

Symantec Endpoint Protection 11.0
Application and Device Control

GUIDANCE DOCUMENT

Content Table

Content	
Introduction	3
Audience	3
Chapter One: Application control policy	4
Creating an Application Control policy:	4
Application Control Rule Sets	5
Rules	7
Conditions	8
Common Mistakes	15
Performances	16
Chapter Two: Device control Policy	17
Standard Devices	17
Custom Device	18
Chapter Three: Use cases for Application and device control	21
Blocking Unwanted network interfaces / Dialup	21
Allowing only corporate purchased USB keys	24
Monitor Device ID centrally	26
Harden Internet Explorer security against drive by downloads	29

Introduction

Device and Application Control is an advanced security feature included in Symantec Endpoint Protection 11.0. This feature has two focuses, firstly Device and Application Control, provides administrators with the ability to monitor and/or control the behaviour of applications. Administrators can grant/deny access to certain registry keys, files, and folders. In addition, administrators can also define which applications are permitted to run, which applications that cannot be terminated through irregular processes, and which applications can call Dynamic Link Libraries. Secondly, Device and application controls can block and control peripherals connected to a SEP client. This gives administrators the flexibility to enable, disable and control access to removable storage and other peripherals. This document will focus on providing several examples on how to create Device and Application Control Policies to take full advantage of all the capabilities of this feature.

Audience

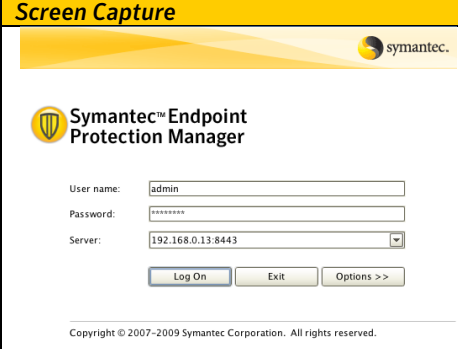
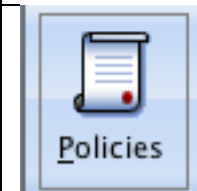
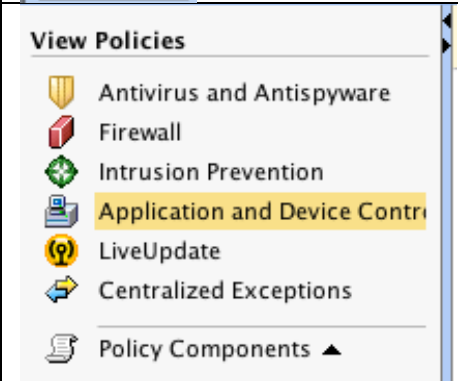
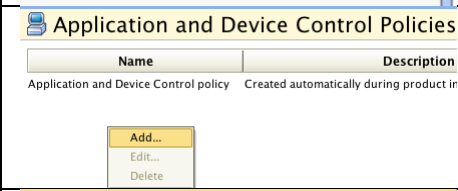
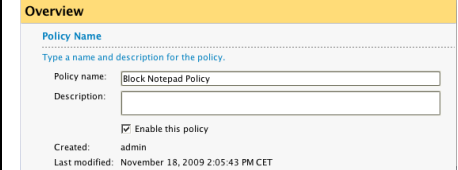

This document is aimed at administrators and product specialist with a working knowledge of Symantec Endpoint Protection 11.0. The reference guide and other documentation provided on the CD are a pre requisite to fully benefit from this document.

Application and Device control.

Chapter One: Application control policy

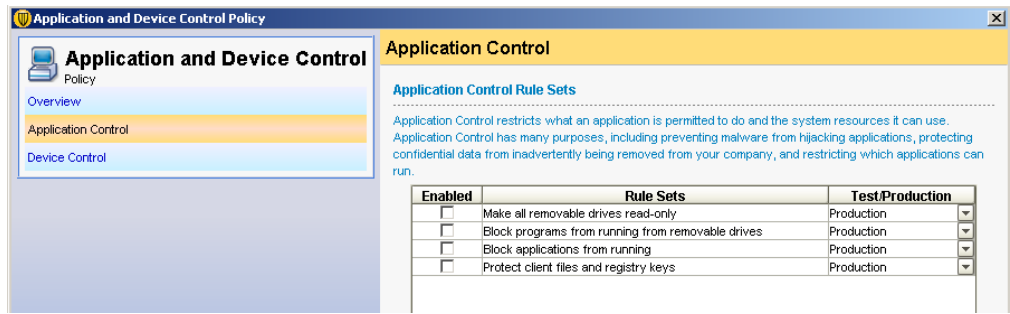
Application Control Policies can only be created and/or modified from the Symantec Endpoint Protection Admin console. Application Control cannot be modified on the Endpoint Protection Client.

Creating an Application Control policy:

Screen Capture	Description
 The screenshot shows the Symantec Endpoint Protection Manager login interface. It includes fields for User name (admin), Password (masked), and Server (192.168.0.13:8443). There are buttons for Log On, Exit, and Options >>. The Symantec logo is at the top right.	Open the SEPM Console and login with an administrator account, or an account with rights
 The screenshot shows a blue icon of a document with a red checkmark, labeled "Policies".	On the Left Tab, Select Policies
 The screenshot shows the "View Policies" sidebar. It lists several policy categories: Antivirus and Antispyware, Firewall, Intrusion Prevention, Application and Device Control (highlighted), LiveUpdate, Centralized Exceptions, and Policy Components.	Under the View Policies Tab, Select Application and Device Control
 The screenshot shows the "Application and Device Control Policies" section. It contains a table with columns "Name" and "Description". The table lists "Application and Device Control policy" with the description "Created automatically during product in". Below the table are buttons for "Add...", "Edit...", and "Delete".	In the tasks section, select Add an Application and Device Control Policy
 The screenshot shows the "Overview" section of a policy. It includes fields for "Policy Name" (Block Notepad Policy) and "Description". There is a checkbox for "Enable this policy" which is checked. At the bottom, it shows "Created: admin" and "Last modified: November 18, 2009 2:05:43 PM CET".	In the Policy Name Type the name you would like to give this policy and enter a description if required.
 The screenshot shows the "Application and Device Control Policy" overview sidebar. It has a blue header "Application and Device Control Policy" and two tabs: "Overview" (selected) and "Device Control".	From the Left pane of the policy windows, select Application Control

Application Control Rule Sets

A Rule Set is a set of controls that allow administrators to allow or block an action. In the example below, you will note that there are currently four rule sets defined. You will also notice that Administrators can choose to create as many rule sets as they would like in a policy. Even though multiple rule sets can be in a given policy, administrators can choose which rule sets are active by toggling the Enabled option. In this example, you will note that none of the rule sets are enabled.



To the right of the rule set name there is an option to configure Test/Production. This feature allows administrators to test rules before actually enabling them. In the Test (log only) configuration, no actions will be applied in the rule, but the action is logged. This allows administrators to see what would have happened if this rule would have been active. All new rule sets are created with the default option configured to test. This reduces potential accidents an administrator may make by not considering all possibilities of the rule.

Enabled	Rule Sets	Test/Production
<input type="checkbox"/>	Make all removable drives read-only	Production
<input type="checkbox"/>	Block programs from running from removable drives	Test (log only)
<input type="checkbox"/>	Block applications from running	Production
<input type="checkbox"/>	Protect client files and registry keys	Production

BEST PRACTICE FOR RULE SETS:

It is recommended to run rules in a test mode for some acceptable period of time before switching them to production. Logs should be reviewed to verify that the correct items are blocked.

Application and Device control.

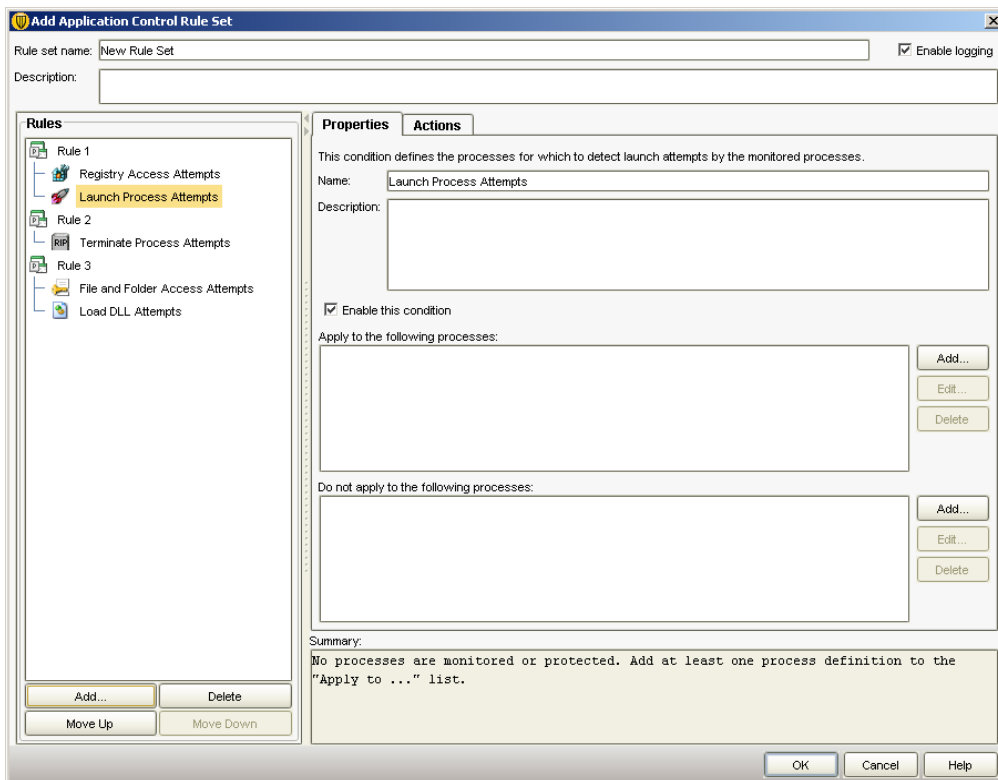
Adding Rule Sets

To add a rule set, simply select the add button. To edit or delete a rule set, select the rule, then click the edit or delete button.



