EC-Council

Building A Culture Of Security

C|ND

Certified Network Defender

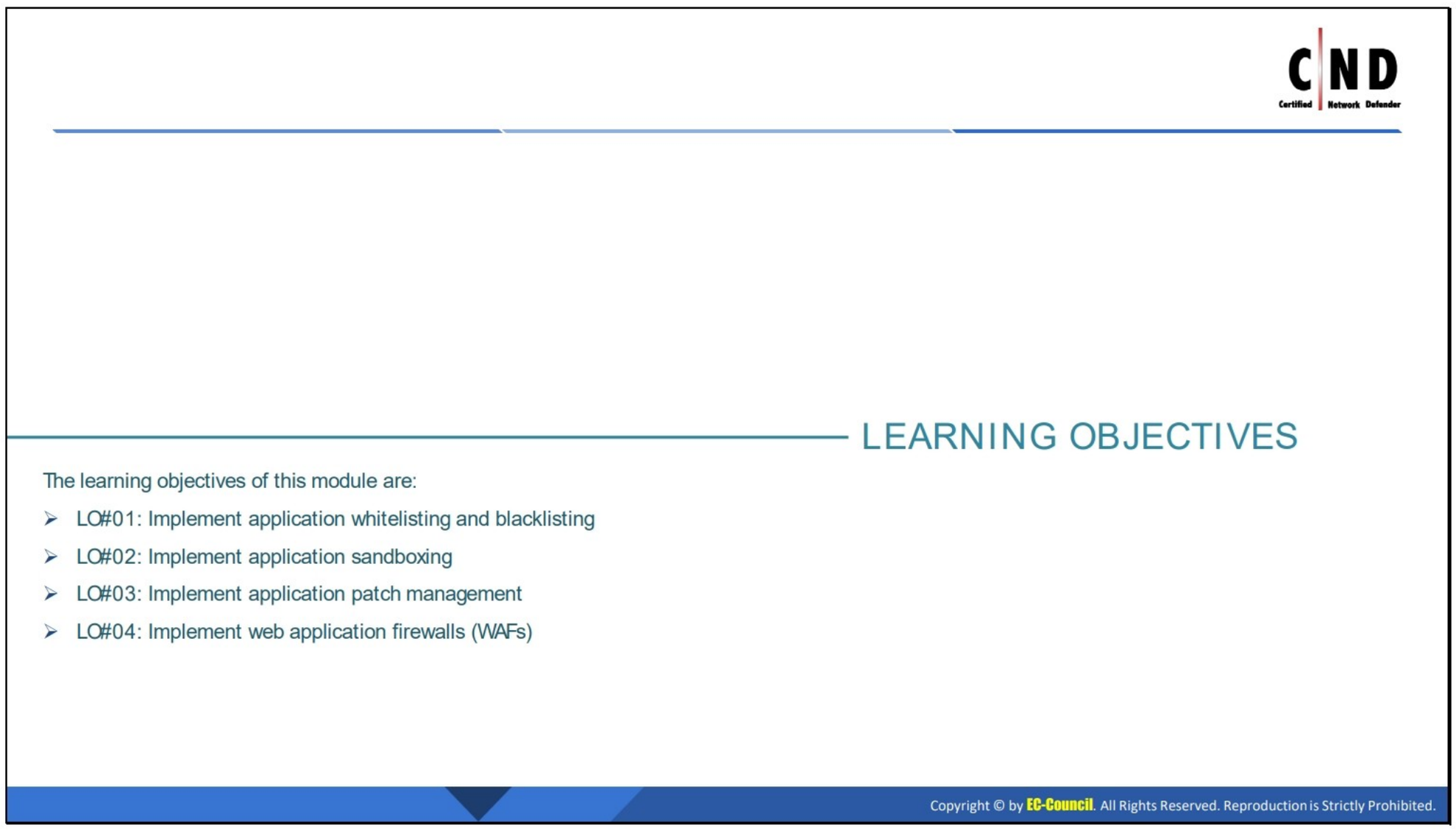
Certified Network Defender v3

MODULE 09

ADMINISTRATIVE APPLICATION SECURITY

EC-Council Official Curricula

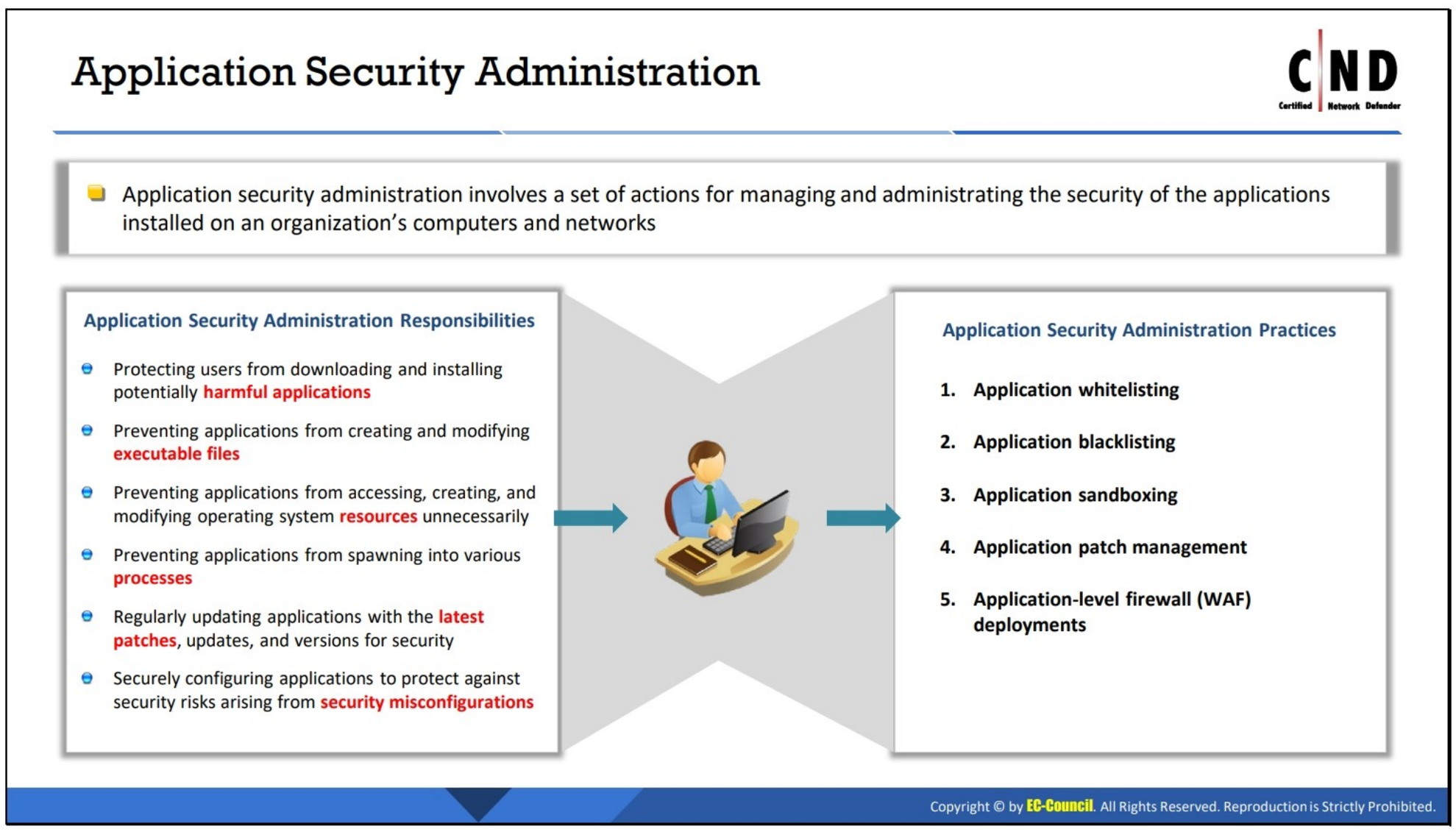
This page is intentionally left blank.



Learning Objectives

An outdated or insecure application installed on a system can pose a serious security threat and, in turn, affect network security. Network defender must manage the security of the deployed applications and constantly monitor, patch, and upgrade the installed applications. The learning objectives of this module are as follows:

- Implement application whitelisting and blacklisting
- Implement application sandboxing
- Implement application patch management
- Implement web application firewalls (WAFs)



Application Security Administration

Organizations must continuously monitor their applications for vulnerabilities to reduce potential risks and maintain the security of applications. Application security administration involves a set of actions for managing the security of the applications installed on an organization's computers and networks. It is one of the several levels of security that companies consider to secure systems.

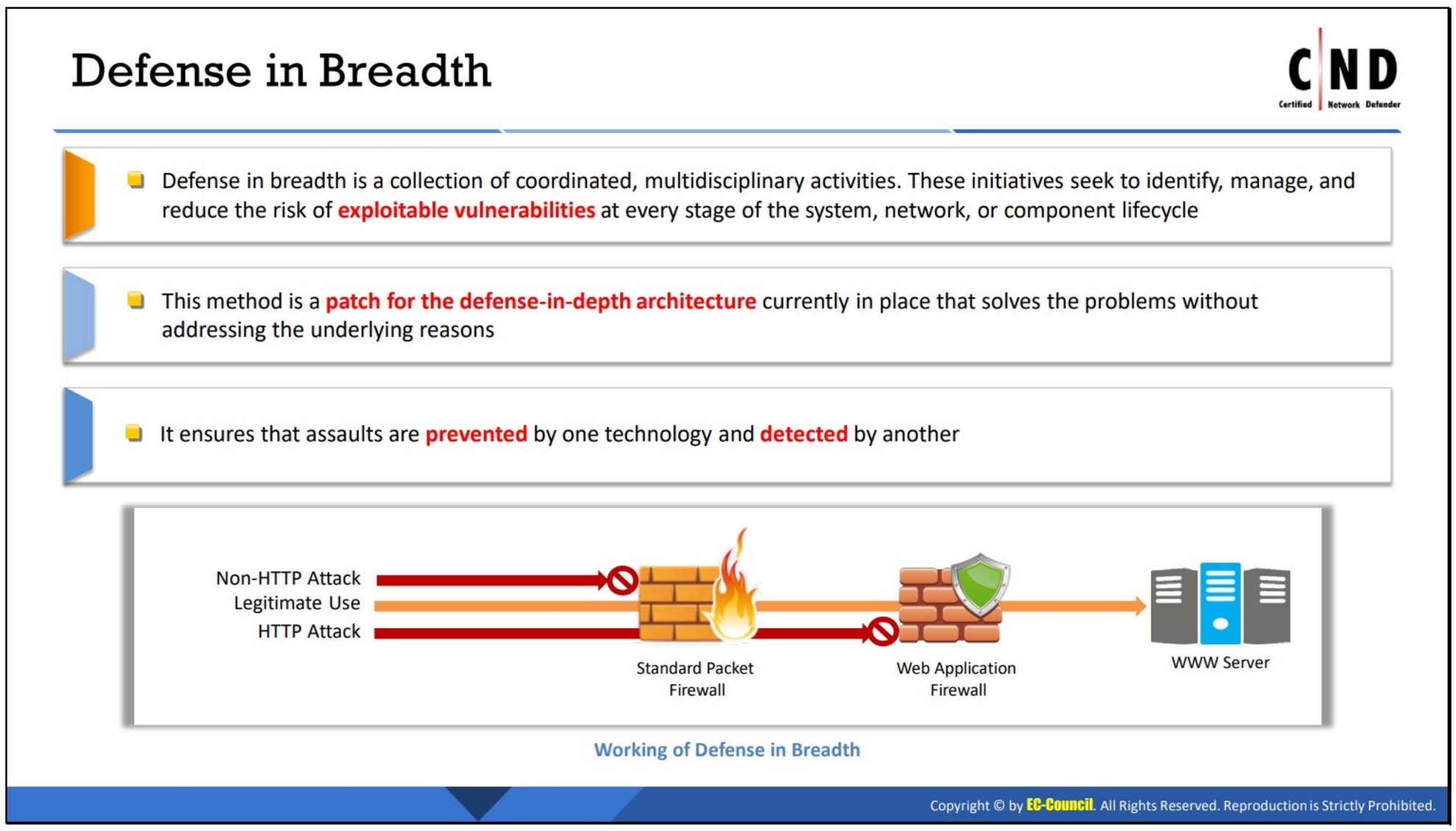
The typical responsibilities of application security administration include the following:

- Users must be protected from downloading and installing potentially harmful applications. Network defenders should not allow the downloading and installation of applications from untrusted sources or third-party sites. Untrusted sources may hide malware inside applications to compromise the system.
- Application must not be allowed to create and modify executable files. Network defenders should ensure the security of an application before it is installed on the system, and applications should be installed using the installation guide provided by the vendor.
- Applications must not be allowed to access, create, or modify OS resources unnecessarily. Network defender should monitor the running applications, and the applications should have only the required permissions to access the system resources to prevent the loss of confidentiality, integrity, and availability.
- Applications must not be allowed to spawn into various processes.
- Applications must be regularly updated with the latest patches and for security. The existence of outdated or insecure applications poses a serious security threat to the organization's network.

- Applications must be configured in a secure manner to protect users from security risks arising from security misconfigurations.

To implement application security in an organization, a network defender performs the following:

- **Application whitelisting/blacklisting**: Control the execution of unwanted or malicious applications.
- **Application sandboxing**: Execute untrusted or untested applications in an isolated environment to protect the system.
- **Application patch management**: Monitor and deploy new or missing patches to ensure the security of applications on hosts.
- **Application-level firewall (WAF) deployment**: Deploy WAF to protect web servers from malicious traffic.



Defense in Breadth

Defense in Breadth is a multi-faceted, coordinated approach encompassing various activities. These efforts are tailored to detect, manage, and reduce the risk of exploitable vulnerabilities at every system, network, or component level. The lifecycle includes the design, development, production, assembly, packaging, distribution, system integration, operations, maintenance, and retirement of a system or product.

However, this approach has its challenges. Administrators often find themselves investing disproportionately more time and resources in specific areas when layering defense systems, leaving other parts of the network sections vulnerable. The method lacks foresight and adaptability to emerging threats. When properly executed, defense in breadth can potentially reroute attacks similarly to defense in depth. Yet, it becomes ineffective and response-lagging when faced with new attack vectors, as addressing these requires considerable resources.

This method serves as a supplementary measure to the existing defense-in-depth architecture, addressing symptoms rather than root causes. It ensures that threats undetected by one technology are intercepted by another.

Figure 9.1: Defense in Breadth

Defense in Breadth vs Defense in Depth

Defense in Breadth	Defense in Depth
The fundamental tenet of Defense in Breadth is layering diverse security systems in the common attack vectors	In defense in depth, an information security plan uses technology, people, and operations to create flexible barriers across different layers of an organization
It ensures that attempts missed by one technology are intercepted by another	It offers varying levels of protection based on the source of the attack (internal or external)
In defense in breadth, planning and adaptability to new threats are absent	By layering in new security controls, defense in depth can adapt to new threats
It provides security at the application layer	It provides security at the network layer

Defense in Breadth vs Defense in Depth

The differences between defense in breadth and defense in depth are described below.

Defense in depth: The information security plan leverages the strengths of technology, people, and operations to create adaptable barriers across multiple layers within the organization. This approach offers security at the network level. Assets are fortified with multiple defensive strategies to prevent unauthorized access. To enhance protection, the defensive layers often overlap, ensuring that traffic undergoes scrutiny by multiple security technologies. This redundancy aims to compensate for any single security control's potential weakness. The defense-in-depth architecture not only thwarts the majority of attacks but also notifies administrators of any breaches. Moreover, it effectively counters automated attacks, as such attacks primarily target assets exposed to the public Internet. The defense in depth method employs security measures including network address translation (NAT), a firewall, a demilitarized zone (DMZ), and encryption to get access from the internet. a gateway intrusion detection system (IDS) to prevent the attack.

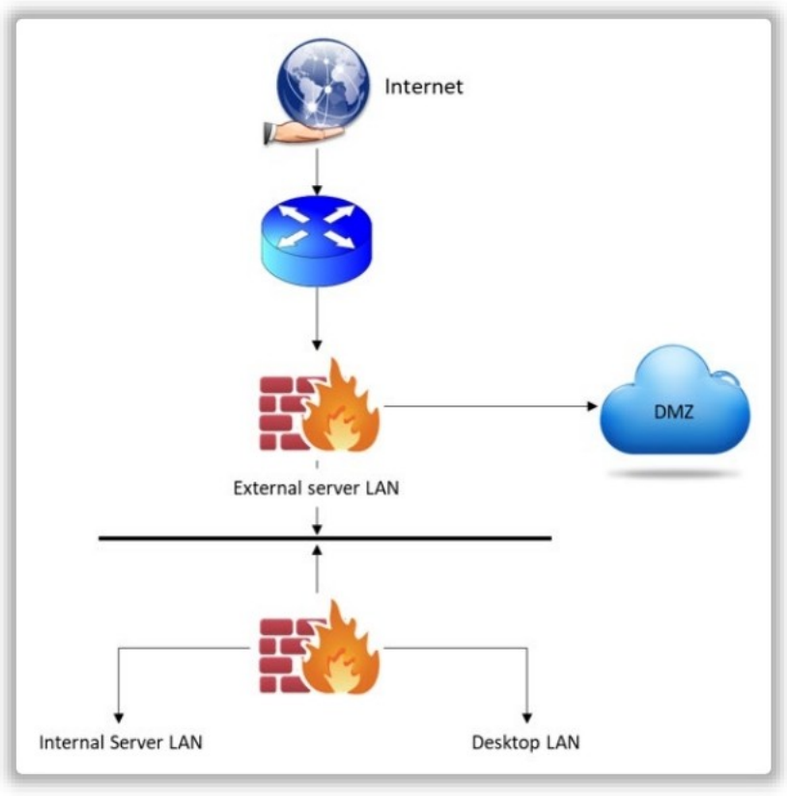


Figure 9.2: Defense-in-Depth Architecture

Defense in breadth: This is more of an extension to the "defense in depth" methodology rather than a fully fleshed-out approach. It tends to provide solutions to issues without necessarily addressing their root causes. The core principle of defense in breadth is to layer diverse security systems across common attack vectors. This ensures that if one technology fails to detect a threat, another will catch it. For example, in defense in breadth, multiple antivirus software programs might be installed on a single host. This way, if one antivirus fails to identify an attack, another might succeed. However, using multiple antiviruses can present challenges, as these programs are often not designed to operate concurrently with competing technologies. Furthermore, defense in breadth may lack proactive planning and agility in the face of emerging threats. While it offers many of the benefits of defense in depth when countering conventional attacks, it can be slow to adapt to novel attack vectors. Defense in breadth primarily focuses on security at the application layer.

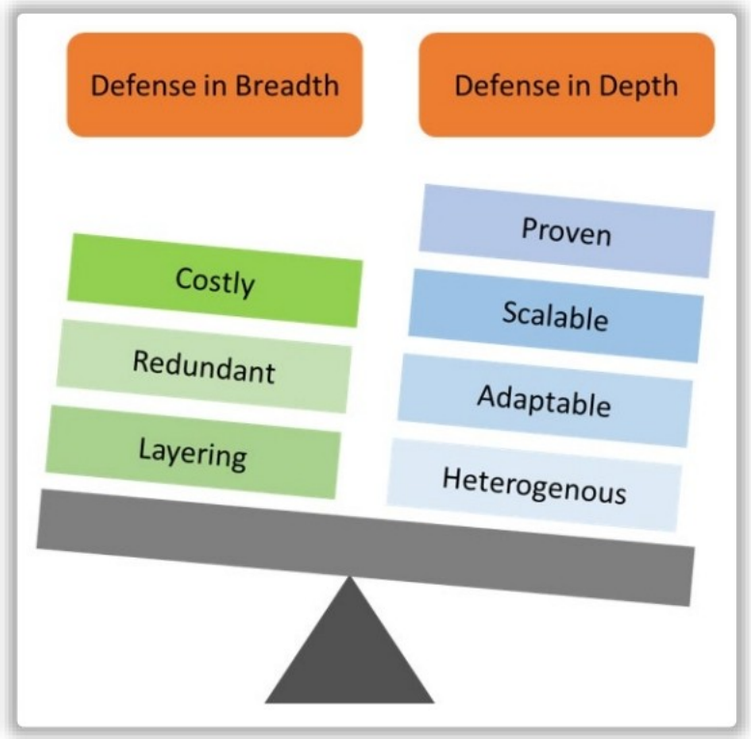
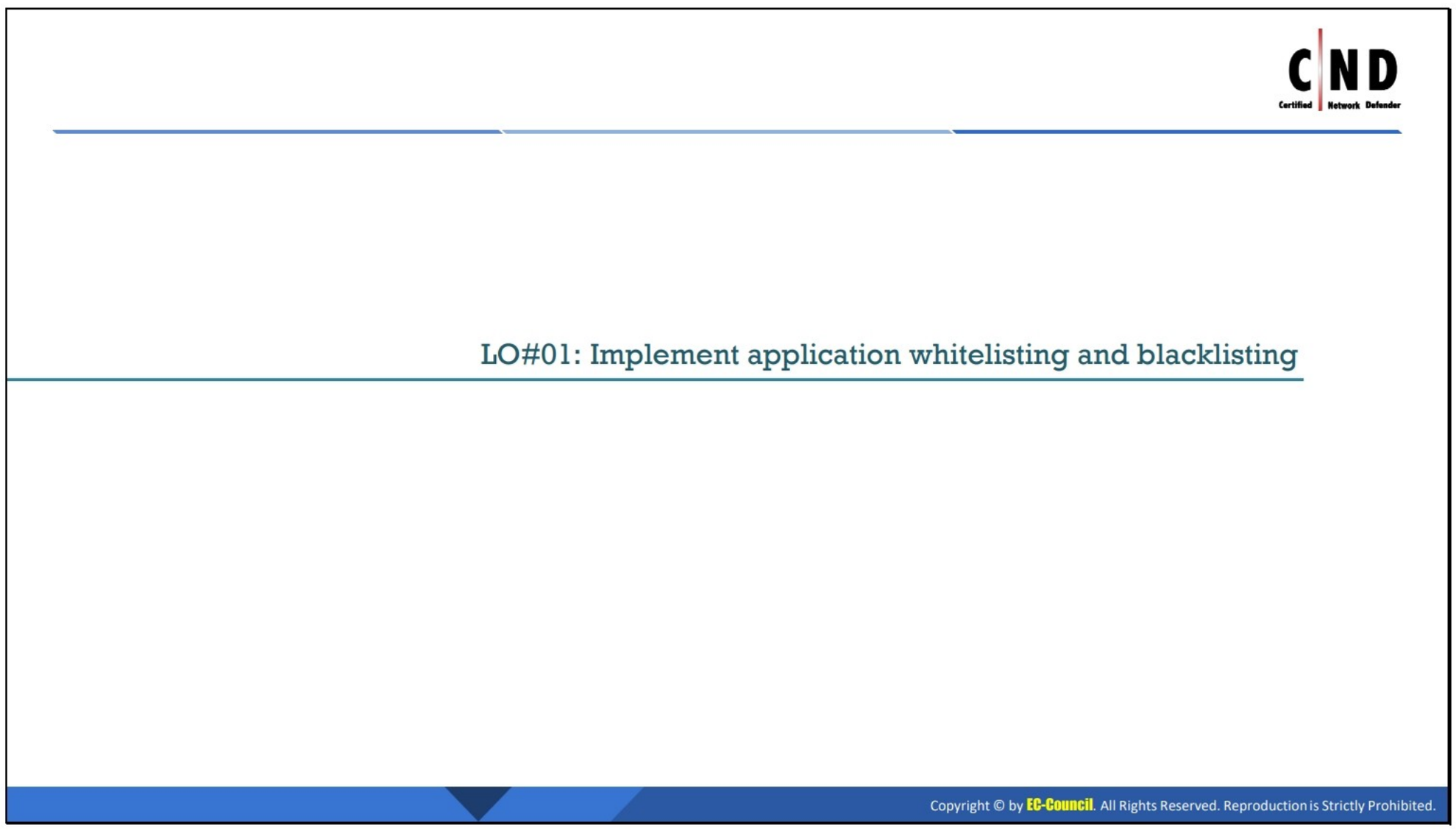
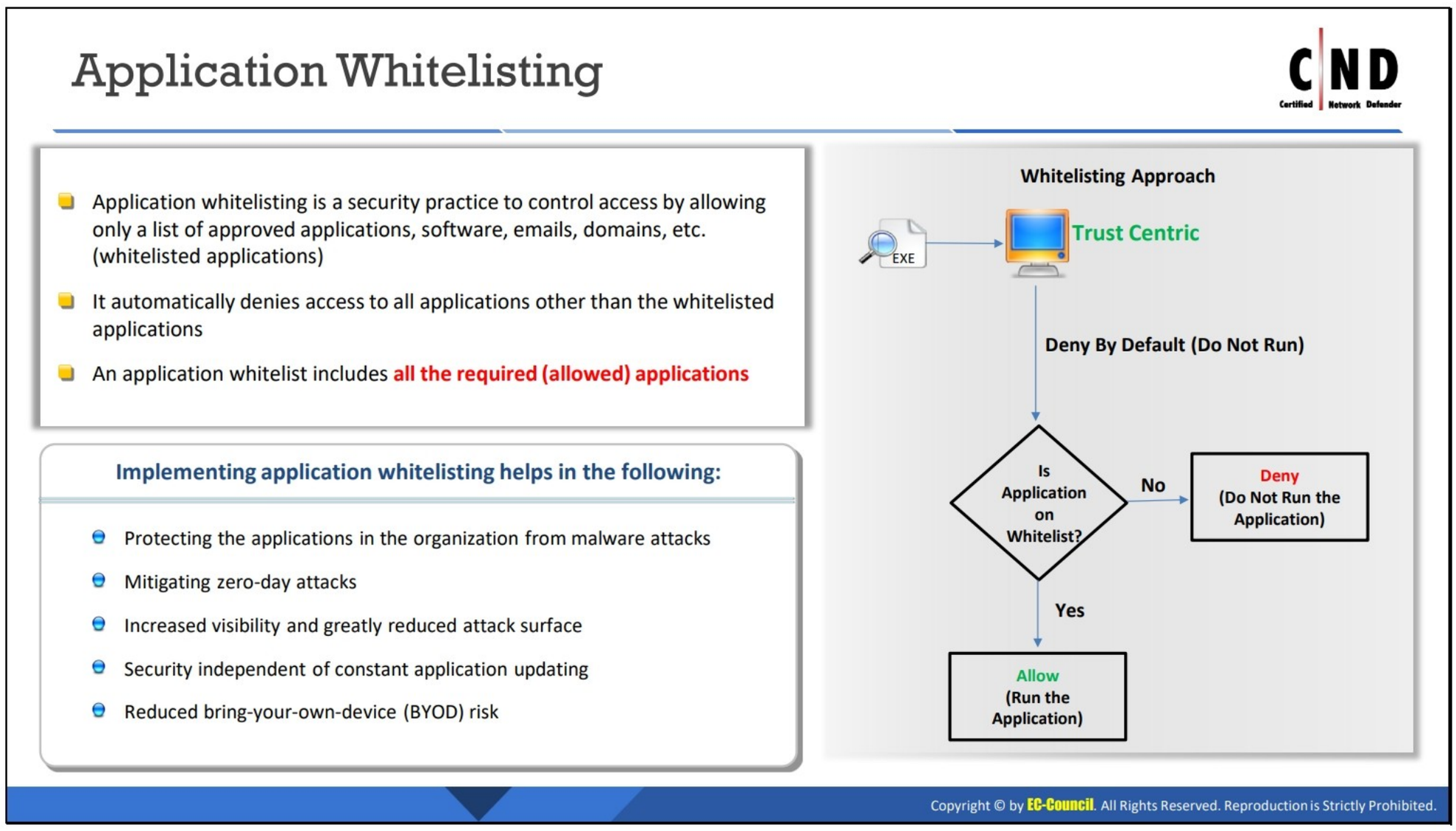


