Windows Applocker Policy – A Beginner’s Guide

hackingarticles.in/windows-applocker-policy-a-beginners-guide

Raj

January 13, 2019

This Post is based on “Microsoft Windows – Applocker Policy” and this topic for System Administrator, defines the AppLocker rules for your application control policies and how to work with them.

Table of Content

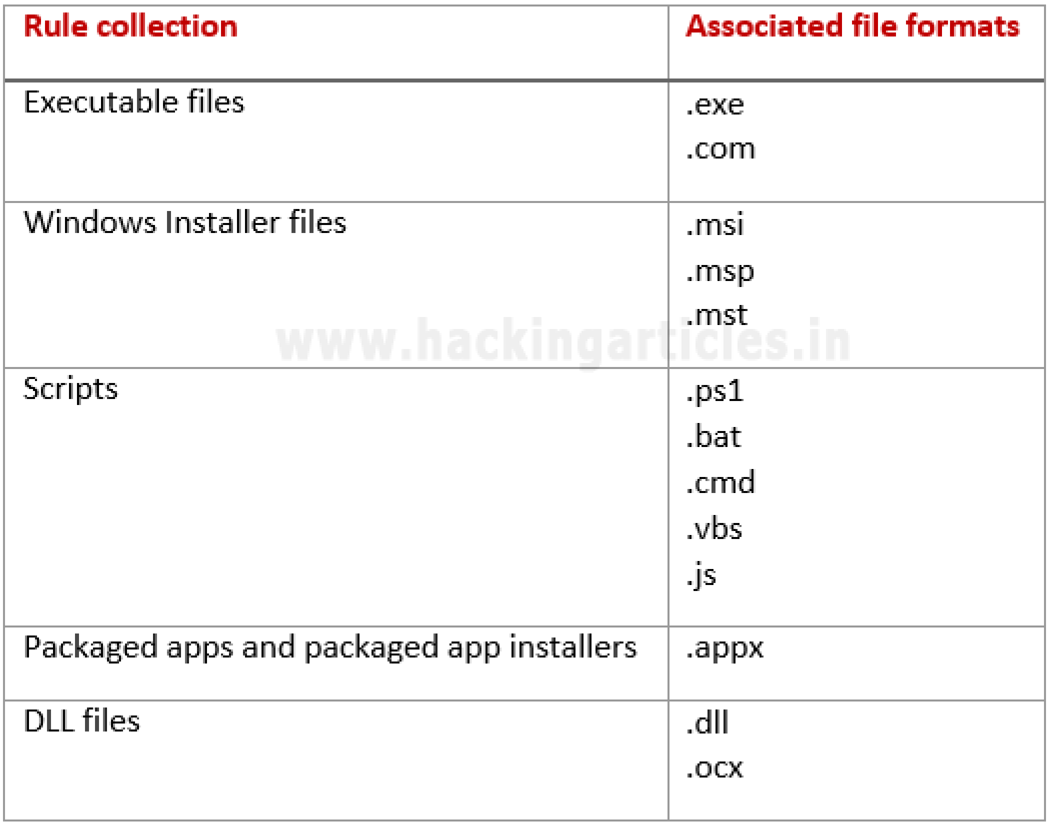
Introduction to Applocker

What is applocker Policy?

Windows Applocker is a function that was introduced in home windows 7 and windows server 2008 r2 as a method to restrict the usage of unwanted Programs. Windows AppLocker lets administrators control which executable files are denied or allowed to be run. With this policy, administrators are able to generate rules based on file names, publishers or file locations on unique identities of files and specify which users or groups can execute those applications.

What can your rules be based upon?

The AppLocker console is ordered into rule collections, which include executable files, scripts, Windows Installer files, packaged apps, and packaged app installers, and DLL files. These collections allow you to easily distinguish rules for different types of applications. The following table lists the file formats included in each rule collection.



Rule collection	Associated file formats
Executable files	.exe .com
Windows Installer files	.msi .msp .mst
Scripts	.ps1 .bat .cmd .vbs .js
Packaged apps and packaged app installers	.appx
DLL files	.dll .ocx

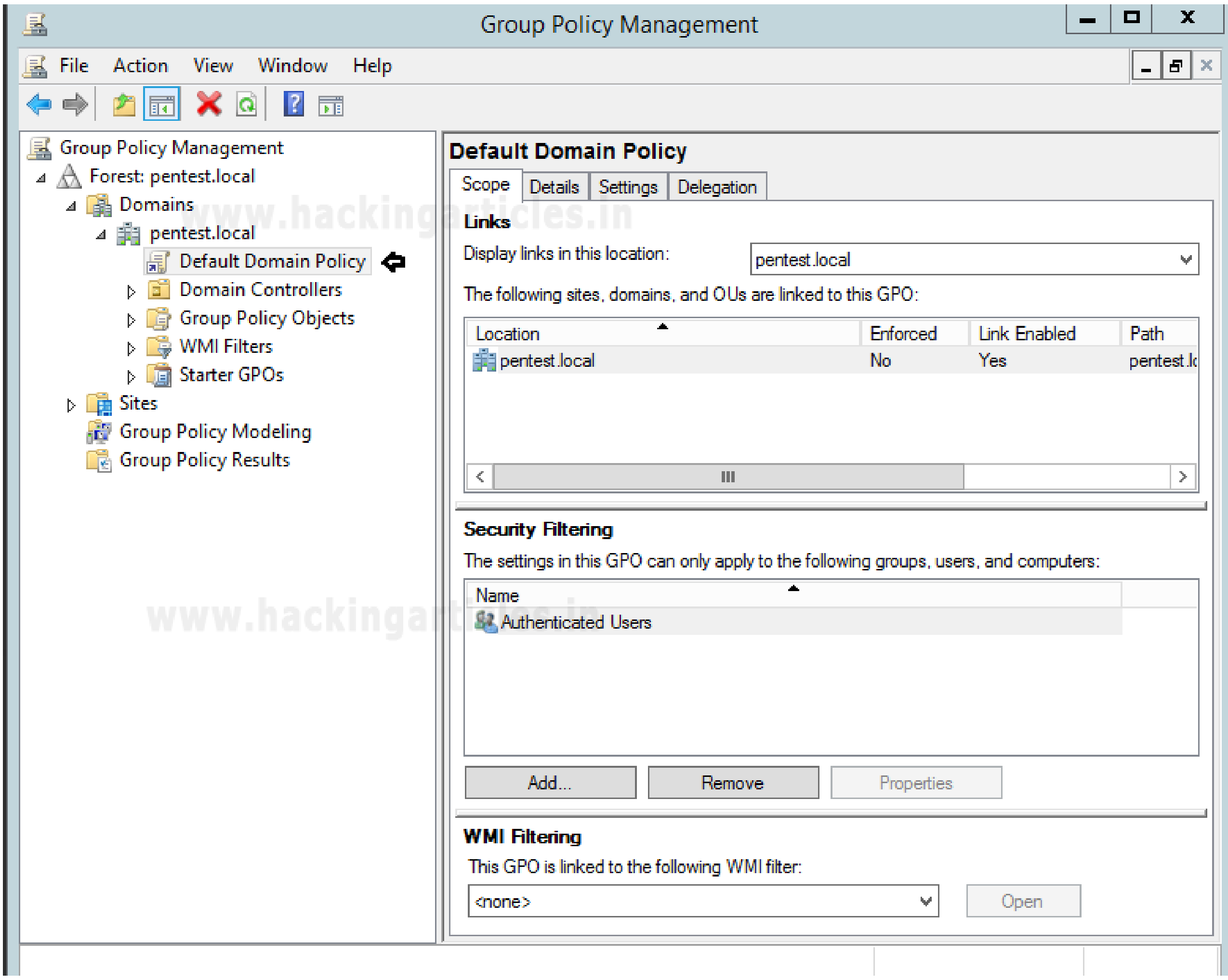
Who Should Use AppLocker?

AppLocker is worthy for organizations that have to accomplish any of the following jobs:

- Check which applications are allowed to run inside the company network.
- Check which users are allowed to use the licensed program.
- Offer an audit log of what kind of applications clients were running.
- Prevent trendy users from installing software per user.

Configure the Applocker to Allow/Deny Execution of an App

In the Group Policy Object Editor at ***Computer Configuration > Windows Settings > Security Settings > Application Control Policies > AppLocker***, the Windows AppLocker settings exist.



Default Domain Policy

Links

Display links in this location: pentest.local

The following sites, domains, and OUs are linked to this GPO:

Location	Enforced	Link Enabled	Path
pentest.local	No	Yes	pentest.lo

Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
Authenticated Users

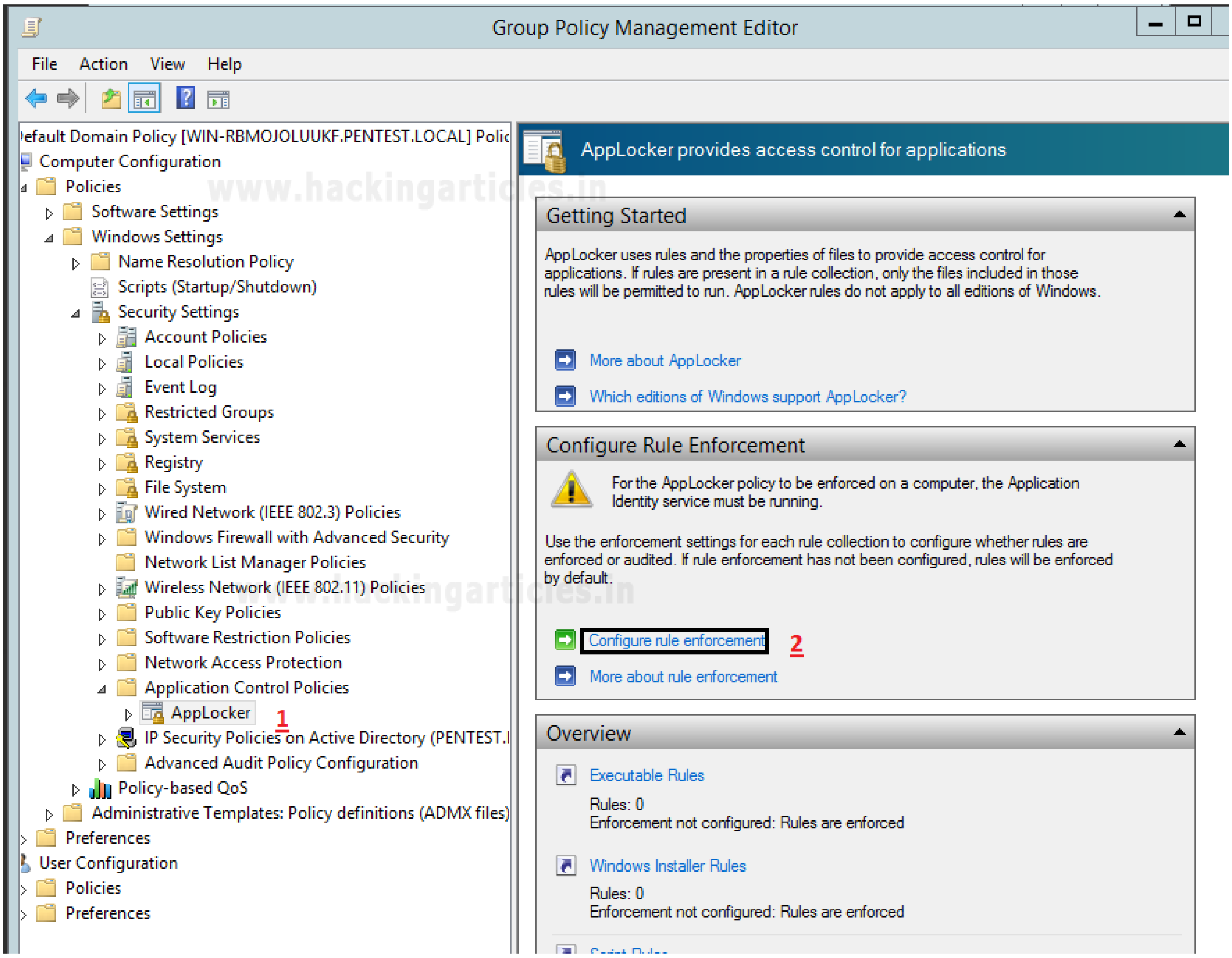
WMI Filtering

This GPO is linked to the following WMI filter:

<none>

Configure Enforcement Rule

Use the enforcement setting for each collection to configure to **Enforce rules**, rules are enforced for the rule collection and all events are audited.



