Nine Ways to Restrict End-Users Who Have Windows Admin Privileges

WHITE PAPER

Contents

This document is provided strictly as a guide. No guarantees can be provided or expected. This document contains the confidential information and/or proprietary property of Ivanti, Inc. and its affiliates (referred to collectively as "Ivanti"), and may not be disclosed or copied without prior written consent of Ivanti.

Ivanti retains the right to make changes to this document or related product specifications and descriptions, at any time, without notice. Ivanti makes no warranty for the use of this document and assumes no responsibility for any errors that can appear in the document nor does it make a commitment to update the information contained herein. For the most current product information, please visit www.Ivanti.com.

Introduction

How many Windows administrator accounts exist in your company? If the answer is lots of them, then it's likely that many of these privileged accounts are used by individuals who don't need this level of access to perform their role; or who only need privileged access for one or two tasks.

Providing full admin rights to users who aren't trained as IT system administrators increases security risk and manageability costs, and makes it difficult to achieve compliance. This white paper presents nine simple restrictions you can implement immediately using Ivanti® Security Controls that, when combined, will:

- Reduce the likelihood an individual will inadvertently change some administrative settings and require IT assistance to fix the issue.
- Make it more difficult for an individual to alter or disable certain protections on your endpoints.

For even better protection, your long-term goal should be to change any unnecessary administrator accounts to standard user accounts and employ a least privilege approach. This can also be accomplished using Ivanti Security Controls but is beyond the scope of this white paper.

The functionality to restrict non-admin admins is part of the Privilege Management sub-feature of Application Control, which is part of Ivanti Security Controls. An Application Control engine that is part of the Security Controls agent runs on each endpoint. This engine enforces a set of rules that are defined within a configuration and specify how Application Control behaves. The screenshots in this paper show the configuration required for each of the restrictions.

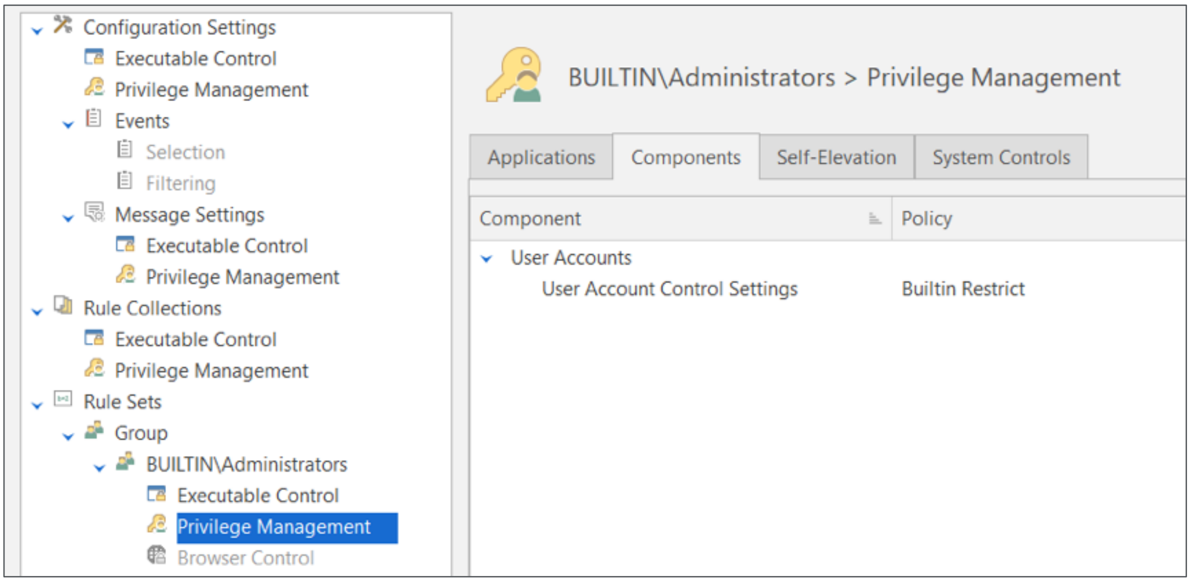
Restriction 1 – Prevent Users from Changing the UAC Setting

Ivanti advises that all your end-users, including those with admin accounts, employ User Account Control. Microsoft states on its website:

> *User Account Control (UAC) helps prevent malware from damaging a PC and helps organizations deploy a better-managed desktop. With UAC, apps and tasks always run in the security context of a non-administrator account, unless an administrator specifically authorizes administrator-level access to the system. UAC can block the automatic installation of unauthorized apps and prevent inadvertent changes to system settings.*

It is advisable that UAC is active on all machines, with the Always Notify setting selected. This is activated via group policy. Ivanti Security Controls can prevent administrators from turning off User Account Control.