Figure 9.3: Comparison of Defense in Breath
and
Defense in Depth



LO#01: Implement Application Whitelisting and Blacklisting

Application whitelisting and blacklisting are two approaches used by network defenders to control and regulate access to system and network resources. This section aims to impart an understanding of application whitelisting and blacklisting approaches as well as their advantages. Furthermore, this section explains how to implement various tools for application whitelisting and blacklisting.



Application Whitelisting

Application whitelisting is a form of access control that allows only specific programs to run. Unless a program is whitelisted, it is blocked on a host. Application whitelisting technologies are also called application control programs or whitelisting programs.

The approach of application whitelisting is trust centric. By default, applications that are not in the whitelist are prevented from being executed. To allow the execution of any program or application, the network defender must add it in the application whitelist.

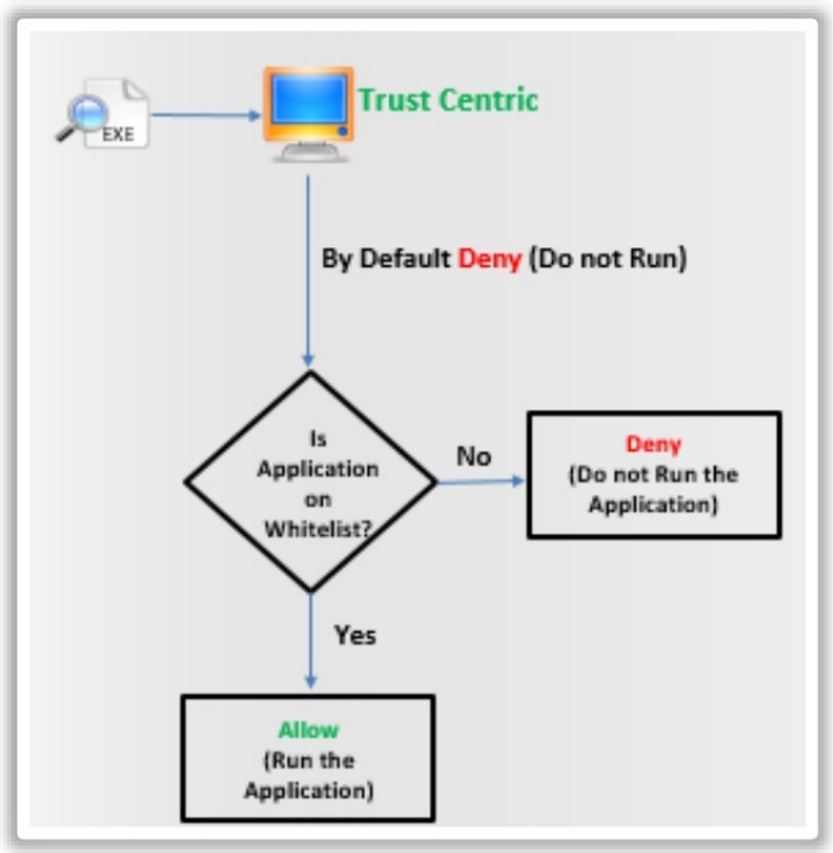


Figure 9.4: Whitelisting Approach

Any runtime process, host, application and application components (plug-ins, configuration files, software libraries, and extensions), email addresses, port numbers, etc. can be used to create a whitelist.

Advantages of Whitelisting

Implementing application whitelisting ensures the confidentiality, integrity, and availability of data. Application whitelisting provides network defenders and organizations the following benefits.

- **Protection against malware attacks**

  The whitelisting of applications in an organization can prevent malware attacks. Any application that is not in the whitelist is blocked.

- **Mitigating zero-day attacks**

  Generally, attackers start exploiting vulnerabilities once a software patch is released. Occasionally, malware for unpatched systems is ready to be deployed in a short time window during which a new patch has not yet been tested or implemented. Antivirus vendors also take time to identify new signatures to produce and distribute. Implementing application whitelisting hinders the execution of such vulnerabilities.

- **Improved efficiency of computers**

  Application whitelisting prevents unauthorized applications from running in organizations, improving the efficiency of computers.

- **Increased visibility and greatly reduced attack surface**

  Application whitelisting removes many basic attacks by protecting against the attack vector of download and execute. Application whitelisting enables organizations to track which applications are running or blocked on company systems. Improving the capability of monitoring and controlling applications greatly reduces the attack surface area, unauthorized changes to applications, and inspection requirements.

- **Reclaiming bandwidth from streaming or sharing applications**

  Application whitelisting avoids the significant use of resources to operate unapproved and unnecessary applications, ensuring the optimal utilization of company resources in organizations. Application whitelisting limits the exposure of social media applications, bans certain websites, eliminates games, and blocks other destructive applications that consume excessive employee time and network bandwidth.

- **Avoiding organizations from facing lawsuits or paying unnecessary license fees**

  Application whitelisting helps organizations avoid troubles such as lawsuits or license fees for unknowingly using unlicensed or illegal applications.

- **Security independent of constant application updating**

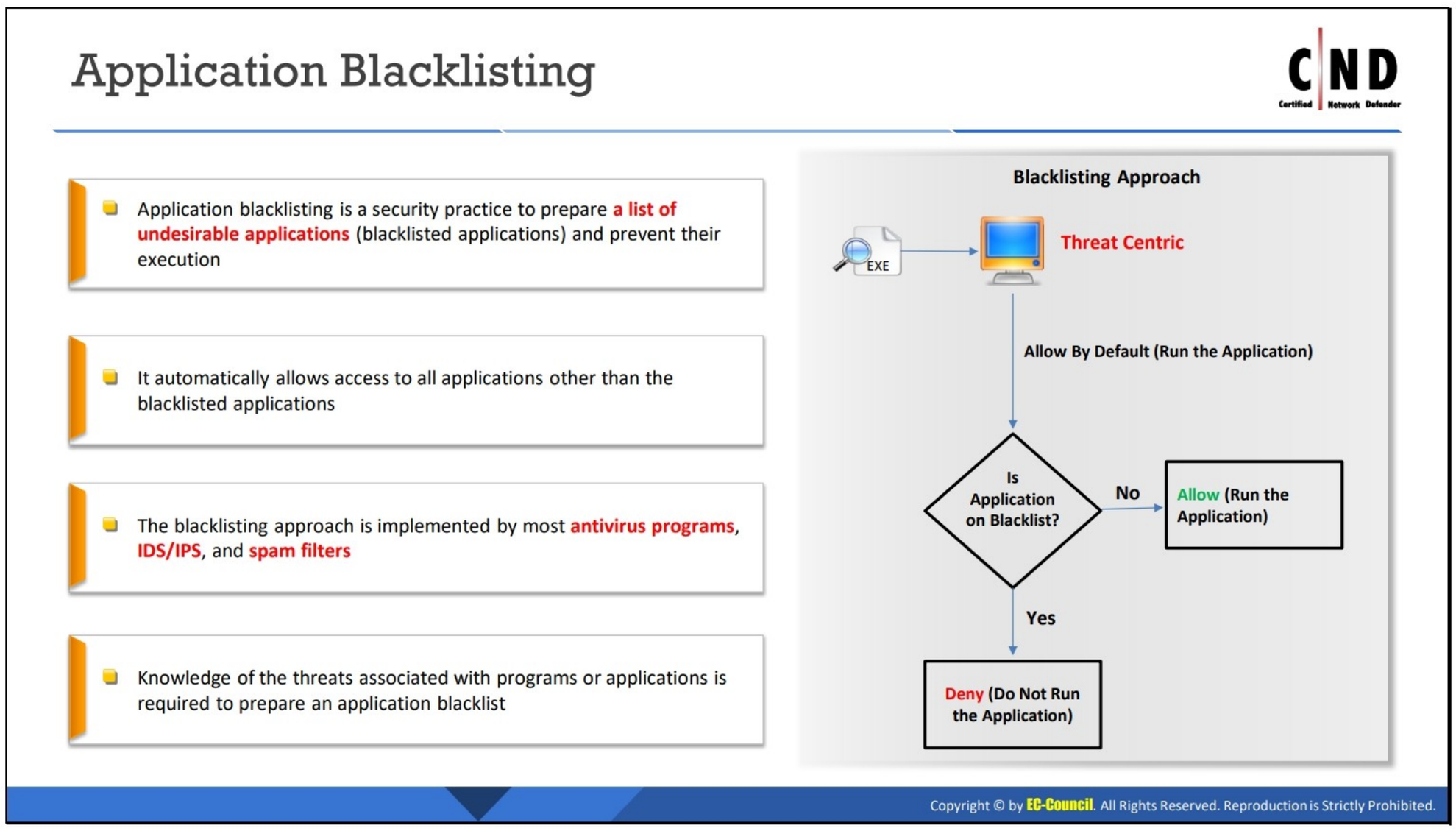
  Unlike antivirus programs, application whitelisting solutions do not need to get updated periodically to be active.

- **Easier attack detection**

  Attack detection becomes easier when many attack activities are blocked, and attacks generate a lot of noise. The noise created by attackers provide valuable information to incident response teams. This helps in measuring how long it takes for an antivirus solution to detect the existence of malware or changes on a system.

- **Reduced bring-your-own-device (BYOD) risk**

  Application whitelisting reduces BYOD risk through the enforcement of mobile-application policies.



Application Blacklisting

Application blacklisting is a security practice of blocking the running and execution of a list of undesirable programs.

Application blacklisting is threat centric. By default, it allows all applications that are not in the blacklist to be executed. To block any program or application, the network defender must add it in the application blacklist.

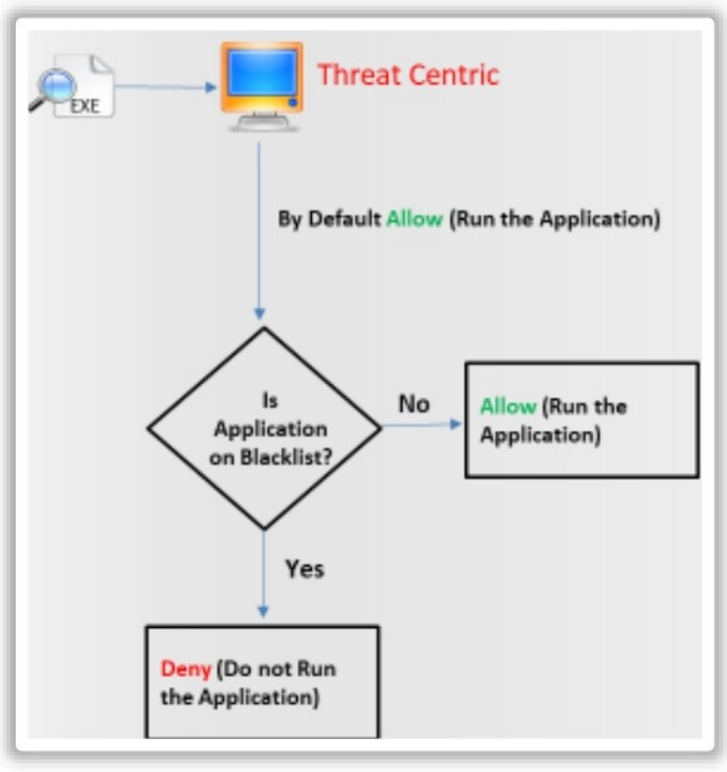


Figure 9.5: Blacklisting Approach

Most antivirus programs, spam filters and other intrusion prevention or detection systems use the application blacklisting method. A blacklist often comprises malware, users, IP addresses, applications, email addresses, domains, etc.

Advantages of Application Blacklisting

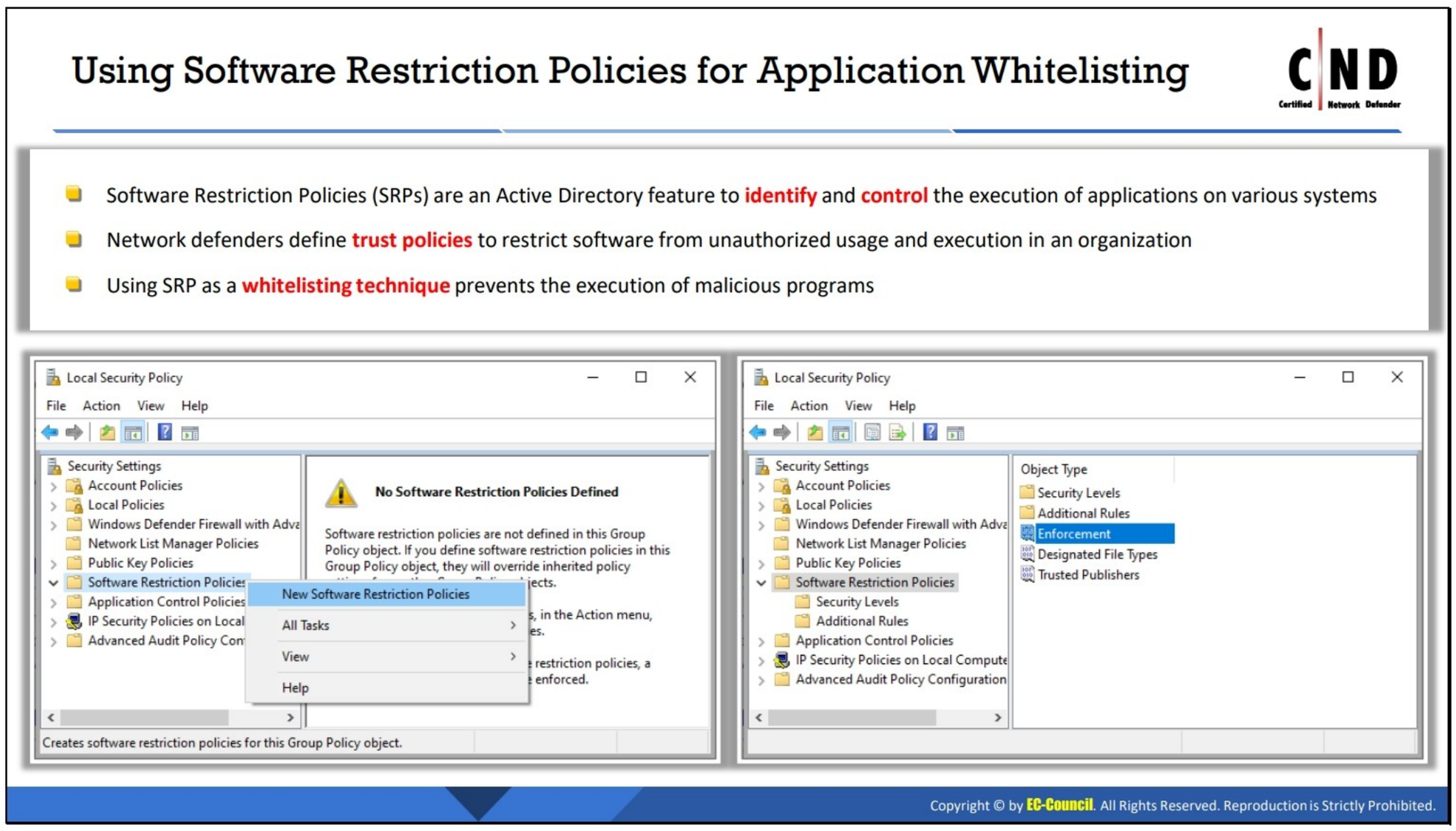
Application blacklisting provides network defenders and organizations the following benefits.

- It is simple to implement. A blacklist simply identifies the blacklisted applications, denies them access, and allows the execution of all other applications not in the blacklist.
- Blacklists need low maintenance since the security software compiles lists and do not ask users for inputs often.

Disadvantages of Application Blacklisting

The following are some of the disadvantages of implementing application blacklisting.

- A blacklist cannot be comprehensive, and the effectiveness of a blacklist is limited as the number of different and complex threats is continuously increasing. Sharing threat information can help make application blacklisting more effective.
- Blacklisting can tackle known attacks well but will not be able to protect against zero-day attacks. If an organization is the first target of new threats, blacklisting cannot stop them.
- Occasionally, hackers create malware to evade detection using blacklisting tools. In these cases, blacklisting fails to recognize the malware and add it to the blacklist.



Using Software Restriction Policies for Application Whitelisting

Windows facilitates software restriction policies (SRPs) to whitelist applications. SRPs are rules set by the network defender to restrict software from the unauthorized usage and execution in an organization. They are integrated with Microsoft Active Directory and Group Policy. The network defender can define policies through the Software Restriction Policies extension of the Local Group Policy Editor or the Local Security Policies related to the Microsoft Management Console (MMC).

There are four types of software restriction policy rules for whitelisting specific applications:

1. Path rules
2. Hash rules
3. Certificate rules
4. Internet zone rules

Path Rules

A path rule locates an application by its file path. Even when a computer is blocked (Disallowed) from all applications, a specific folder can be allowed (Unrestricted) for each user is possible by creating a path rule through the file path.

It is possible to create registry path rules, which use the registry key of an application as its path.

If an application is moved from a path, the path rule no longer applies to that application.

- **Steps to create path rules for whitelisting applications:**
  - Open **Software Restriction Policies**.

- Right-click **Additional Rules** in the console tree or the details pane and select **New Path Rule**.

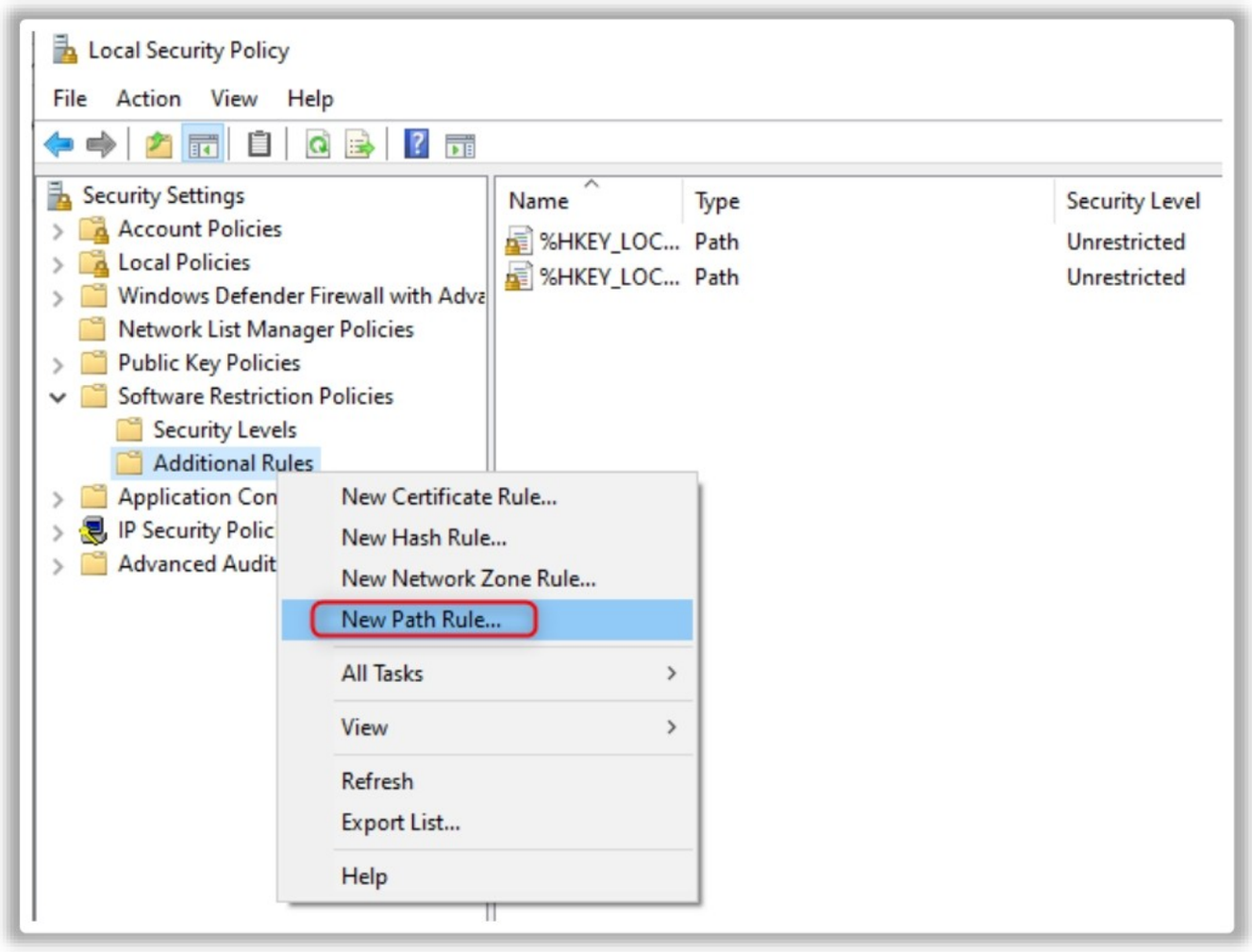


Figure 9.6: Selecting New Path Rule

- In **Path**, type a path or click **Browse** to find a file or folder.
- Based on the requirement, click **Disallowed** or **Unrestricted** in **Security level**.