Group Policy Management Editor
File
Action
View
Help
'efault Domain Policy [WIN-RBMOJOLUUKF.PENTEST.LOCAL] Polic
Computer Configuration
Policies
Software Settings
Windows Settings
Name Resolution Policy
Scripts (Startup/Shutdown)
Security Settings
Account Policies
Local Policies
Event Log
Restricted Groups
System Services
Registry
File System
Wired Network (IEEE 802.3) Policies
Windows Firewall with Advanced Security
Network List Manager Policies
Wireless Network (IEEE 802.11) Policies
Public Key Policies
Software Restriction Policies
Network Access Protection
Application Control Policies
AppLocker
1
IP Security Policies on Active Directory (PENTEST.L
Advanced Audit Policy Configuration
Policy-based QoS
Administrative Templates: Policy definitions (ADMX files)
Preferences
User Configuration
Policies
Preferences
AppLocker provides access control for applications
Getting Started
More about AppLocker
Which editions of Windows support AppLocker?
Configure Rule Enforcement
Configure rule enforcement
2
More about rule enforcement
Overview
Executable Rules
Windows Installer Rules
Script Rules

AppLocker uses rules and the properties of files to provide access control for applications. If rules are present in a rule collection, only the files included in those rules will be permitted to run. AppLocker rules do not apply to all editions of Windows.

For the AppLocker policy to be enforced on a computer, the Application Identity service must be running.

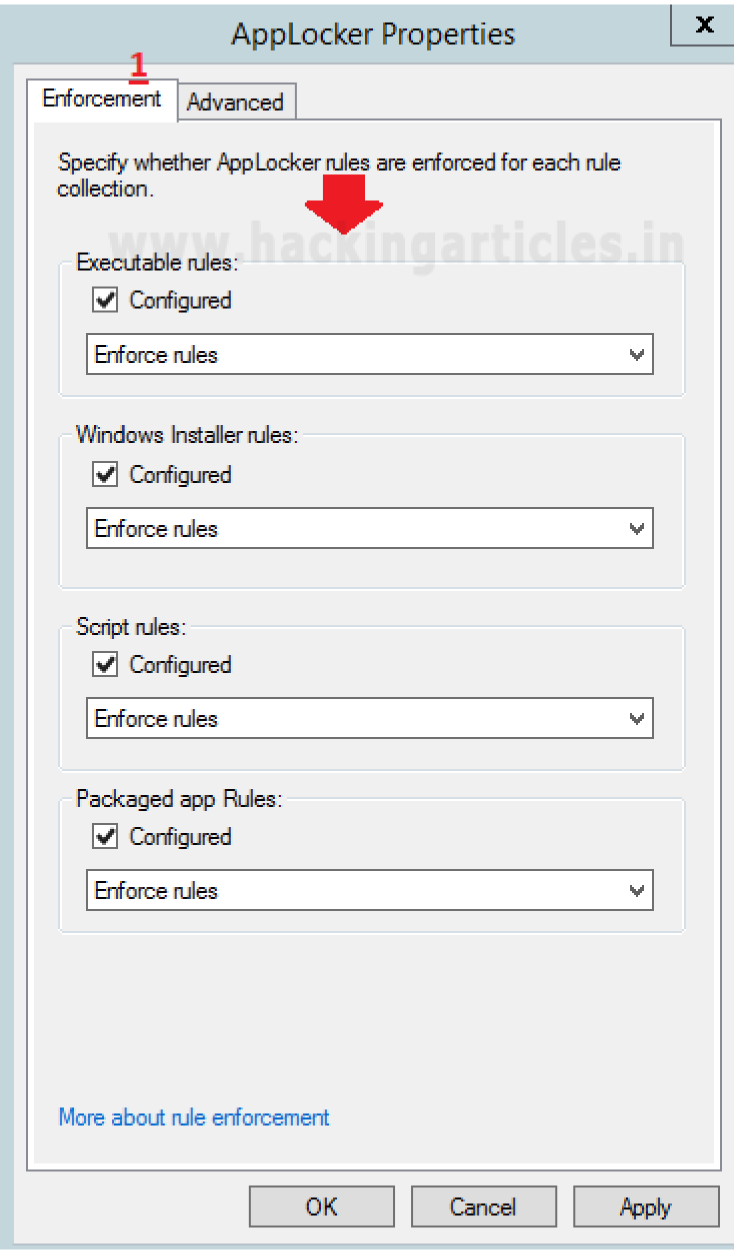
Use the enforcement settings for each rule collection to configure whether rules are enforced or audited. If rule enforcement has not been configured, rules will be enforced by default.

Rules: 0
Enforcement not configured: Rules are enforced

Rules: 0
Enforcement not configured: Rules are enforced

1. Select the **Configured** check box for the rule collection that you are editing, and then verify that **Enforce rules** are selected.
2. Click OK.

Open the **Advanced** tab and enable the DLL rule collection.



AppLocker Properties

Specify whether AppLocker rules are enforced for each rule collection.

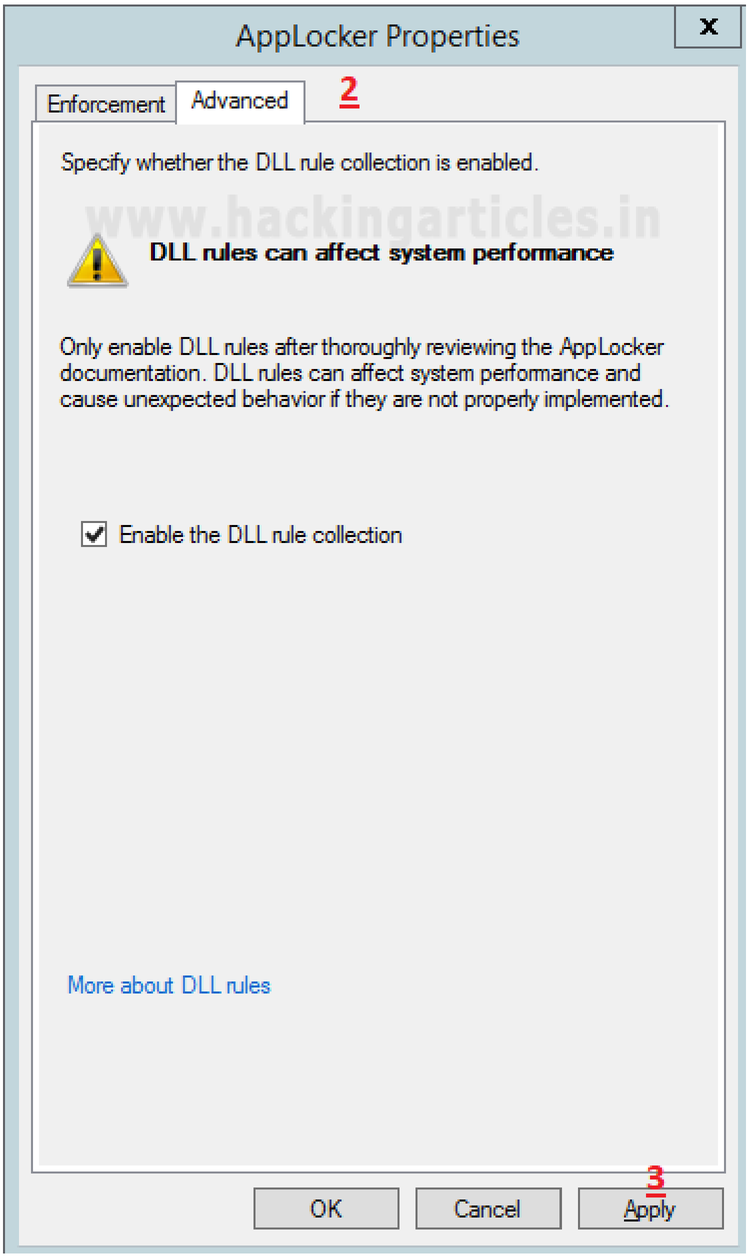
Executable rules:
☑ Configured
Enforce rules

Windows Installer rules:
☑ Configured
Enforce rules

Script rules:
☑ Configured
Enforce rules

Packaged app Rules:
☑ Configured
Enforce rules

More about rule enforcement



Specify whether the DLL rule collection is enabled.

DLL rules can affect system performance

Only enable DLL rules after thoroughly reviewing the AppLocker documentation. DLL rules can affect system performance and cause unexpected behavior if they are not properly implemented.