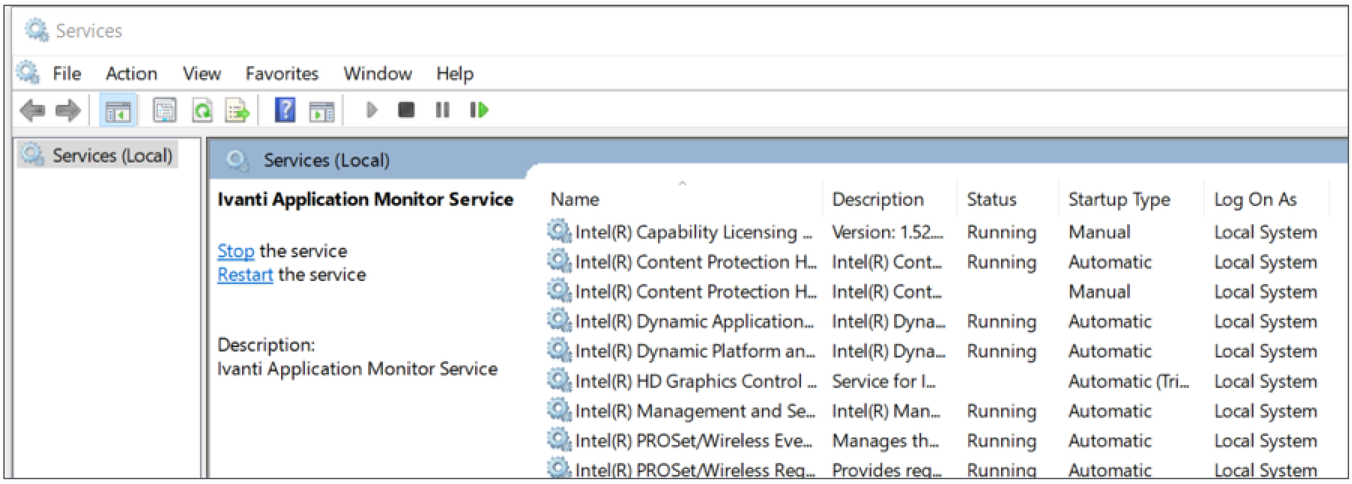
In the Configuration Editor shown below, navigate to the Components tab and select the 'User Account Control Settings' Windows component. Then set the policy of the rule item to Builtin Restrict.



Component	Policy
User Accounts	
User Account Control Settings	Builtin Restrict

Restriction 2 – Prevent Users from Running the MMC with Admin Privileges

Microsoft's management console (MMC) is a framework that provides end-users with an interface for management and configuration of the operating system. It allows the end-user to load snap-ins. Each snap-in is a tool to manage a particular Windows feature, e.g., the Services snap-in provides a tool to manage Windows services.



