Australian Government
Australian Signals Directorate

ASD AUSTRALIAN SIGNALS DIRECTORATE
ACSC Australian Cyber Security Centre

Implementing Application Control

First published: June 2011
Last updated: November 2023

Introduction

Application control is one of the most effective mitigation strategies in ensuring the security of systems. As such, application control forms part of the Essential Eight from the *Strategies to Mitigate Cyber Security Incidents*.

This publication provides guidance on what application control is, what application control is not, and how to implement application control.

What application control is

Application control is a security approach designed to protect against malicious code (also known as malware) executing on systems. When implemented robustly, it ensures only approved applications (e.g. executables, software libraries, scripts, installers, compiled HTML, HTML applications, control panel applets and drivers) can be executed.

While application control is primarily designed to prevent the execution and spread of malicious code, it can also prevent the installation or use of unapproved applications.

What application control is not

The following approaches are not considered to be application control:

- providing a portal or other means of installation for approved applications
- using web or email content filtering to prevent users from downloading applications from the internet
- checking the reputation of an application using a cloud-based service before it is executed
- using a next-generation firewall to identify whether network traffic is generated by an approved application.

How to implement application control

Implementing application control involves the following high-level steps:

- identifying approved applications
- developing application control rules to ensure only approved applications are allowed to execute

- maintaining application control rules using a change management program
- validating application control rules and their implementation on an annual or more frequent basis.

When determining how to enforce application control, the following methods are considered suitable if implemented correctly:

- cryptographic hash rules
- publisher certificate rules (combining both publisher names and product names)
- path rules (ensuring file system permissions are configured to prevent unauthorised modification of folder and file permissions, folder contents and individual files).

Conversely, the use of file names, package names or any other easily changed application attribute is not considered suitable as a method of application control.

In addition to preventing the execution of unapproved applications, application control can contribute to the identification of attempts by malicious actors to execute malicious code. This can be achieved by configuring application control to generate event logs for allowed and blocked executions. Such event logs should ideally include information such as the name of the file, the date/time stamp and the username of the user attempting to execute the file.

Finally, it is important that application control does not replace antivirus and other security software already in place on systems. Using multiple security solutions together can contribute to an effective defence-in-depth approach to preventing the compromise of systems.

Implementing application control within Microsoft Windows environments

Using Windows Defender Application Control

Windows Defender Application Control (WDAC), a security feature of Microsoft Windows 10 and Microsoft Windows 11, uses code integrity policies to restrict what code can run in both user mode and kernel mode. WDAC can also use virtualisation to protect itself from being disabled by malicious actors that have obtained administrative privileges.

If WDAC is used for application control, the following Group Policy settings can be implemented, noting additional hardware requirements need to be met for the use of virtualisation-based security.

Group Policy Setting	Recommended Option
Computer Configuration\Policies\Administrative Templates\System\Device Guard	
Deploy Windows Defender Application Control	Enabled Code Integrity Policy file path: *<organisation defined>*
Turn On Virtualization Based Security	Enabled Virtualization Based Protection of Code Integrity: Enabled with UEFI lock

Additional information on WDAC is available from Microsoft in their *WDAC design guide*, *WDAC deployment guide* and *WDAC operational guide*. Furthermore, the WDAC policy wizard can assist organisations in creating WDAC policies.

Using Microsoft AppLocker

If Microsoft AppLocker (the predecessor of WDAC) is used for application control, the following rules can be used for a basic path-based implementation. In support of this, the rules, enforcement of rules and the automatic starting of the Application Identity service should be set via Group Policy at a domain level.

