request by matching the Auth Request URI, allowing you to target that specific Auth request URI and apply your own controls.

This matching criteria can be used in combination with other criteria to target authorization requests from specific applications, if more than one application uses the same Auth Request URI.

When this matching criteria is used in a definition, it will only match the authorization request of the application, and not the execution of the application. If you want to apply rules to both the application execution and application authorization request, then separate definitions must be created for each.

If you want to apply different rules to application execution and application authorization requests, then definitions must be added to different Application groups and applied to different Application Rules.

This matching criteria includes the following matching options:

- Auth Request URI (for example system.preferences.datetime)
- Perform Match Using
  - Exact Match
  - Starts With
  - Ends With
  - Contains
  - Regular Expressions (see **Regular Expressions Syntax** detailed on the next page)

Each option supports the use of wildcards; “?” will match any one character, “*” will match any string of characters.

This definition can be used with the following application types:

- Binaries
- Bundles
- Packages
- System Preferences (mandatory)

A.2 - Regular Expressions Syntax

Defendpoint can control applications at a granular level by using regular expression syntax. Defendpoint for Mac uses **RegEx** modelled on the Perl programming language and fully supports Unicode. Below is a summary of the regular expression syntax used by this library.

Metacharacter	Meaning	Example
Any character except [\^$.|?*+()	All characters except the listed special characters match a single instance of themselves. To match one of these listed characters use a backslash escape character (see below)	“abc” matches “abc”
\ (backslash)	Escape character: interpret the next character literally.	“a\+b” matches “a+b”
. (dot)	Matches any single character.	“a.b” matches “aab”, “abb” or “acb”, etc.
[ ]	Indicates a character class. Matches any character inside the brackets (for example, [abc] matches "a", "b", and "c").	“[abc]” matches "a", "b", or "c"
^ (caret)	If this metacharacter occurs at the start of a character class, it negates the character class. A negated character class matches any character except those inside the brackets (for example, [^abc] matches all characters except "a", "b", and "c"). If ^ is at the beginning of the regular expression, it matches the beginning of the input (for example, ^[abc] will only match input that begins with "a", "b", or "c").	“[^abc]” matches all characters except "a", "b", and "c"
- (minus character)	In a character class, indicates a range of characters (for example, [0-9] matches any of the digits "0" through "9").	“[0-9]” matches any of the digits "0" through "9"
?	Indicates that the preceding expression is optional: it matches once or not at all (for example, [0-9][0-9]? matches "2" and "12").	“ab?c” matches "ac" or "abc"
+	Indicates that the preceding expression matches one or more times (for example, [0-9]+ matches "1", "13", "666", and so on).	“ab+c” matches "abc" and "abbc", “abbbc”, etc.
* (asterisk)	Indicates that the preceding expression matches zero or more times	“ab*c” matches "ac" and "abc", “abbc”, etc.
| (vertical pipe)	Alternation operator: separates two expressions, exactly one of which matches.	“a|b” matches “a” or “b”

Metacharacter	Meaning	Example
??, +?, *?	Non-greedy versions of ?, +, and *. These match as little as possible, unlike the greedy versions which match as much as possible. Example: given the input "<abc><def>", <.*?> matches "<abc>" while <.*> matches "<abc><def>".	Given the input "<abc><def>", <.*?> matches "<abc>" while <.*> matches "<abc><def>".
( )	Grouping operator. Example: (\d+,)*\d+ matches a list of numbers separated by commas (such as "1" or "1,23,456").	“(One)|(Two)” matches "One" or "Two"
{ }	Indicates a match group. The actual text in the input that matches the expression inside the braces can be retrieved through the CAtlREMatchContext object.	
\	Escape character: interpret the next character literally (for example, [0-9]+ matches one or more digits, but [0-9]\+ matches a digit followed by a plus character). Also used for abbreviations (such as \a for any alphanumeric character; see table below). If \ is followed by a number n, it matches the nth match group (starting from 0). Example: <{.*?}>.*?</\0> matches "<head>Contents</head>". Note that in C++ string literals, two backslashes must be used: "\\+", "\\a", "<{.*?}>.*?</\\0>".	<{.*?}>.*?</\0> matches "<head>Contents</head>"
$	At the end of a regular expression, this character matches the end of the input. Example: [0-9]$ matches a digit at the end of the input.	[0-9]$ matches a digit at the end of the input
|	Alternation operator: separates two expressions, exactly one of which matches (for example, T|the matches "The" or "the").	T|the matches "The" or "the")
!	Negation operator: the expression following ! does not match the input. Example: a!b matches "a" not followed by "b".	a!b matches "a" not followed by "b"

A.3 - Application Templates

Defendpoint ships with some standard application templates to simplify the definition of applications. The standard application templates are split into three categories:

- Binaries
- Bundles
- System Preference Panes

A.3.1 - Creating Custom Application Templates

Application templates are stored as XML files in:

```
%ALLUSERSPROFILE%\Application Data\Avecto\Privilege Guard Templates\
```

The Standard Application Templates are stored in a single file named `OSXTemplates.xml`, and it is highly recommended that you do not change these templates.

Instead, you should create your own XML template files. Application templates are a set of Application groups that have been exported from the Defendpoint management console as an XML file.