Note: The edit, delete, move up, and move down button will remain greyed out until a rule set is selected.

Rule Sets consists of Rules and Conditions. A rule is a set of conditions and actions that apply to a given process or processes. For organizational purposes, it is recommended to create a rule set that includes all of the actions that you want to allow/block/monitor a given task. For example, if an administrator wanted to block write attempts to all removable drives and block people/applications from tampering with a specific application, it would be recommended to create two distinct rule sets versus creating all of the necessary rules to do both tasks under one rule set.



In order to get high performance environment it is recommended not to exceed 100 rules in a rule set.

Rules

Rules define the application(s) that you are monitoring. Conditions define what specifically you want to allow or block an application from doing, and actions determine what action to take when the condition is met.

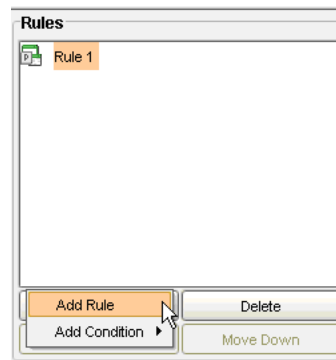
Before getting into rule development, it is best to understand how rules work. A rule applies to an application or multiple applications. Rules contain conditions that monitor specified operations for the application(s) defined in the rule. The condition also contains the actions to take when the specified operation is observed. A majority of the issues encountered by new administrators when configuring Application Control is caused by not realizing that Actions always apply to the process defined in the rule and not the Condition.

BEST PRACTICE FOR RULES:

Actions always apply to the process defined in the rule, not by the Condition.

Adding a Rule

1. Open the Rule Set
2. Click the Add button in the Rules pane
3. Click the Add Rule option.
4. In the Rule Name, enter a name for the rule
5. In the Description Field, enter a description for the rule (optional)



Once the rule created, it must be tied to an application or multiple applications. This is done in the Rules Properties window. You will also notice that you have an option to enable or disable the rule by toggling the Enable this Rule Check Box.

A screenshot of a 'Rules Properties' window. It has two main sections. The top section is titled 'Apply this rule to the following processes:' and contains a large empty text box. To its right are three buttons: 'Add...', 'Edit...', and 'Delete'. The bottom section is titled 'Do not apply this rule to the following processes:' and also contains a large empty text box with similar 'Add...', 'Edit...', and 'Delete' buttons. At the bottom left, there is a checkbox labeled 'Sub-processes inherit conditions' which is currently unchecked.

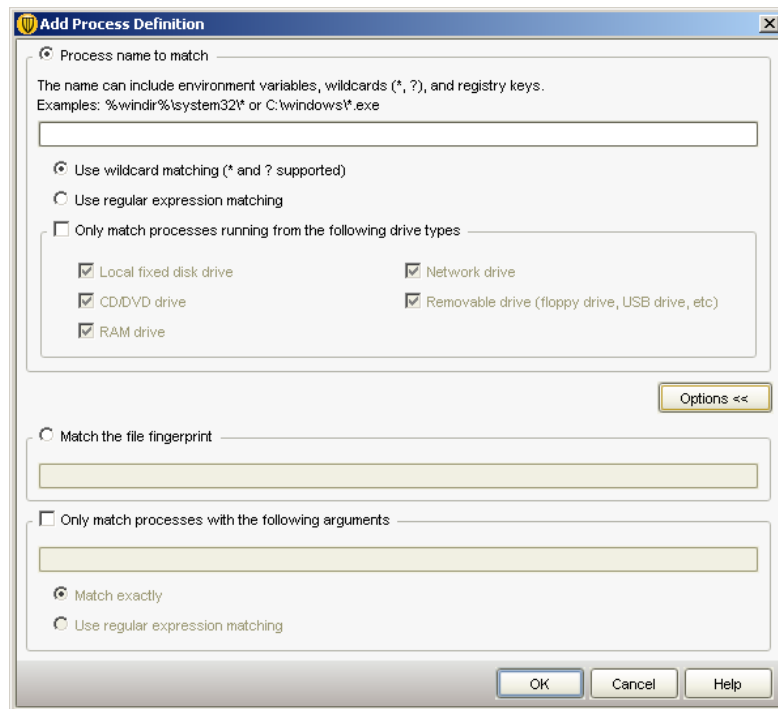
There are two sections that deal with tying the rule to an application or multiple applications. One process definitions list contains processes to which the rule applies. The other process definitions list contains processes to which the rule does not apply. If an administrator wanted to tie the rule to all application except for a given set of applications, then they would define a wildcard for all (*) in the top section, and list the applications that need an exception in the bottom section.

NOTE: In every configuration, the top section must have at least one application defined. When adding applications to a rule, administrators can use the process name, wildcards, regular expressions, fingerprints, and/or drive types from where the application was launched.

Adding an Application to a Rule

Application and Device control.

1. Determine if the Application being added is the application to tie the rule to or if the application is going to be an exception to the rule.
2. Select the Add for the appropriate the section.



3. In the Add Process Definition, you can use the criteria of choice to define the application(s)

Administrators can define as many applications as they would like to a given rule.

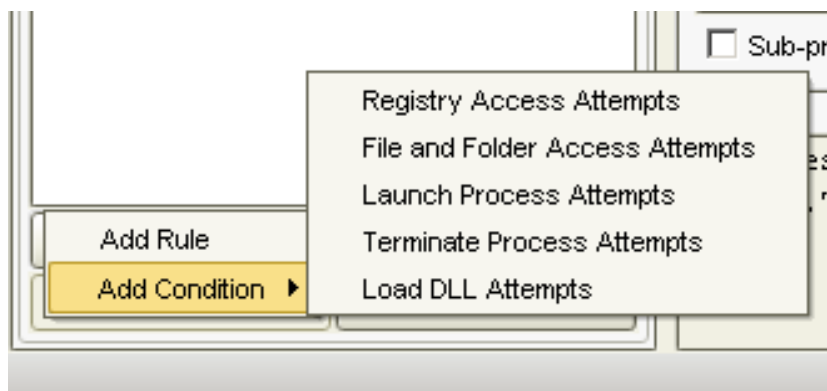
Conditions

Conditions are operations that can be allowed or denied for an application or multiple applications. There are several condition types that can be configured. These include the following:

- **Registry Access Attempts** - Allow or block access to a client computer's registry settings
- **File and Folder Access Attempts** - Allow or block access to defined files or folders on a client computer
- **Launch Process Attempts** - Allow or block the ability to launch a process on a client computer
- **Terminate Process Attempts** - Allow or block the ability to terminate a process on a client computer. For example, you may want to block a particular application from being stopped.
***NOTE:** This Condition does not prevent an application from being terminated using normal methods of quitting an application (i.e. Alt-F4, or the Program's native exit routine). It will prevent the process from being terminated by other applications or procedures.*
- **Load DLL Attempts** - Allow or block the ability to load a DLL on a client computer

To add a condition to a rule, select the add Click in the Rules pane and then select the condition type you want to add.

Application and Device control.

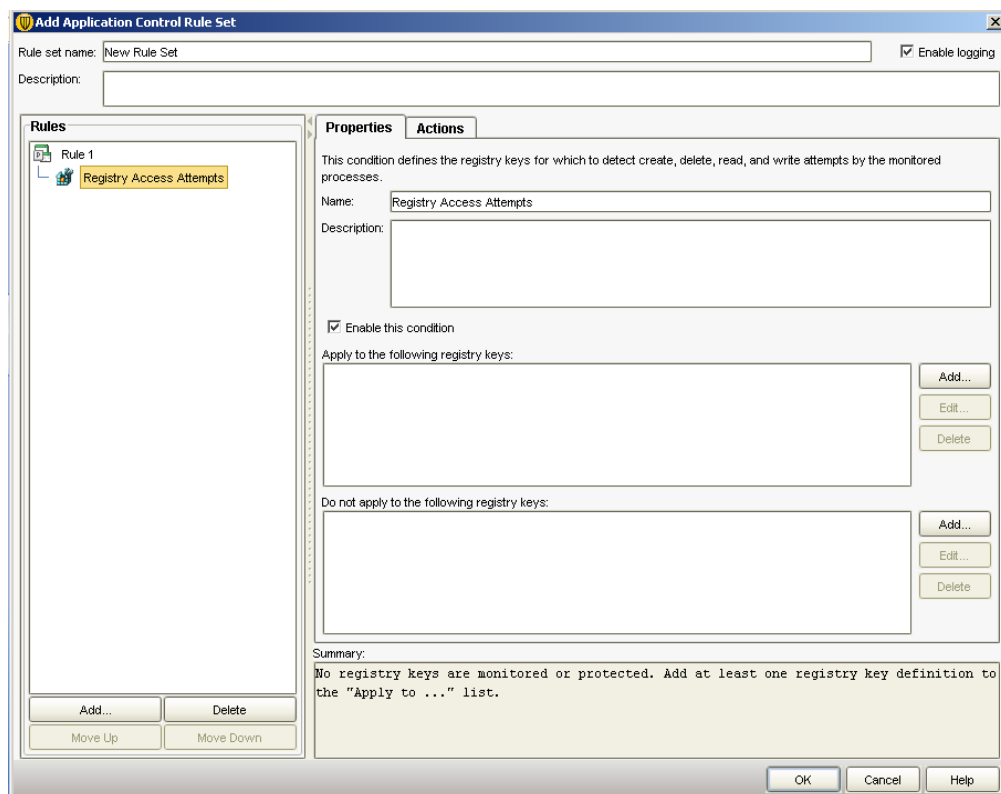


Multiple conditions can be added to a given rule. As conditions are added, administrators will need to specify the specific properties of the condition and what actions to take when the condition is met. Each condition type will have different properties.

Application and Device control.