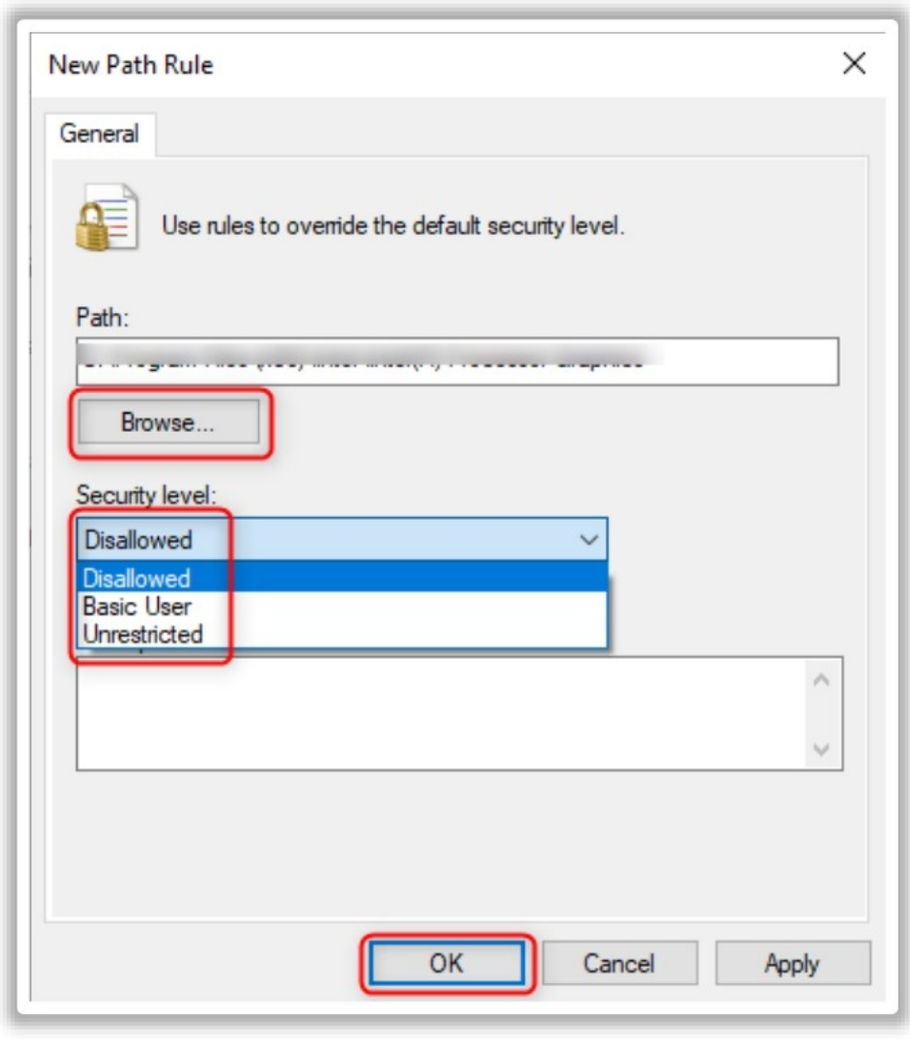


Figure 9.7: Specify the "Security level"

- Add a description of the rule in the **Description** box and click **OK**.

- **Steps to create a registry path rule for whitelisting applications:**
  - Enter "regedit" in the **Start** menu and open the Registry Editor.
  - In the console tree, right-click the registry key on which the path rule applies and click **Copy Key Name**. Note the value name in the details pane.

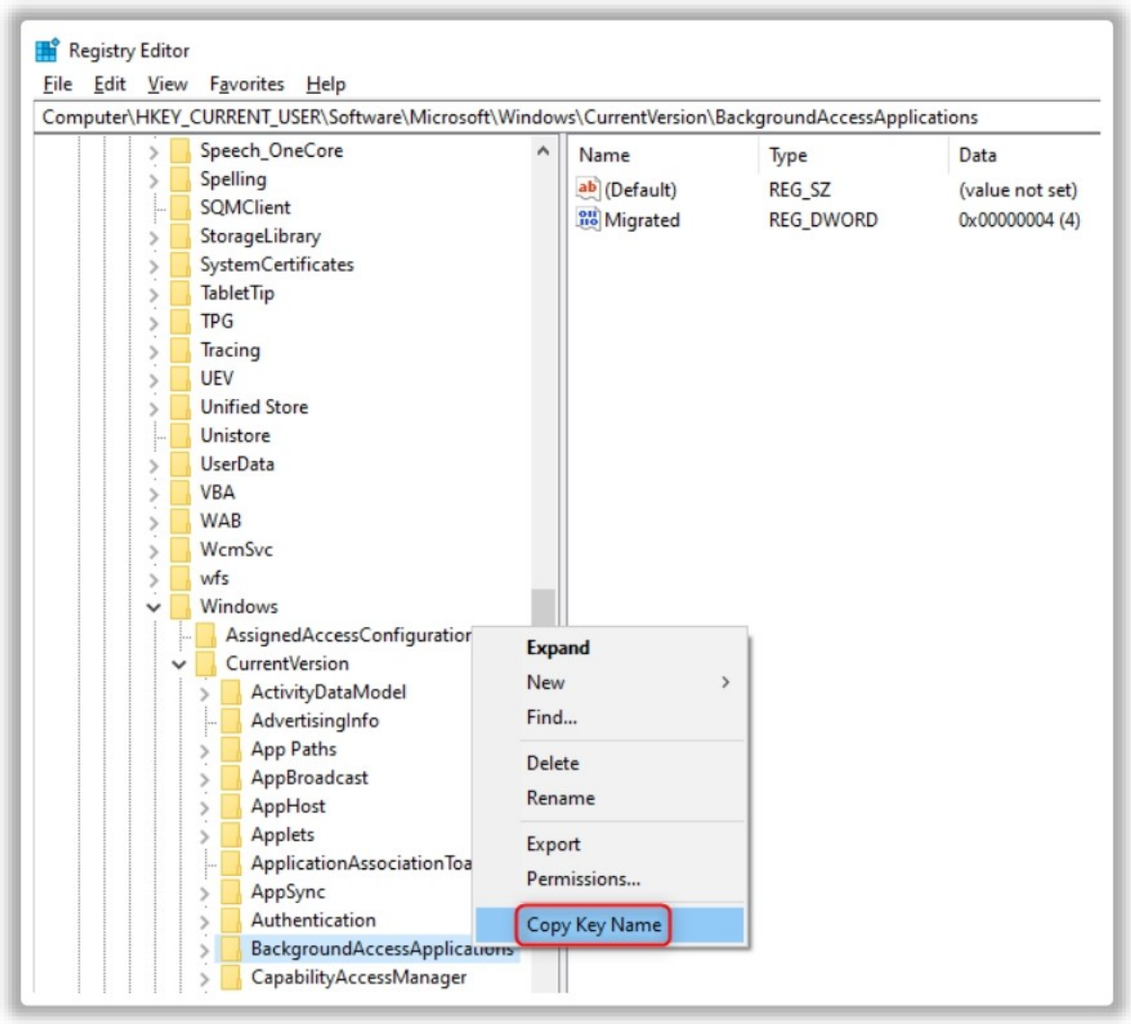


Figure 9.8: Copy Key Name from Registry Editor

- Open **Software Restriction Policies**.
- Right-click **Additional Rules** in the console tree or the details pane and click **New Path Rule**.

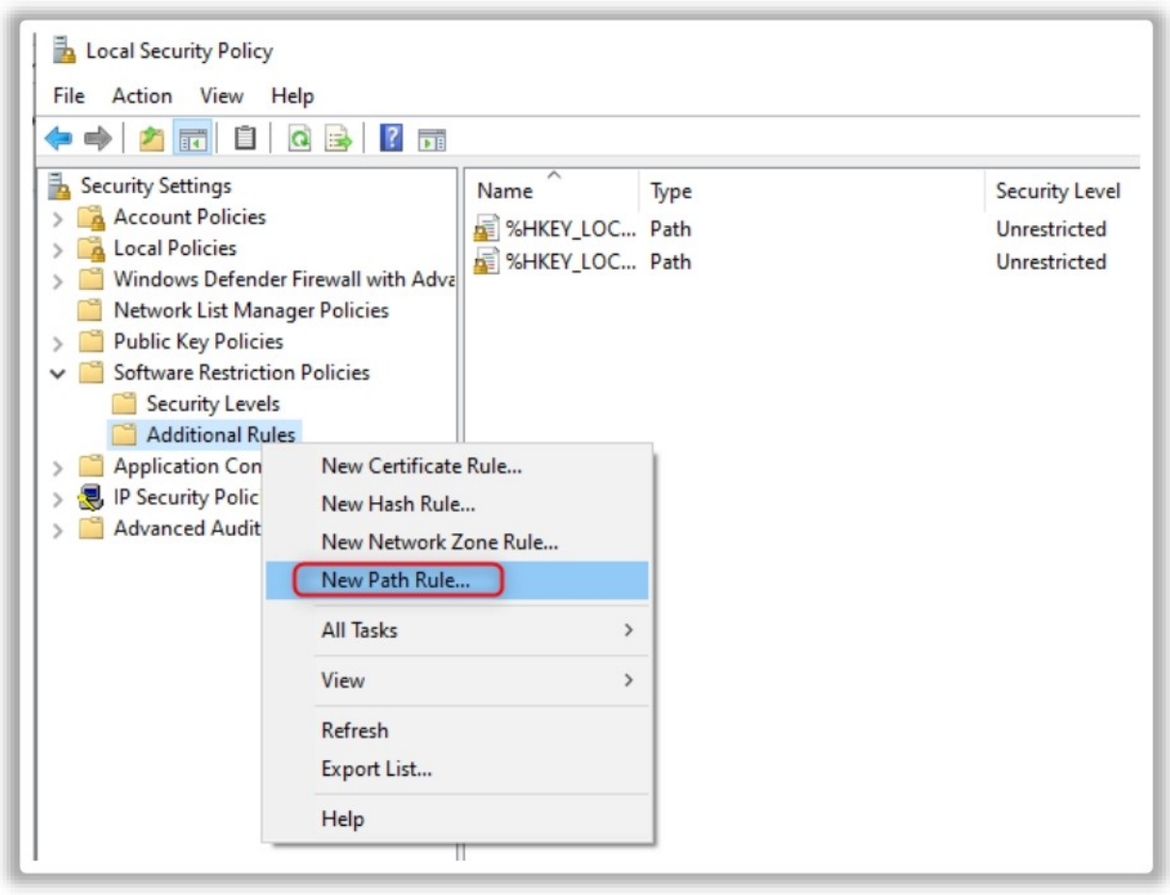


Figure 9.9: Create New Path Rule

- Paste the registry key name in **Path** followed by the value name.
- Enclose the registry path in percent signs (%). For example,

  %HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Background AccessApplications%
- Based on the requirement, select **Disallowed** or **Unrestricted** in **Security level**.

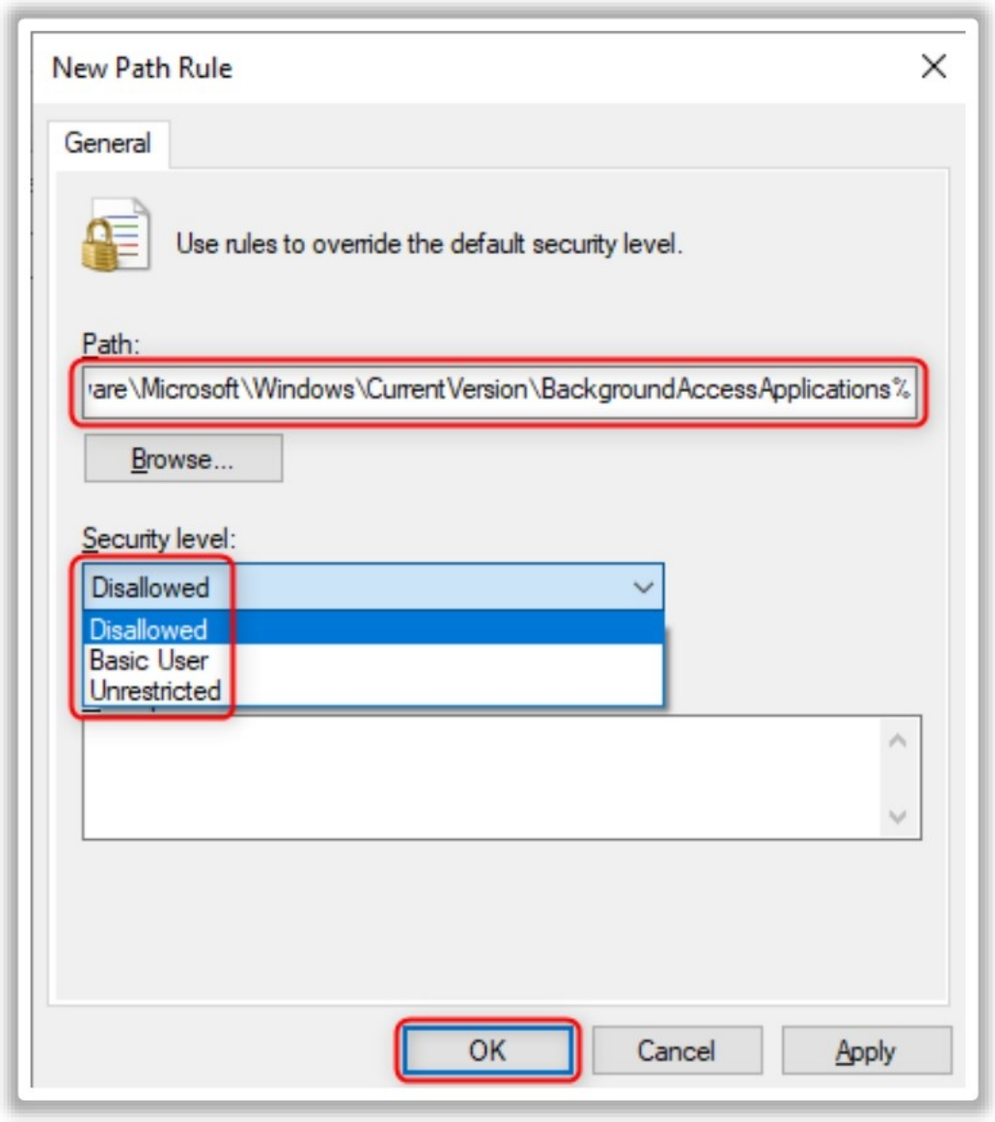


Figure 9.10: Specify the "Security level"

- Add a description of the rule in the **Description** box and click **OK**.

Hash Rule

Software restriction policies identify files by their hash. When opening an application, the existing hash rules of software restriction polices are compared to the hash of the application. The hash rule allows an application to run regardless of its location. For example, creating a hash rule and setting the security level to Disallowed stop users from running a specific file. The hash of an application remains the same irrespective of its location. Even renaming or moving the file to another location cannot change the hash value. If the user upgrades to a newer version of the application, the hash rule would no longer apply, even when the filename remains the same. Only the changes to the file (including a change to even one byte in the file) can change its hash value and allow it to bypass restrictions.

- **Steps to create a hash rule for whitelisting applications:**
  - Open **Software Restriction Policies.**
  - In the details pane, right-click **Additional Rules** and select **New Hash Rule**.

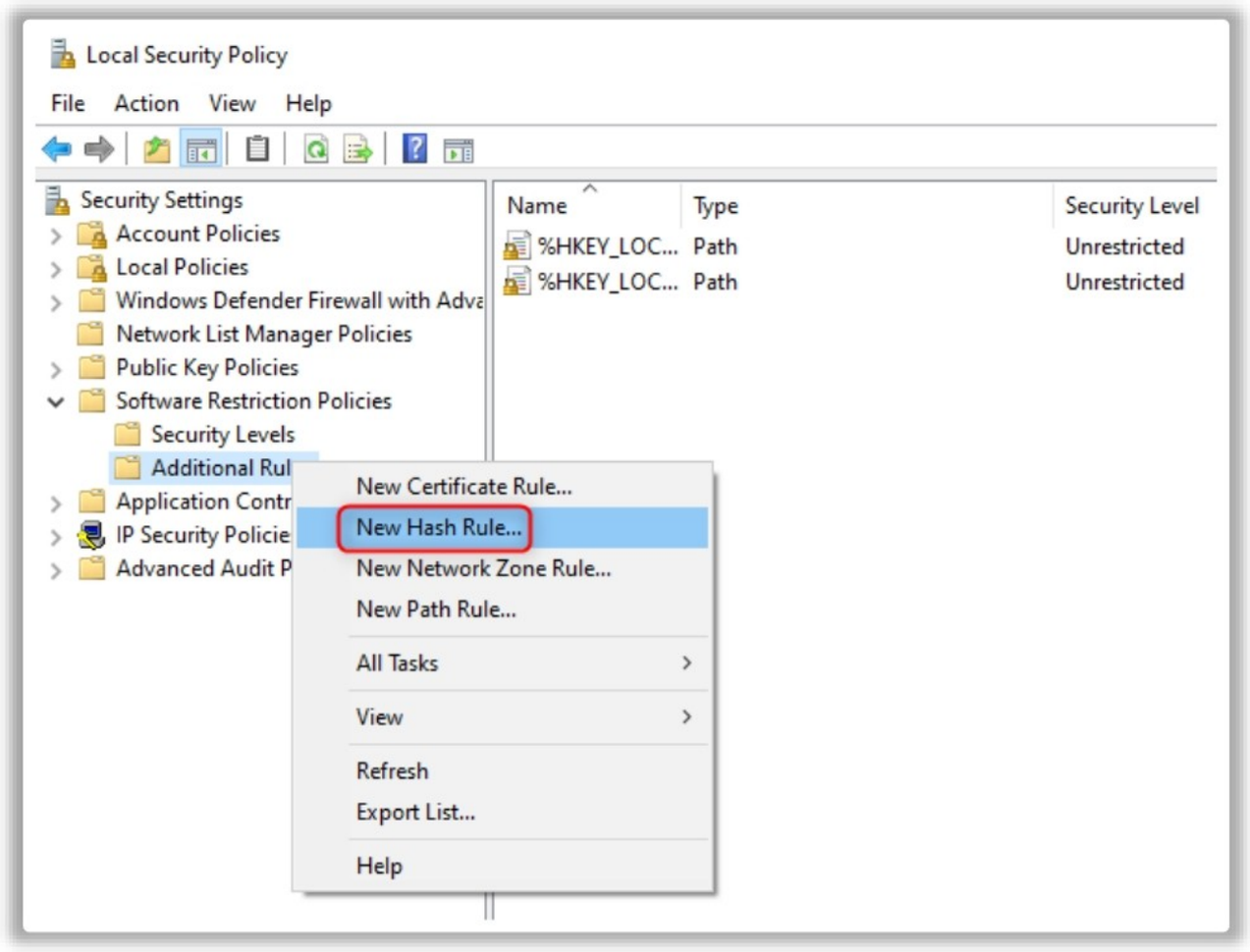


Figure 9.11: Create New Hash Rule

- Click **Browse** to find a file.

Note: It is possible to paste a pre-calculated hash in **File hash** in Windows XP. This option is not available in Windows Server 2008 R2, Windows 7, and later versions.

- Based on the requirement, select either **Disallowed** or **Unrestricted** in **Security level.**
- Add a description of the rule in the **Description** box and click **OK**.

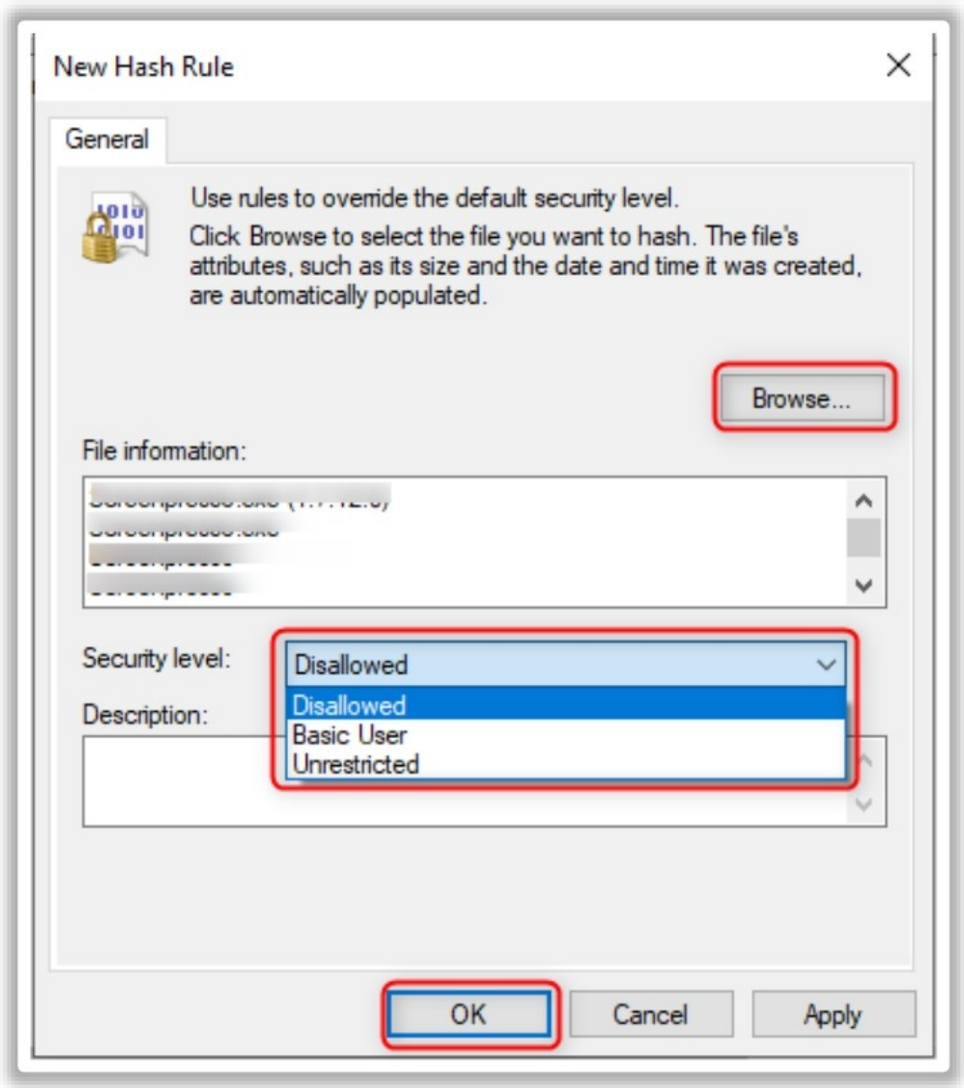


Figure 9.12: Specify the "Security level"

- **Note:**
  - Permissions are needed to create or modify a Group Policy Object when creating rules for the domain using Group Policy.
  - Never email the virus itself if other users wish to use a hash rule to prevent virus attacks. Calculate the hash of the virus by using software restriction policies and email the hash value to other users.
  - Users must log out from and login to systems to update policy settings and start applying the software restriction policies.
  - Creating a new software restriction policy setting for the Group Policy Object might be necessary if it has not yet been done.
  - If more than one software restriction policy rule is created, there will be a precedence of rules for handling conflicts.