To create a set of application templates, create some Application groups and populate the Application groups with applications. The Application groups will become the Categories, and the applications in each application group will be the list of Applications for that Category. Once you have defined your application templates, export the settings to an XML file:

1. Select the **Defendpoint Settings** node.
2. Right-click and select **Export…**.

The XML file that you export must be saved with a prefix of OSX e.g. OSX_My_Templates.xml.

To import an application template file back into the management console for editing:

1. Select the **Defendpoint Settings** node.
2. Right-click and click **Import…**.
3. When prompted click **No** to overwrite the current workstyles.

Remember to re-export your application templates once you've modified them.

The final step is to copy your application templates to the application templates directory on any machines where the management console is being used to create Defendpoint settings. The management console automatically loads all of the application templates in the application templates directory and merges them to create a single list of categories.

A.4 - Advanced Settings

Defendpoint includes some advanced settings that are configured by editing a configuration file on disk. In order to edit the configuration file, you will need root privileges on the following file:

```
/Library/Application Support/Avecto/Defendpoint/defendpoint.plist
```

It is recommended that you edit the configuration file using a commandline editor, such as `vi`:

```
sudo vi /Library/Application Support/Avecto/Defendpoint/defendpoint.plist
```

A.4.1 - Debug Logging

To enable debug logging on OS X, the logs need to be registered by executing the following command:

```
sudo /usr/local/libexec/Avecto/Defendpoint/1.0/configure/installdbglogs.sh
```

Defendpoint has now registered the following log files:

```
/private/var/log/pguard/pgdaemon.log

/private/var/log/pguard/pggui.log

/private/var/log/pguard/pgpolicyserver.log
```

By default, the debug logging level is set to 6 in the `defendpoint.plist` configuration file. This provides a basic level of debug logging.

To enable full debug logging, edit the following section in the `defendpoint.plist` configuration file to set the debug logging level to 7:

```
<key>LogLevel</key>
<integer>7</integer>
```

To unregister the Defendpoint debug logs, execute the following command:

```
sudo /usr/local/libexec/Avecto/Defendpoint/1.0/configure/uninstalldbglogs.sh
```

A.4.2 - Anonymous Logging

By default, Defendpoint will include user and computer specific information in all audit events. You can set your application rules to not log this information for events associated with your rules by setting the **Raise an Event** option to **On (Anonymous)** on each rule.

You can also set whether user or computer information is kept anonymous for audit events that are not associated with a rule, such as events raised for having an invalid license.

To enable anonymous auditing for events not associated with a rule, edit the following section in the `defendpoint.plist` configuration file:

```
<key>AnonymousLogging</key>
<string>true</string>
```

To disable anonymous auditing for events not associated with a rule, edit the following section in the `defendpoint.plist` configuration file:

```
<key>AnonymousLogging</key>
<string>false</string>
```

A.4.3 - Application Compatibility

Defendpoint for Mac uses process hooking techniques to control the behaviour of applications. In certain circumstances, process hooking may cause erroneous behaviour in the application, which in turn could cause the application to fail. In these circumstances, the process should not be hooked.

Defendpoint includes an advanced setting that allows processes to be excluded from hooking. Exclusions are set as either a full or relative path of the applications process. For each exclusion, Defendpoint will perform a comparison of the application process path to determine if the process should be hooked or excluded.

To edit the exclusion list, edit the following section in the `defendpoint.plist` configuration file. This example, Defendpoint has been configured to exclude “MyApplication.app”:

```
<key>HookException</key>
<array>
	<string>/Applications/Xcode.app</string>
</array>
```

To add additional exclusions to the list, add a new line to the <array>:

```
<key>HookException</key>
<array>
	<string>/Applications/MyApplication.app</string>
	<string>/Applications/MySecondApplication.app</string>
</array>
```

If you wish to exclude multiple applications in the same directory, then you can add the directory where the applications are located. When checking the exclusion list, Defendpoint checks whether the application process *starts with* the string(s) in the array.

To exclude all applications in the directory "/Applications/MyApps/", add the following new line to the `<array>`:

```
<string>/Applications/MyApps/</string>
```

By excluding an application process from hooking, some privilege management functionality will no longer be available. It is recommended that Application Compatibility settings are only applied under instruction from Avecto Support.

A.5 - Sudo Licenses

Sudo is distributed under the following license:
Copyright (c) 1994-1996, 1998-2017

Todd C. Miller <Todd.Miller@courtesan.com>

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Sponsored in part by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement number F39502-99-1-0512.

The file redblack.c bears the following license:
Copyright (c) 2001 Emin Martinian

Redistribution and use in source and binary forms, with or without modification, are permitted provided that neither the name of Emin Martinian nor the names of any contributors are be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The file reallocarray.c bears the following license:
Copyright (c) 2008 Otto Moerbeek <otto@drijf.net>

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The files getcwd.c, glob.c, glob.h, snprintf.c and sudo_queue.h bear the following license:
Copyright (c) 1989, 1990, 1991, 1993

The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The file fnmatch.c bears the following license:
Copyright (c) 2011, VMware, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or withoutmodification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the VMware, Inc. nor the names of its contributors without specific prior written permission.
- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the VMware, Inc. nor the names of its contributors without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL VMWARE, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The file getopt_long.c bears the following license:
Copyright (c) 2000 The NetBSD Foundation, Inc.

All rights reserved.

This code is derived from software contributed to The NetBSD Foundation by Dieter Baron and Thomas Klausner.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditionsare met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The file inet_pton.c bears the following license:
Copyright (c) 1996 by Internet Software Consortium.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.