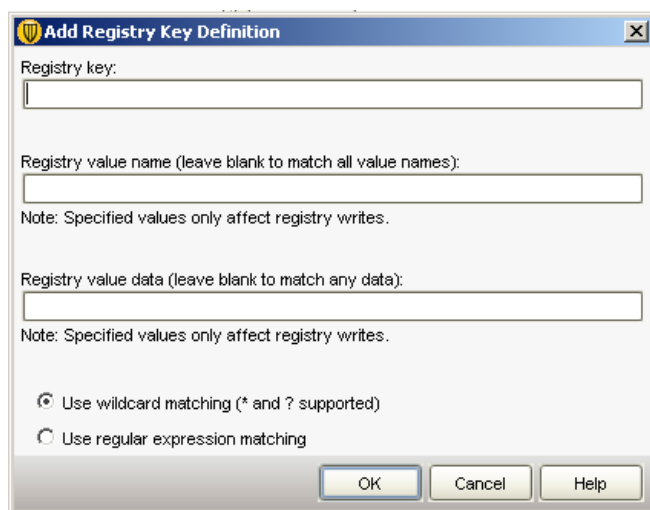
Condition Properties

Each condition type has its own Condition Properties to specify what the condition is looking for. Each condition also has its own specific actions to take when the condition is true. To edit the properties and action for a condition, select the condition in the rules windows pane.



Registry Access Attempts

Administrators can define specific registry keys, values, and data to monitor as depicted in the graphic below. Administrators can also use wildcards and regular expressions to define keys, values, and data.



Application and Device control.

The screenshot shows the 'Properties' tab of a Windows Security rule. It is divided into two main sections: 'Read Attempt' and 'Create, Delete, or Write Attempt'. Each section has a description, a set of radio buttons for actions, and checkboxes for logging and user notification. The 'Severity' is set to 'Critical -- 0' in both sections. Below the settings are two large empty text boxes for additional configuration.

Section	Action	Enable logging	Notify user	Severity
Read Attempt	<input checked="" type="radio"/> Continue processing other rules	<input type="checkbox"/>	<input type="checkbox"/>	Critical -- 0
	<input type="radio"/> Allow access			
	<input type="radio"/> Block access			
	<input type="radio"/> Terminate process			
Create, Delete, or Write Attempt	<input checked="" type="radio"/> Continue processing other rules	<input type="checkbox"/>	<input type="checkbox"/>	Critical -- 0
	<input type="radio"/> Allow access			
	<input type="radio"/> Block access			
	<input type="radio"/> Terminate process			

For registry access attempts, administrators can define different actions to take for read and/or create/delete/write attempts.

File and Folder Access Attempts : Administrators can define files and folders. Administrators can also use wildcards and regular expressions. In addition, Administrators can also restrict the monitoring of files and folders to specific drive types.

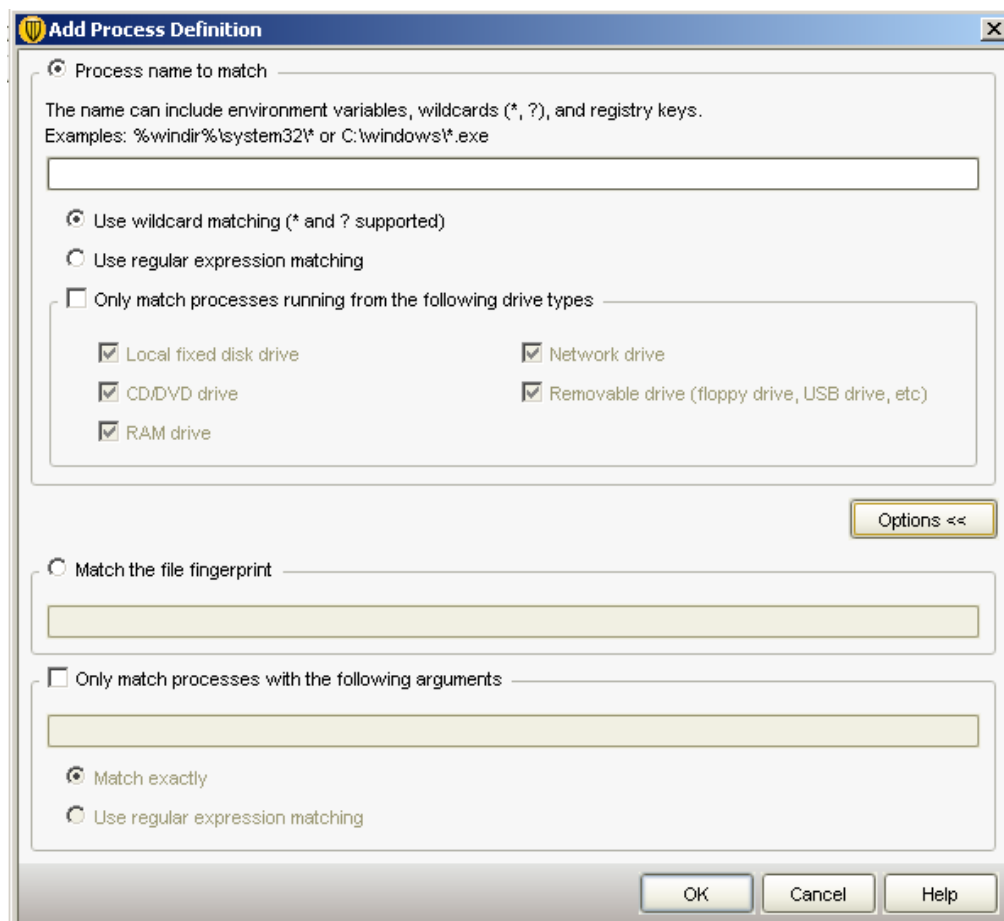
The screenshot shows the 'Add File or Folder Definition' dialog box. It has a title bar with a shield icon and a close button. The main area is titled 'File or Folder Name To Match' and contains a text box for the name. Below the text box are three radio buttons: 'Use wildcard matching (* and ? supported)' (selected), 'Use regular expression matching', and 'Only match files on the following drive types' (checked). The 'Only match files on the following drive types' section is expanded, showing a list of drive types with checkboxes: 'Local fixed disk drive', 'CD/DVD drive', 'RAM drive', 'Network drive', and 'Removable drive (floppy drive, USB drive, etc)'. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

Application and Device control.

NOTE: When applying a condition to everything in a given folder, it is best to {folder name}*. In many cases, administrators forget to include the wildcard to include all files similar to Registry Attempt Access, administrators can choose to take different actions for read and/or Create/Delete/Write attempts.

Launch Process Attempts

Administrators can define processes that they want to prevent or allow to start. When defining processes, administrators can use specific file names, wildcards, and regular expressions. Administrators can also choose to limit monitoring to applications being launched from a particular drive type, arguments being passed to an application, and or applications with a particular file fingerprint.



The "Add Process Definition" dialog box is used to configure process matching rules. It features a title bar with a shield icon and a close button. The main content is organized into two primary sections, each with a radio button for selection. The first section, "Process name to match", includes a text input field, explanatory text about wildcards and registry keys, and a list of drive types with checkboxes. The second section, "Match the file fingerprint", includes another text input field and options for matching arguments. An "Options <<" button is located between the two sections. At the bottom, there are "OK", "Cancel", and "Help" buttons.

Add Process Definition

☒ Process name to match

The name can include environment variables, wildcards (*, ?), and registry keys.
Examples: %windir%\system32* or C:\windows*.exe

☒ Use wildcard matching (* and ? supported)

☐ Use regular expression matching

☐ Only match processes running from the following drive types

- ☒ Local fixed disk drive
- ☒ Network drive
- ☒ CD/DVD drive
- ☒ Removable drive (floppy drive, USB drive, etc)
- ☒ RAM drive

Options <<

☐ Match the file fingerprint

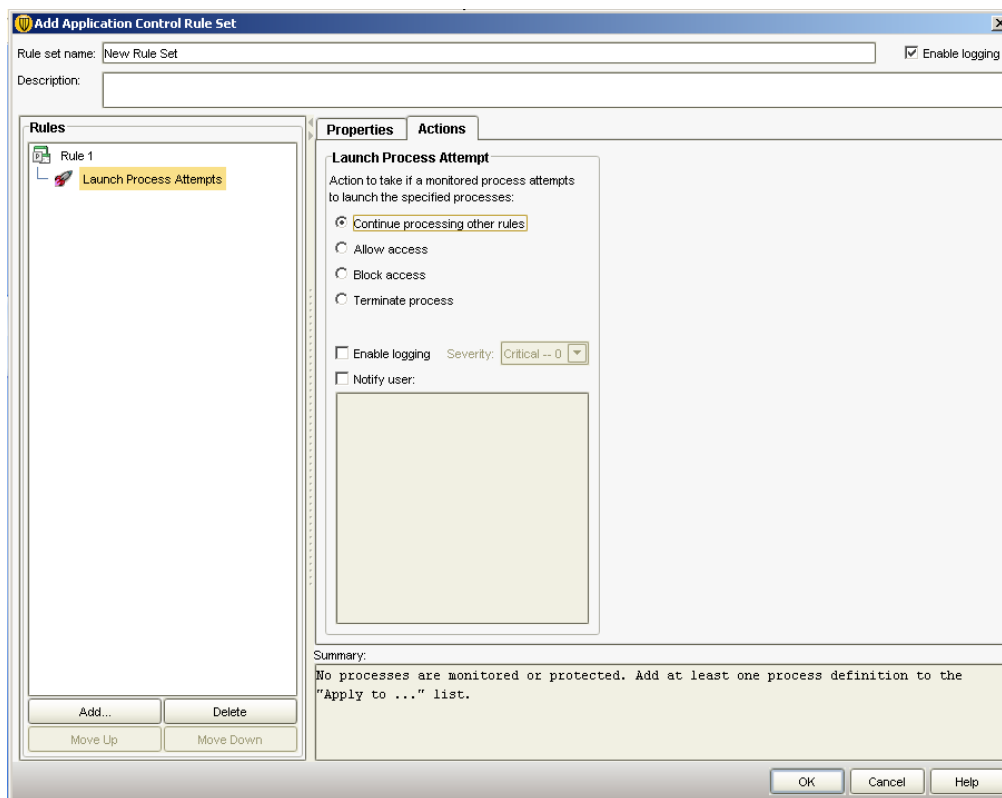
☐ Only match processes with the following arguments

☒ Match exactly

☐ Use regular expression matching

OK Cancel Help

Application and Device control.