Ivanti Application Monitor Service

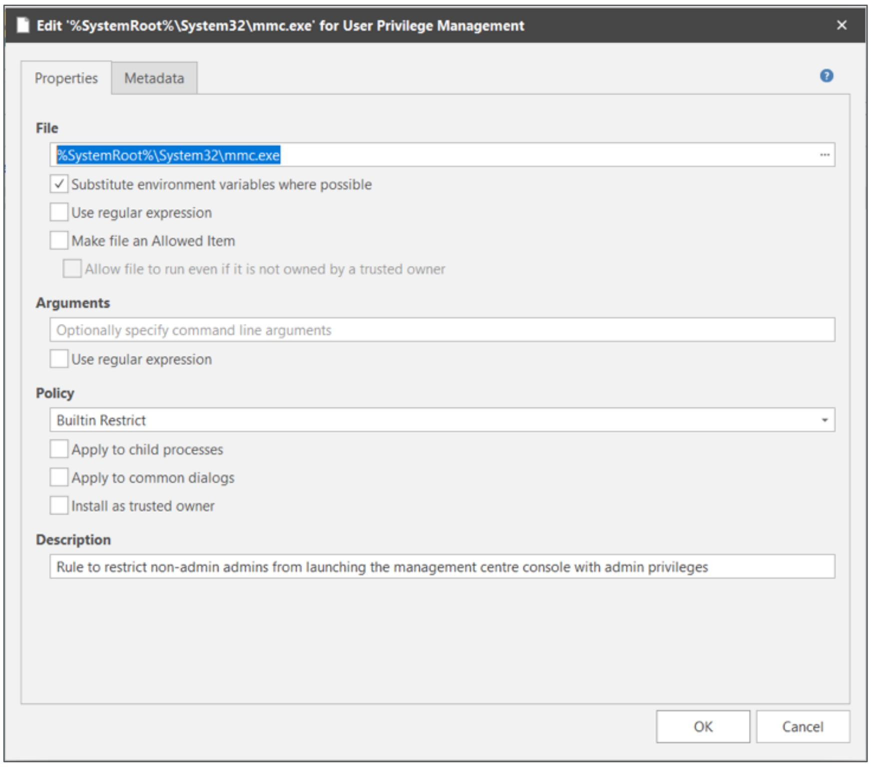
Stop the service
Restart the service

Description:
Ivanti Application Monitor Service

Name	Description	Status	Startup Type	Log On As
Intel(R) Capability Licensing ...	Version: 1.52....	Running	Manual	Local System
Intel(R) Content Protection H...	Intel(R) Cont...	Running	Automatic	Local System
Intel(R) Content Protection H...	Intel(R) Cont...		Manual	Local System
Intel(R) Dynamic Application...	Intel(R) Dyna...	Running	Automatic	Local System
Intel(R) Dynamic Platform an...	Intel(R) Dyna...	Running	Automatic	Local System
Intel(R) HD Graphics Control ...	Service for I...		Automatic (Tri...	Local System
Intel(R) Management and Se...	Intel(R) Man...	Running	Automatic	Local System
Intel(R) PROSet/Wireless Eve...	Manages th...	Running	Automatic	Local System
Intel(R) PROSet/Wireless Req...	Provides req...	Running	Automatic	Local System

This can be very powerful. For example, in the Services snap-in, an end-user with admin privileges can stop services. If a stopped service was part of your antivirus software, it could disable antivirus scanning on any downloads, increasing the threat from malware.

With Ivanti Security Controls, you can prevent end-users with administrator accounts from running the MMC with administrator privileges. In the Configuration Editor, create a file rule in the Applications tab. Then set the policy of the rule item to Builtin Restrict.



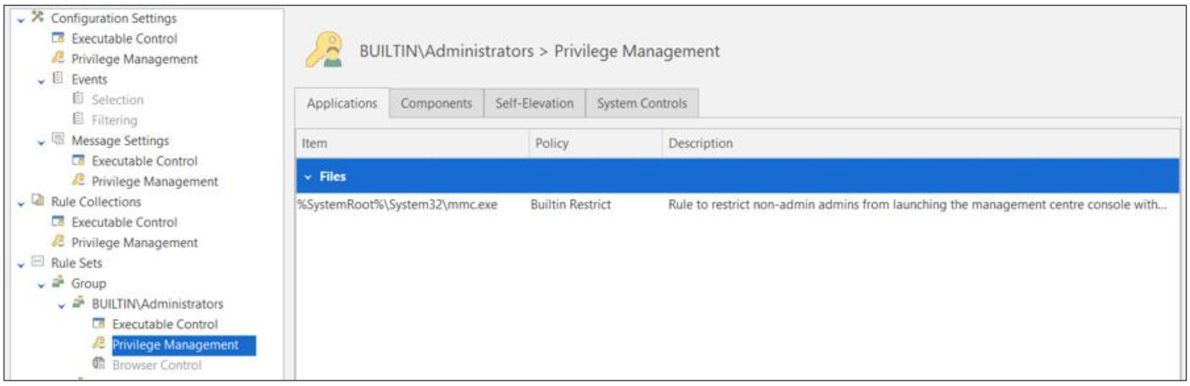
Edit '%SystemRoot%\System32\mmc.exe' for User Privilege Management