Certificate Rule

Software restriction policies can identify applications by their signing certificate. Depending on the security level, software restriction policies allow or disallow any application to run.

For example, certificate rules are used to automatically trust software from trusted software vendors in a domain without user interference.

- **Steps to create a certificate rule for whitelisting applications:**
  - Open **Software Restriction Policies.**
  - In the details pane or console tree, right-click **Additional Rules** and select **New Certificate Rule**.

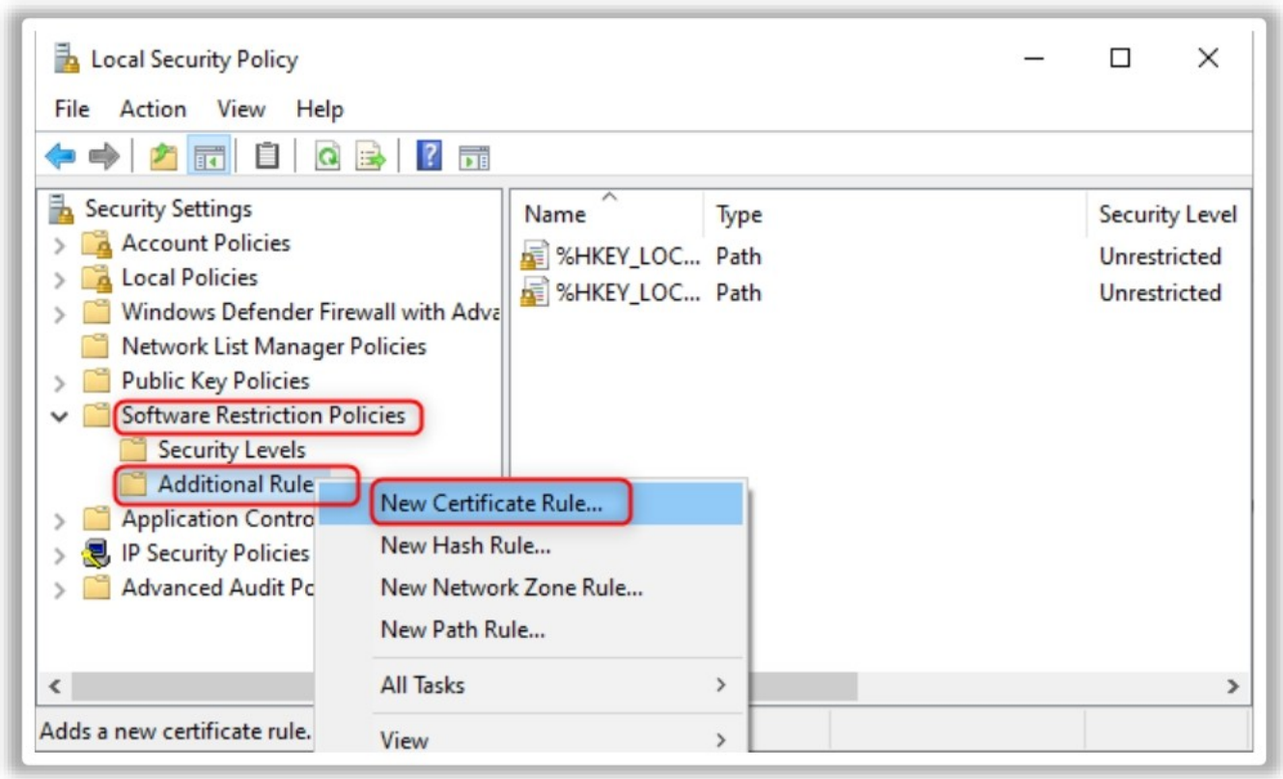


Figure 9.13: Create New Certificate Rule

  - Click **Browse** to select a certificate or signed file.
  - Based on priority, select either **Disallowed** or **Unrestricted** in **Security level**.

- Add a description of the rule in the **Description** box and click **OK**.

- **Note:**
  - Certificate rules are not enabled by default.
  - Administrative credentials are necessary for creating rules for local computers.
  - Permissions are needed to create or modify a Group Policy Object when creating rules for the domain using Group Policy.
  - Users must log out from and login to systems to update policy settings and start applying software restriction policies.
  - If more than one software restriction policy rule is created, there will be a precedence of rules for handling conflicts.
  - Creating a new software restriction policy setting for the Group Policy Object might be necessary if it has not yet been done.
  - Certificate rules can cause performance issues. Therefore, use them only when necessary.

Internet Zone Rules

The Internet Zone Rule locates software from zones identified through Internet Explorer and stops users from downloading and installing software. The zones are My Computer, Internet, trusted sites, local intranet, and restricted sites.

- **Steps to create internet zone rules for whitelisting applications:**
  - Open **Software Restriction Policies**.
  - In the details pane or console tree, right-click **Additional Rules** and select **New Internet Zone Rules.**

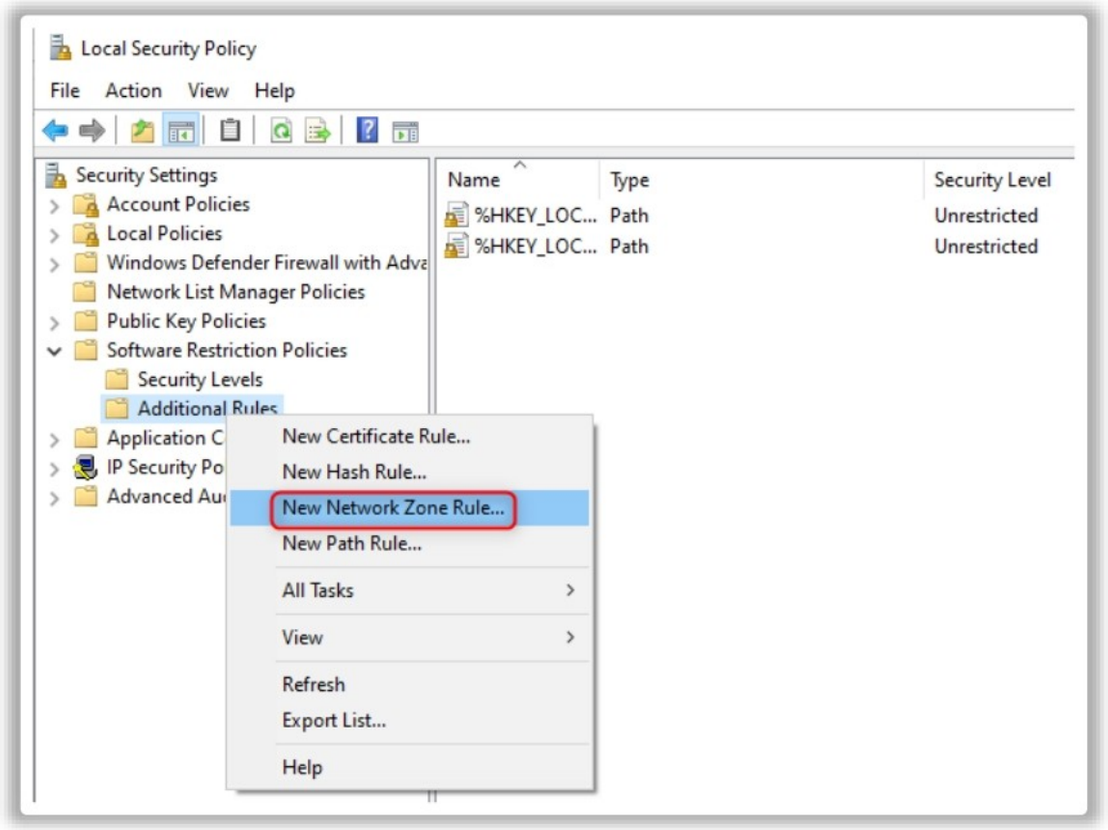


Figure 9.14: Create New Network Zone Rule

- Select Internet in **Network zone**.

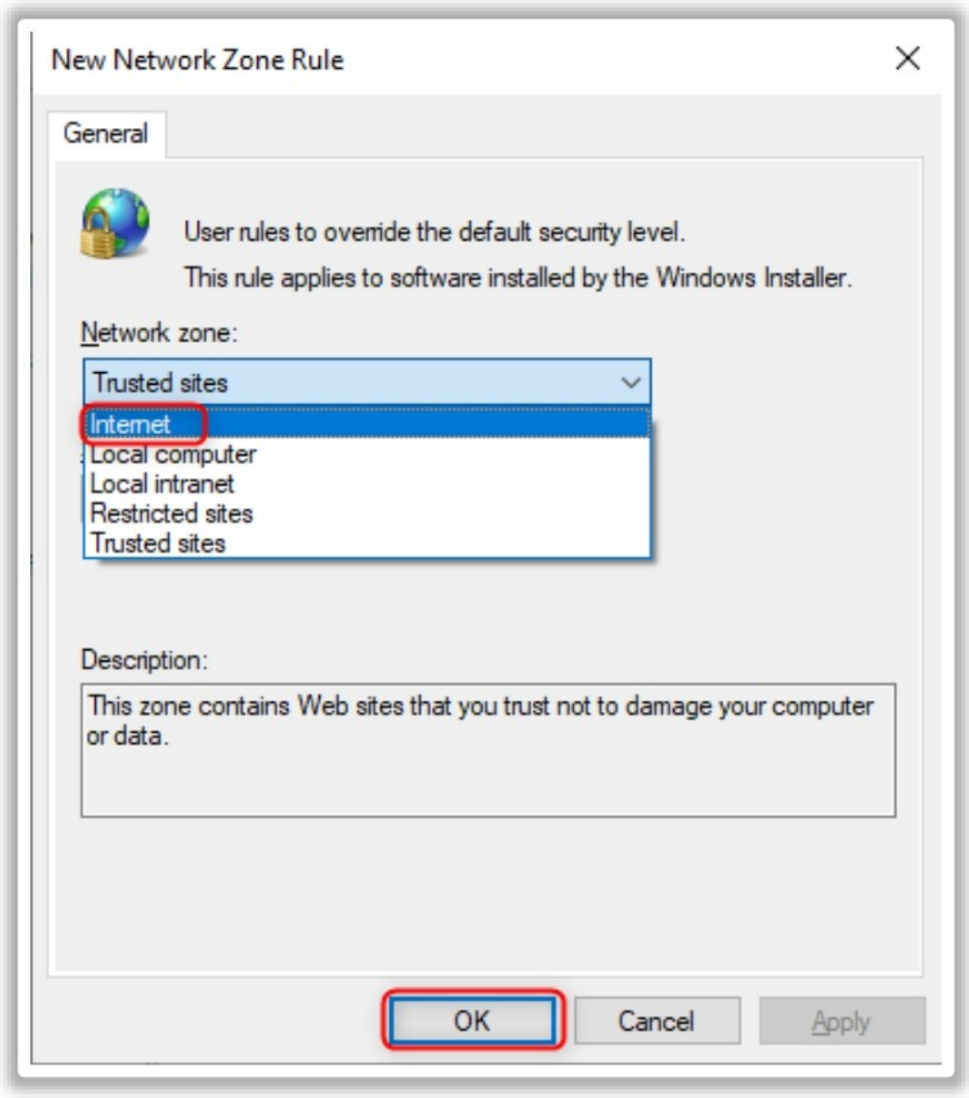


Figure 9.15: Specify the "Network zone:"

- Depending on the requirement, select **Disallowed** or **Unrestricted** in **Security level**.
- Add a description of the rule in the **Description** box and click **OK**.

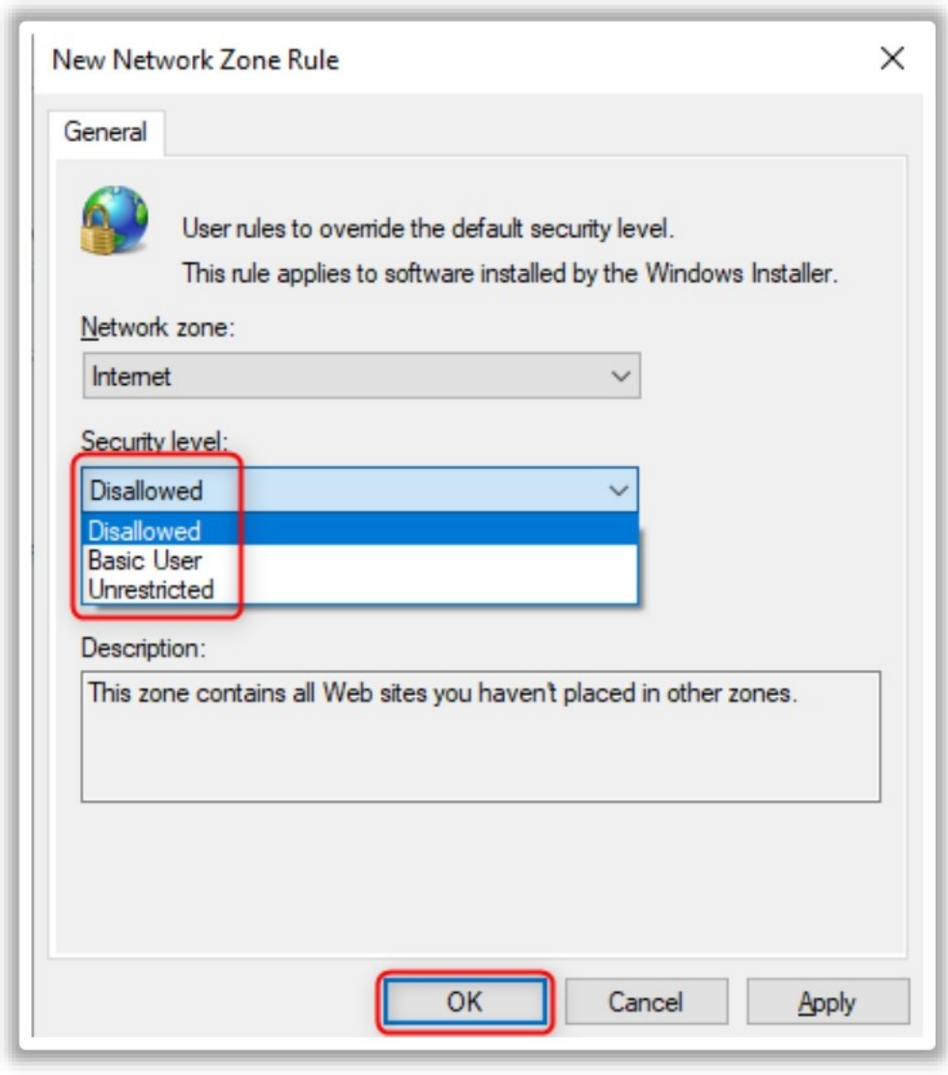


Figure 9.16: Specify the "Security level"

- **Note:**
  - Internet zone rules only apply to Windows Installer packages (.msi).
  - Creating a new software restriction policy setting for the Group Policy Object might be necessary if it has not yet been done.
  - Users must log out from and login to systems to update policy settings and start applying the software restriction policies.
  - If more than one software restriction policy rule is created, there will be a precedence of rules for handling conflicts.

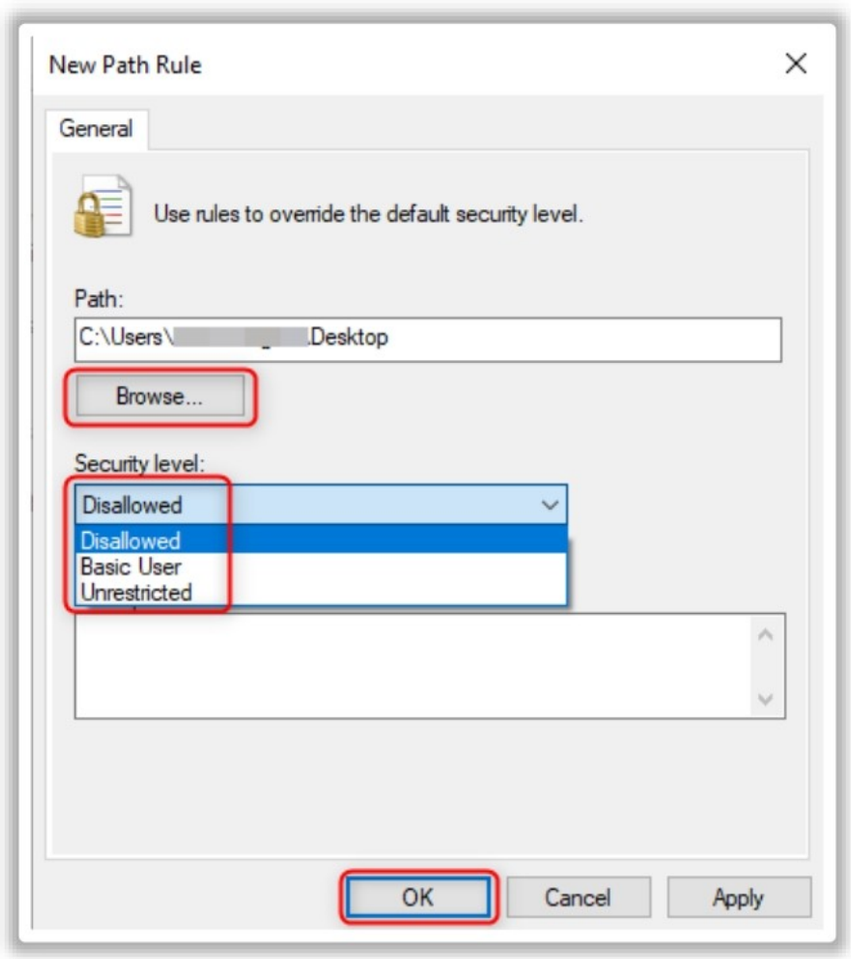
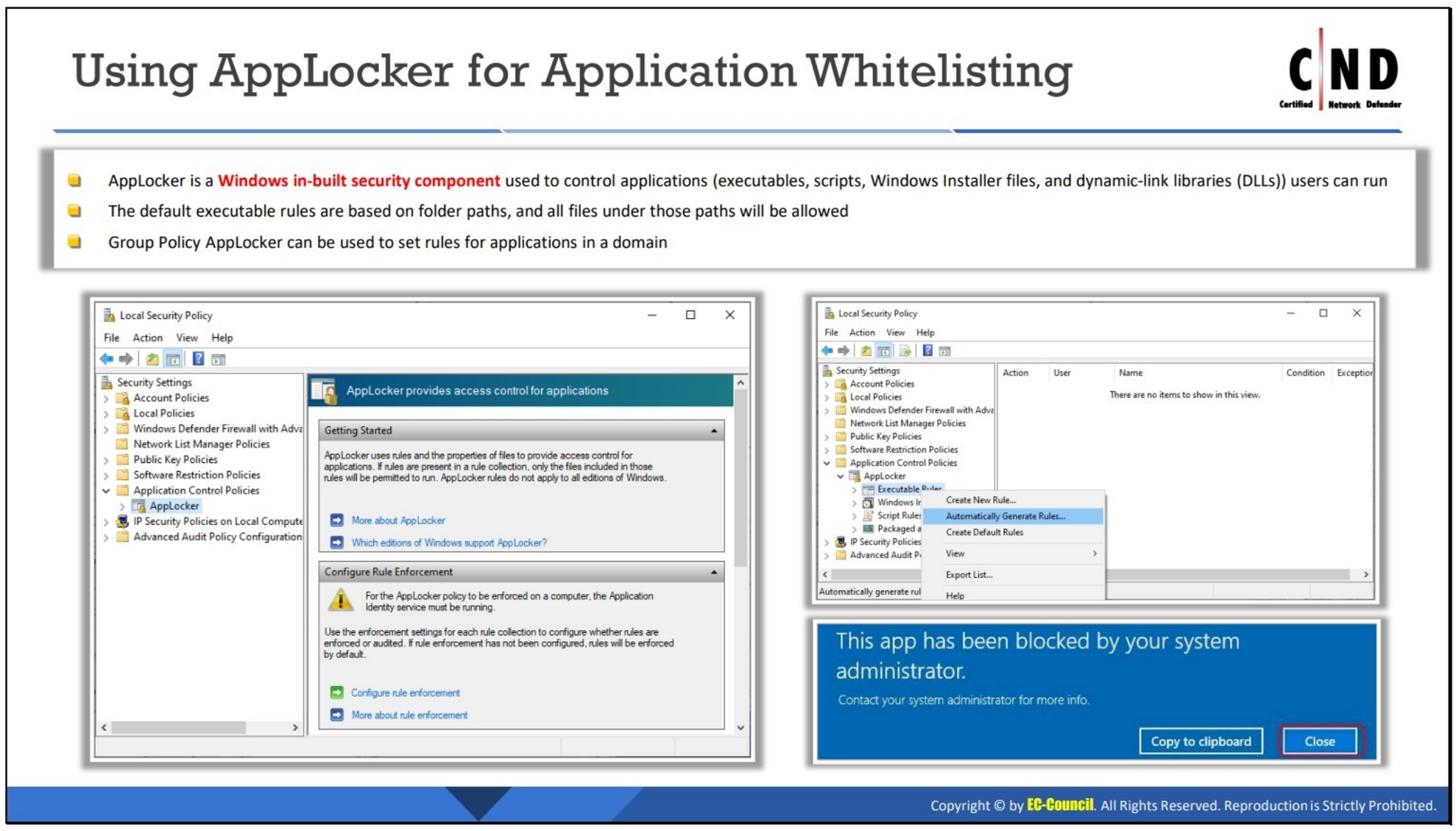


Figure 9.17: Specify the "Security level"

  - Wildcards make it possible to specify what programs should be allowed to run when creating path rules. A question mark (?) is used to denote a single wildcard character, and an asterisk (*) is used to denote a series of wildcard characters.
    - For example, C:\MyFiles*.exe will allow the execution of all files that end with .exe but not executables in subfolders.
  - In the path rule, use environment variables such as %temp%, %windir%, %programfiles%, %appdata%, %systemroot%, and %userprofile%.
  - If a user has the registry key of a software but does not know where it is stored in the system, the user can still create a path rule for the software by creating a registry path rule.
  - If a software is not allowed to run (Disallowed), the user can still run the software by copying it to another location.
  - Set folders such as the Windows folder to Disallowed as allowing them can affect the OS operation.