The actions for Launch Process Attempts are limited to allowing the process, blocking the process from being launched, or terminate the calling application.

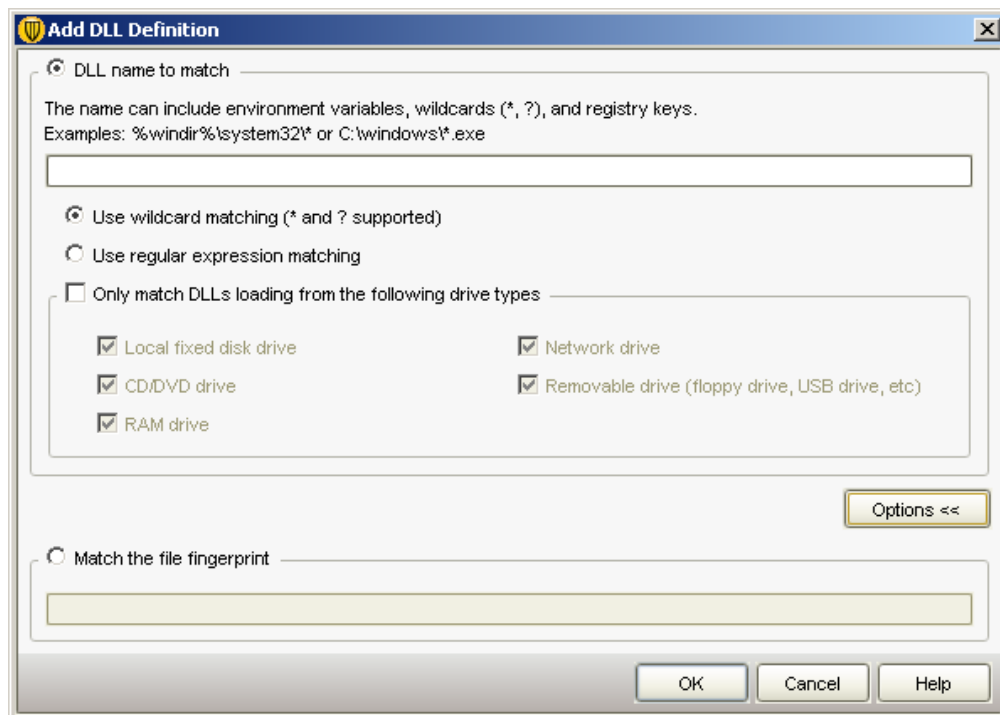
Terminate Process Attempts

Properties and actions for Terminate Process Attempts are similar to Process Launch Attempts. The only difference between the two is that Terminate Process Attempts looks for applications/processes that try to kill a specified process vs. looking for a process to start.

Application and Device control.

Load DLL Attempts

Administrators can define Dynamic Link Library files that they want to prevent or allow to be loaded into an application. When defining DLLs, administrators can use specific file names, wildcards, fingerprint and regular expressions. Administrators can also choose to limit monitoring of DLLs to DLLs being launched from a particular drive type.



The "Add DLL Definition" dialog box is used to configure monitoring for DLLs. It features two main sections: "DLL name to match" and "Match the file fingerprint".

- DLL name to match:** This section is selected with a radio button. It includes a text box for the DLL name, with a note: "The name can include environment variables, wildcards (*, ?), and registry keys. Examples: %windir%\system32* or C:\windows*.exe". Below the text box are three radio buttons: "Use wildcard matching (* and ? supported)" (selected), "Use regular expression matching", and "Only match DLLs loading from the following drive types". The latter is a checkbox that, when selected, reveals a list of drive types: "Local fixed disk drive", "CD/DVD drive", "RAM drive", "Network drive", and "Removable drive (floppy drive, USB drive, etc)".
- Match the file fingerprint:** This section is unselected. It contains a text box for the file fingerprint.

At the bottom right of the "DLL name to match" section is an "Options <<" button. At the bottom of the dialog are "OK", "Cancel", and "Help" buttons.

In the Actions, administrators can choose to allow the DLL to load, block the DLL from being loaded, or terminate the application that is attempting to load the DLL.



The "Load DLL Attempt" action configuration is shown in the "Actions" tab of a properties window. It includes a description: "Action to take if a monitored process attempts to load the specified DLLs:". Below this are four radio buttons: "Continue processing other rules" (selected), "Allow access", "Block access", and "Terminate process". There are also checkboxes for "Enable logging" and "Notify user". The "Enable logging" checkbox is checked, and the "Severity" is set to "Critical -- 0". The "Notify user" checkbox is unchecked.

Application and Device control.

Common Mistakes

There are two common mistakes made by individuals configuring Application Control for the first time. The first is configuring the wrong action and the second is neglecting the order of rules.

Wrong Action

In every action setting, there are four options for the action to take: Continue processing other rules, Allow access, Block access.

- **Continue processing** – This action allows administrators the ability to log the event and continue processing other rules in the stack. The standard operation is to stop processing rules once the first criteria matches.
- **Allow** – Allows the operation to continue
- **Block** – Prevents the operation
- **Terminate process** – Kills the application making the request.

Although these options seem simple, many people will accidentally choose to terminate the process. This can lead to undesired results. To fully understand the common mistake, consider the scenario below:

An Administrator wants to block individuals from modifying the secret.doc on client machines. The administrator does the following:

1. Creates a new Rule Set
2. Adds a new rule to the rule set
3. The rule is tied to the application *
4. Adds a condition, File and Folder Attempt Access
5. Adds the file secret.doc to the Condition
6. Configures the Write Action to Terminate Process

To test the policy, the administrator opens MS Word. The administrator then proceeds to use Word to navigate to the folder where secret.doc is located. The administrator opens the file. The Administrator makes some changes and then attempts to save the file. The End Result, MS Word terminates. Although no writes were allowed, the administrator did not expect MS Word to close. The reason this occurred is due to the Administrator choosing the Terminate option vs. the Block option.

BEST PRACTICE FOR ACTIONS:

It is recommended to use the Block Action to prevent a condition vs. Terminate. Terminate should be only be used in advanced configurations.

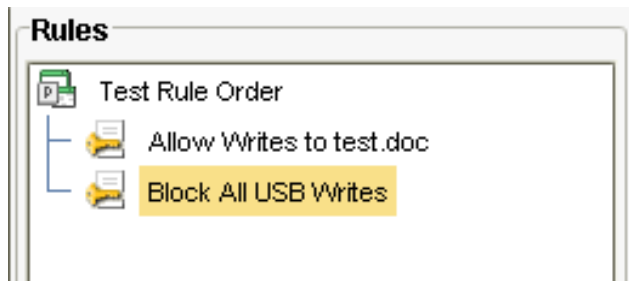
Order of Rules

Many new administrators fail to notice that Application Control rules function very similar to the way that most network based firewalls work with the first rule match feature. When there are multiple rules where the conditions are true, the rule list on the top will be the only condition/action that will be applied. It is important to understand the order of rules being configured. Neglecting the order could lead to wrong expectations.

Consider the following scenario:

Suppose an administrator wanted to block everyone from moving/copying/creating files on USB drives, so the Administrator adds a rule to an existing rule set as depicted below.

Application and Device control.



In the above scenario, clients would be able to create/modify a file called test.doc on USB drives. Because the Allow Writes to test.doc is ordered before the Block All USB Writes, the Block ALL USB Writes never gets processed in the case where rules above it are true.

Performances

The more rules you have the more it can slow you down. The effect of the number of rules on the client performance is very gradual.

The performance of Application and Device Control really depends on what the rules are applied to. For example if ones creates 1,000 rules only for Word then none of the other applications should function any differently, though Word may be slower.

The best practice would be to limit a rule set 100 rules and the policy to 1000 Rule Sets.

Note that when doing a block all for read & write actions, **smcgui.exe** can cause higher than normal CPU utilization. The workaround is to exclude smcgui.exe from this rule. This appears to impact systems limited RAM (512MB or less typically.)

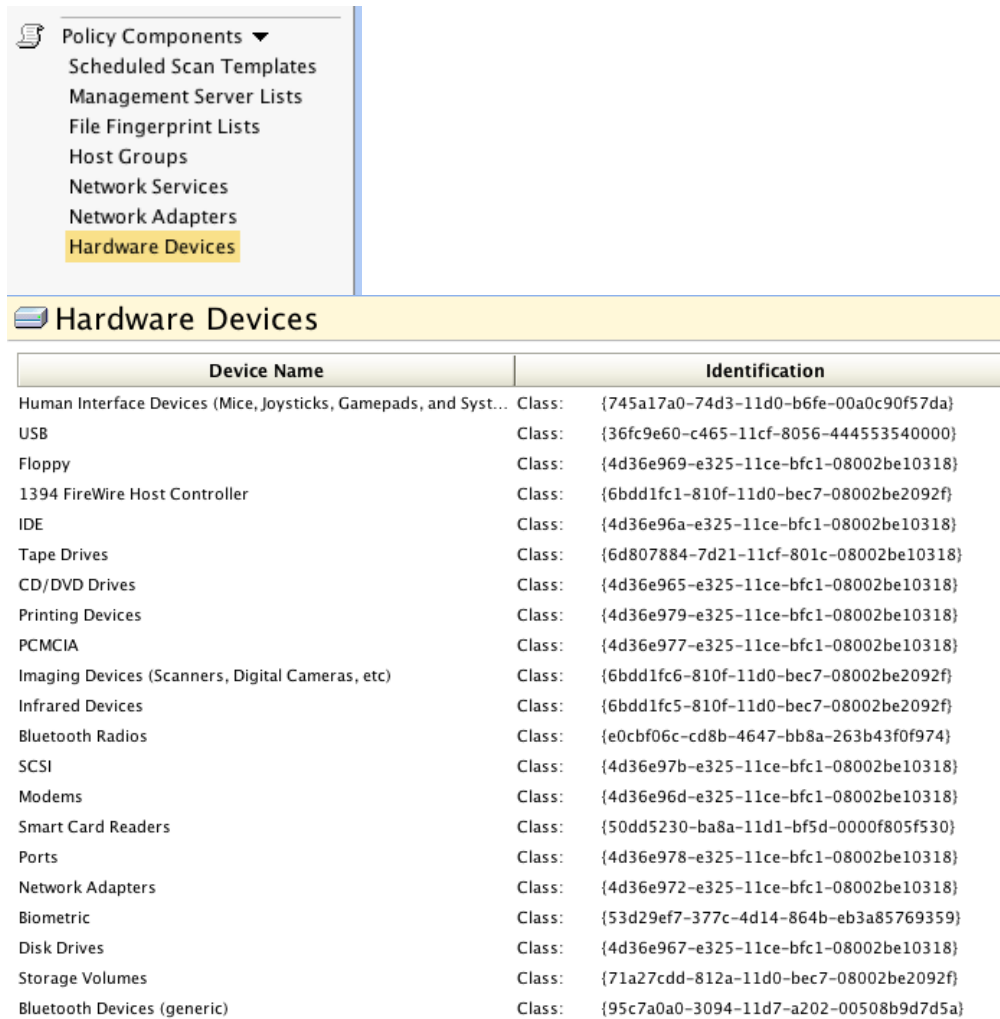
Application and Device control.