File
%SystemRoot%\System32\mmc.exe
☑ Substitute environment variables where possible
□ Use regular expression
□ Make file an Allowed Item
□ Allow file to run even if it is not owned by a trusted owner

Arguments
Optionally specify command line arguments
□ Use regular expression

Policy
Builtin Restrict
□ Apply to child processes
□ Apply to common dialogs
□ Install as trusted owner

Description
Rule to restrict non-admin admins from launching the management centre console with admin privileges



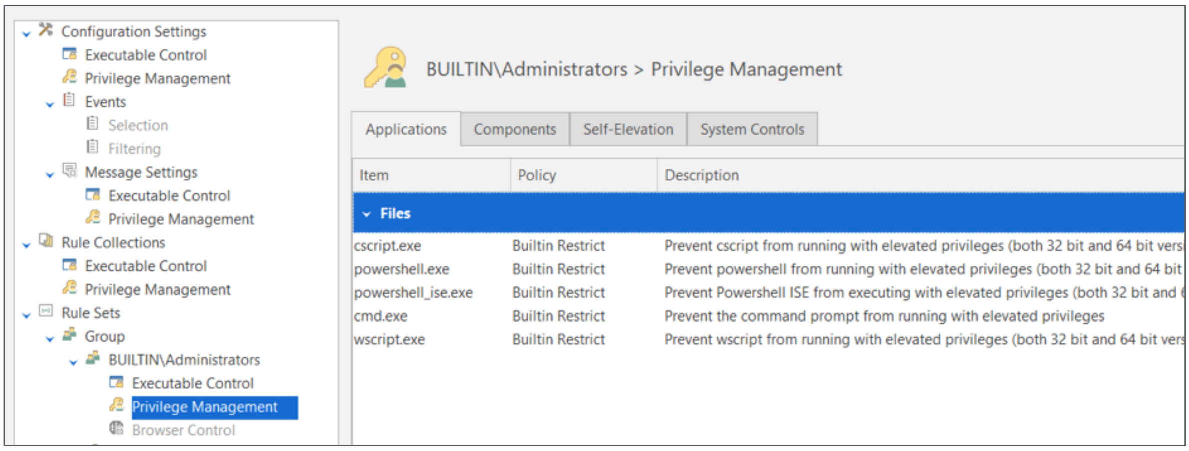
Item	Policy	Description
%SystemRoot%\System32\mmc.exe	Builtin Restrict	Rule to restrict non-admin admins from launching the management centre console with...

Restriction 3 – Prevent Users from Running Commands or Scripts with Admin Privileges

Windows supports some alternative ways to interact with the operating system by issuing specific commands via command line interpreters or by executing scripts. The former is used for management purposes and the latter is typically used for automation. The command line interfaces that come with Windows are the Command Prompt and Windows PowerShell. Also included with the operating system is the Windows Script Host that can be used to run scripts in a variety of scripting languages.

Anyone with administrative privileges can execute commands or scripts, and this provides an alternative method in many cases to override the GUI-based restrictions outlined in this paper. For this reason, restrict command line interpreters and the Windows Script Host.

Application Control can help with this. To configure this restriction, create a file rule for each command line interpreter and the Windows and console versions of the Windows Script Host. Create each rule item in the Applications tab and set the policy of each of them to Builtin Restrict.