Using AppLocker for Application Whitelisting

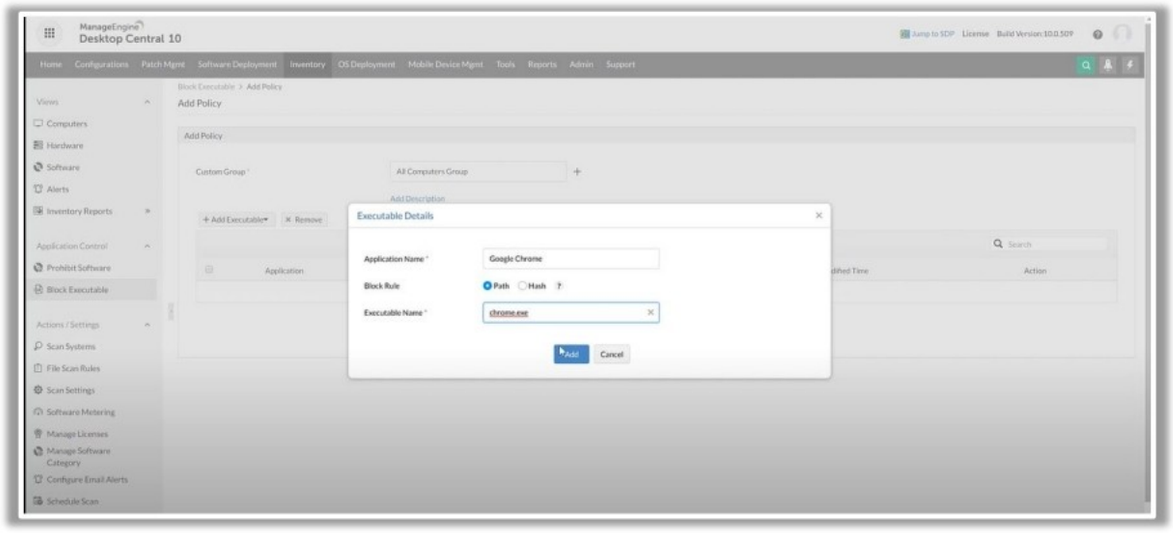
AppLocker is an in-built Windows security component that can be used to control which applications users can run. When AppLocker rules are enforced, apps excluded from the list of allowed apps are prevented from running. The files include executables, Windows Installer files, and dynamic-link library (DLL) files. The default executable rules are based on paths, and all files under those paths are allowed. Group Policy AppLocker is used to set rules for applications in a domain.

Using ManageEngine Endpoint Central for Application Blacklisting

- Endpoint Central helps in restricting the usage of **blacklisted applications** as well as **portable executables**, which can be accessed without installation

Block Executable Features

- Enables network defenders to block the required applications/executables
- Block applications using the following:
  - Path rules
  - Hash values

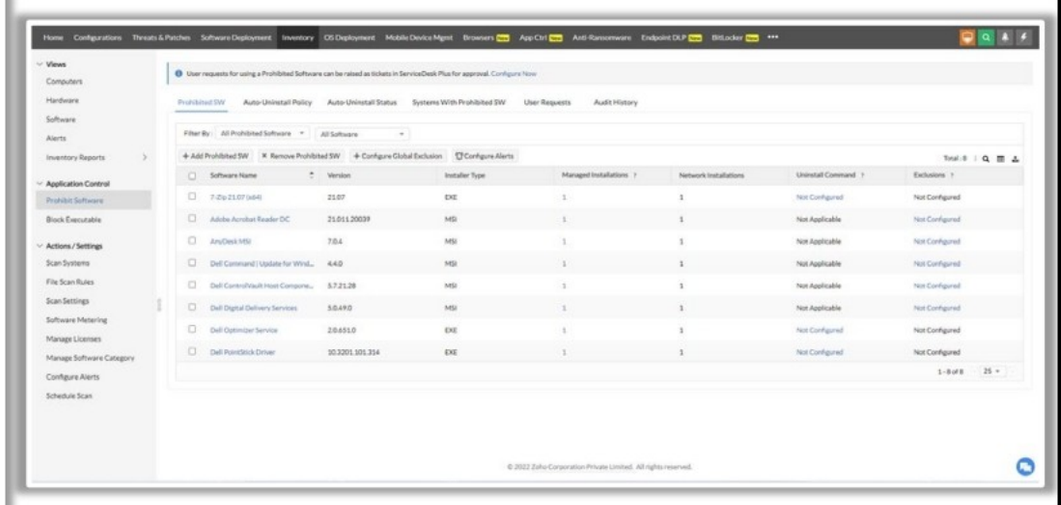


Source: *https://www.manageengine.com*

Using ManageEngine Endpoint Central for Application Blacklisting (Cont'd)

Prohibited Software Feature

- Enables automatic **detection** and removal of **blacklisted** applications (prohibited applications)
- Network defenders can perform the following:
  - Blacklist applications and block blacklisted applications
  - Identify blacklisted application in the network
  - Auto-uninstall the blacklisted applications
  - Exempt computers from the auto-uninstallation routine
  - Generate a report on prohibited software



Source: *https://www.manageengine.com*

Using ManageEngine Endpoint Central for Application Blacklisting

ManageEngine Endpoint Central prevents blacklisted applications based on the organization's policies. It helps in restricting the usage of blacklisted applications as well as portable executables, which can be accessed without installation. The Block Executable and Prohibit Software features of ManageEngine Endpoint Central can be used for Application Blacklisting.

Block Executable Feature

The Block Executable feature enables network defender to block applications/executables. It is possible to block executables in all computers or block them for specific users/computers.

There are two methods to block an executable/application.

- **A path rule** can be used to block all versions of specific applications based on the name of the executable and its file extension.
- **A hash value** can be used to block executables even if they are renamed.

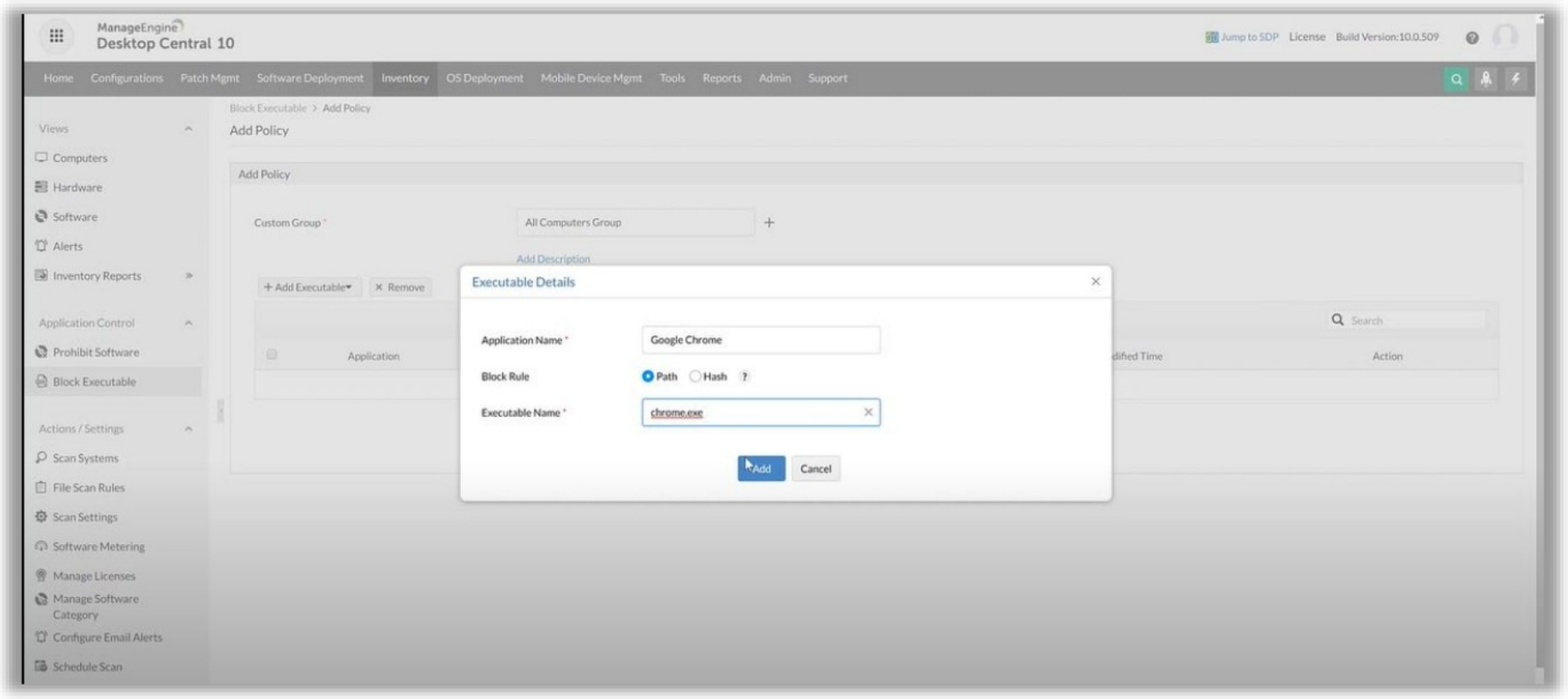


Figure 9.18: Desktop Central

Steps to Block an Executable/Application Using the Block Executable Feature

- **Create a policy**

  Create a policy to block an executable for a specific target. Creating a policy involves selecting the target system, selecting and adding the executable to the list, and applying the block rule as a path or hash. It is possible to create two different policies for a single executable, where one uses a path rule and the other uses a hash rule. The system must be restarted to let the changes take effect.

- **Block executables for all users/computers**

  By default, the Endpoint Central features a custom group that comprises all the managed systems. Choose the **All Managed Computers** group and select the executable to be blocked if the blocking is to be applied to all the managed systems. Create a policy by specifying the target and the executable to be blocked.

- **Block executable for specific users/computers**

  Create a new custom group or use existing custom groups to block an executable only for specific targets (users or computers). Create a policy by specifying the target and the executable to be blocked.

Prerequisites for Blocking Executables/Applications

- **Enable Local Group Policy on the target machine**
  - Go to **Run.**
  - Enter **gpedit.msc.**
  - Click **Group Policy**.
  - Click **Turn Off Local Group Policy Objects Processing** in the right pane.

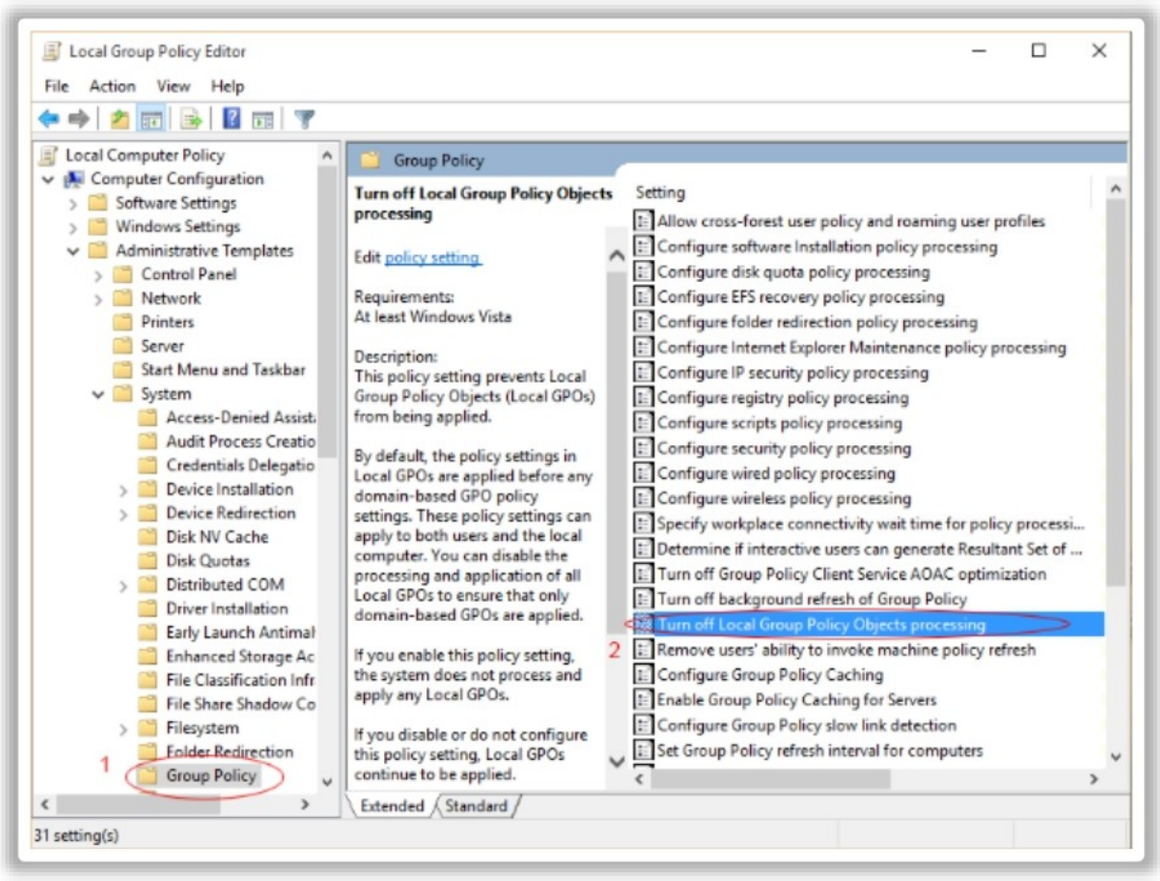


Figure 9.19: Select "Turn Off Local Group Policy Objects Processing" Policy Setting

- Choose **Not Configured** and click **OK.**

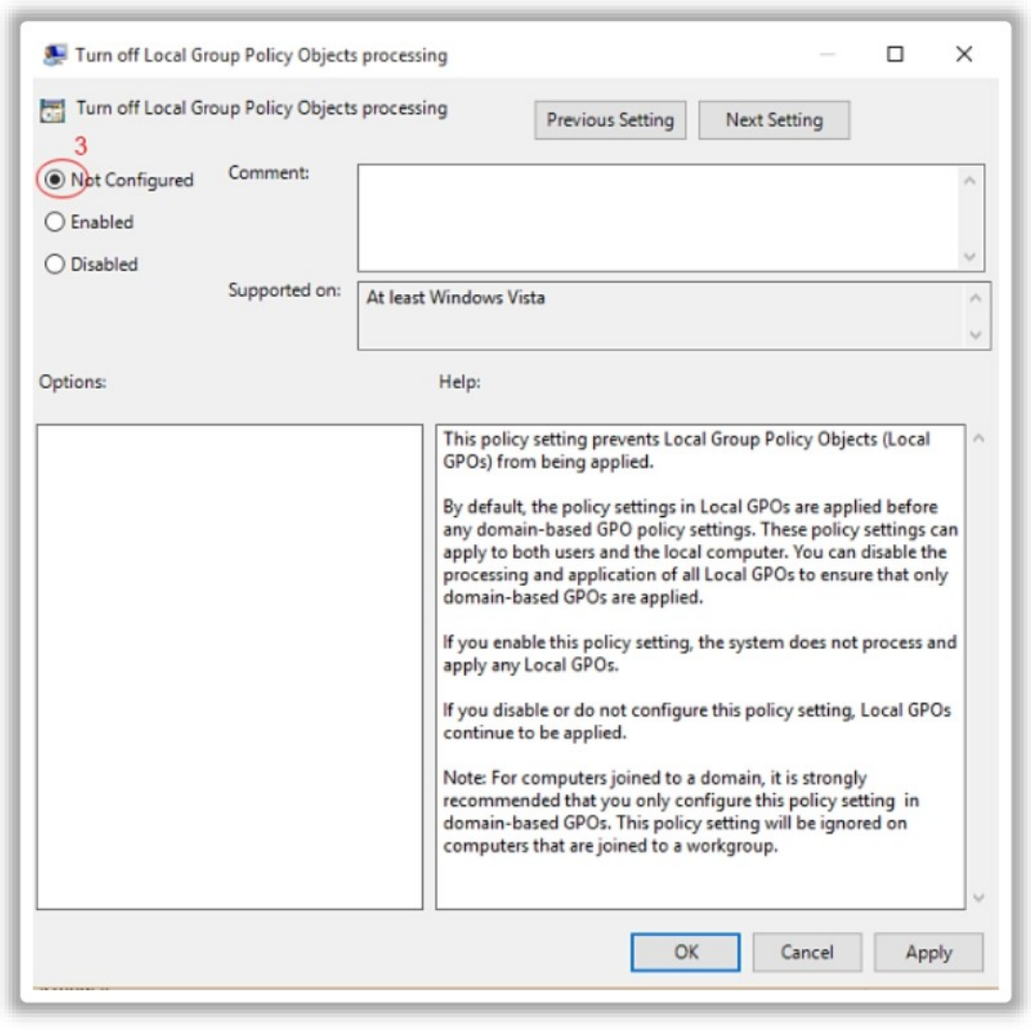


Figure 9.20: Blocking Applications

- **Enable Local Group Policy on the target system**
  - Right-click **Local Computer Policy** in the **Local Group Policy Editor**, select **Properties**, and check **Disable Computer Configuration Settings**.

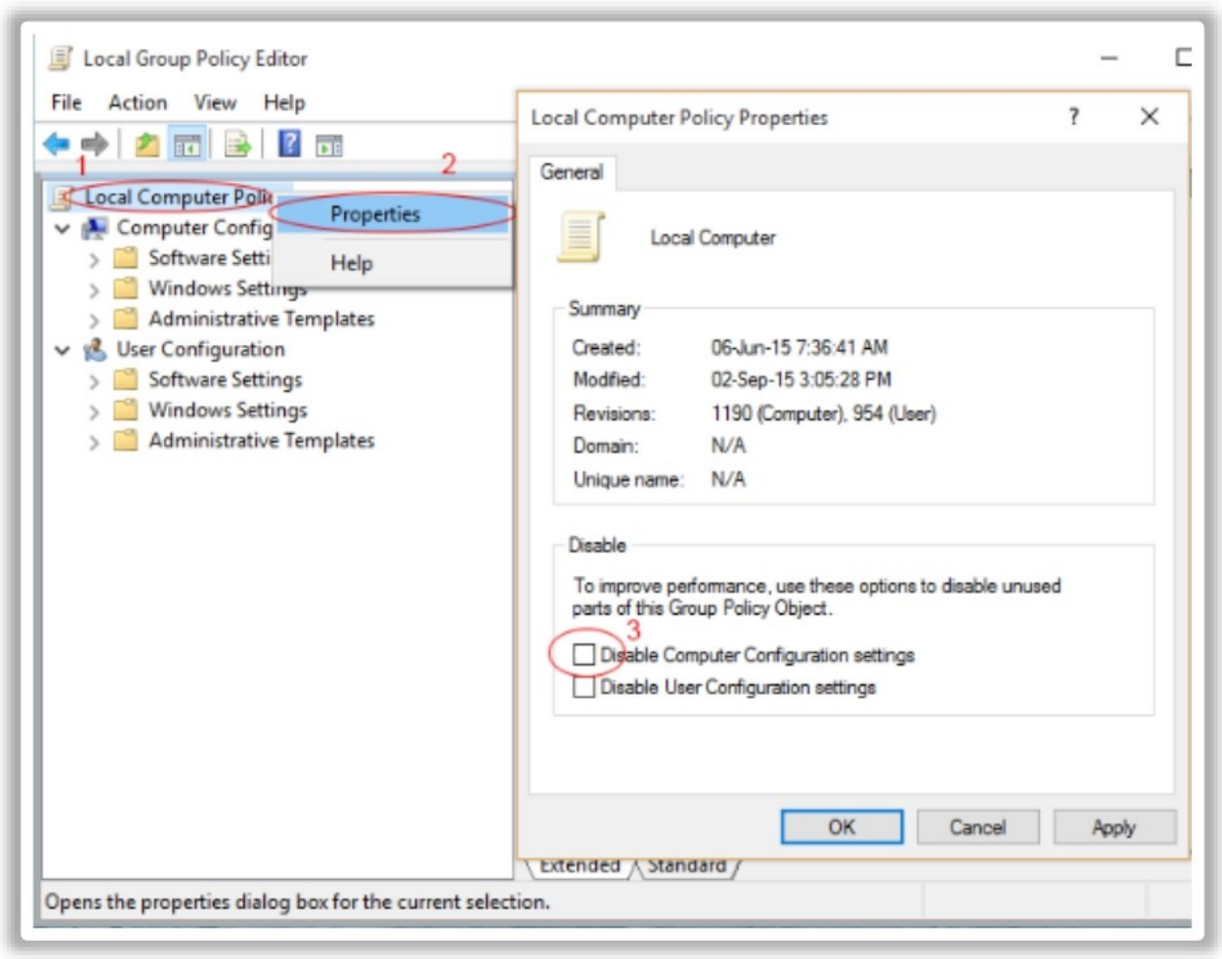


Figure 9.21: Disabling Computer Configuration Settings

- **Set the default security policy as "Unrestricted"**
  - Go to **Local Computer Policy->Windows Settings->Security Settings->Software Restriction Policies.**
  - Click **Security levels** and double-click **Unrestricted** in the right-side pane.

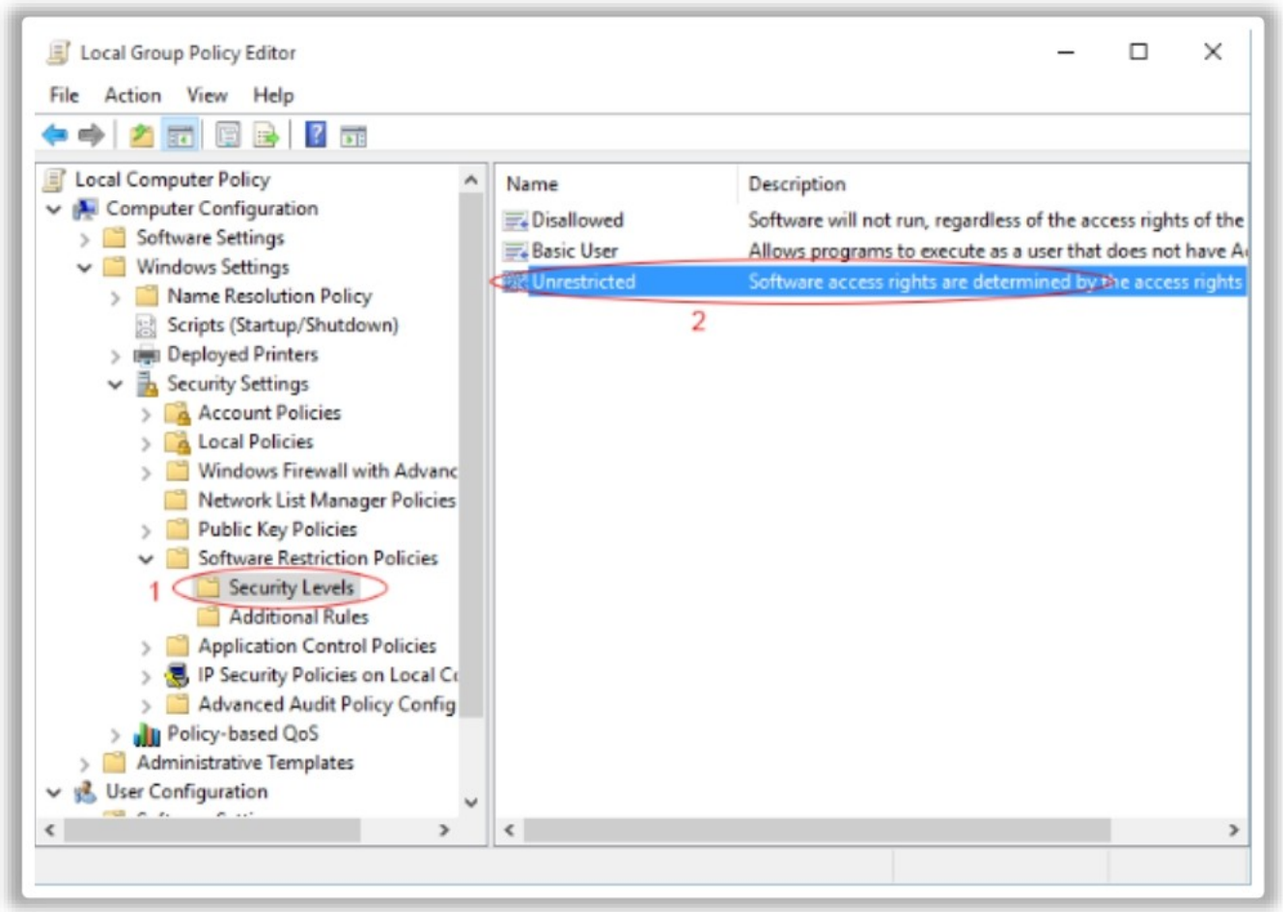


Figure 9.22: Setting Security Levels

- Click **Set as Default** in **Unrestricted Properties** window and click **OK**.