Chapter Two: Device control Policy

Device control allows SEP administrators to restrict the access of usage of a given peripheral by policy. The features acts in the registry and disable on the device manager level the chosen peripheral. Once a peripheral is blocked you need to re-enable it by policy, uninstalling SEP will not re-enable the device.

Standard Devices

A large number of devices controlled by SEP are pre listed on the SEPM for the standard entries under Policy >Policy Component >Hardware Devices:



The screenshot shows the Windows Policy Components console. On the left, a tree view lists various policy components, with 'Hardware Devices' selected and highlighted in yellow. The main pane displays a table titled 'Hardware Devices' containing a list of device classes and their corresponding identification numbers (GUIDs).

Device Name	Identification
Human Interface Devices (Mice, Joysticks, Gamepads, and Syst...	Class: {745a17a0-74d3-11d0-b6fe-00a0c90f57da}
USB	Class: {36fc9e60-c465-11cf-8056-444553540000}
Floppy	Class: {4d36e969-e325-11ce-bfc1-08002be10318}
1394 FireWire Host Controller	Class: {6bdd1fc1-810f-11d0-bec7-08002be2092f}
IDE	Class: {4d36e96a-e325-11ce-bfc1-08002be10318}
Tape Drives	Class: {6d807884-7d21-11cf-801c-08002be10318}
CD/DVD Drives	Class: {4d36e965-e325-11ce-bfc1-08002be10318}
Printing Devices	Class: {4d36e979-e325-11ce-bfc1-08002be10318}
PCMCIA	Class: {4d36e977-e325-11ce-bfc1-08002be10318}
Imaging Devices (Scanners, Digital Cameras, etc)	Class: {6bdd1fc6-810f-11d0-bec7-08002be2092f}
Infrared Devices	Class: {6bdd1fc5-810f-11d0-bec7-08002be2092f}
Bluetooth Radios	Class: {e0cbf06c-cd8b-4647-bb8a-263b43f0f974}
SCSI	Class: {4d36e97b-e325-11ce-bfc1-08002be10318}
Modems	Class: {4d36e96d-e325-11ce-bfc1-08002be10318}
Smart Card Readers	Class: {50dd5230-ba8a-11d1-bf5d-0000f805f530}
Ports	Class: {4d36e978-e325-11ce-bfc1-08002be10318}
Network Adapters	Class: {4d36e972-e325-11ce-bfc1-08002be10318}
Biometric	Class: {53d29ef7-377c-4d14-864b-eb3a85769359}
Disk Drives	Class: {4d36e967-e325-11ce-bfc1-08002be10318}
Storage Volumes	Class: {71a27cdd-812a-11d0-bec7-08002be2092f}
Bluetooth Devices (generic)	Class: {95c7a0a0-3094-11d7-a202-00508b9d7d5a}

This list allows having an out of the box solution for standard requests.

Class ID:

To facilitate device installation, devices that are set up and configured in the same manner are grouped into a device setup class. For example, SCSI media changer devices are grouped into the Medium Changer device setup class. The device setup class defines the class installer and class co-installers that are involved in installing the device.

Microsoft defines setup classes for most devices. IHVs and OEMs can define new device setup classes, but only if none of the existing classes apply. For example, a camera vendor does not have to define a new setup class because cameras fall under the Image setup class. Similarly, uninterruptible power supply (UPS) devices fall under the Battery class.

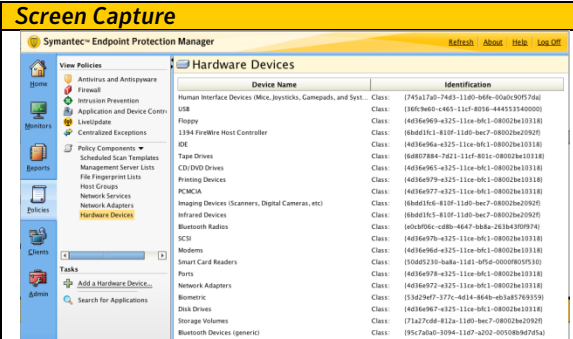
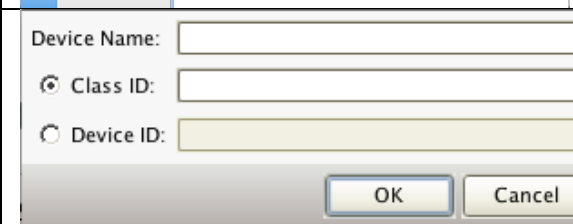
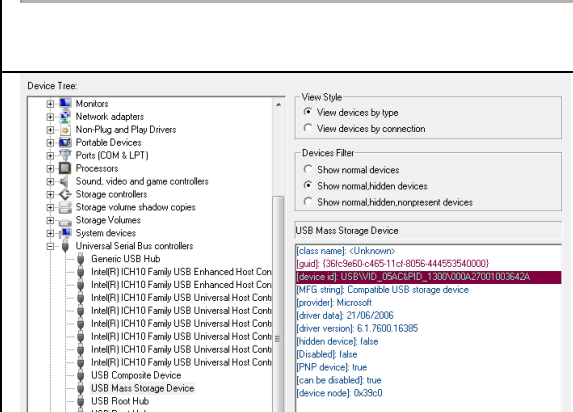
Application and Device control.

There is a GUID associated with each device setup class. System-defined setup class GUIDs are defined in *Devguid.h* and typically have symbolic names of the form *GUID_DEVCLASS_XXX*. The device setup class GUID defines the *..\CurrentControlSet\Control\Class\ClassGUID* registry key under which to create a new subkey for any particular device of a standard setup class.

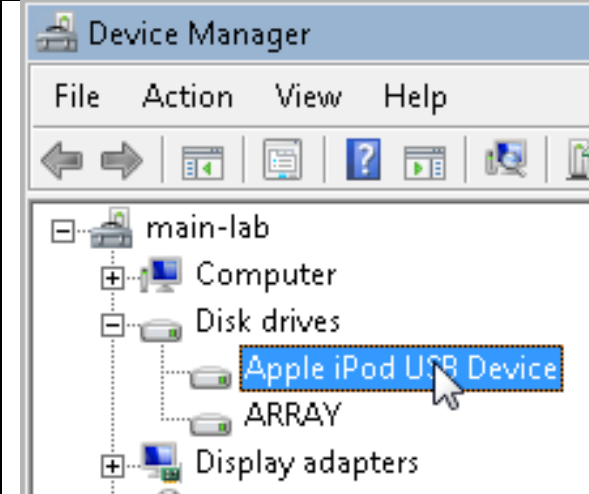
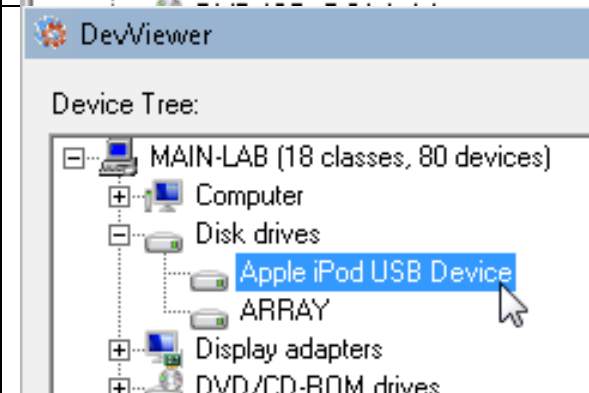

Custom Device

Sometimes these categories do not suffice as you need to allow on ly a specific hardware to work with your SEP clients. (Block iPods, Allow specific WIFI dongle...). For this reason the administrator can add a custom device to integrate in the device control policy alongside with the preconfigured ones.

Adding a new custom device

Screen Capture	Description
	From the policy tab, expand the policy components object and select hardware device. Click Add a hardware device.
	The device name is required to be displayed on the hardware list alongside with the Class ID or the device ID of that specific custom device. Class ID or Device ID can be retrieved from a test client where the device is plugged by using a tool provided on the SEP CD2 called devviewer.
	On the test client plug the hardware you want to control with SEP. Run DevViewer from the CD2 under \Tools\Nosupprt\DevViewer

Application and Device control.