☑ Enable the DLL rule collection

More about DLL rules

Create Default Rules

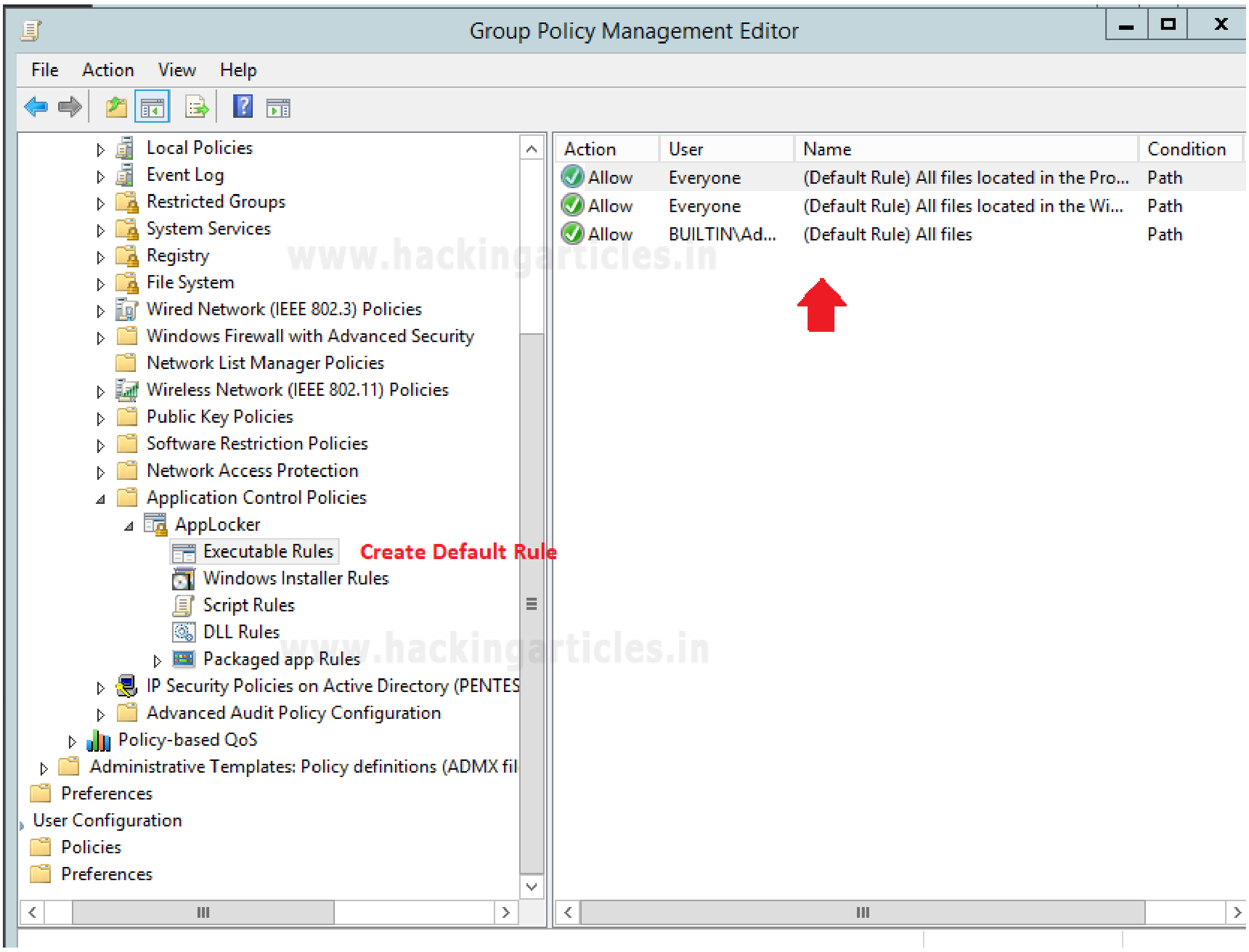
AppLocker includes default rules for each rule collection. These rules are intended to help ensure that the files that are required for Windows to operate properly are allowed in an AppLocker rule collection.

- Open the AppLocker console.
- Right-click the appropriate rule type for which you want to generate default rules automatically. You can automatically create executable rules, Windows Installer rules, script rules, and packaged application rules.
- Click Create Default Rules.

Executable Default Rule Types Include:

- Allow members of the local **Administrators** group to run all apps.
- Allow members of the **Everyone** group to run apps that are located in the Windows folder.

- Allow members of the **Everyone** group to run apps that are located in the Program Files folder.


Group Policy Management Editor
File Action View Help
Local Policies
Event Log
Restricted Groups
System Services
Registry
File System
Wired Network (IEEE 802.3) Policies
Windows Firewall with Advanced Security
Network List Manager Policies
Wireless Network (IEEE 802.11) Policies
Public Key Policies
Software Restriction Policies
Network Access Protection
Application Control Policies
AppLocker
Executable Rules
Windows Installer Rules
Script Rules
DLL Rules
Packaged app Rules
IP Security Policies on Active Directory (PENTES
Advanced Audit Policy Configuration
Policy-based QoS
Administrative Templates: Policy definitions (ADMX fil
Preferences
User Configuration
Policies
Preferences
Create Default Rule
www.hackingarticles.in

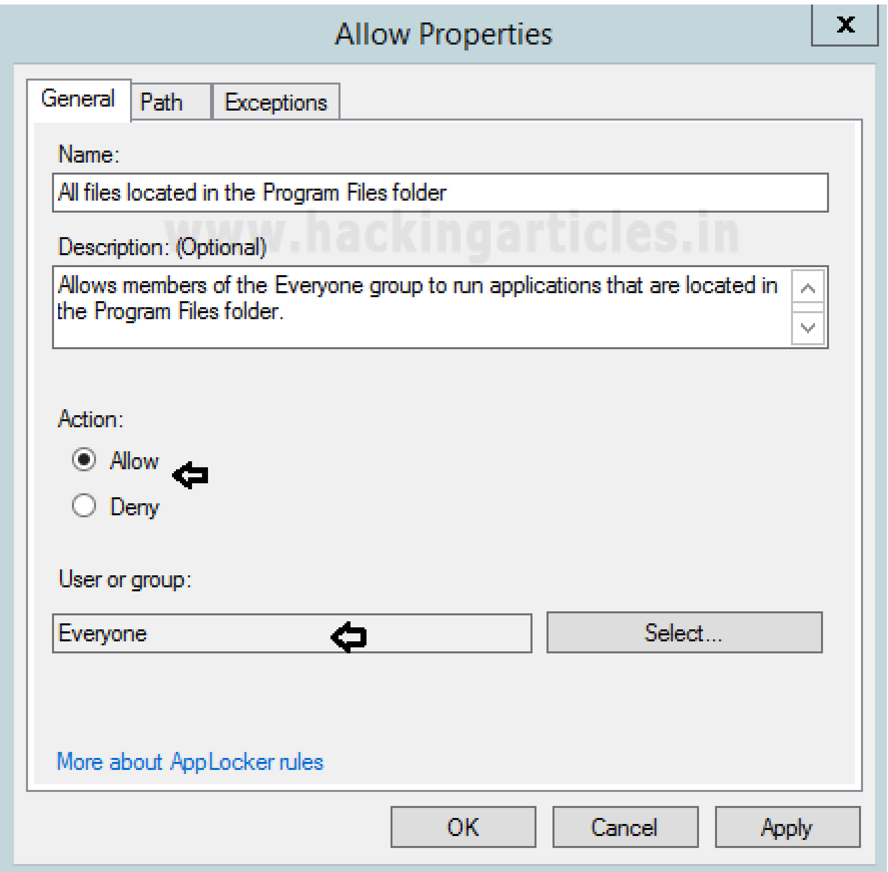
Action	User	Name	Condition
Allow	Everyone	(Default Rule) All files located in the Pro...	Path
Allow	Everyone	(Default Rule) All files located in the Wi...	Path
Allow	BUILTIN\Ad...	(Default Rule) All files	Path

Modify Executable Default Rules to Allow an App

A rule can be configured to use allow or deny actions:

- **ALLOW:** You can specify which files are allowed to run in your environment, and for which users or groups of users.
- **DENY:** You can specify which files are *not* allowed to run in your environment, and for which users or groups of users.

Once you have configured default rules as done above, then you can modify it as per your requirement. For example, if you want to modify rule: “*Allow members of the **Everyone** group to run apps that are located in the Program Files folder*” for specific user or group to allow a specific program file execution, then go its property by making right click on that rule and follow below steps.



Allow Properties

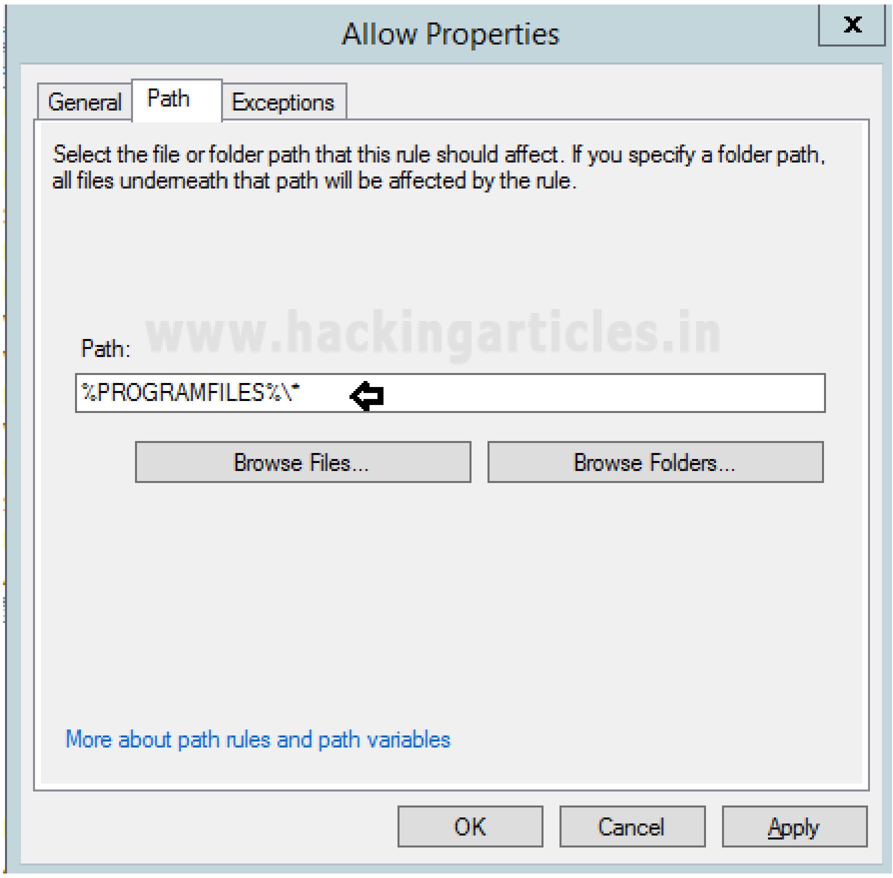
Name: All files located in the Program Files folder

Description: (Optional) Allows members of the Everyone group to run applications that are located in the Program Files folder.

Action:
● Allow
○ Deny

User or group: Everyone

Select the file or folder path that this rule should affect. The asterisk (*) can be used as a wildcard in the rules of the path. For example, %ProgramFiles% * indicates that all files and subfolders within that path.



Allow Properties

Select the file or folder path that this rule should affect. If you specify a folder path, all files underneath that path will be affected by the rule.

Path: %PROGRAMFILES%*

Rule conditions

Conditions of rules are criteria for AppLocker to identify the applications to which the rule applies. The three main rules are the publisher, path, and hash of the file.

Publisher

Identifies a digital signature- based application. The digital signature encloses information about the company (the publisher) who created the application.

Wildcard characters can be used as values in the publisher rule fields according to the following specifications:

Advantage:

- Frequent updating is not required.
- You can apply different values within a certificate.
- You can use a single rule to allow a complete product suite.
- Within the publisher rule, you can use the asterisk (*) wildcard character to specify that any value should match.

Disadvantage:

> While a single rule can be used to allow a complete product suite, all files in the suite must be uniformly signed.

Path

Identify an app in the computer file system or on the network by its location. For well-known paths such as Program Files and Windows, AppLocker uses custom path variables.

Advantages:

- Many folders or a single file can be easily controlled.
- The asterisk (*) can be used as a wildcard in the rules of the path. For example, %ProgramFiles%\Microsoft Office* indicates that all files and subfolders within the Microsoft Office folder will be affected by the rule.

Disadvantage:

> It could be at risk if a rule that is organized to use a folder path holds subfolders that are writable by the local user.