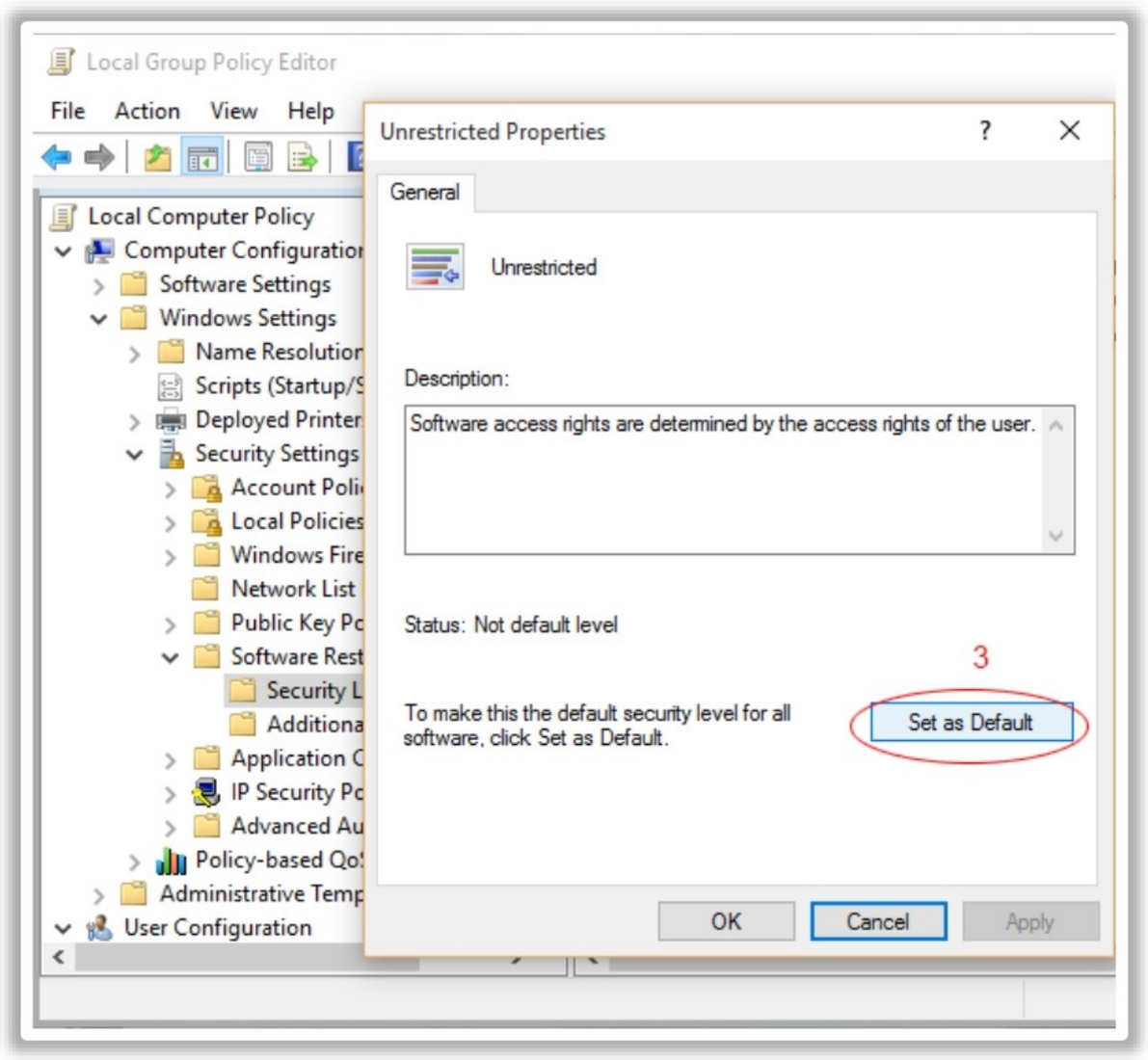


Figure 9.23: Setting the Properties of "Unrestricted" Security Level

- **Enable Local Group Policy for the administrator**
  - Go to **Local Computer Policy->Computer Configuration->Windows Settings->Security Settings->Software Restriction Policy.**
  - Double-click **Unrestricted** in the right-side pane.
  - Click **Set as Default** in **Unrestricted Properties** window and click **OK**.

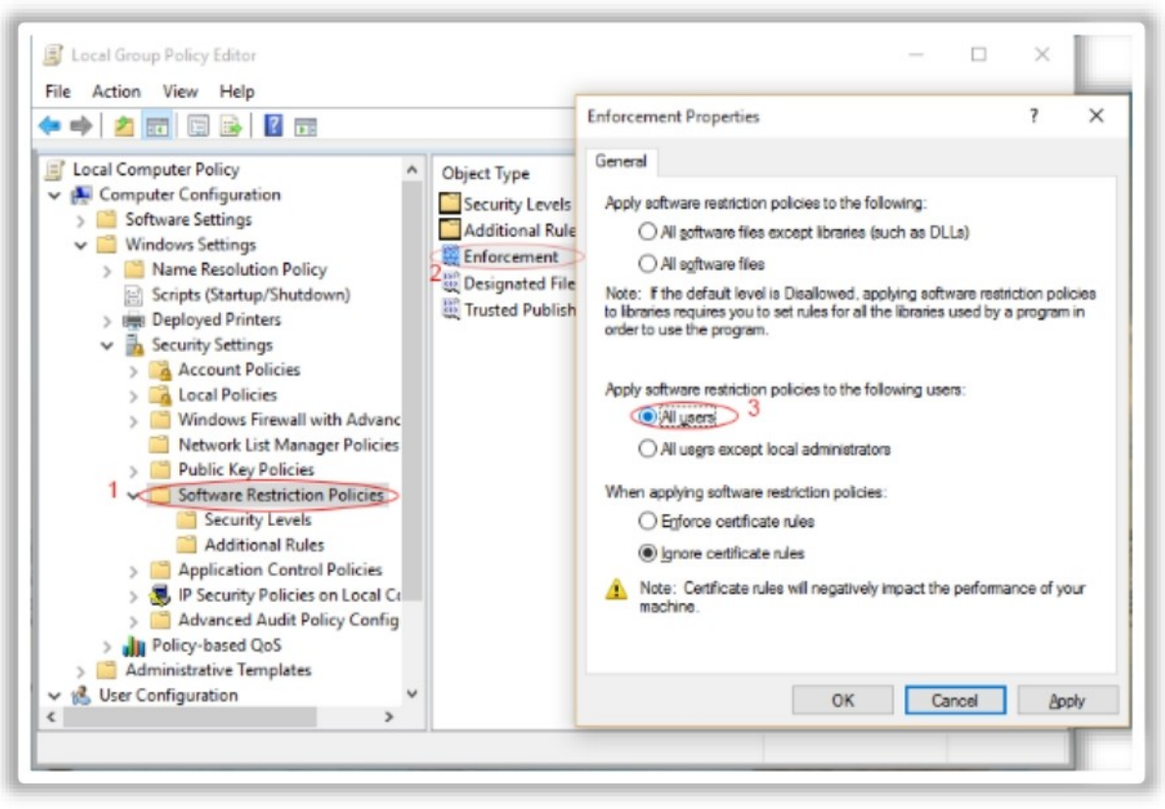


Figure 9.24: Applying Software Restriction Policies to All Users

- Double-click **Enforcement** and select **All Users.**

- Click **OK**.

Prohibit Software Features

Endpoint Central's Prohibited Software feature or module fully automates the detection and removal of prohibited applications.

Steps to Prohibit Applications Using Prohibit Software Feature

- **Add prohibited software to a list**
  - Navigate to **Prohibit Software** from the **Inventory** tab to view the details of all the software that have already been prohibited.
  - Click **Add Prohibited Software**. The dialog **Add Prohibited Software** lists all the software detected in the managed systems. Scan the OS at least once to know the details of the software here.

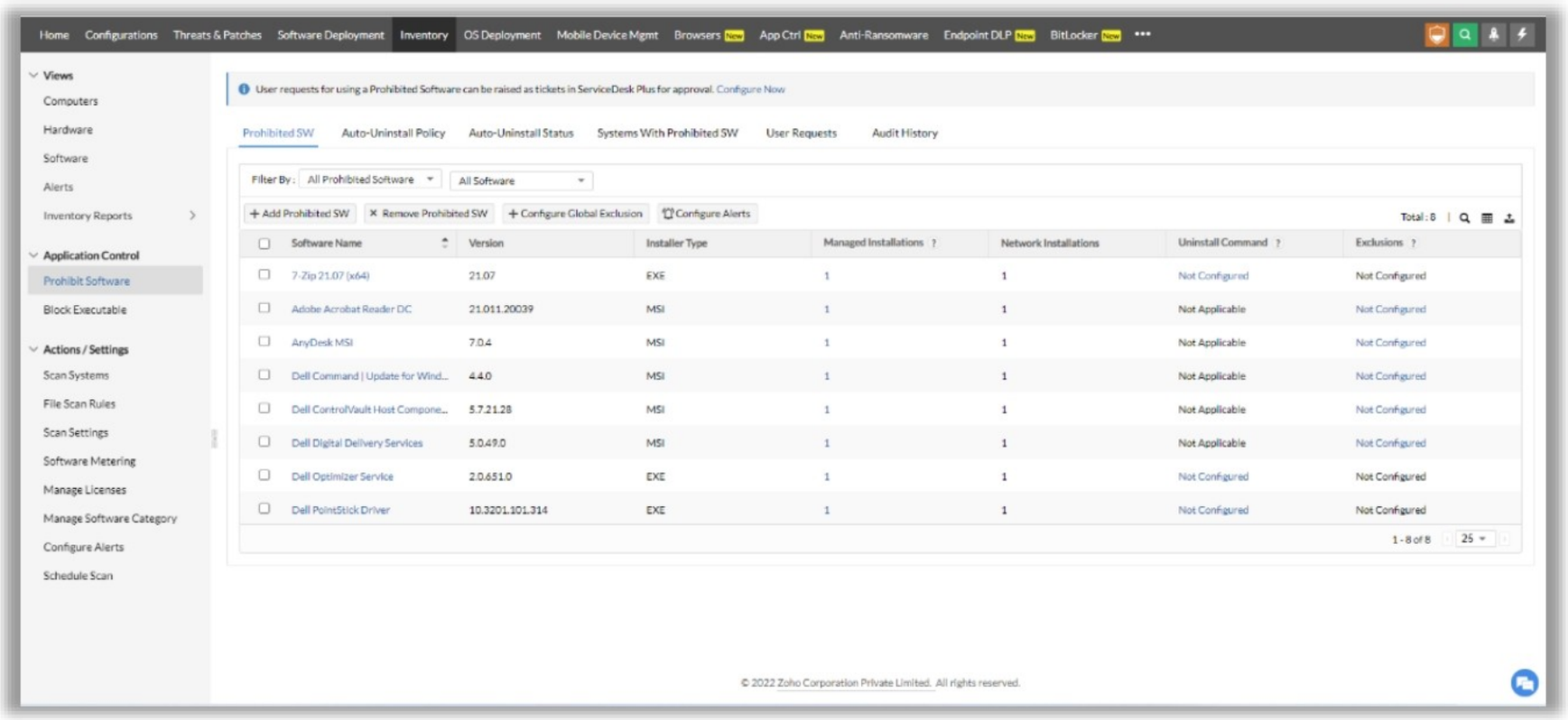


Figure 9.22: Add Available Software to Prohibited Software

  - Select the software and move it to the **Prohibited List** to be blacklisted. Adding a **Software Group** under the **Prohibited Software List** blacklists all the software in that group.
  - Click **Update** to confirm the addition of the software to the prohibited list.
- **Auto-uninstall the identified prohibited application**

  The steps below should be followed to configure the auto-uninstall policy to automatically uninstall prohibited software detected on the system.
  - Select the **Auto-Uninstall Policy** tab and check **Enable Automatic Uninstallation**.
  - Specify the maximum number of software that can be uninstalled from a system during the subsequent refresh cycle.

    Note: Increasing the number of software will cause high CPU usage during uninstallation. If the prohibited software count is detected to exceed the allowed

maximum number of software to be uninstalled, the remaining software will be uninstalled during the subsequent startup.

- Check **Notify User before Uninstalling** and specify a custom message to prompt the user before the software uninstallation.

  Note: The user is given an alert message during login and whenever the agent identifies prohibited software. This functionality is applicable only if the **Notify User Settings** is configured.

- Specify a number for the wait window for software uninstallation if the software are to be removed a few days after detection.

- Click **Save**.

By default, the auto-uninstallation option is available for **.msi** and **.exe** applications and requires silent switches.

- **Steps to auto-uninstall .exe-based software**

  - Select the **Prohibited SW** tab and click **Not Configured link** under **Uninstall command** against the **.exe** application.
  - The **Add/Edit Uninstall Command** window pops up.
  - Choose any one of the following options:
    - **Pre-fill Uninstall Command**—This command fetches the uninstall command of the Add/Remove Programs application and displays it. Specify only the silent switch.
    - **I Will Specify Myself**—Enter the uninstall command and silent switch manually. Test the uninstallation command manually to verify its correctness.
  - Click **Save.**
  - Verify the status in the **Auto Uninstallation Status** tab.

    Note: The uninstallation occurs based on the configured auto-uninstall policy.

  - Select **Detailed View** under **Auto Uninstallation Status** to view the status and remarks.

    Note: Uninstalling a software by configuring the auto-uninstall policy does not prevent users from installing a software. Once a software is installed, it will get uninstalled automatically.

- **Exempt computers from auto-uninstallation routine**

  The following are the steps to exempt computers from the auto-uninstallation routine to allow the usage of prohibited software for certain users:

  - Navigate to **Prohibit Software** from the **Inventory** tab to view the details of all the software that are already prohibited.

- Select the checkbox corresponding to the specified software and click the link under the **Exclusions** column to open the **Add Exclusions** dialog.
- Select whether to exclude **custom groups** or **computers**, select **the groups/computers**, and move them to the **Excluded** list.
- Click **Save**.

- **Approve requests to use prohibited software**
  - Select the specific prohibited software from the list of prohibited software from the agent tray icon.
  - Handle requests from **Desktop Central web console->Inventory->Prohibit Software->User Requests**.

  Users are allowed to install and use the prohibited software they request once the request is approved.

- **Notify admin and end users when prohibited software is detected**

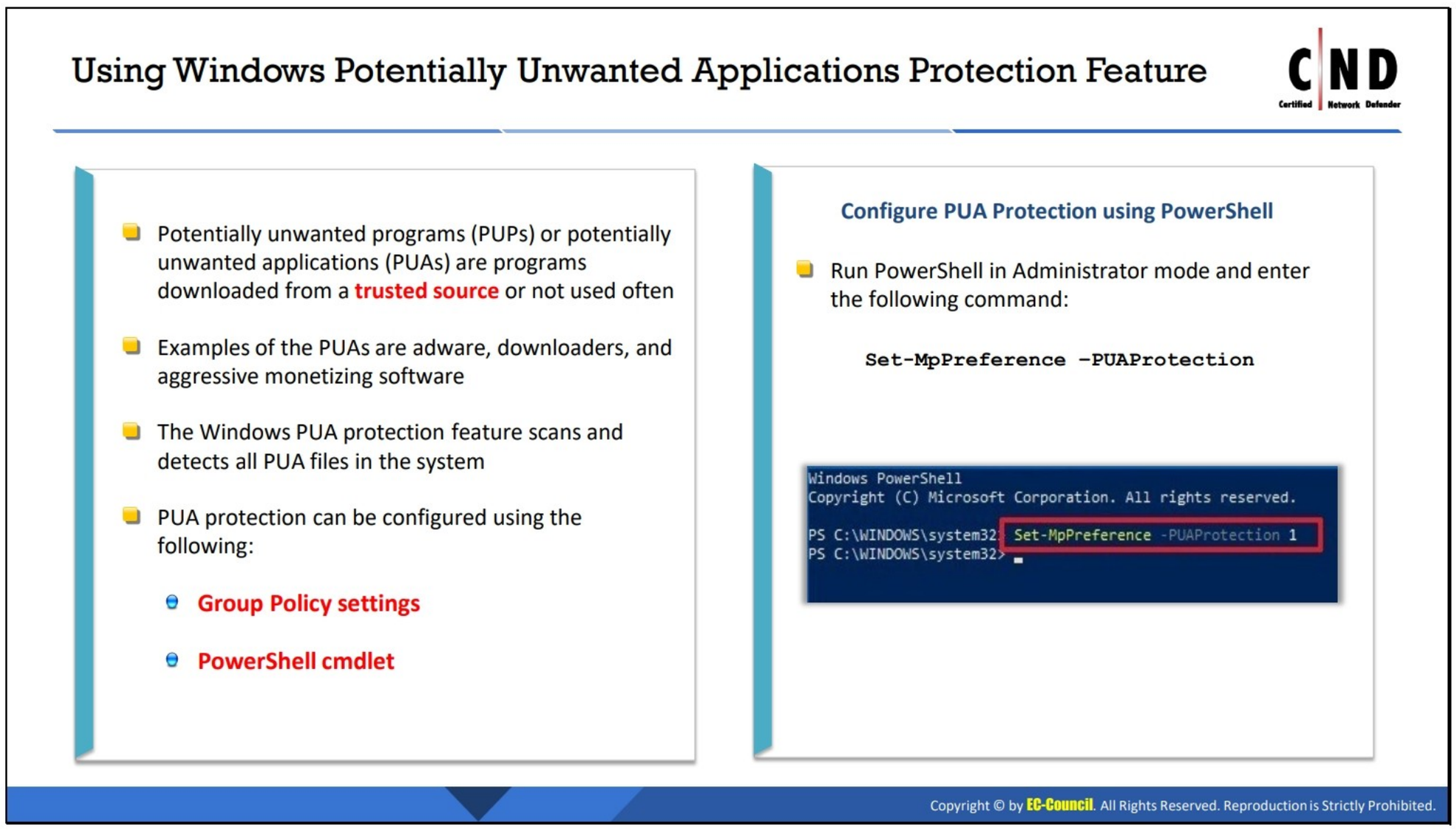
  The following are the steps to notify the admin and end users when prohibited software is detected:

  - Navigate to the **Inventory** tab.
  - Click **Configure E-mail Alerts** in the left pane under Actions/Settings.
  - Under **Notifications**, specify when the notifications should be sent, and configure alerts based on requirements.
  - Specify the email address or addresses to which the notifications must be sent.
  - Click **Save.**

- **Generate a report on prohibited software**

  The following are the steps to generate a report on prohibited software to find the computers in the network using applications at any point of time:

  - Select the **Inventory** tab.
  - Choose the **Prohibited Software** link under the **Software Reports** category by moving the mouse over **Inventory Reports**.



Using Windows Potentially Unwanted Applications Protection Feature

Potentially unwanted programs (PUPs) or potentially unwanted applications (PUAs) are programs downloaded from a trusted source but not used often. The PUA feature allows Windows Defender to detect and block certain unwanted apps. These apps are not categorized as threats, but they can increase the chances of malware attacks and decrease the system's performance.

Examples of the PUAs are adware, downloaders, and aggressive monetizing software.

The Windows PUA protection feature scans and detects all PUA files in the system.

The methods to configure the Windows PUA protection are detailed below.

- **Steps to use Group Policy Settings**
  - In the **Local Group Policy Editor**, navigate to **Computer Configuration\Administrative Templates\Windows Components\Windows Defender Antivirus**.

- Double-click the policy **Configure detection for potentially unwanted applications** in the right-hand side pane to edit it.

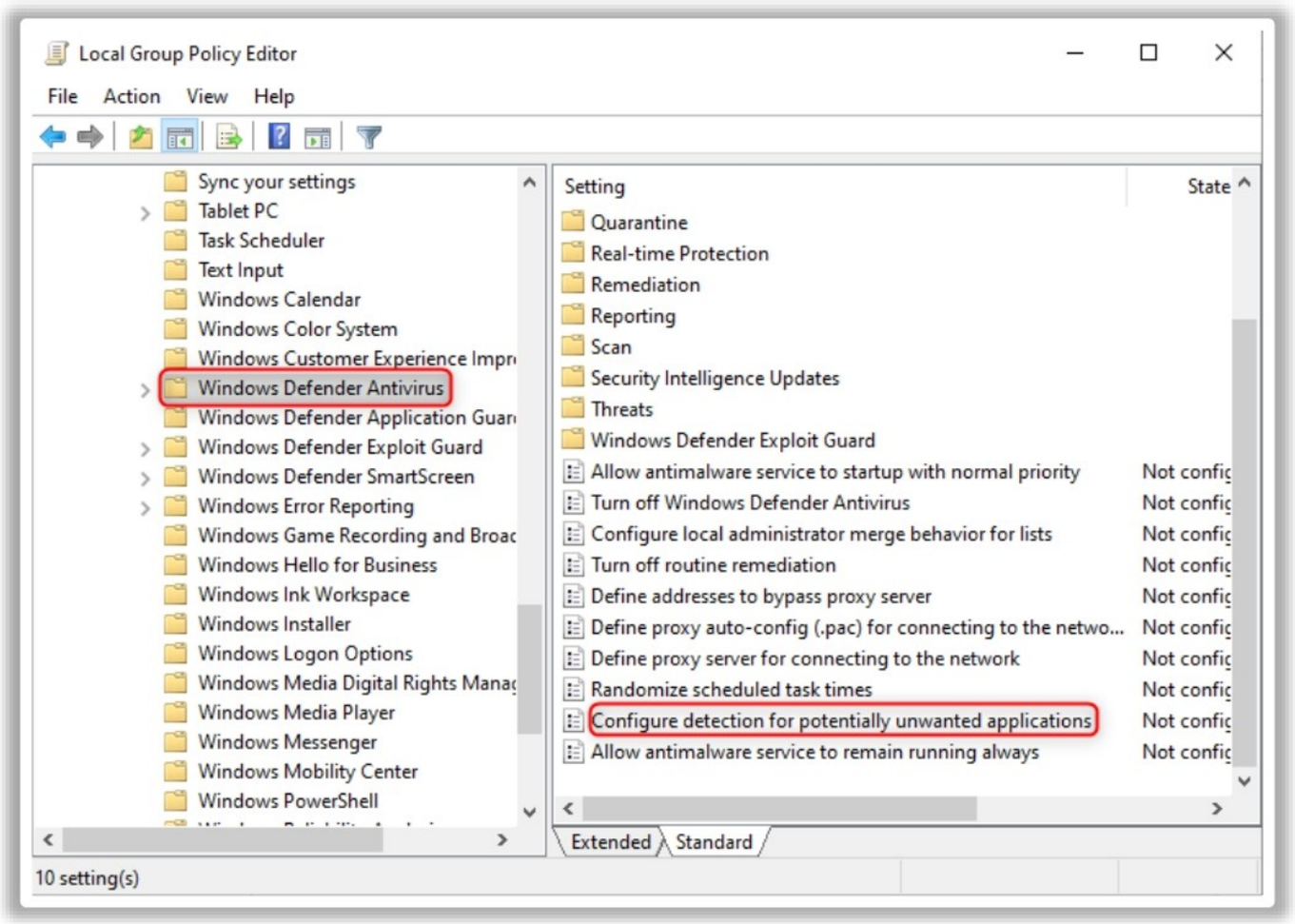


Figure 9.25: Selecting "Configure detection for potentially unwanted applications" Policy Setting

- Select **Enabled** and select **Block** under the **Options** section.
- Click **Apply** and **OK**.