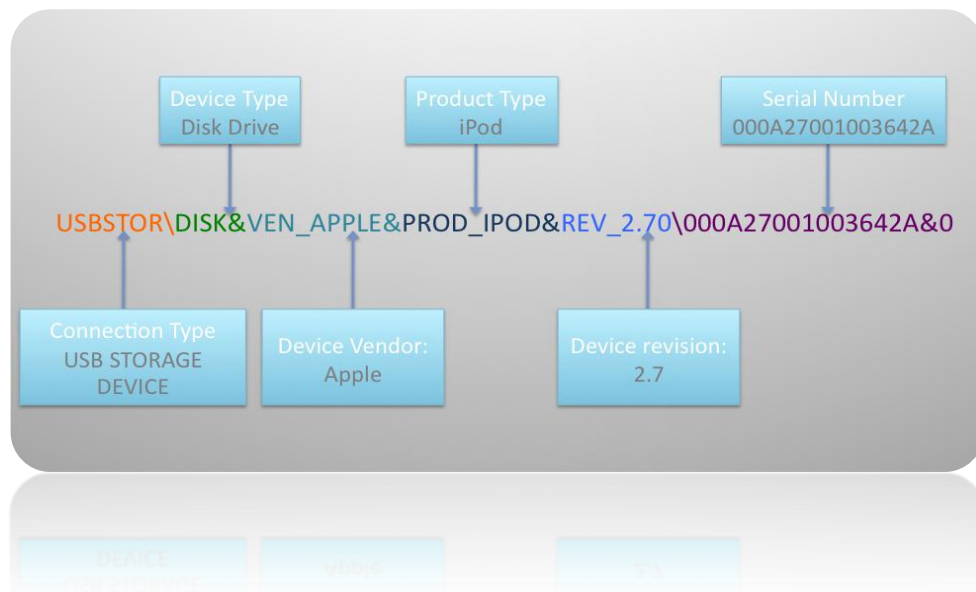
	<p>In this example we've used an iPod Shuffle 1st generation. The iPod is recognized as a USB Mass Storage device.</p>
	<p>Opening devviewer we can also locate the iPod under Disk drive</p>
<p>[class name]: <Unknown> [guid]: {4d36e967-e325-11ce-bfc1-08002be10318} [device id]: USBSTOR\DISK&VEN_APPLE&PROD_IPOD&REV_2.70\000A [MFG string]: (Standard disk drives) [provider]: Microsoft [driver data]: 21/06/2006 [driver version]: 6.1.7600.16385 [hidden device]: false [Disabled]: false [PNP device]: true [can be disabled]: true [device node]: 0x2c2c</p>	<p>On the right pane we can see the device ID for that iPod</p>
<p>[class name]: <Unknown> [guid]: {4d36e967-e325-11ce-bfc1-08002be10318} [device id]: USBSTOR\DISK&VEN_APPLE&PROD_IPOD&REV_2.70\000A2 [MFG string]: (Standard disk drives) [provider]: Microsoft [driver data]: 21/06/2006 [driver version]: 6.1.7600.16385</p> 	<p>By right clicking on the device ID we can copy the required string</p>
<p>USBSTOR\DISK&VEN_APPLE&PROD_IPOD&REV_2.70\000A27001003642A&0</p>	<p>The resulting Device ID for this iPod is now in the clipboard. If you want to block only that kind of iPod then you can directly paste the string in the new hardware definition in SEPM.</p>

Application and Device control.

Device ID

A device ID is a vendor-defined identification string that is the most specific ID that Setup uses to match a device to an INF file. A device has only one device ID. A device ID has the same format as hardware ID. When an enumerator reports a list of hardware IDs for a device, the device ID should be the first hardware ID in the list.

A device id is like a URL as it gives you the connection path to the device manager. For this string you can gather, the port used by the peripheral, the type of device, the manufacturer, the device version and ultimately in some case the serial number.



if you want to block all iPods then you need to add a new hardware description with a wildcard (*) on the device ID string.

USBSTOR\DISK&VEN_APPLE&PROD_IPOD*

Device Name:

☐ Class ID:

☒ Device ID:

Useful Device ID:

Device type	Device ID
Apple iPod	USBSTOR\DISK&VEN_APPLE&PROD_IPOD*
Blackberry SD card	USBSTOR\DISK&VEN_RIM&PROD_BLACKBERRY*
USB Key / Disk	USBSTOR\DISK&
Linksys USB Wi-Fi Dongle	USB\VID_13B1&*

Chapter Three: Use cases for Application and device control

Blocking Unwanted network interfaces / Dialup

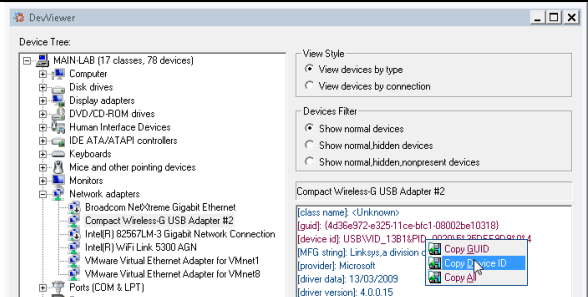
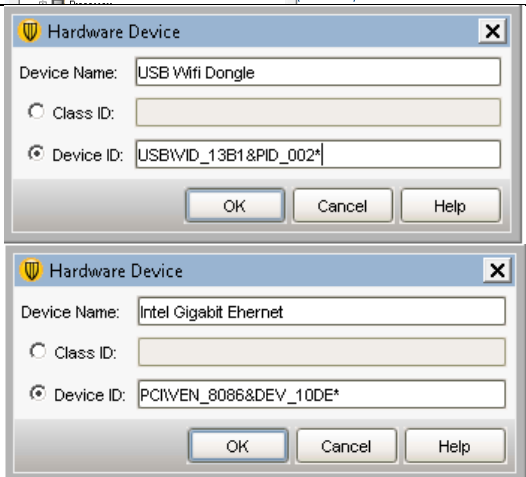
Sometimes, administrators want to ensure that only one network interface is used to avoid leaking corporate information on a private network at home or on the road.

Working in pair with Location awareness it is possible to set Location where specific device will be blocked, like modem or network interface.

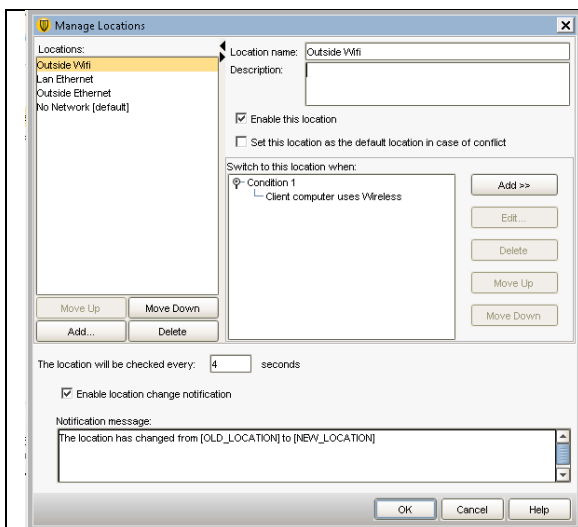
The challenge comes when you want to block Wi-Fi while using Ethernet. You cannot use the Class ID as both interfaces have the same. Two choices are then possible:

- a) **You have a restricted amount of hardware to manage:** Then you can use the Device control policy to add all the Wi-Fi interfaces Devices ID and block these according to what you need.

Example:

Screen Capture	Instructions
	Gather the Devices ID with DevViewer.
	Add the specific Wireless and Ethernet Cards as Hardware in SEPM

Application and Device control.



Create 4 locations in this order :

- Outside Wifi:
 - Computer uses Wireless
- Lan Ethernet
 - Computer uses Ethernet
 - DHCP IP address:192.168.0.254
- Outside Ethernet
 - Computer Uses Ethernet
- No Network (Default)
 - No criteria

Blocked Devices

Use this pane to manage the list of devices to which you want to block access.

Device Name	Identification
Intel Gigabit Ethernet	Device: PCI\VEN_8086&DEV_10DE*

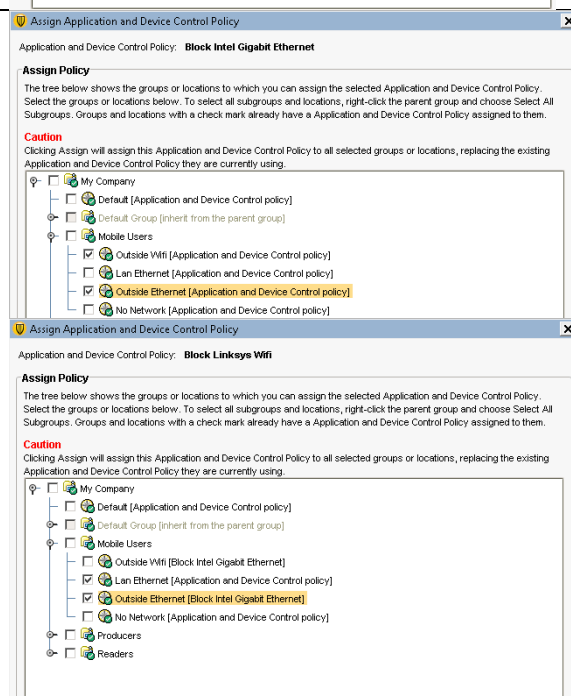
Blocked Devices

Use this pane to manage the list of devices to which you want to block access.

Device Name	Identification
USB Wifi Dongle	Device: USB\VID_13B1&PID_002*

Create 2 Policies for device & application control:

- Block Eth
- Block Wifi



Assign the Policies to the appropriate locations

Application and Device control.