File Hash

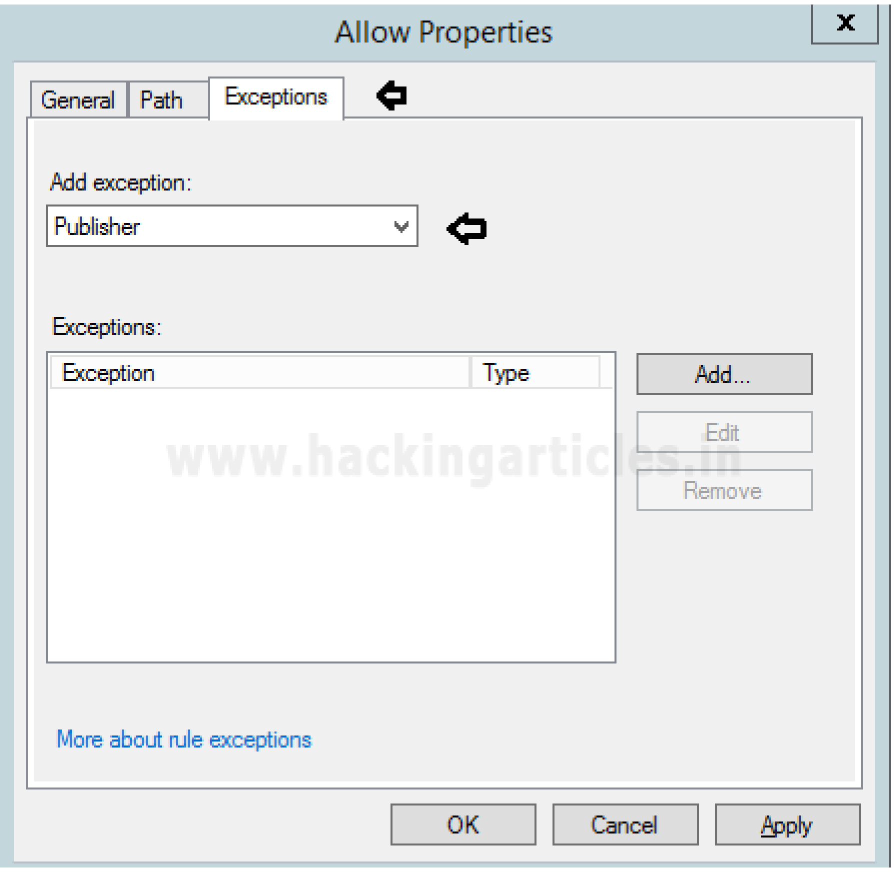
Represents the calculated cryptographic hash system of the identified file. For non-digitally signed files, file hash rules are safer than path rules.

Advantage:

Since each file has a unique hash, a file hash condition only applies to one file.

Disadvantage:

Whenever the file is updated (such as security updates or upgrades), the hash of the file changes. Consequently, you have to manually update the rules for file hash.

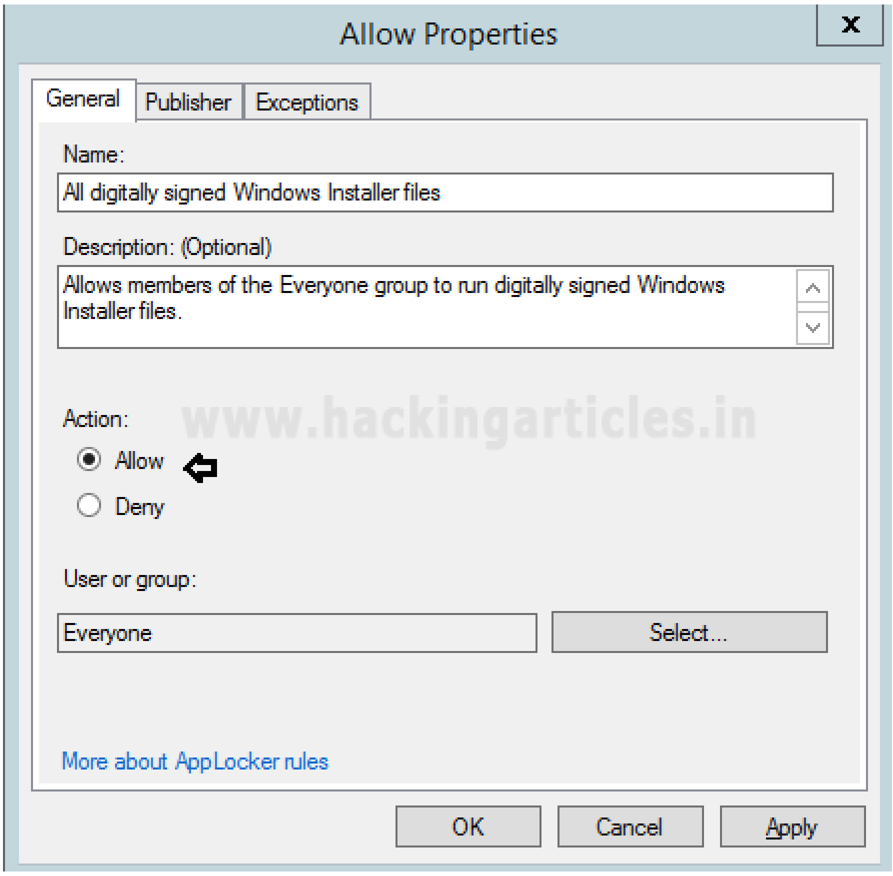


Modify Windows Installer Default Rules to Allow an App

Windows Installer Default Rule Types Include:

- Allow members of the local **Administrators** group to run all Windows Installer files.
- Allow members of the **Everyone** group to run all digitally signed Windows Installer files.
- Allow members of the **Everyone** group to run all Windows Installer files that are located in the Windows\Installer folder.

Similarly, if you want to modify Windows Install default rules, then repeat above steps.



Name:
All digitally signed Windows Installer files

Description: (Optional)
Allows members of the Everyone group to run digitally signed Windows Installer files.

Action:
● Allow
○ Deny

User or group:
Everyone

Wildcard characters can be used as values in the publisher rule fields according to the following specifications:

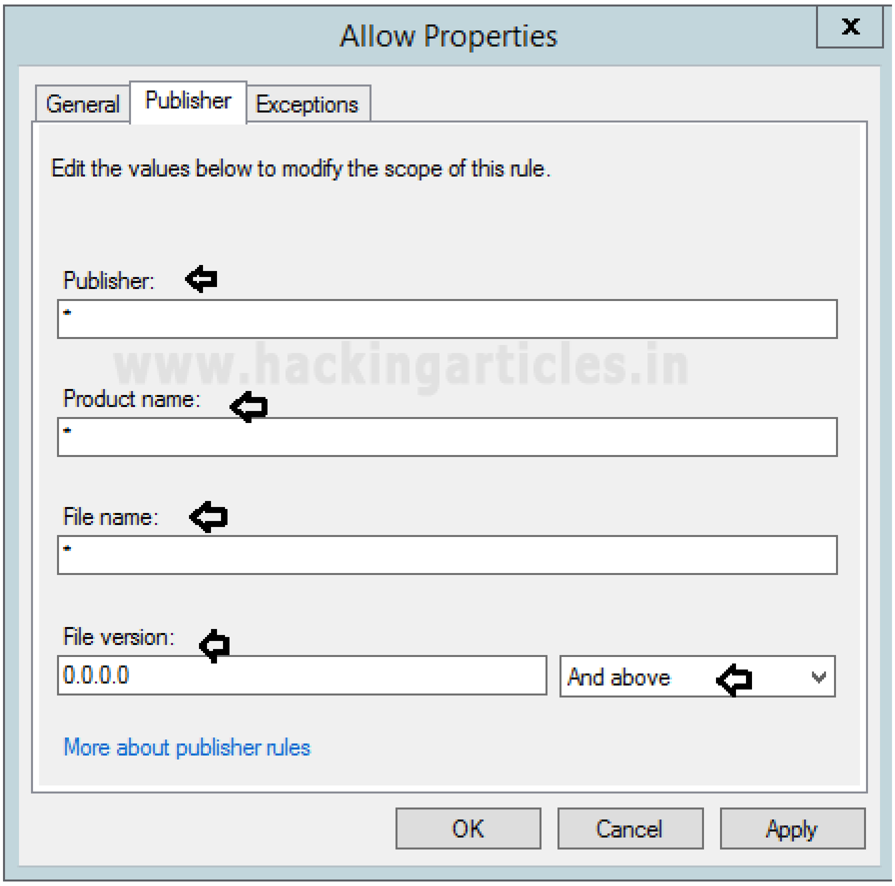
Publisher: The asterisk (*) character used by itself represents any publisher.

Product name: The asterisk (*) character used by itself represents any product name.

File name: Either the asterisk (*) or question mark (?) characters used by themselves represent any and all file names.

File version: The asterisk (*) character used by itself represents any file version. If you want to limit the file version to a specific version or as a starting point, you can state the file version and then use the following options to apply limits:

- **Exactly**. The rule applies only to this version of the app
- **And above**. The rule applies to this version and all later versions.
- **And Below**. The rule applies to this version and all earlier versions.



Allow Properties

General | Publisher | Exceptions

Edit the values below to modify the scope of this rule.

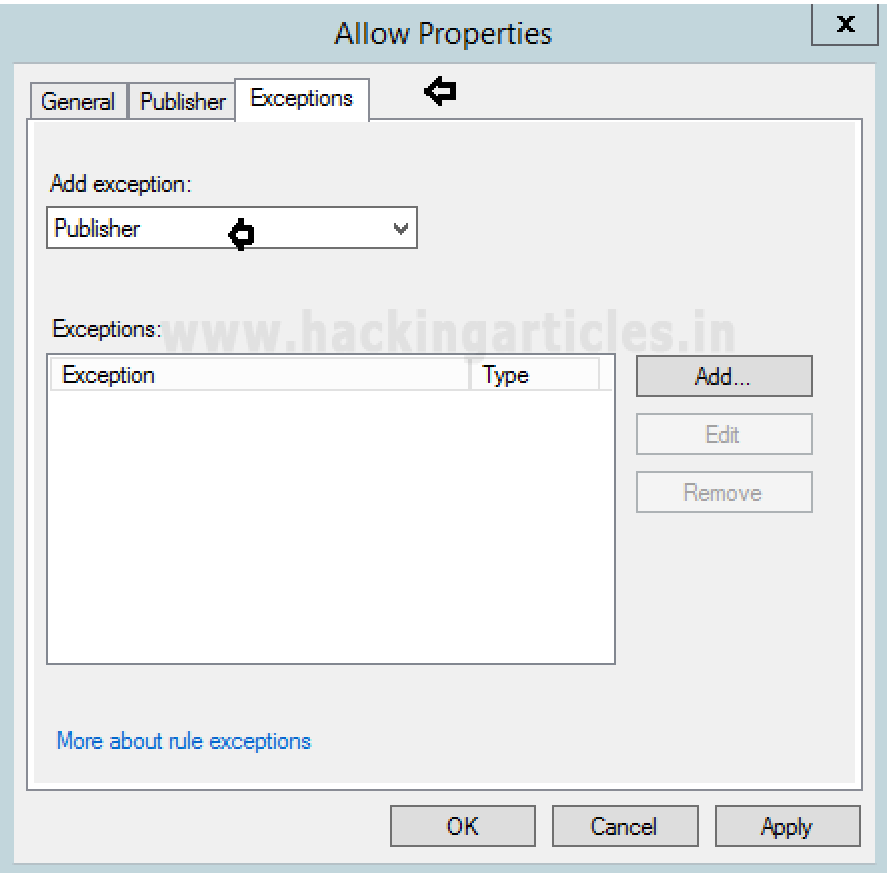
Publisher: *

Product name: *

File name: *

File version: 0.0.0.0 And above

Open Exceptions and then again select Publisher.