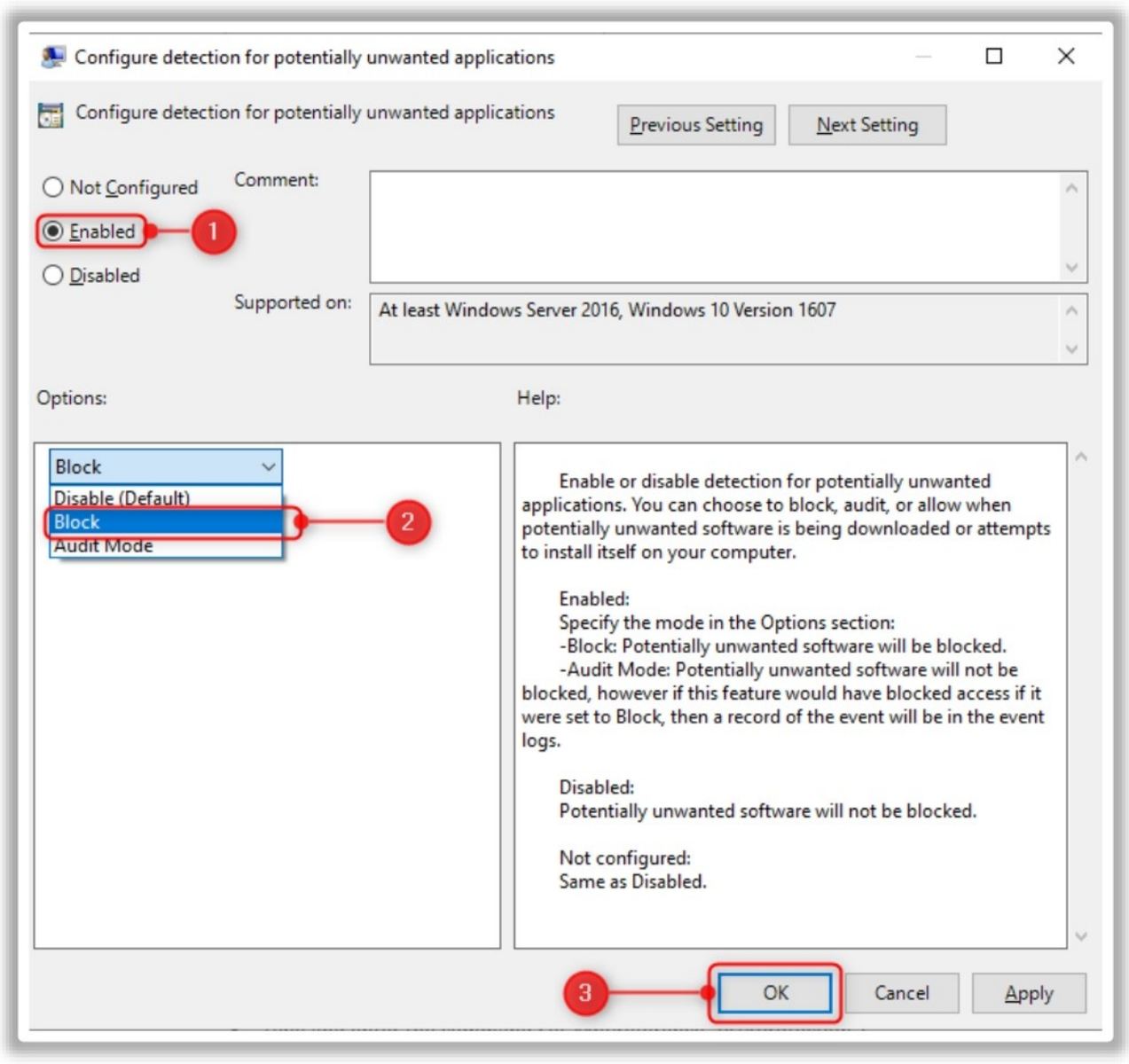


Figure 9.26: Enabling "Configure detection for potentially unwanted applications" Policy Setting

- **Steps to use PowerShell cmdlet**
  - Run **Windows PowerShell** as an administrator.
  - Enter the command Set-MpPreference -PUAProtection 1

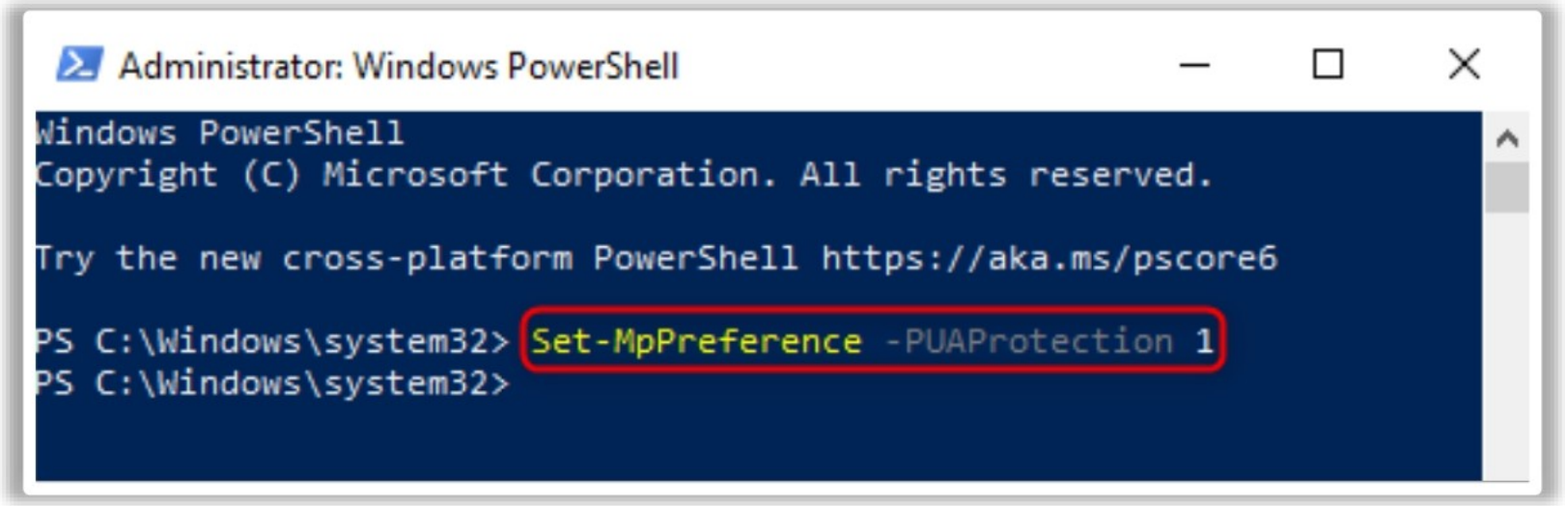
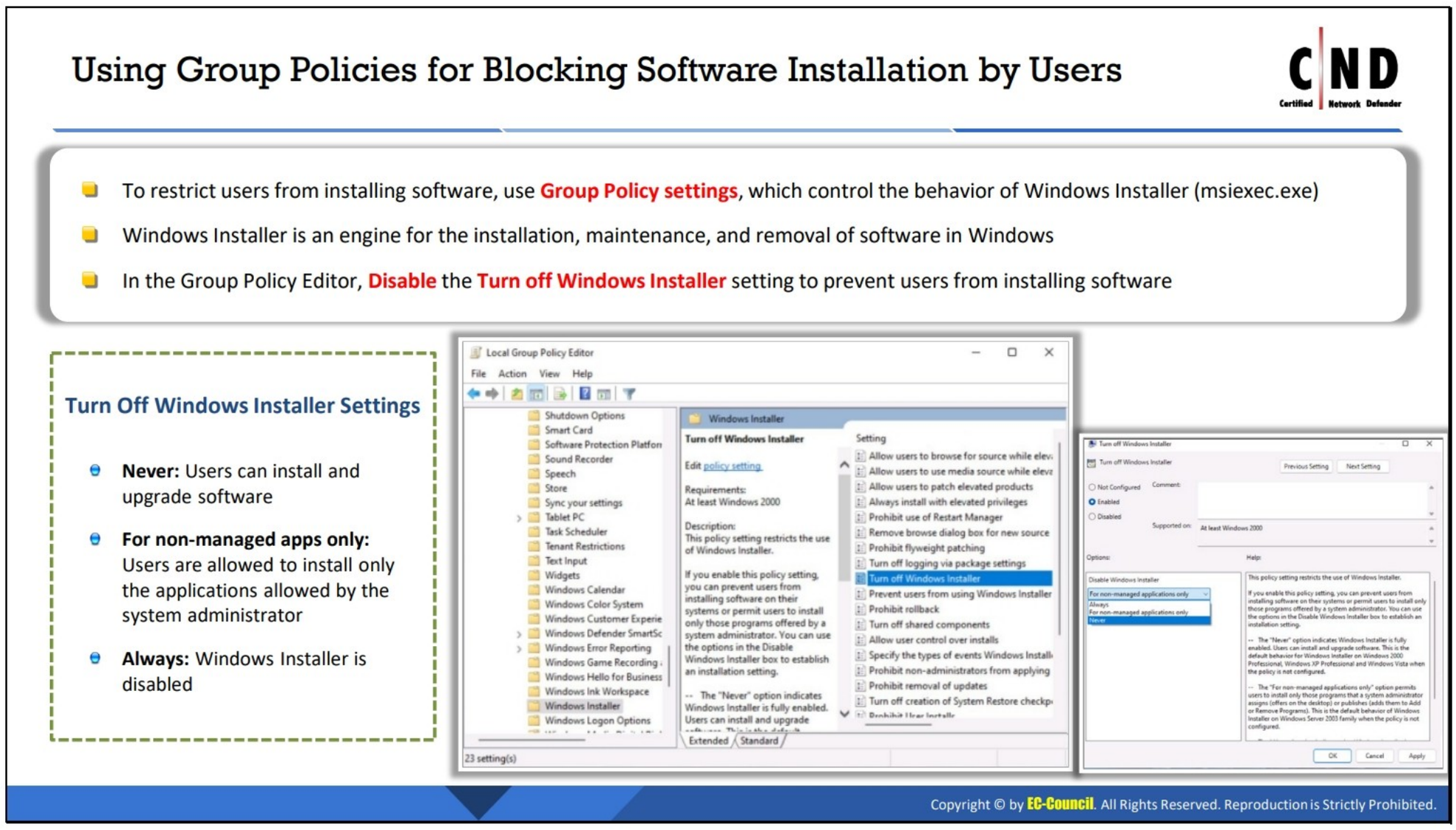


Figure 9.27: Using PowerShell for using PUA Features

Now, Windows Defender is configured to detect and block suspicious or unwanted applications.



Using Group Policies for Blocking Software Installation by Users

Group Policy Settings can control the behavior of the Windows Installer. The Windows Installer/Microsoft Installer (*msiexec.exe)* is an engine for the installation, maintenance, and removal of programs.

Steps to Block Unwanted Installations using Group Policy

- Go to **Computer Configuration->Administrative Templates->Windows Component->Windows Installer** in **Group Policy Management Editor**.
- Double-click the **Turn off Windows Installer** policy in the right pane.

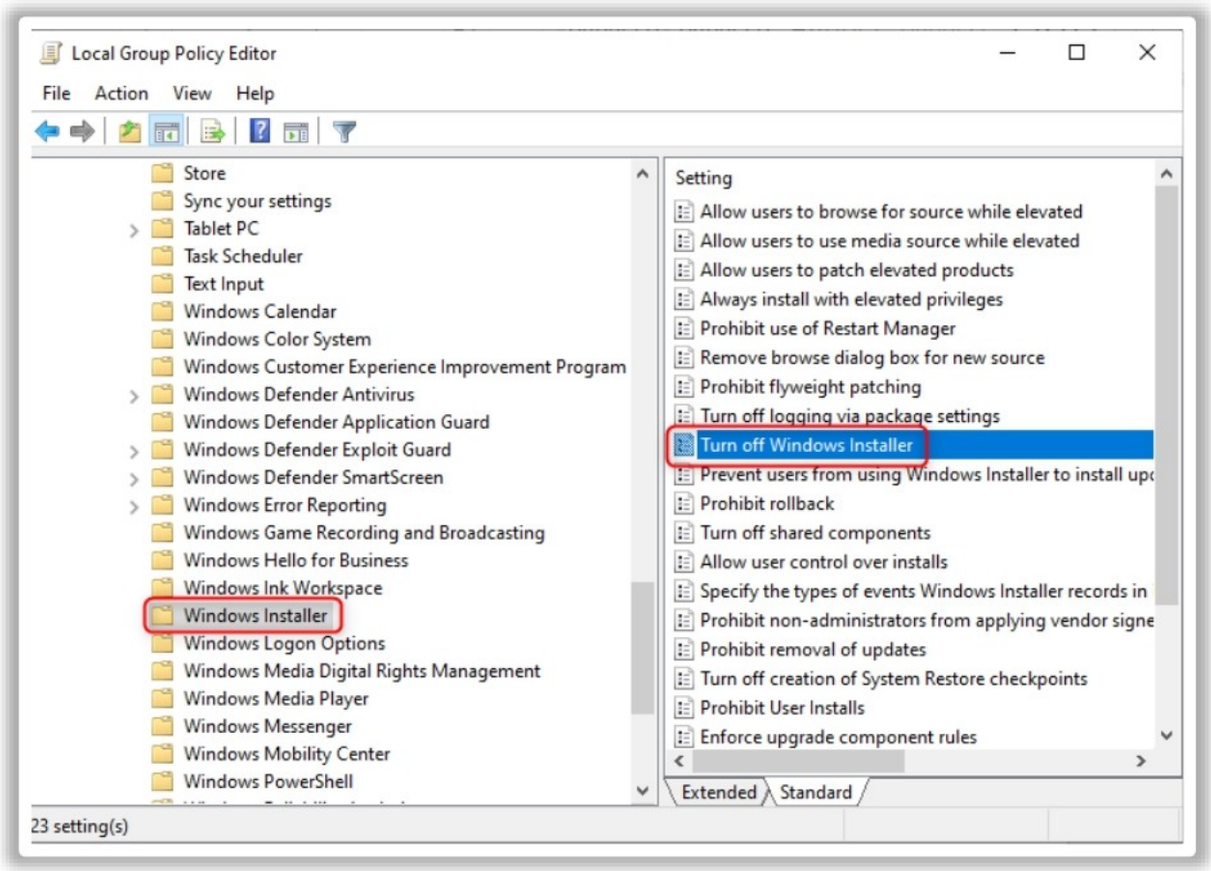


Figure 9.28: Selecting “Turn off Windows Installer” Policy Setting

- Click **Enabled** and configure **Disable Windows Installer**. **Turn off Windows Installer** can be configured to the following settings:
  - **Never**—Users can install and upgrade software.
  - **For non-managed apps only**—Users are allowed to install only the applications allowed by the system administrator.
  - **Always**—Disables Windows Installer.
- Select **Always** to disable **Windows Installer** and click **Apply** and **OK**.

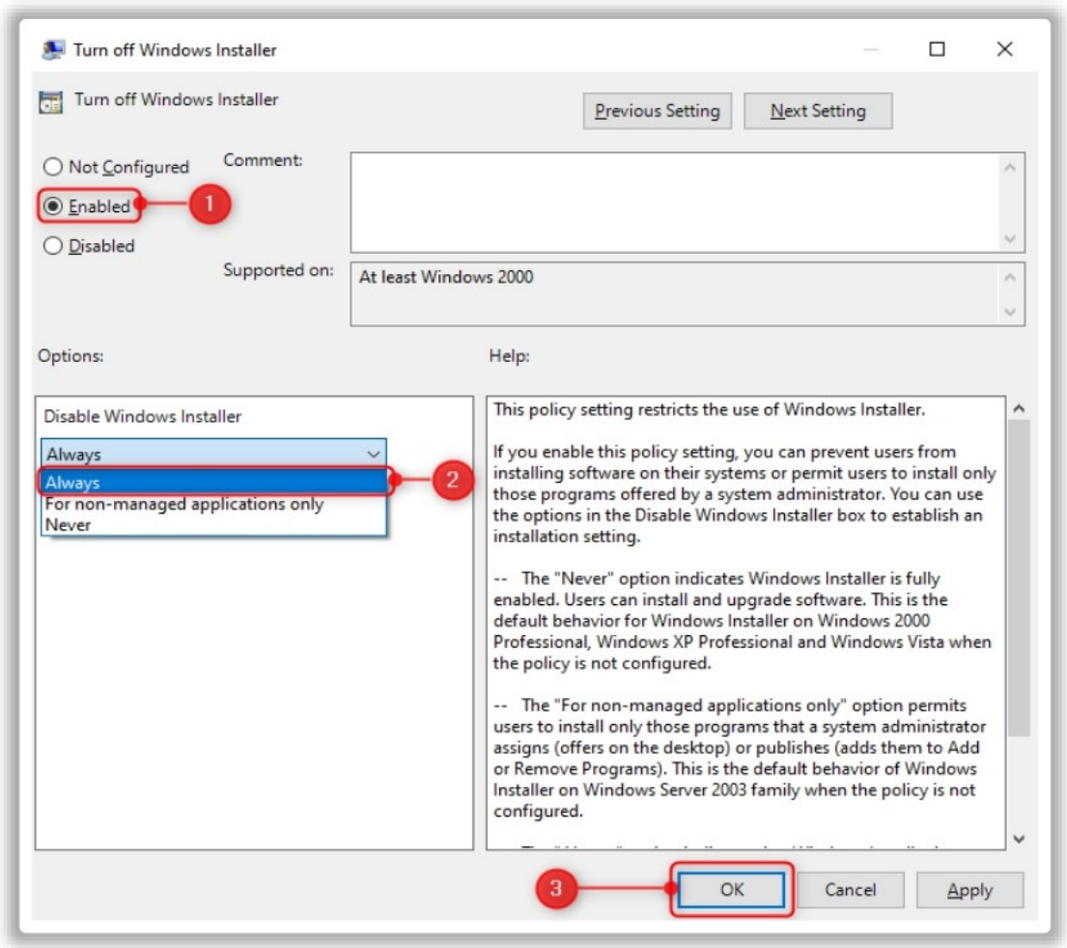


Figure 9.29: Enabling "Turn off Windows Installer" Policy Setting

Block Windows Installer (msiexec.exe) using the Block Specific Application Group Policy

- Navigate to **Group Policy Editor->User Configuration->Administrative Templates->System**.

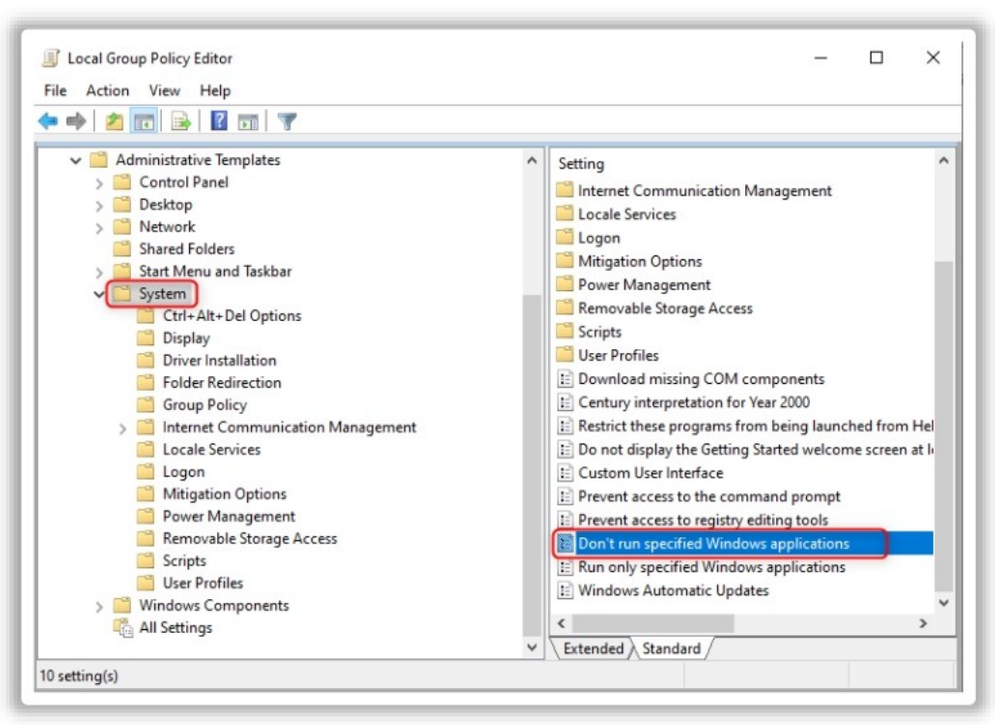


Figure 9.30: Selecting "Don't run specified Windows applications" Policy Setting

- Double-click **Don't run specified Windows applications** in the right-hand side pane.
- Select **Enabled**.
- Under **Options**, click **Show**.
- Enter the path of the application to disallow.