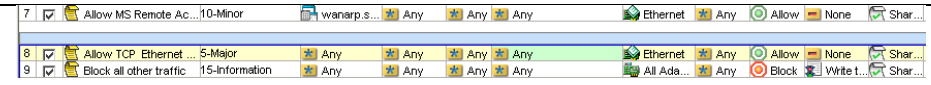
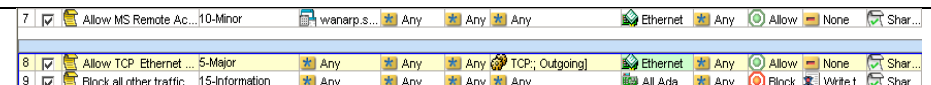
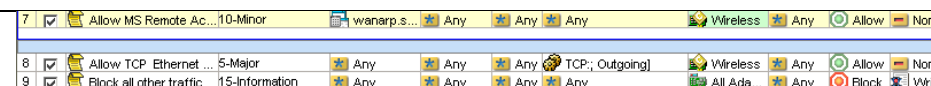
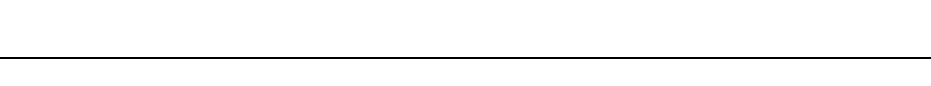
When such procedure is required ensure that all hardware and network connection types are available when creating the policies.

b) You have a large amount of hardware to cover with this requirement:

The main issue with the first option is to gather the Device ID for Network interface. If you work on a large environment and new laptops, 3G cards, are being purchased every so often then maintaining the policy current for all hardware will not be a manageable solution.

A possible solution is to use Location awareness and the firewall rules in order to block DHCP on the interface you are not meant to use. (Eg. Block DHCP on Eth while using a Wifi connection).

Example:

Screen Capture	Intructions
	Create 3 Firewall policies: FW-Lan FW-Eth_out FW-Wifi_out
	FW-Lan : allow all traffic in/out on the Ethernet card. Block all on Wifi
	FW-Eth_out: Allow all traffic outbound on Ethernet. Block all on Wifi
	FW-Wifi_out: Allow all traffic outbound on Wifi. Block all on Ethernet
	Assing the Policies to the appropriate locations
	Test

Note: Should you have issue with unwanted IP address on blocked interface, think of disabling the smart traffic rules.

Application and Device control.

Allowing only corporate purchased USB keys

Organizations can restrict the usage of personal USB keys / Portable devices with the device control policy. For example: a removable storage policy can be implemented to avoid execution of files from a USB storage device and the write access given only to certain individuals.

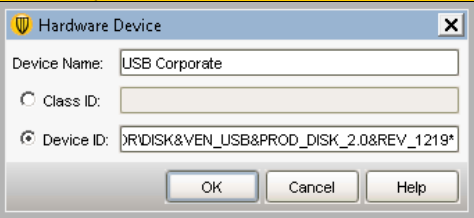

Example:

The company Acme, LTD purchased a stock of “PNY Attaché” USB devices for the employees to exchange documents outside of the corporate network. All other USB storage devices are blocked in every location (Home, Office, VPN). Authorized users are separated in 2 Pools:

- Content Producers: On the Office location and VPN they can write to the corporate USB device
- Content Readers: In all locations they can read documents from the corporate USB device



Device and application control Policy Producers:

Screen Capture	Comments
	Definition of the corporate USB storage device by adding it as a custom Hardware device.
	Creation of a Device & Application control policy assigned to the Locations Office and VPN for the Producers group

Application and Device control.

Blocked Devices

Use this pane to manage the list of devices to which you want to block access.

Device Name	Identification
USB	Class: {36fc9e60-c465-11cf-8056-444553540000}

Add...Delete

Devices Excluded From Blocking

Use this pane to manage the list of devices to which you want to allow access.

Device Name	Identification
Human Interface De...	Class: {745a17a0-74d3-11d0-b6fe-00a0c90f57da}
USB Corporate	Device: USBSTOR\DISK&VEN_USB&PROD_DISK_2.0&REV_1219*

Add...Delete

Only advanced administrators should create Application Control rule sets.

Enabled	Rule Sets	Test/Production
<input type="checkbox"/>	Make all removable drives read-only	Production
<input type="checkbox"/>	Block programs from running from removable drives	Production
<input checked="" type="checkbox"/>	Block applications from running	Production
<input type="checkbox"/>	Protect client files and registry keys	Production
<input type="checkbox"/>	Block writing to USB drives	Production
<input checked="" type="checkbox"/>	Log files written to USB drives	Production
<input type="checkbox"/>	Block modifications to hosts file	Production

Edit Application Control Rule Set

Rule set name: Block applications from running

Description: This rule will block listed applications from running.

Rules

Block applications from running

Block these applications

Properties

This rule defines processes which Symantec Endpoint Protection monitors for attempts specified in the rule conditions.

Rule name: Block applications from running

Description:

☒ Enable this rule

Apply this rule to the following processes:

Do not apply this rule to the following processes:

☐ Sub-processes inherit conditions

Summary:
Block applications from running applies to processes watching *

Edit Process Definition

Process name to match

The name can include environment variables, wildcards (*, ?), and registry keys.
Examples: %windir%\system32* or C:\windows*.exe

☒ Use wildcard matching (* and ? supported)

☐ Use regular expression matching

☐ Only match processes running from the following drive types

☒ Local fixed disk drive☒ Network drive

☒ CD/DVD drive☒ Removable drive (floppy drive, USB drive, etc)

☒ RAM drive

☒ Only match processes running on the following device id type

Device Selection

Device	Device Instance Name	Device Instance ID
Apple iPod		Device: USBSTOR\DISK&VEN_APPLE&PROD_IPOD*
USB Corporate		Device: USBSTOR\DISK&VEN_USB&PROD_DISK_2.0*

Device control tab:

Block all USB

Allow USB corporate custom Hardware and Human Interface Devices (for Keyboard ad Mouse)

Application Control tab:

Select Block applications from running

Select Log File written to USB drives for accountability of information’s held on corporate USB sticks.

Block application from running ruleset:

No modification from the template on the main screen.

Click on the * for “Apply the rule to the following processes”.

Deselect “Only match processes running from the following drive types”

Select “Only match processes running on the following device type” and select USB corporate.

Monitor Device ID centrally

Another use case of the device and application policy is to monitor the registry for devices used in an environment. Each time a user plugs a new device to its system a new registry key is created under the PnP enumerator for USB devices. By monitoring the appropriate registry key we can then gather in a log the device ID for this peripheral in a log.

Example:

Create a new rule set called Monitor USB and disk Drive Regkeys

Only advanced administrators should create Application Control rule sets.

Enabled	Rule Sets	Test/Production
<input checked="" type="checkbox"/>	Monitor USB and Disk Drive Regkeys	Production

Create rule monitoring reg keys

Create a registry key access attempt rule for the following keys:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB*\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB*\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceClasses\{53f56307-b6bf-11d0-94f2-00a0c91efb8b}\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceClasses\{53f56307-b6bf-11d0-94f2-00a0c91efb8b}\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceClasses\{53f56308-b6bf-11d0-94f2-00a0c91efb8b}\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceClasses\{53f56308-b6bf-11d0-94f2-00a0c91efb8b}\*\*
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceClasses\{53f5630d-b6bf-11d0-94f2-00a0c91efb8b}\*\*
```

Application and Device control.

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceClasses\{53f5630d-b6bf-11d0-94f2-00a0c91efb8b}\*\*.
```

For each of these key set the following actions:

The screenshot shows the 'Properties' and 'Actions' tabs for a device class. The 'Read Attempt' section on the left and the 'Create, Delete, or Write Attempt' section on the right both have the same configuration: 'Continue processing other rules' is selected, 'Allow access', 'Block access', and 'Terminate process' are unselected, 'Enable logging' is checked, 'Severity' is set to 'Info -- 15', 'Send Email Alert' is unchecked, and 'Notify user' is unchecked. There are also empty text boxes for additional configuration.

Apply the policy to the group of client you want to monitor and then consult the logs from the console.

The screenshot shows the 'Logs' tab for 'Application and Device Control Logs: Application Control'. The 'What type of log would you like to see?' section is expanded, showing filters for log type, content, and filter settings. The 'Log type' is set to 'Application and Device Control', 'Log content' is set to 'Application Control', and 'Use a saved filter' is set to 'Default'. The 'What filter settings would you like to use?' section is also expanded, showing filters for time range, severity, test mode, event type, action, and various system properties like Site, Domain, Group, Server, Computer, User, and Caller process. The 'How would you like to view this log?' section is collapsed, showing a limit of 20 entries. There are buttons for 'Save Filter...' and 'View Log'.

Application and Device control.

Time	Action	Domain Computer	User	Severity	Rule Name	Caller Process	Target
11/27/2009 14:19:17	Continue	Default secclient	SYSTEM	Info	Registry Access Attempts_Read Registry	C:/WINDOWS/system32/services.exe	/REGISTRY/MACHINE/SYSTEM/CurrentControlSet/Enum/USB/ROOT_HUB/5&1DC927FF&0
11/27/2009 14:19:17	Continue	Default secclient	SYSTEM	Info	Registry Access Attempts_Read Registry	C:/WINDOWS/system32/services.exe	/REGISTRY/MACHINE/SYSTEM/CurrentControlSet/Enum/USB/ROOT_HUB20/5&2F792170&0

In the following a Blackberry has been plugged into a monitored system.