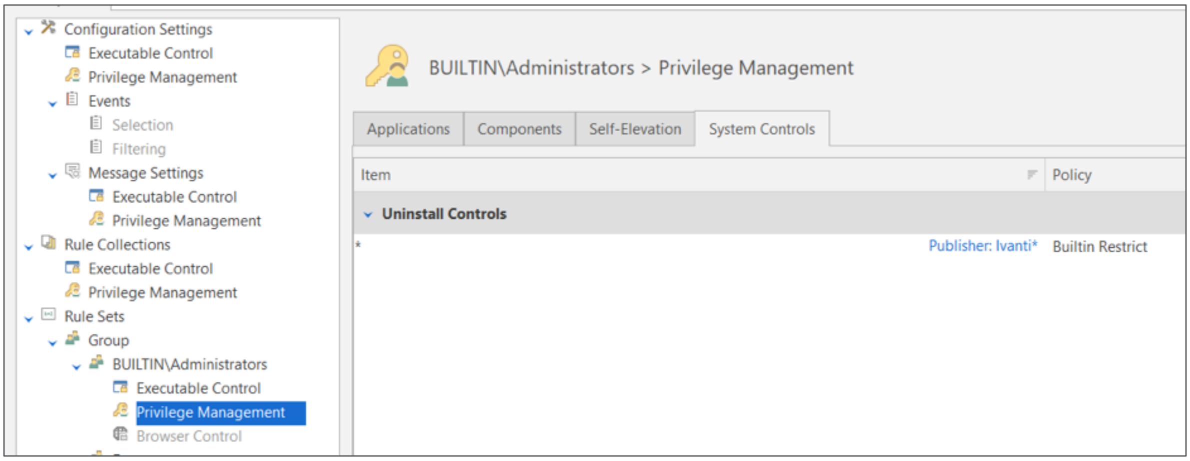
Item	Policy	Description
cscript.exe	Builtin Restrict	Prevent cscript from running with elevated privileges (both 32 bit and 64 bit vers
powershell.exe	Builtin Restrict	Prevent powershell from running with elevated privileges (both 32 bit and 64 bit
powershell_ise.exe	Builtin Restrict	Prevent Powershell ISE from executing with elevated privileges (both 32 bit and 6
cmd.exe	Builtin Restrict	Prevent the command prompt from running with elevated privileges
wscript.exe	Builtin Restrict	Prevent wscript from running with elevated privileges (both 32 bit and 64 bit vers

Should it become necessary to run a script with elevated privilege, then Application Control can also be configured to allow this to run only for that script.

Restriction 4 – Prevent Users from Uninstalling Third-party Software Protecting Your System

Ivanti Security Controls works through an agent that is installed onto your endpoints to protect them. This agent could be uninstalled by an end-user with admin privilege, causing that protection to be lost. To prevent this from happening, configure an uninstall rule within the System Controls tab. It’s possible to target the uninstall rule to a specific version of an application, or wildcards can be used to make the rule apply more generally.

In the screenshot below, only the publisher is set. Any application that matches the publisher cannot be uninstalled by an end-user with admin privilege.



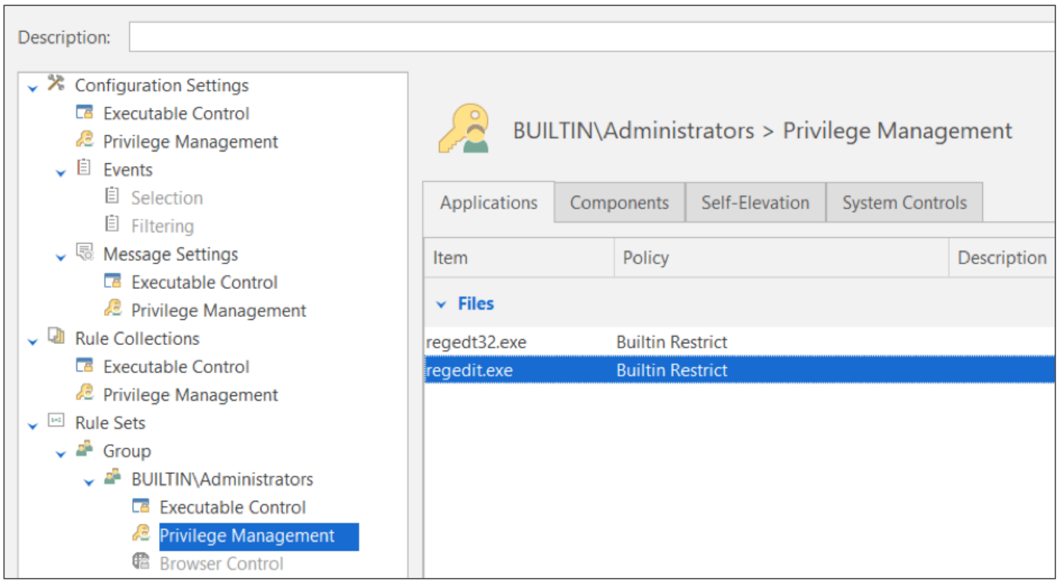
Item	Policy
Uninstall Controls	
* Publisher: Ivanti*	Builtin Restrict

You can also use Ivanti Security Controls to prevent other third-party software from being uninstalled by creating a similar rule.