Allow Properties

General | Publisher | Exceptions

Add exception: Publisher

Exceptions:

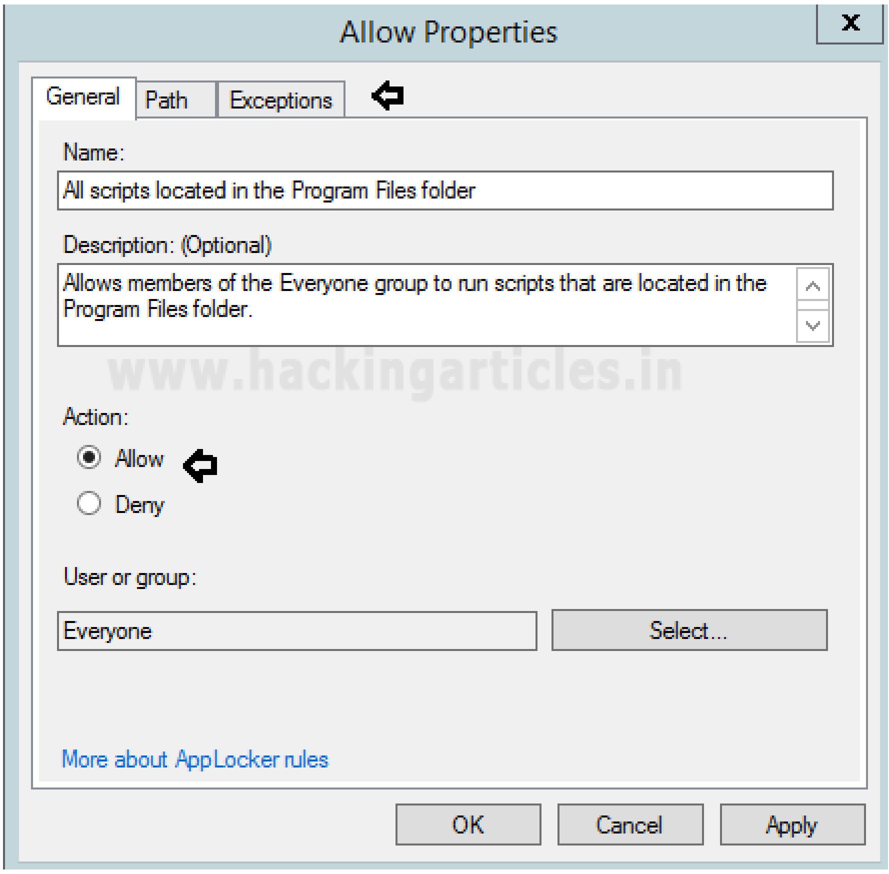
Exception	Type

Modify Script Default Rules to Allow an App

Script Default Rule Types Include:

- Allow members of the local **Administrators** group to run all scripts.
- Allow members of the **Everyone** group to run scripts that are located in the Program Files folder.
- Allow members of the **Everyone** group to run scripts that are located in the Windows folder.

Similarly, if you want to modify Script default rules, then repeat above steps.



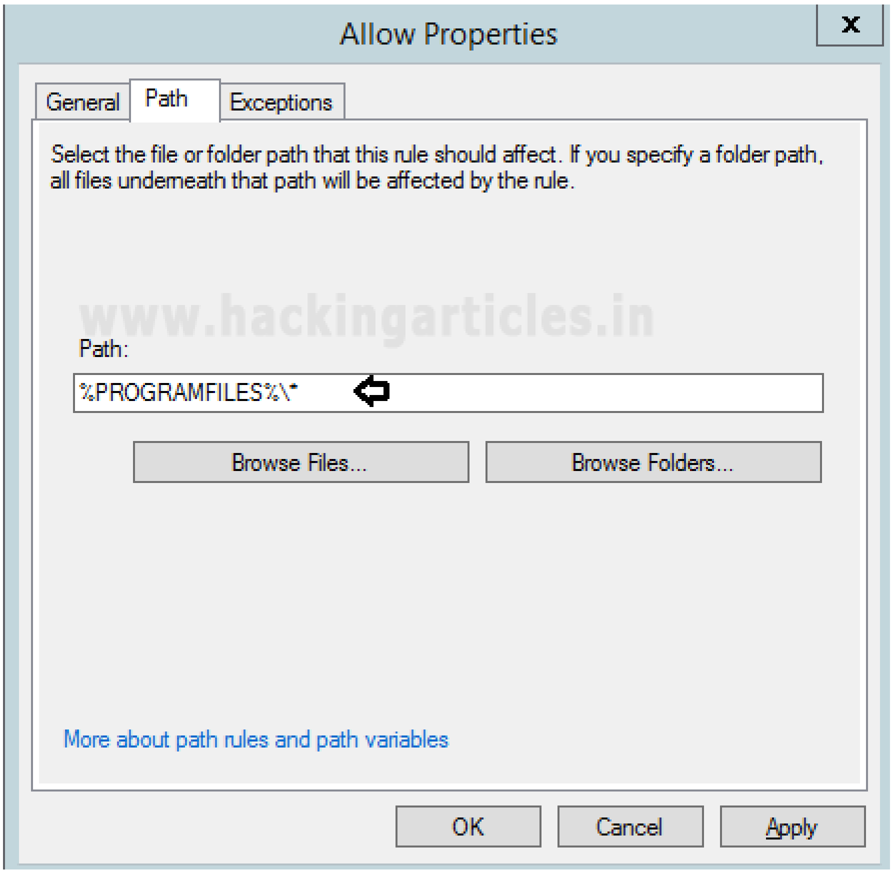
Name:
All scripts located in the Program Files folder

Description: (Optional)
Allows members of the Everyone group to run scripts that are located in the Program Files folder.

Action:
● Allow
○ Deny

User or group:
Everyone

Select the file or folder path that this rule should affect.

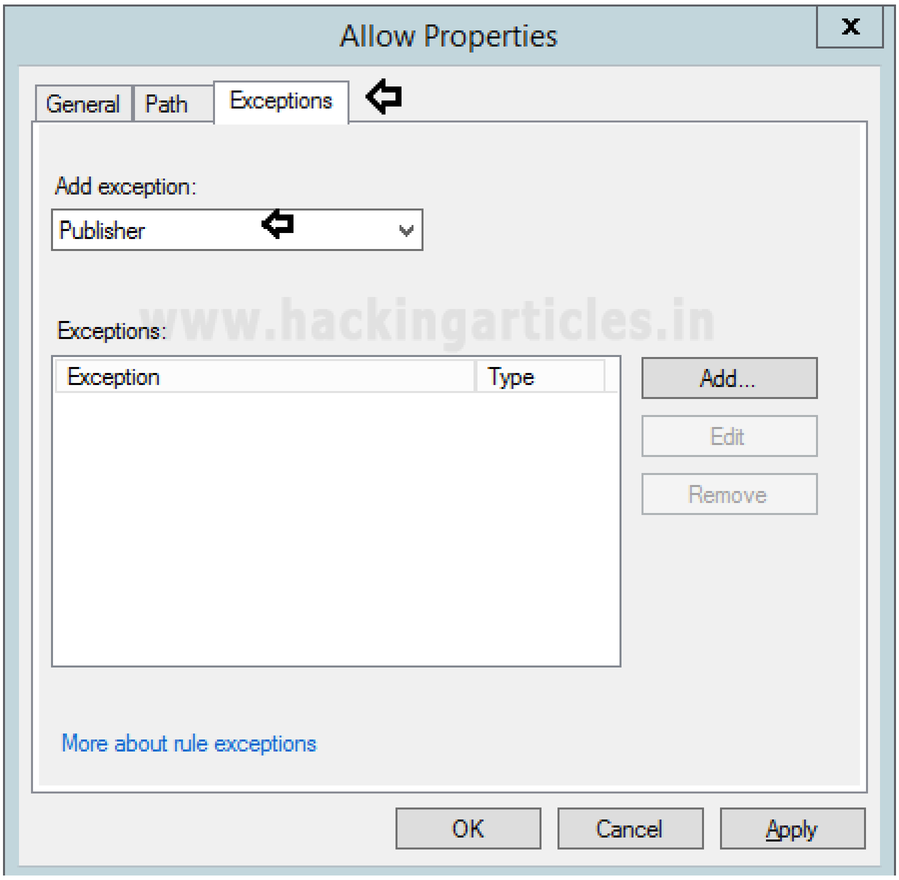


Select the file or folder path that this rule should affect. If you specify a folder path, all files underneath that path will be affected by the rule.

Path:
%PROGRAMFILES%*

Open Exceptions and then again select Publisher.

In this way, you can implement Default rules and modify them for Executable file, Script rules or Windows Installer files according to your situation.



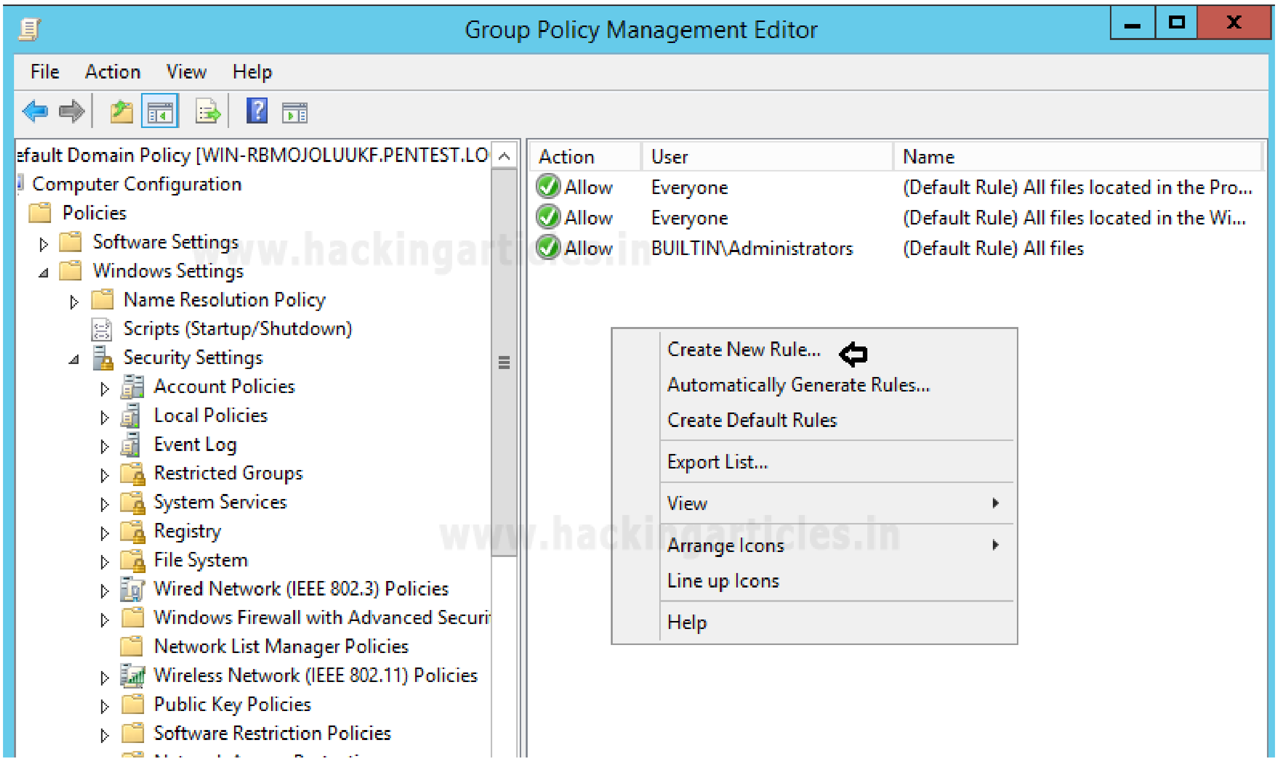
Add exception:
Publisher

Exceptions:

Exception	Type

Creating New Rules to Block an APP

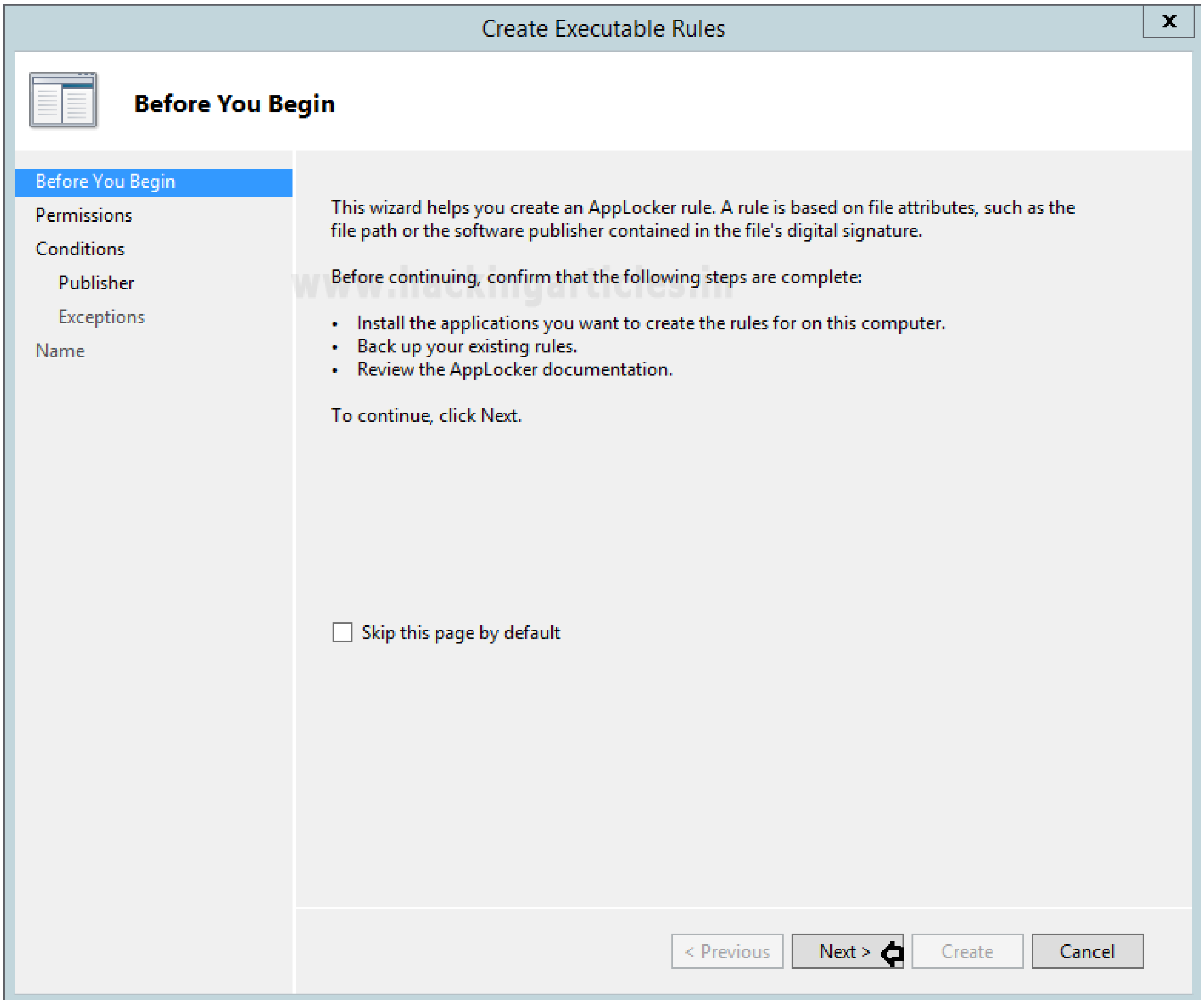
If you want to make your own rule in order to allow or deny an action for any application, you can choose the options " Create New Rule" below. Let's say, I want to create a new Executable file rule to restrict command prompt execution for everyone.



Action	User	Name
Allow	Everyone	(Default Rule) All files located in the Pro...
Allow	Everyone	(Default Rule) All files located in the Wi...
Allow	BUILTIN\Administrators	(Default Rule) All files

Then, you will get a wizard that helps you to create an Applocker rule, which will truly be based on the file attribute such as the file path and digital signature.

***NOTE:** Install the applications you want to create the rules for on this computer.*



Create Executable Rules

Before You Begin

This wizard helps you create an AppLocker rule. A rule is based on file attributes, such as the file path or the software publisher contained in the file's digital signature.

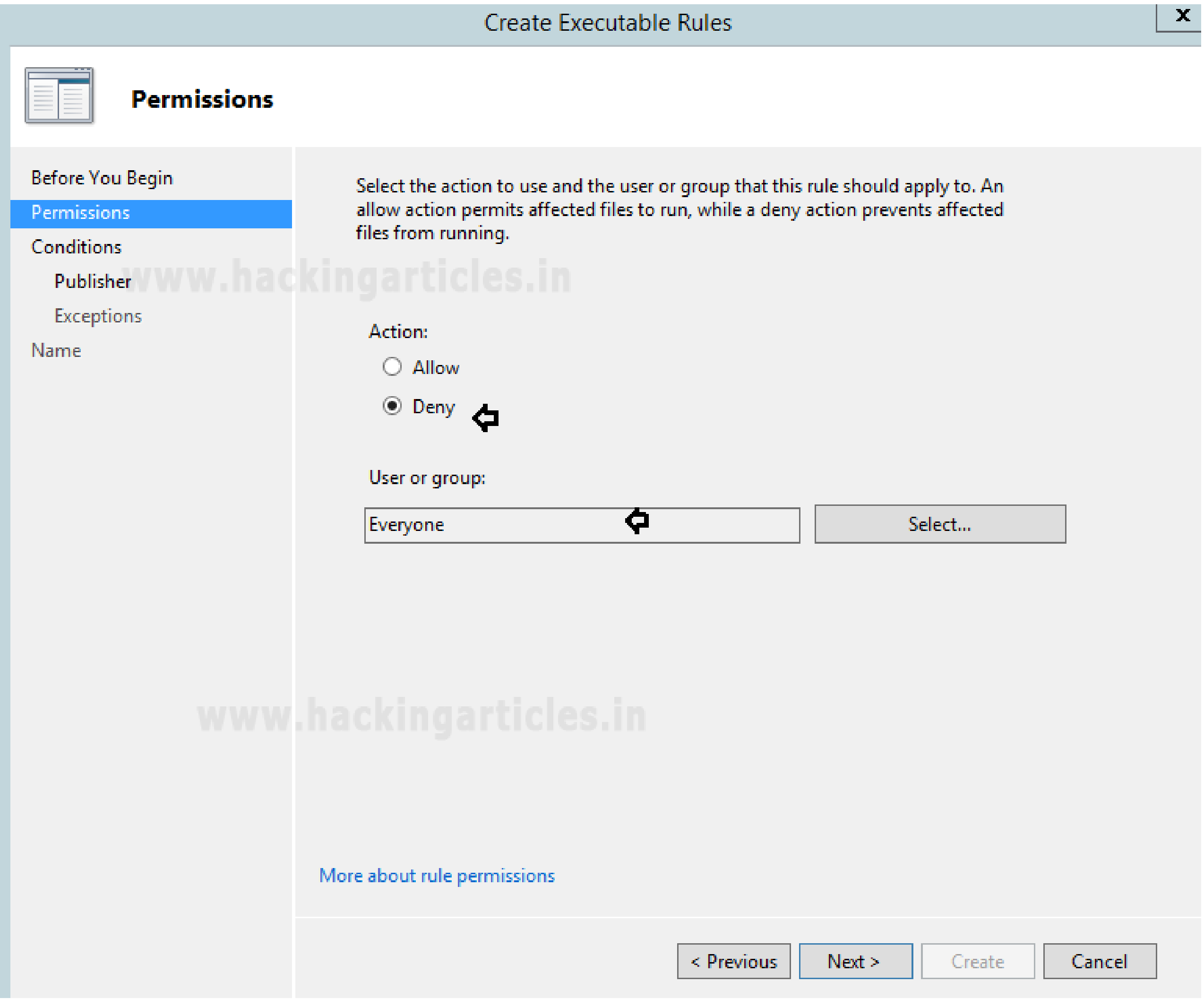
Before continuing, confirm that the following steps are complete:

- Install the applications you want to create the rules for on this computer.
- Back up your existing rules.
- Review the AppLocker documentation.

To continue, click Next.

□ Skip this page by default

Now the action to use and the user or group that this rule should apply to. A **deny** action prevent affected file from running.



Select the action to use and the user or group that this rule should apply to. An allow action permits affected files to run, while a deny action prevents affected files from running.