Application Control Rule	Recommended Option
Computer Configuration\Policies\Windows Settings\Security Settings\Application Control Policies\AppLocker\DLL Rules	
[Path] %ProgramFiles%*	Allow Everyone
[Path] %WinDir%*	Allow Everyone Exceptions: %System32%\Microsoft\Crypto\RSA\MachineKeys\* %System32%\spool\drivers\color\* %System32%\Tasks\* %WinDir%\Tasks\* %WinDir%\Temp\*
Computer Configuration\Policies\Windows Settings\Security Settings\Application Control Policies\AppLocker\Executable Rules	
[Path] %ProgramFiles%*	Allow Everyone
[Path] %WinDir%*	Allow Everyone Exceptions: %System32%\Microsoft\Crypto\RSA\MachineKeys\* %System32%\spool\drivers\color\* %System32%\Tasks\* %WinDir%\Tasks\* %WinDir%\Temp\*
Computer Configuration\Policies\Windows Settings\Security Settings\Application Control Policies\AppLocker\Packaged app Rules	
[Publisher] CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US	Allow Everyone
[Publisher] CN=Microsoft Windows, O=Microsoft Corporation, L=Redmond, S=Washington, C=US	Allow Everyone
Computer Configuration\Policies\Windows Settings\Security Settings\Application Control Policies\AppLocker\Script Rules	
[Path] %ProgramFiles%*	Allow Everyone

[Path] %WinDir%*	Allow Everyone Exceptions: %System32%\Com\dmp\* %System32%\FxsTmp\* %System32%\Microsoft\Crypto\RSA\MachineKeys\* %System32%\spool\drivers\color\* %System32%\spool\PRINTERS\* %System32%\spool\SERVERS\* %System32%\Tasks\* %WinDir%\Registration\CRMLog\* %WinDir%\Tasks\* %WinDir%\Temp\* %WinDir%\tracing\*
Computer Configuration\Policies\Windows Settings\Security Settings\Application Control Policies\AppLocker\Windows Installer Rules	
[Publisher] CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US	Allow Everyone
[Publisher] CN=Microsoft Windows, O=Microsoft Corporation, L=Redmond, S=Washington, C=US	Allow Everyone

Note, for those organisations using the latest version of Microsoft Windows 10, the following paths no longer need to be included as exceptions:

- %WinDir%\servicing\Packages\*
- %WinDir%\servicing\Sessions\*.

Preventing malicious actors from bypassing application control

To prevent malicious actors from 'living off the land' through the use of Microsoft executables that are typically approved by default under most application control implementations, Microsoft's recommended application blocklist should be implemented.

In addition, to prevent the exploitation of vulnerable drivers by malicious actors, Microsoft's vulnerable driver blocklist should be implemented. This can be achieved either through the use of WDAC, or by enabling memory integrity functionality via the Windows Security app.

Implementing application control within Linux environments

Implementing application control within Linux environments can be achieved using the File Access Policy daemon (fapolicyd). The fapolicyd framework allows Linux system administrators to control which applications are allowed (or denied) execution based on either path, hash, MIME type or if they are trusted (i.e. properly installed by the system package manager and registered in the RPM database). Red Hat's *Security Hardening* publication provides advice on how to configure and manage the use of the fapolicyd framework within Red Hat Enterprise Linux 9.

Further information

The *Information Security Manual* is a cyber security framework that organisations can apply to protect their systems and data from cyber threats. The advice in the *Strategies to Mitigate Cyber Security Incidents*, along with its Essential Eight, complements this framework.

Contact details

If you have any questions regarding this guidance you can write to us or call us on 1300 CYBER1 (1300 292 371).

Disclaimer

The material in this guide is of a general nature and should not be regarded as legal advice or relied on for assistance in any particular circumstance or emergency situation. In any important matter, you should seek appropriate independent professional advice in relation to your own circumstances.

The Commonwealth accepts no responsibility or liability for any damage, loss or expense incurred as a result of the reliance on information contained in this guide.

Copyright

Use of the Coat of Arms

For more information, or to report a cyber security incident, contact us:

cyber.gov.au | 1300 CYBER1 (1300 292 371)

Australian Government

Australian Signals Directorate