Restriction 5 – Prevent Users from Being Able to Edit System Settings in the Registry

The Windows registry is important because it stores vital information about a Windows endpoint and its configuration, as well as information about all application programs that are installed. By offering the ability to directly access and change registry keys, admin privilege allows end-users to navigate around central management policies whenever they choose and change the settings. This access provides another route for users to circumvent many of the protections described in this paper.

At a minimum, you should restrict privileged access to the registry. Using Ivanti Security Controls, this restriction is configured by creating file rules in the Applications tab for each registry-editing application and setting their policy to Builtin Restrict.



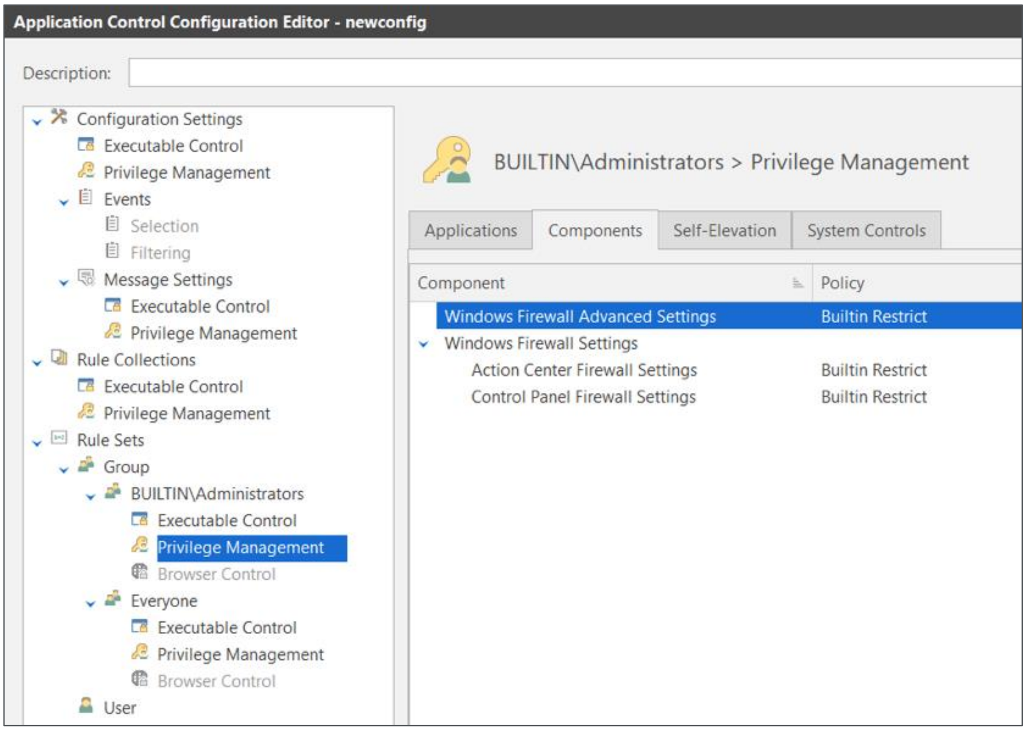
Item	Policy	Description
Files		
regedt32.exe	Builtin Restrict	
regedit.exe	Builtin Restrict	

Going one stage further, Ivanti Security Controls can prohibit a user—with or without admin privilege—from even accessing the application used to edit the registry.

Restriction 6 – Prevent Users from Disabling or Changing Endpoint Firewall Settings

A firewall is a network-security device that monitors traffic to or from your network and allows or blocks traffic based on a defined set of security rules. Firewalls make it more difficult for malicious software to spread throughout a network.

If you're using the Windows Defender firewall, then we recommend preventing this functionality from being disabled. To do this, select the following Windows components under the Privilege Management node and set the policy on each to Builtin Restrict.