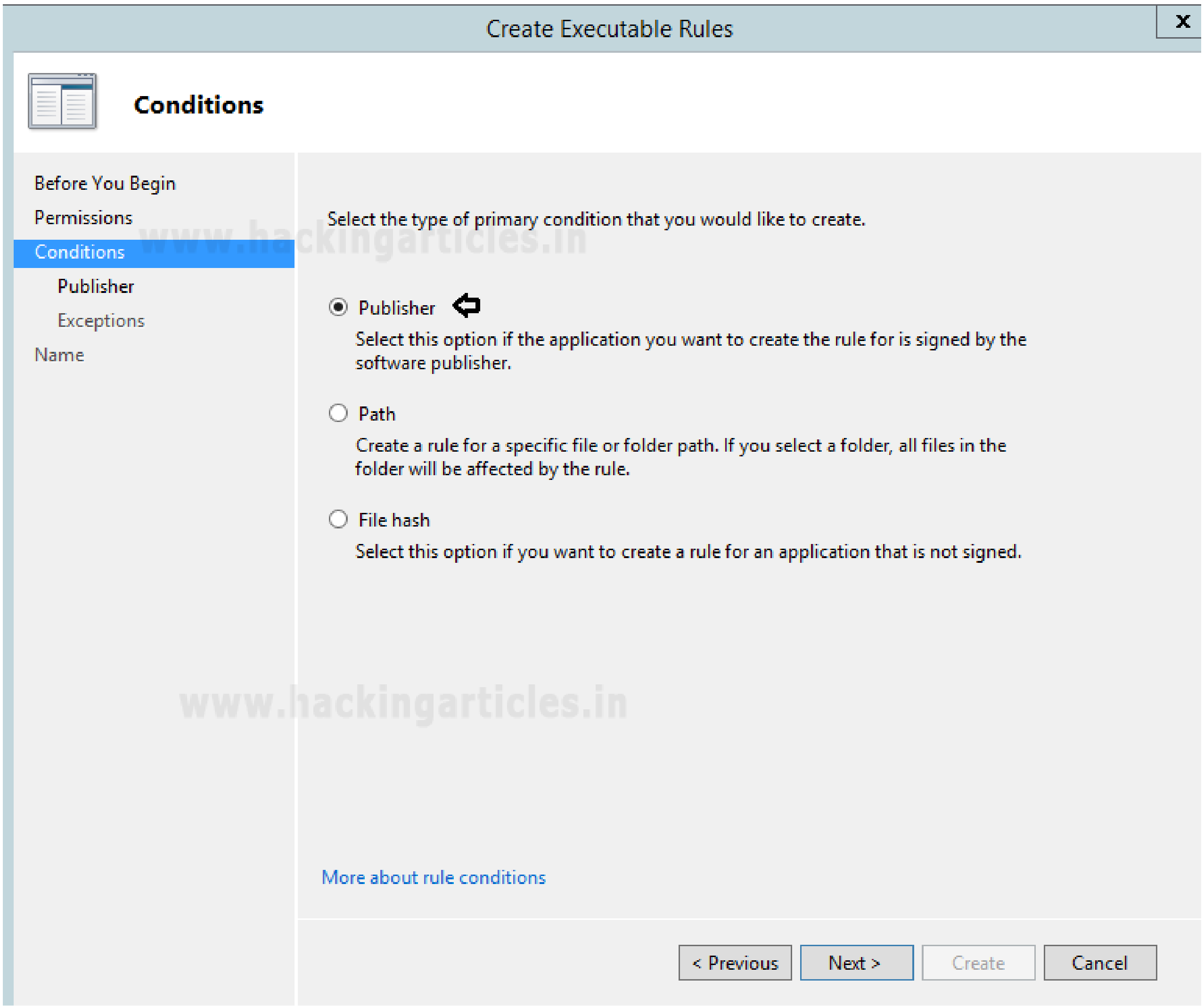
Action:

○ Allow

● Deny

User or group: Everyone

Select the type of primary condition that you would like to create. Here we have chosen “**Publisher**” options.



Select the type of primary condition that you would like to create.

● Publisher

Select this option if the application you want to create the rule for is signed by the software publisher.

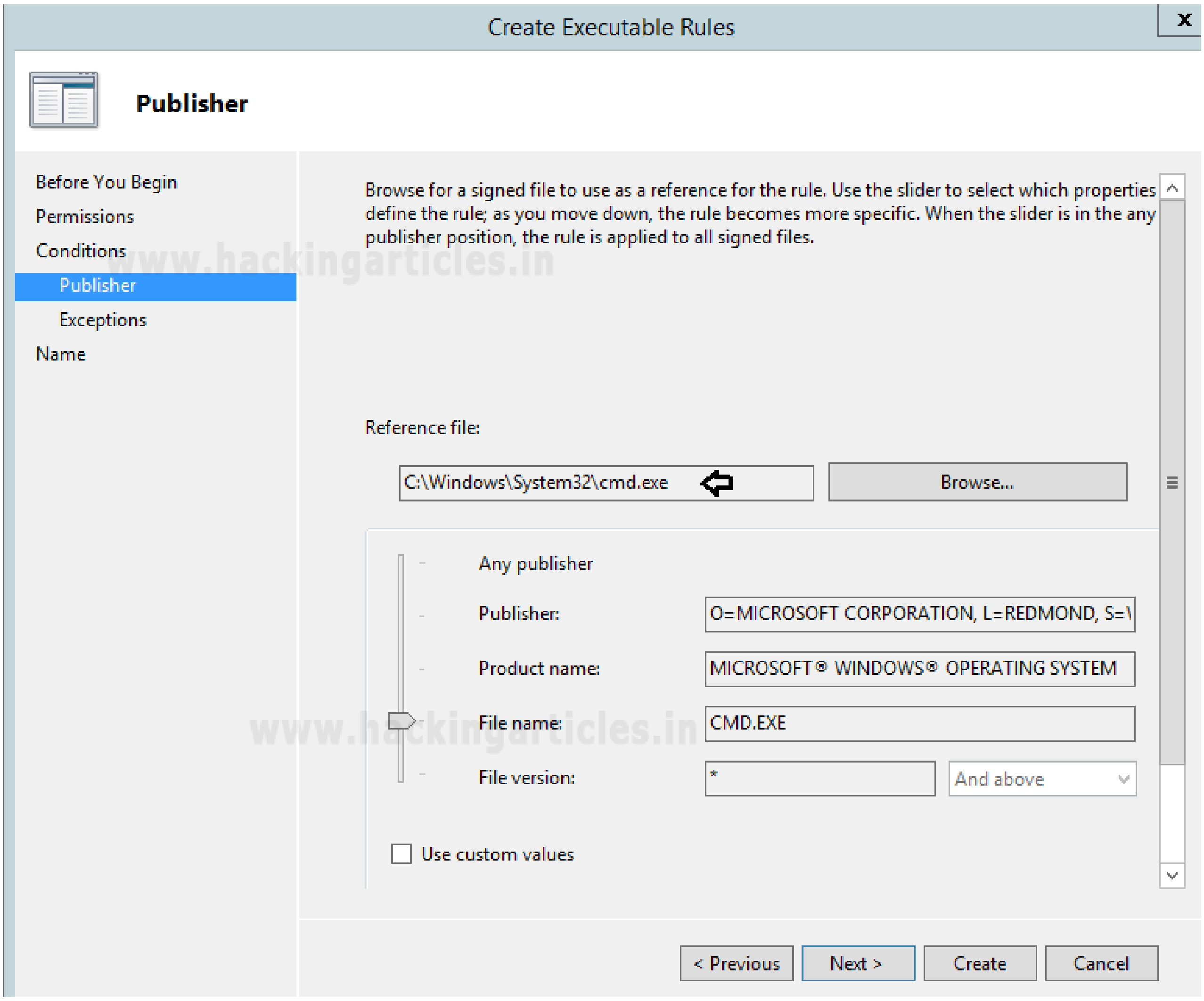
○ Path

Create a rule for a specific file or folder path. If you select a folder, all files in the folder will be affected by the rule.

○ File hash

Select this option if you want to create a rule for an application that is not signed.

Browse for a signed file to use as a reference for the rule. Here we have browsed the **cmd.exe** and then click on next.



Browse for a signed file to use as a reference for the rule. Use the slider to select which properties define the rule; as you move down, the rule becomes more specific. When the slider is in the any publisher position, the rule is applied to all signed files.

Reference file: C:\Windows\System32\cmd.exe

Any publisher

Publisher: O=MICROSOFT CORPORATION, L=REDMOND, S=\

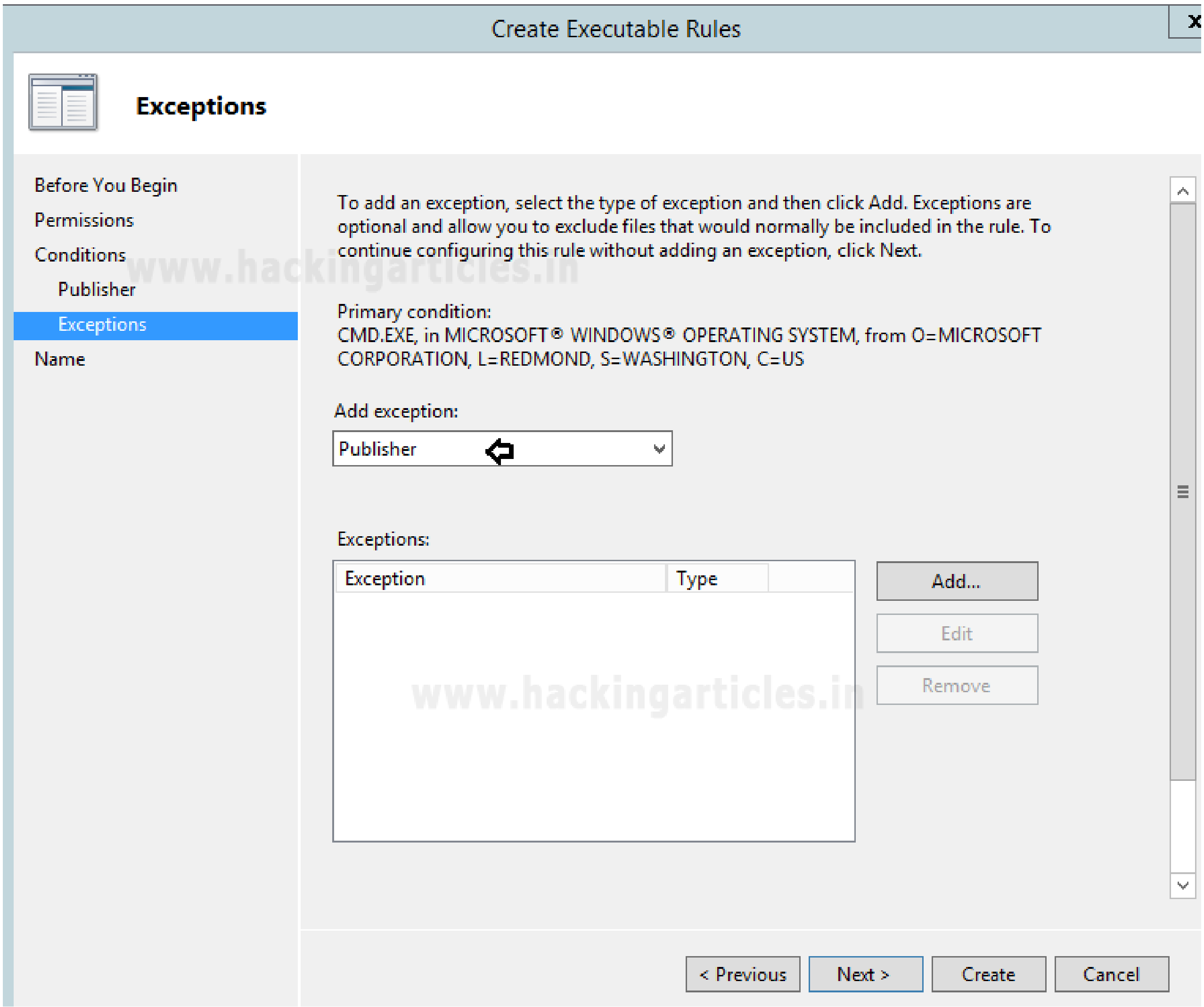
Product name: MICROSOFT® WINDOWS® OPERATING SYSTEM

File name: CMD.EXE

File version: * And above

□ Use custom values

Choose the Publisher as an exception and then click Next.



Create Executable Rules

Exceptions

To add an exception, select the type of exception and then click Add. Exceptions are optional and allow you to exclude files that would normally be included in the rule. To continue configuring this rule without adding an exception, click Next.