Domain name: Default
 Site name: Site One
 API: Registry Read
 Action: Continue
 Test mode: No
 Windows domain: SEETEST
 User: SYSTEM
 Server name: WIN-9DZKDMZW8PD
 Group name: My Company/Default Group
 Computer Name
 Current: secclient
 When event occurred: secclient

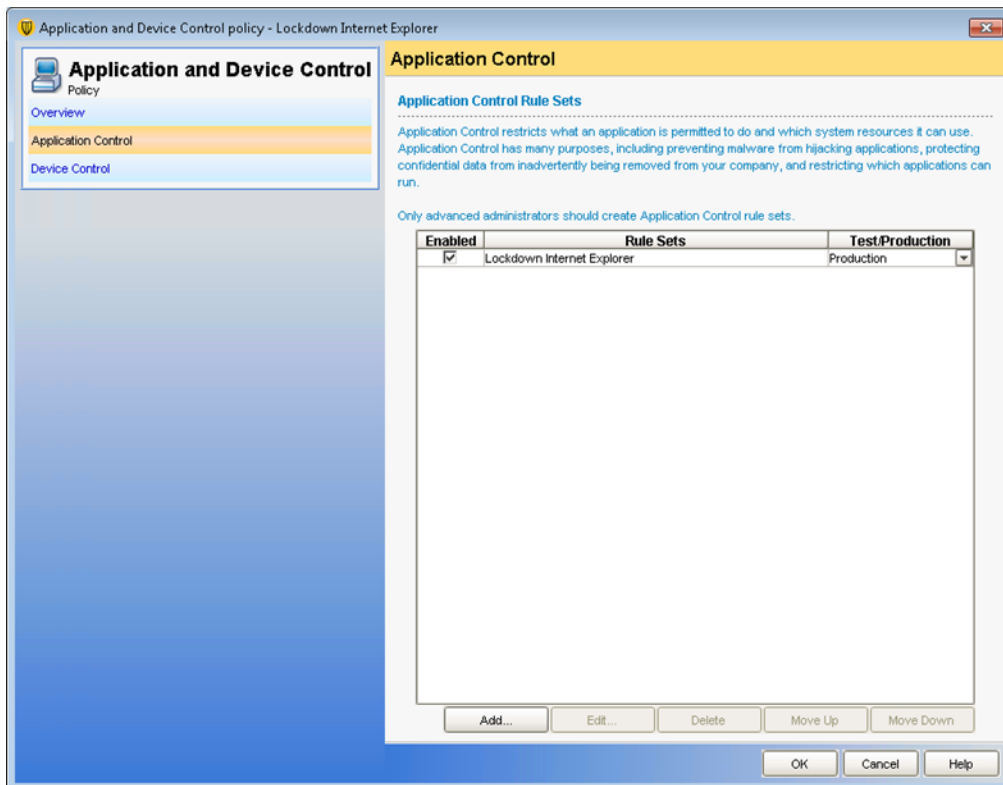
Event type: Application Control Rules
 Event time: 11/27/2009 14:19:17
 Severity: Info
 Begin time: 11/27/2009 14:18:10
 End time: 11/27/2009 14:18:10
 Rule name: Registry Access Attempts_Read Registry
 Alert: No
 Send SNMP trap: 0
 Caller Process ID: 1324
 Caller Process Name: C:/WINDOWS/system32/services.exe
 Target: /REGISTRY/MACHINE/SYSTEM/CurrentControlSet/Enum/USB/STOR/DISK&VEN_RIM&PROD_BLACKBERRY&REV_1002/6&2568F3B3&0&36427353B74AE1A28A489E93F9A2B53DD8531B1B&0
 User name: SYSTEM
 Description:

Application and Device control.

Harden Internet Explorer security against drive by downloads

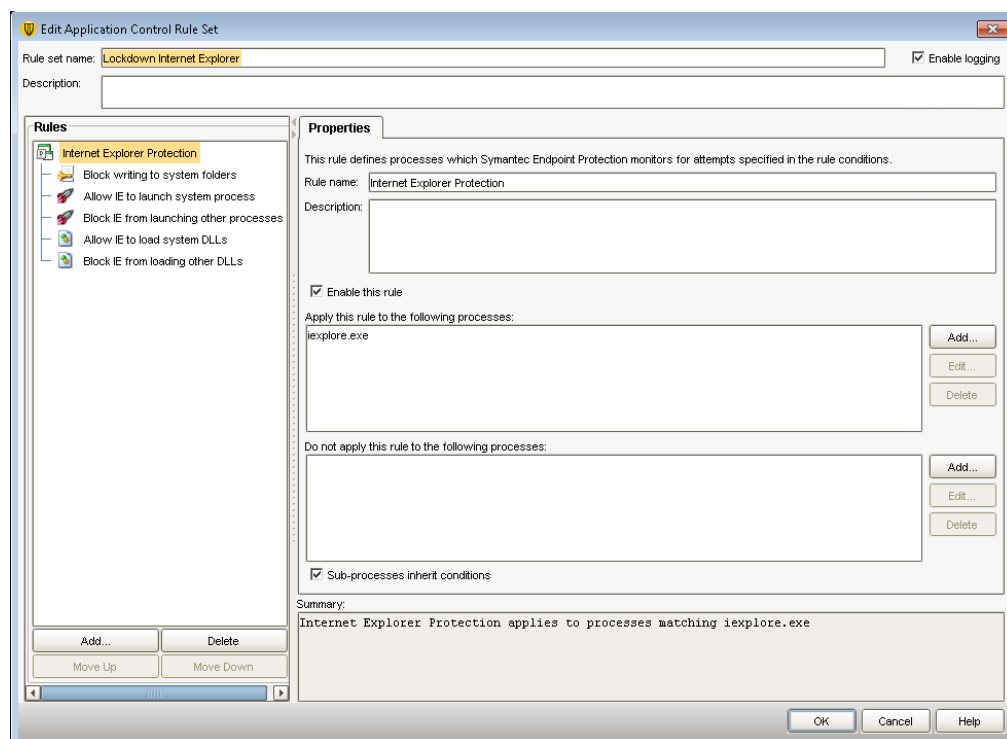
Internet explorer and other browsers are the most common target for attackers to infiltrate malicious code on endpoint. With Application control you can add extra security by allowing the browser to write only where necessary to build the image/content cache to display pages while restricting the execution of scripts.

The following example illustrate Internet explorer, but you can easily adapt it to the browser of your choice.

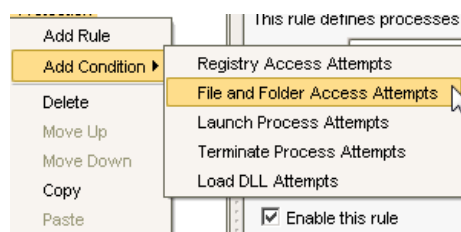
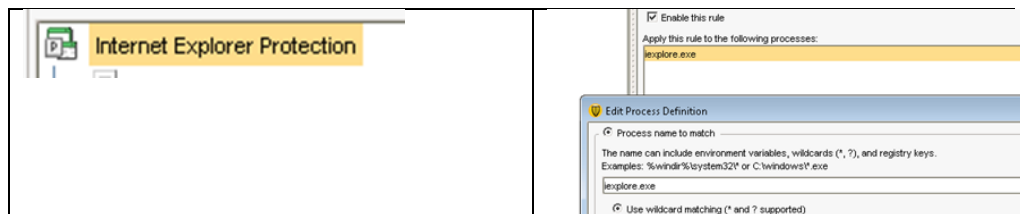


Create a new rule set for the application control Panel.

Application and Device control.



The rule set applies only to the process **iexplorer.exe**



Create a new file and folder access attempt condition.

Block writing to system folders applies to files and folders matching:

`%windir%**,%programfiles%**`

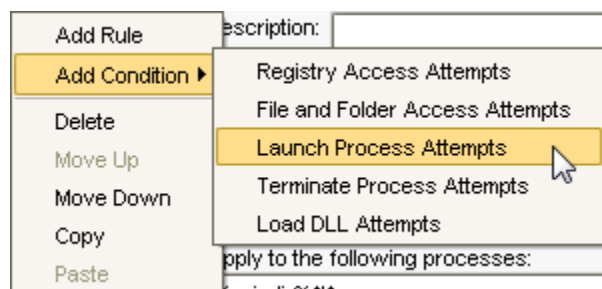
Exclude files and folders matching:

`**softwaredistribution*,**softwaredistribution***,**windowsupdate*,**windowsupdate***,%windir%\profile***`

Action:

Create, delete, or write attempt: Block (Log)

Application and Device control.



Create a new Launch process attempt to prevent IE launching scripts.

Allow IE to launch system process applies to processes matching:

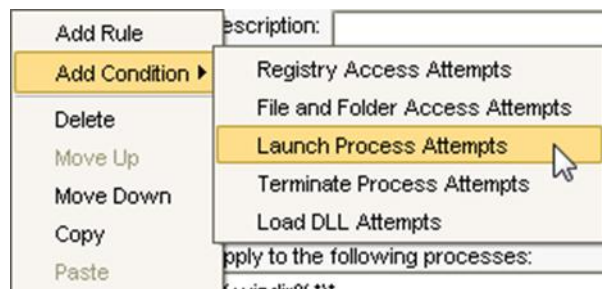
%windir%**,%programfiles%**

Exclude:

Processes matching *script*.exe

Action:

Launch process attempt: Allow (Log)



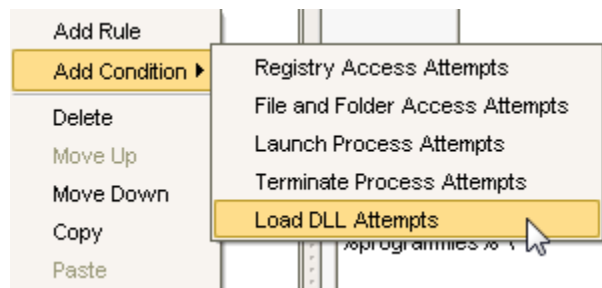
Create a new Launch process attempt to block IE from launching any other processes (like a malware for example).

Block IE from launching other processes applies to processes matching:

*

Action:

Launch process attempt: Block (Log)



Create a new Load DLL attempt condition to allow IE to launch system dlls required to run.

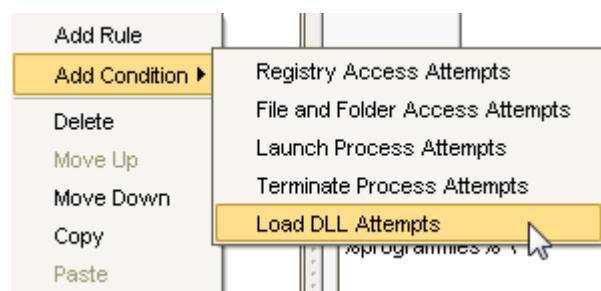
Allow IE to load system DLLs applies to DLLs matching:

%windir%**,%programfiles%**

Action:

Load DLL attempt: Allow

Application and Device control.



Create a new Load DLL attempt condition to prevent IE from launching any other dlls.

Block IE from loading other DLLs applies to DLLs matching:

*

Action:

Load DLL attempt: Block (Log)