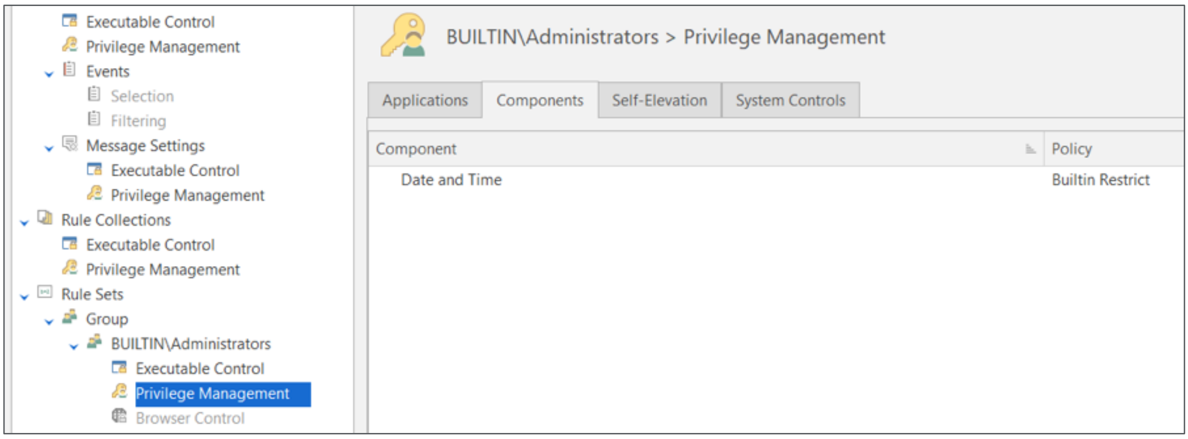


Component	Policy
Windows Firewall Advanced Settings	Builtin Restrict
Windows Firewall Settings	
Action Center Firewall Settings	Builtin Restrict
Control Panel Firewall Settings	Builtin Restrict

Restriction 7 – Prevent Users from Changing the Date and Time

Non-admin administrators can change the date and time, which may sound like a small thing, but it could have some profound effects. If the date or time is wrong, then many applications could behave incorrectly. Furthermore, any logging will have incorrect timestamps, which potentially invalidates auditing and makes troubleshooting much more difficult. Another reason to restrict this is that some people may try to use this to get around license restrictions and keep using the same trial license over and over by setting the clock back.

Ivanti Security Controls can prevent administrators from changing the date and time. Select the Date & Time component found under the Windows Components tab. Then set the policy on the component to Builtin Restrict.



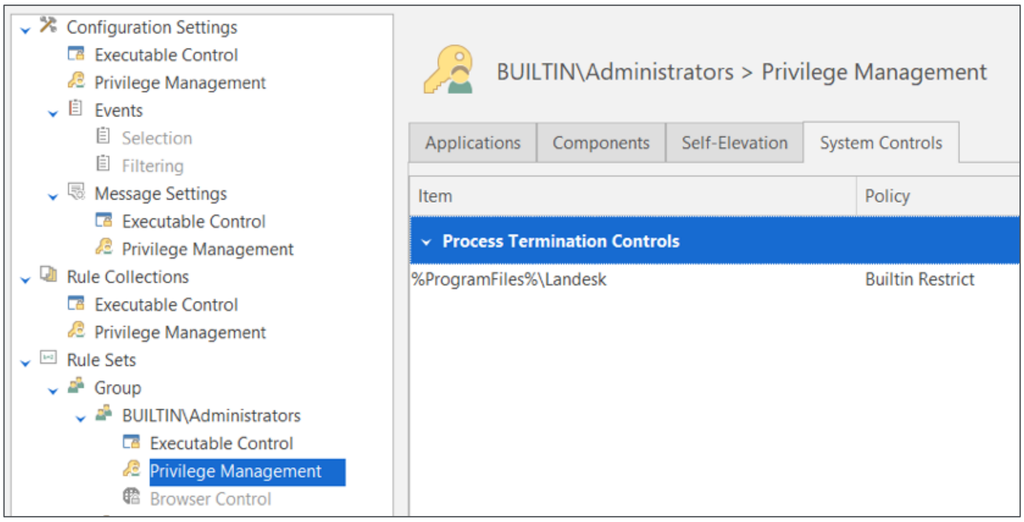
Component	Policy
Date and Time	Builtin Restrict

It’s also possible to change the date and time via an elevated command line using the date and time commands. Protection against this is covered by Restriction 3.