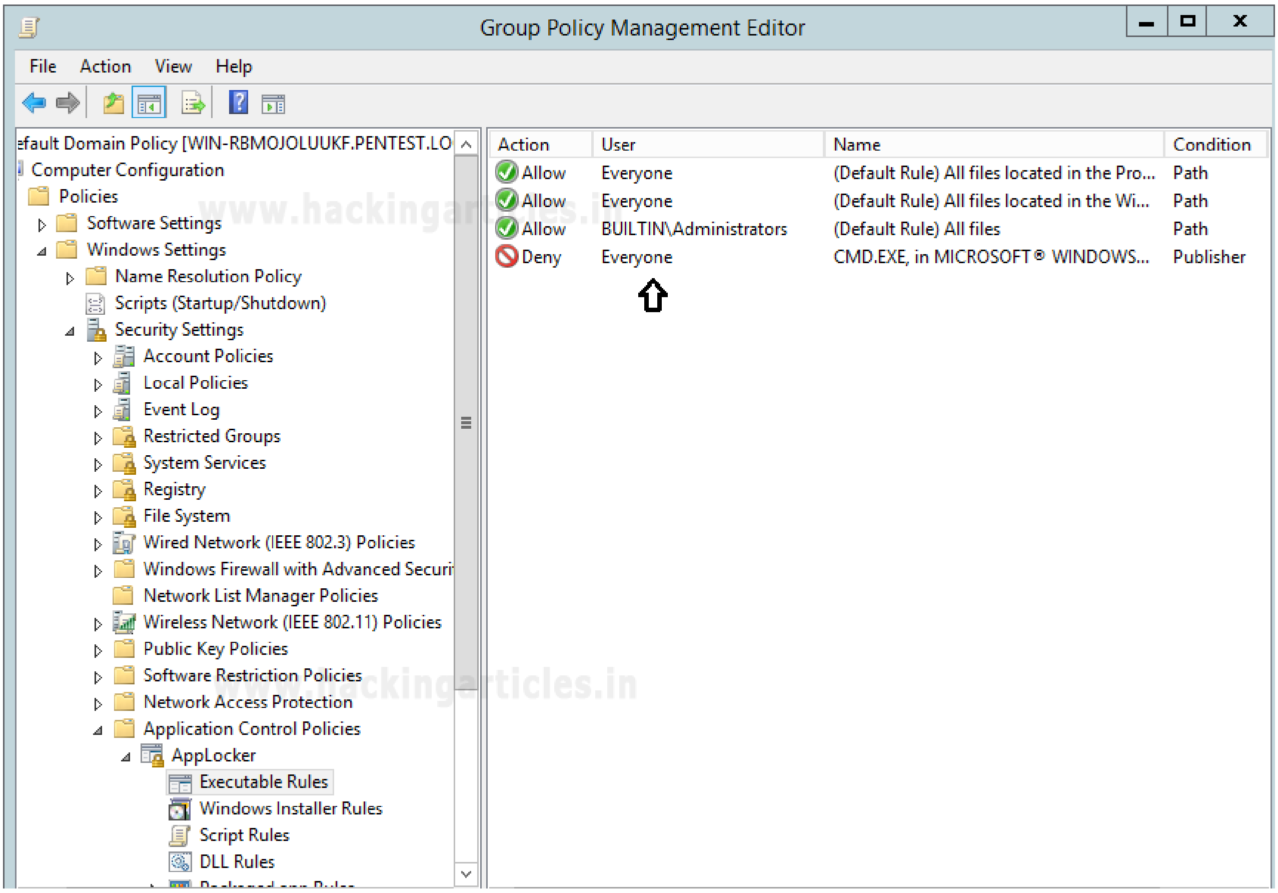
Primary condition:
CMD.EXE, in MICROSOFT® WINDOWS® OPERATING SYSTEM, from O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US

Add exception:
Publisher

Exceptions:

Exception	Type

And finally, this will add your rule to restrict the cmd.exe.



- efault Domain Policy [WIN-RBMOJOLUUKF.PENTEST.LO
  - Computer Configuration
    - Policies
      - Software Settings
      - Windows Settings
        - Name Resolution Policy
        - Scripts (Startup/Shutdown)
        - Security Settings
          - Account Policies
          - Local Policies
          - Event Log
          - Restricted Groups
          - System Services
          - Registry
          - File System
          - Wired Network (IEEE 802.3) Policies
          - Windows Firewall with Advanced Securi
          - Network List Manager Policies
          - Wireless Network (IEEE 802.11) Policies
          - Public Key Policies
          - Software Restriction Policies
          - Network Access Protection
          - Application Control Policies
            - AppLocker
              - Executable Rules
              - Windows Installer Rules
              - Script Rules
              - DLL Rules

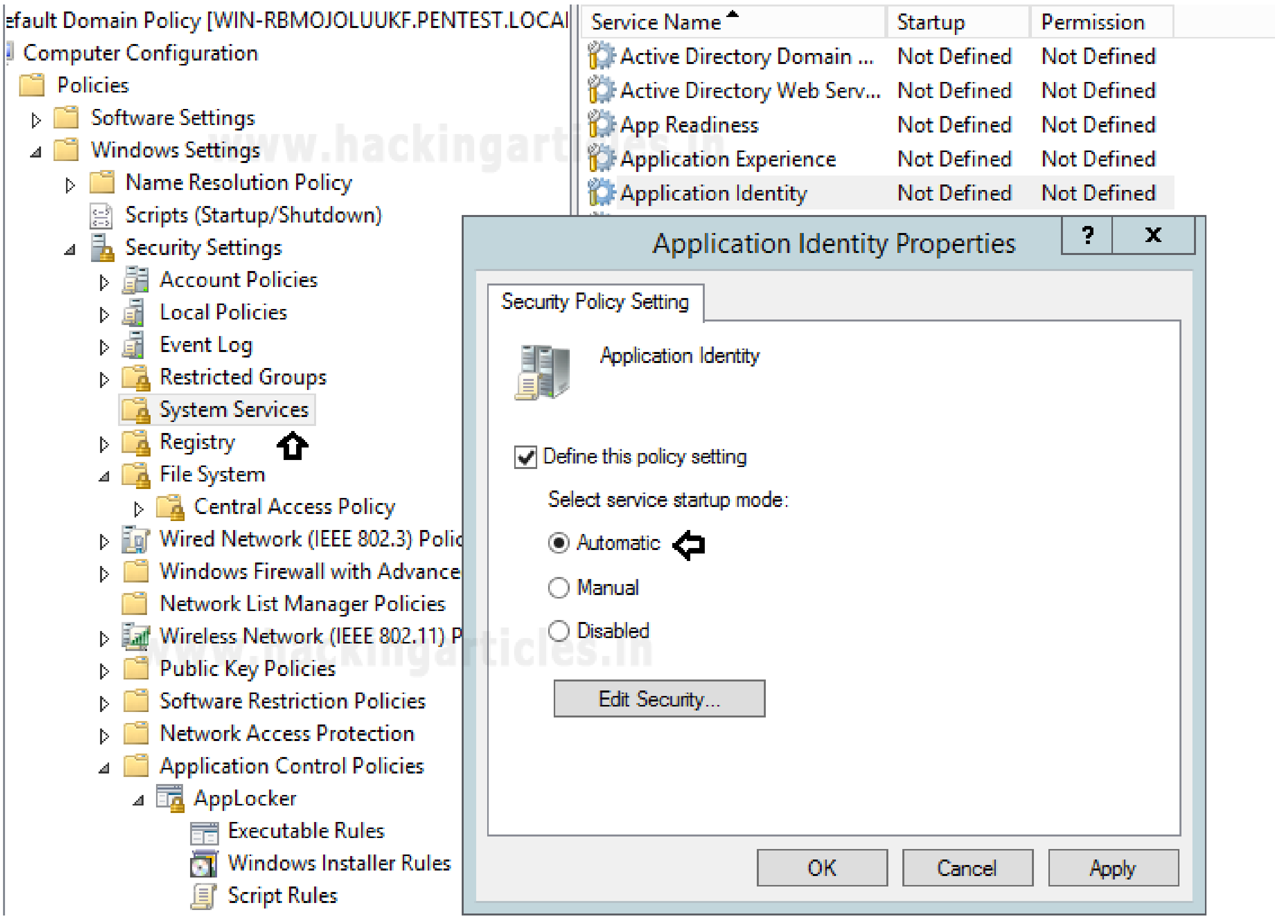
Action	User	Name	Condition
Allow	Everyone	(Default Rule) All files located in the Pro...	Path
Allow	Everyone	(Default Rule) All files located in the Wi...	Path
Allow	BUILTIN\Administrators	(Default Rule) All files	Path
Deny	Everyone	CMD.EXE, in MICROSOFT® WINDOWS...	Publisher

Set Application identity to Automatic mode:

Then navigate to “Application identity Property” through ***Computer Configuration > Policies > Windows Settings > Security Settings > System Services > Application identity***.

Then enable the “**Automatic**” option as the service startup mode.

Now update the Group policy with the help of **gpupdate command**.



efault Domain Policy [WIN-RBMOJOLUUKF.PENTEST.LOCAl
- Computer Configuration
  - Policies
    - Software Settings
    - Windows Settings
      - Name Resolution Policy
      - Scripts (Startup/Shutdown)
      - Security Settings
        - Account Policies
        - Local Policies
        - Event Log
        - Restricted Groups
        - System Services
        - Registry
        - File System
          - Central Access Policy
        - Wired Network (IEEE 802.3) Polic
        - Windows Firewall with Advance
        - Network List Manager Policies
        - Wireless Network (IEEE 802.11) P
        - Public Key Policies
        - Software Restriction Policies
        - Network Access Protection
        - Application Control Policies
          - AppLocker
            - Executable Rules
            - Windows Installer Rules
            - Script Rules

Service Name	Startup	Permission
Active Directory Domain ...	Not Defined	Not Defined
Active Directory Web Serv...	Not Defined	Not Defined
App Readiness	Not Defined	Not Defined
Application Experience	Not Defined	Not Defined
Application Identity	Not Defined	Not Defined

Application Identity Properties

Security Policy Setting

Application Identity

☑ Define this policy setting

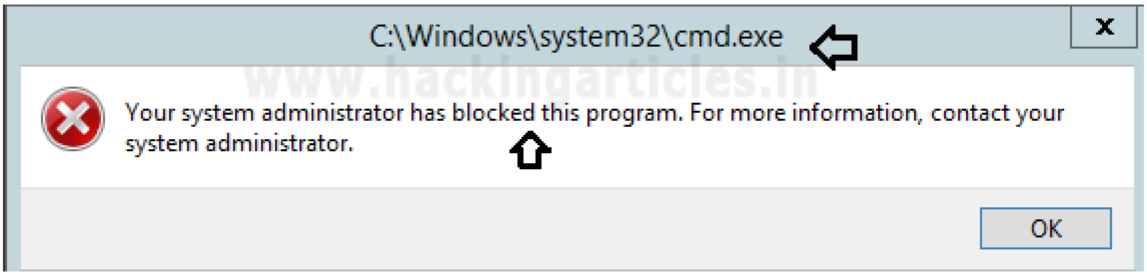
Select service startup mode:

● Automatic

○ Manual

○ Disabled

Now when you will try to open command prompt “cmd.exe” then you will get services restriction prompt as shown.



C:\Windows\system32\cmd.exe

Your system administrator has blocked this program. For more information, contact your system administrator.

Note: If you are configuring these rule on a single machine then it will take some time to impose the rule over the machine.

Reference: //docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/working-with-applocker-rules

Author: AArti Singh is a Researcher and Technical Writer at Hacking Articles an Information Security Consultant Social Media Lover and Gadgets. Contact **here**