Restriction 8 – Prevent Users from Terminating Processes

End-users with an admin privilege can terminate running processes; for example: 1) using Task Manager and choosing to End Task; or 2) running Process Explorer and choosing to Kill Process. These represent another way that end-users could disable protection software running on their system, and by so doing, increase their security risk.

To protect Ivanti Security Controls agent software from being terminated, add a Process Termination Control folder rule item to the configuration in the System Controls tab. The rule should specify the folder location where the software you want to protect is stored (in this case, %ProgramFiles%\Landesk). Its policy should be set to Builtin Restrict.



Item	Policy
Process Termination Controls	
%ProgramFiles%\Landesk	Builtin Restrict

You can also add other rules to the AC configuration to prevent termination of any other software.

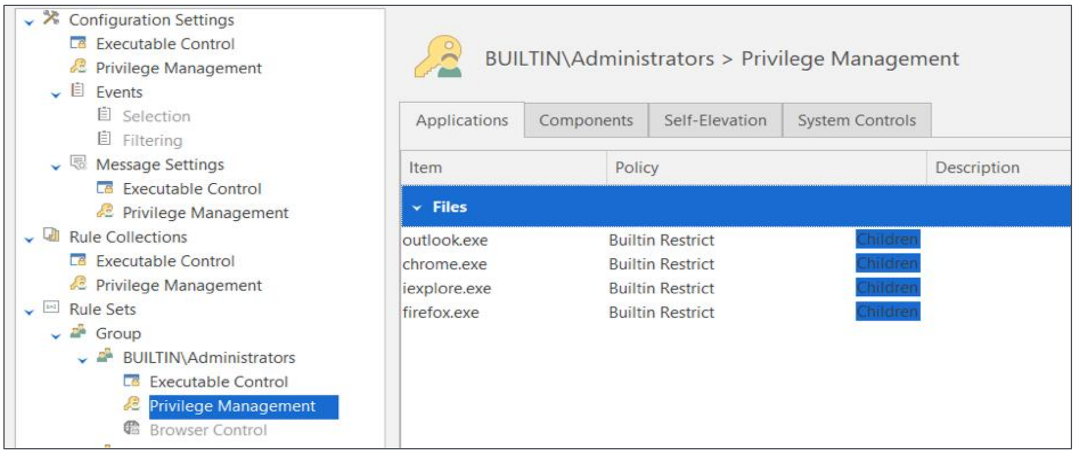
Note that this restriction prevents terminating processes from certain GUI applications, but termination can also be done from an elevated command prompt, e.g., using the ‘taskkill’ command. However, Restriction 3 will prevent this.

Restriction 9 – Prevent Users from Elevating Applications that Could Introduce Malware

End-users with admin privilege can launch any application with elevated privileges. Some of these applications—email and browser applications, for example—can introduce malware to an endpoint. Typically these applications do not run elevated unless a user with admin privilege elevates it on purpose. This restriction will prevent them from doing so.

If one of these applications is running with admin privilege, then any child processes spawned containing malware would also run with admin privilege. Therefore, we want to restrict these applications so that they only ever run with standard privileges.

Using Ivanti Security Controls, this restriction is configured by creating file rules in the Applications tab for each browser or email application and setting their policy to Builtin Restrict. In addition, the option to apply to child process should also be checked for each item. We have included the most common ones in the screenshot below:



Item	Policy	Description
Files		
outlook.exe	Builtin Restrict	Children
chrome.exe	Builtin Restrict	Children
iexplore.exe	Builtin Restrict	Children
firefox.exe	Builtin Restrict	Children

What Next?

This completes the overview of the restrictions that will: 1) reduce the likelihood non-admin admins will change some administrative settings inadvertently and require IT assistance to fix the issue; or 2) increase the difficulty for non-admin admins to alter or disable certain protections on your endpoints.

Want to see firsthand what Application Control can do for you? Request a demo.

Learn More | **www.ivanti.uk** | ☎ **+44 (0) 1344 442100** | ✉ **sales@ivanti.com**