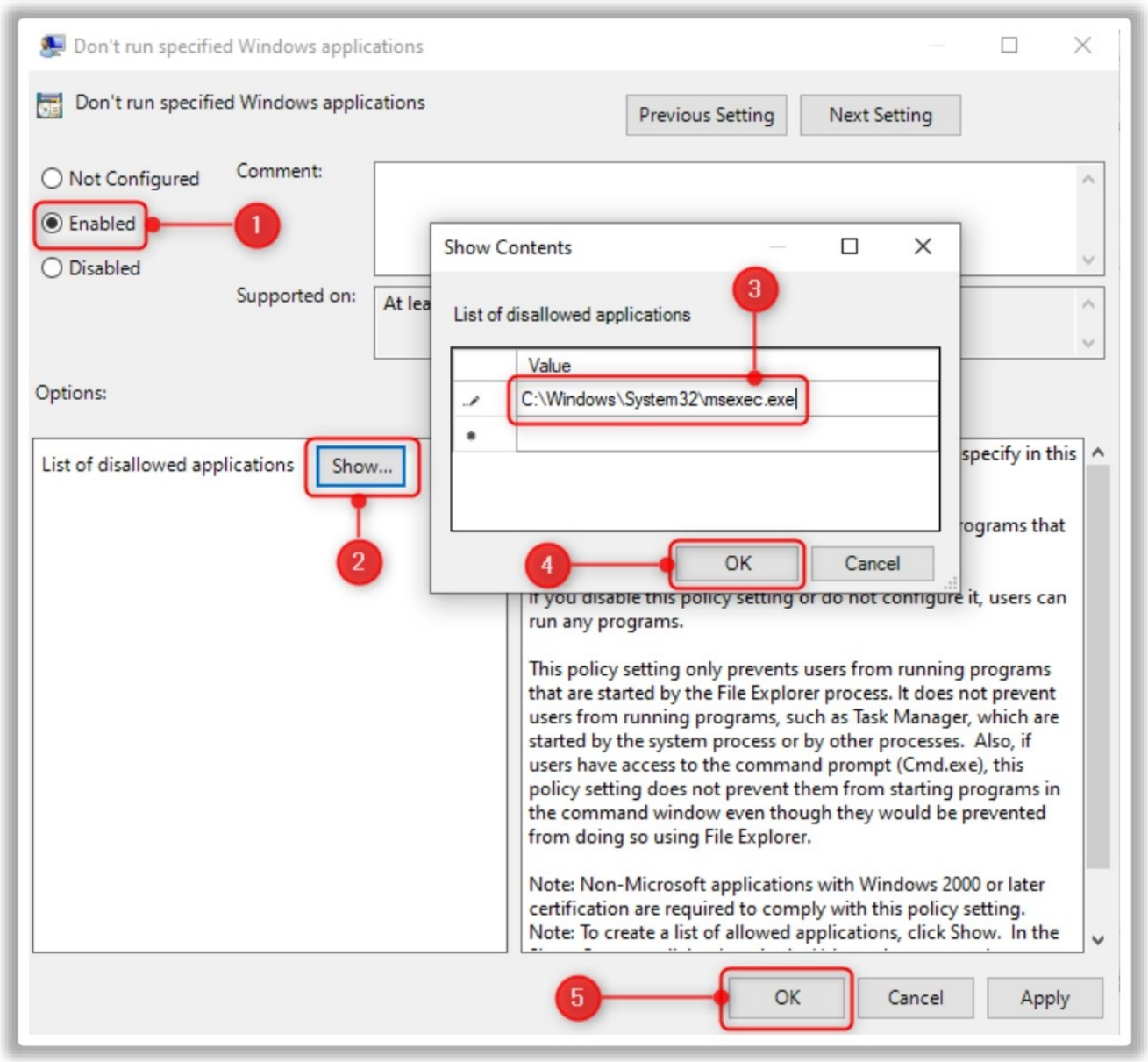
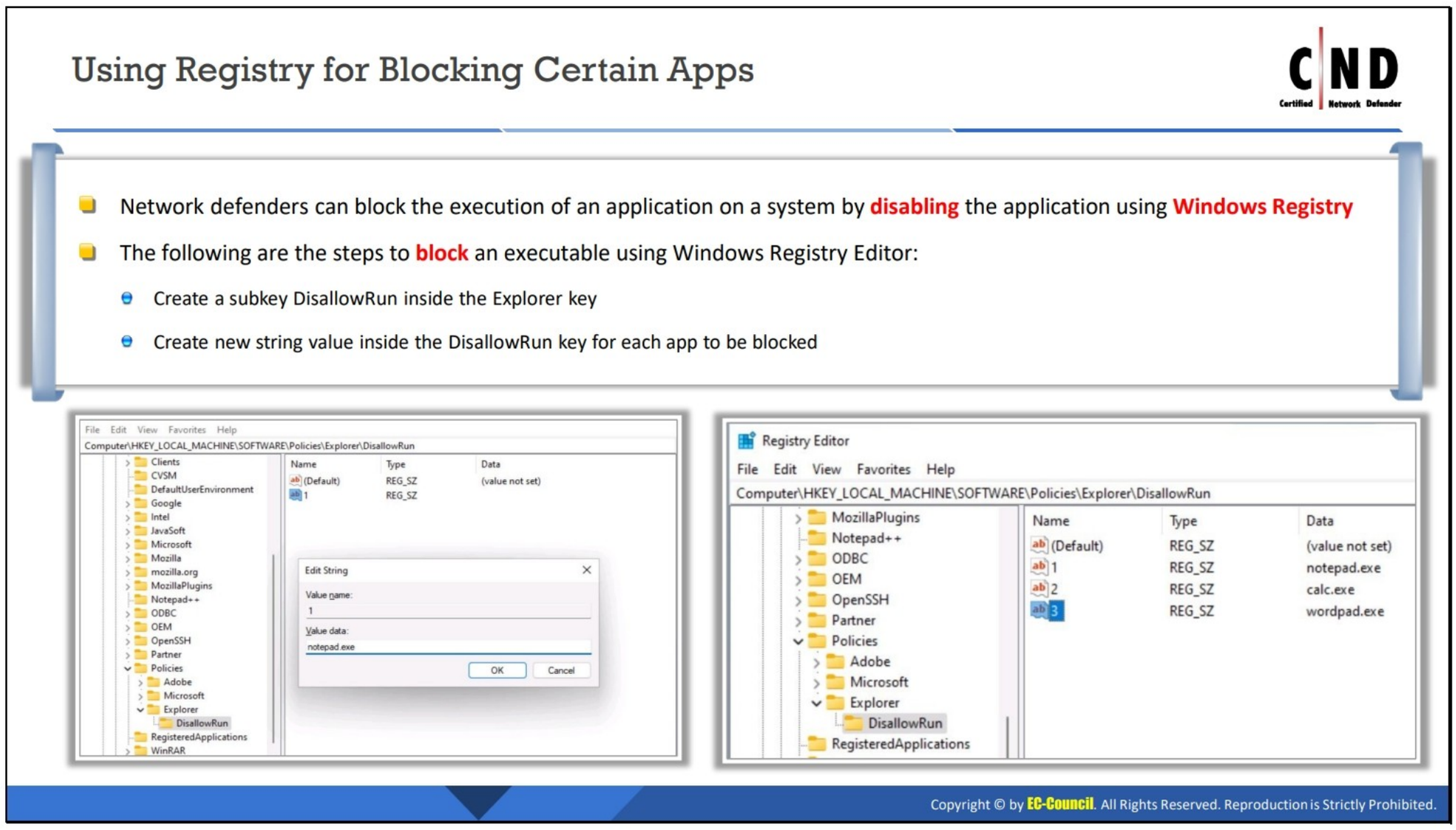


Figure 9.31: Enabling "Don't run specified Windows applications" Policy Setting

- Click **OK**.

Note: The Block Specific Application setting only prevents users from running programs that are started by the Windows Explorer process.



Using Registry for Blocking Certain Apps

Network defenders can block the execution of an application on a system by disabling the application using the Windows Registry Editor.

The following are the steps to block an executable using the Windows Registry Editor:

- Run the **Registry Editor (regedit)** as an administrator.

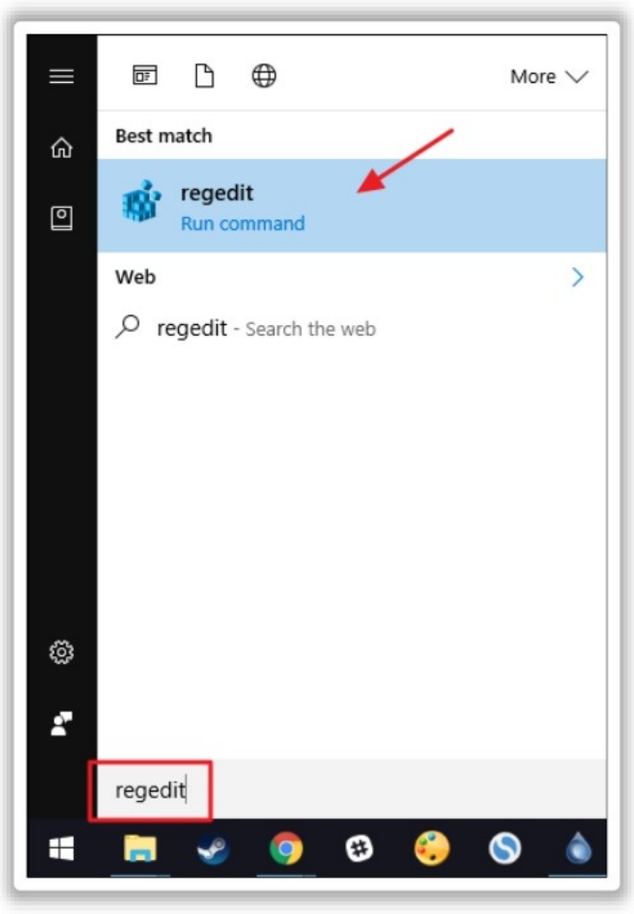


Figure 9.32: Running Registry Editor

- Navigate to the following key:

  HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies

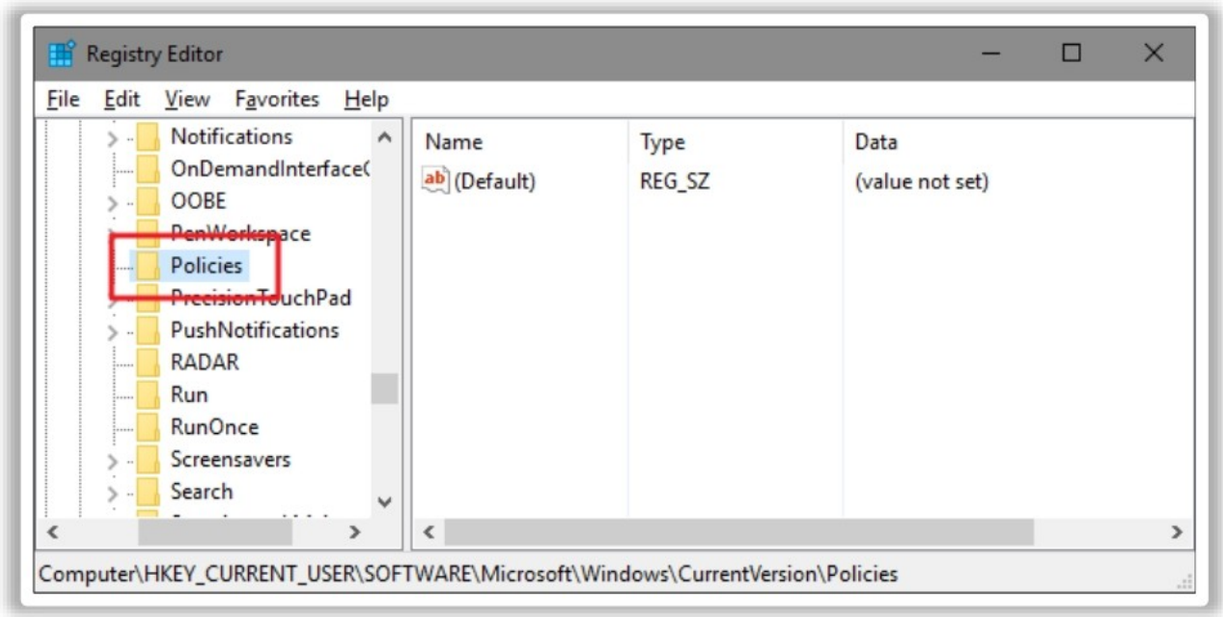


Figure 9.33: Selecting “Policies” Key

- Right-click **Policies key,** choose **New Key**, and name the new key “Explorer.”

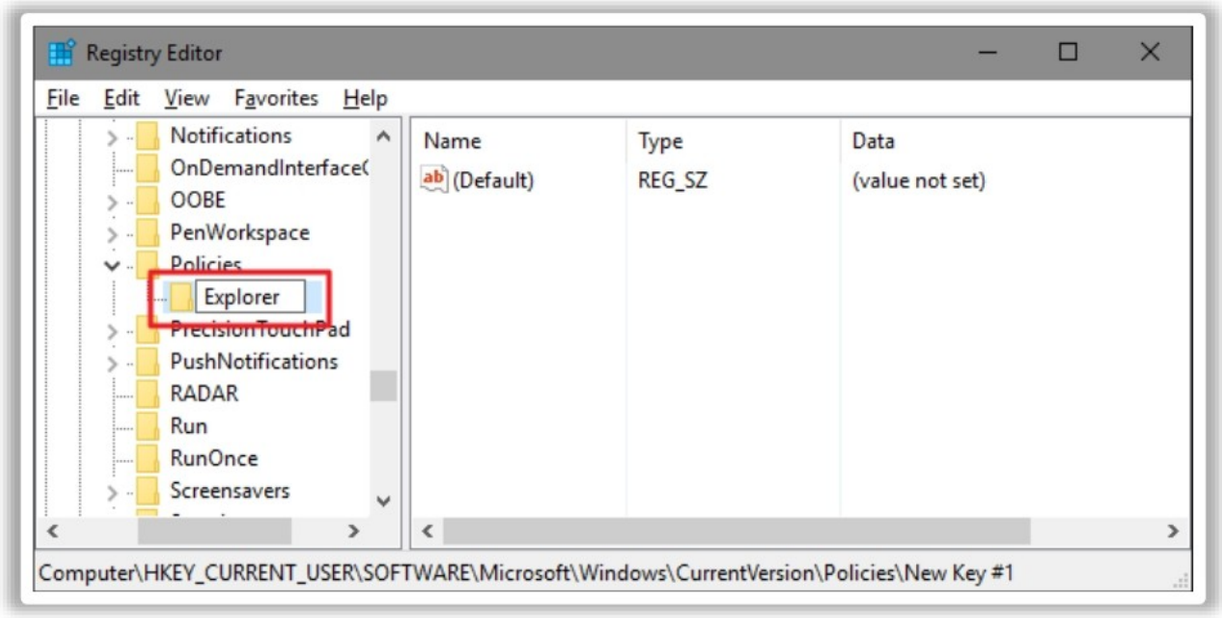


Figure 9.34: Creating a New Key for ‘Polices” Key

- Right-click the **Explorer** key and choose **New DWORD (32-bit) value**. Name the new value **DisallowRun**.

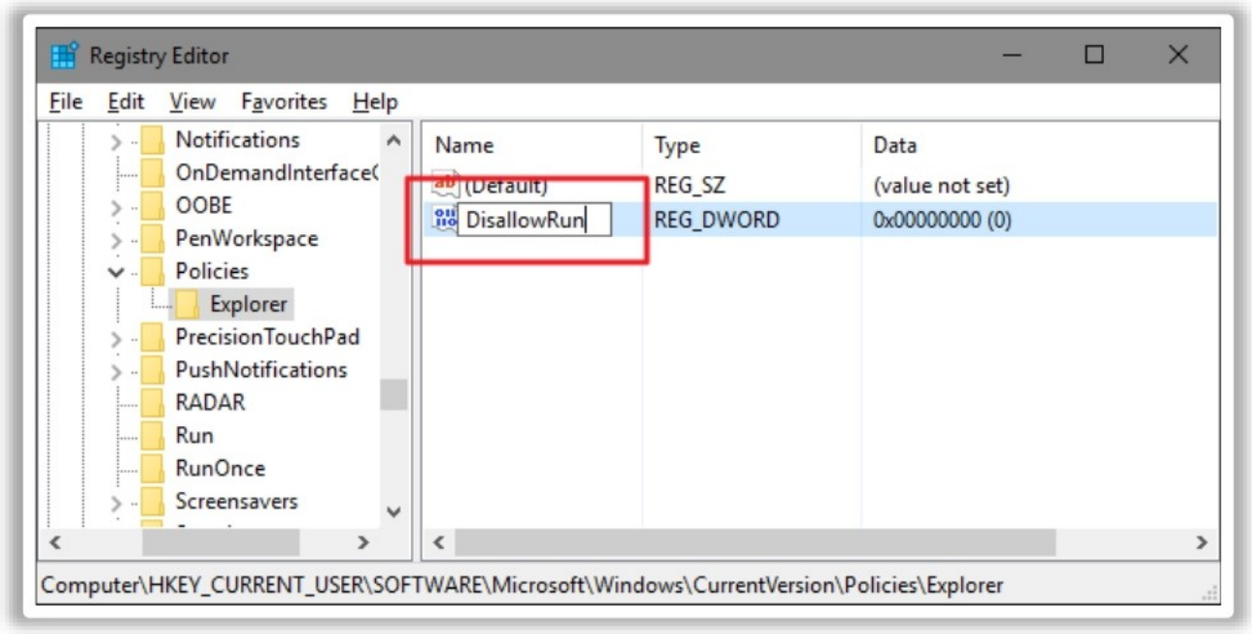


Figure 9.35: Creating New Value “DisallowRun”

- Double-click the new **DisallowRun** value to open its **Properties** dialog. Change the value from 0 to 1 in the **Value data** box and then click **OK**.

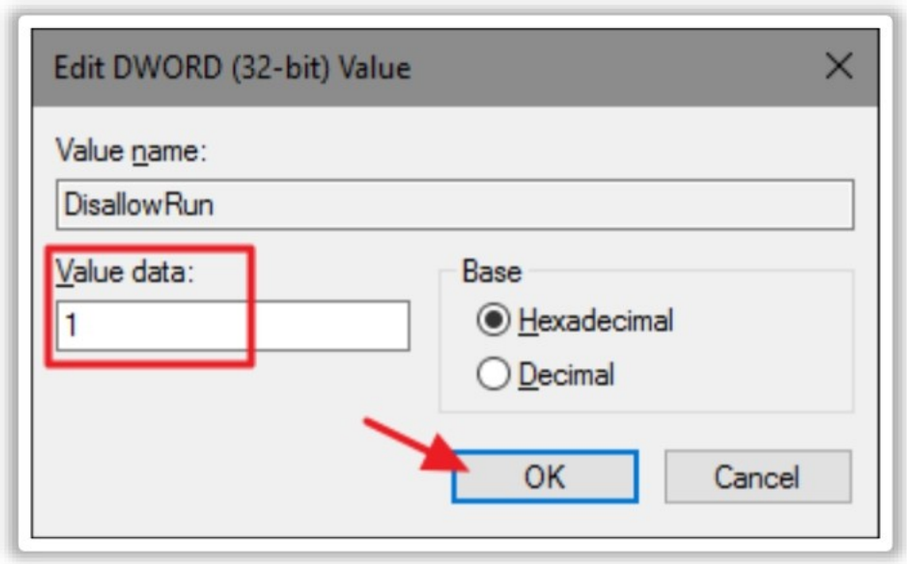


Figure 9.36: Set Value data for "DisallowRun"

- Create a **new subkey** inside the **Explorer** key in the main Registry Editor window.
- Right-click the **Explorer** key and choose **New Key**. Name the new key **DisallowRun**, which is identical to the previously created value.

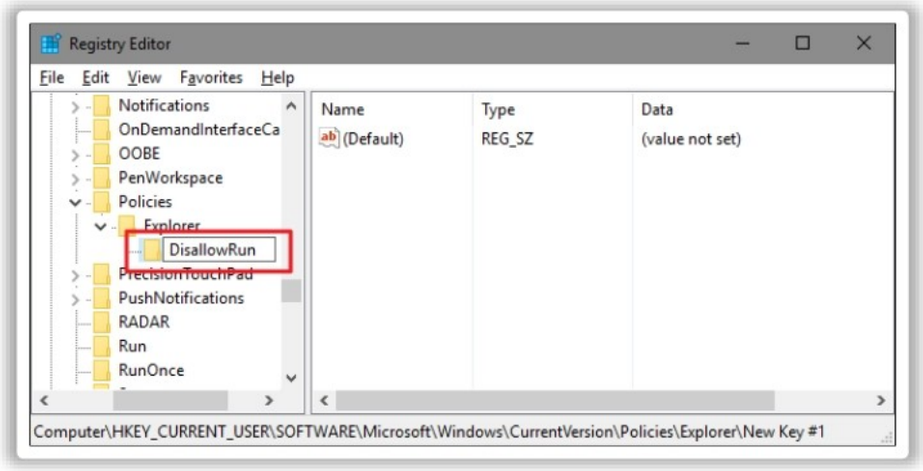


Figure 9.37: Selecting Explorer Key and Create DisallowRun Key

- Start adding apps to be blocked by creating a new string value inside the **DisallowRun** key for each app.
- Right-click the **DisallowRun** value and then choose **New String Value**. Name the first value as **1**.

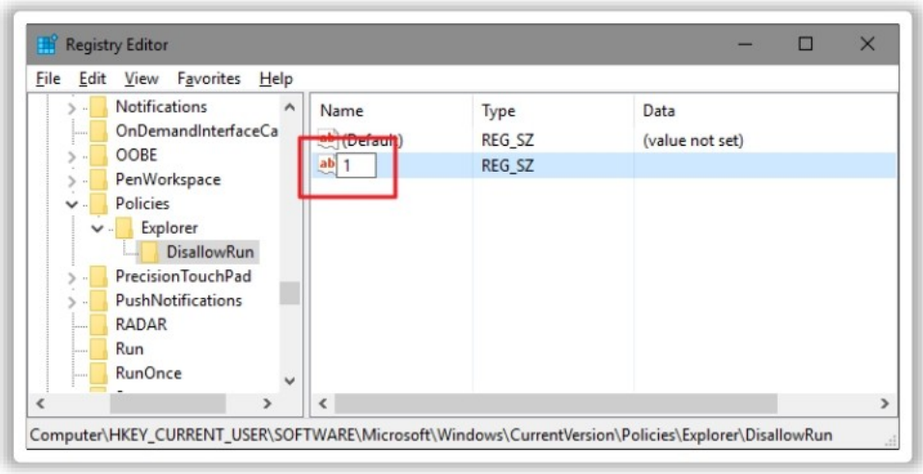


Figure 9.38: Set New String Value for DisallowRun

- Double-click the new value to open its **Properties** dialog, enter the name of the executable to block in the **Value data** box (e.g., notepad.exe) and click **OK**.

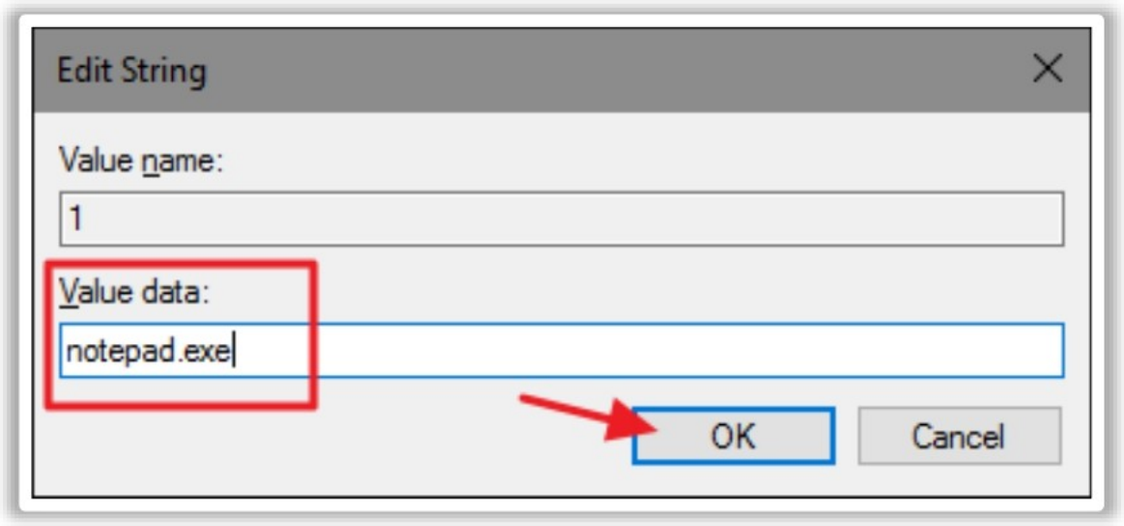


Figure 9.39: Set Value data for New String

- Repeat this process, naming the second string value "2," the third "3," and so on. Add the executable filenames to be blocked to each value.

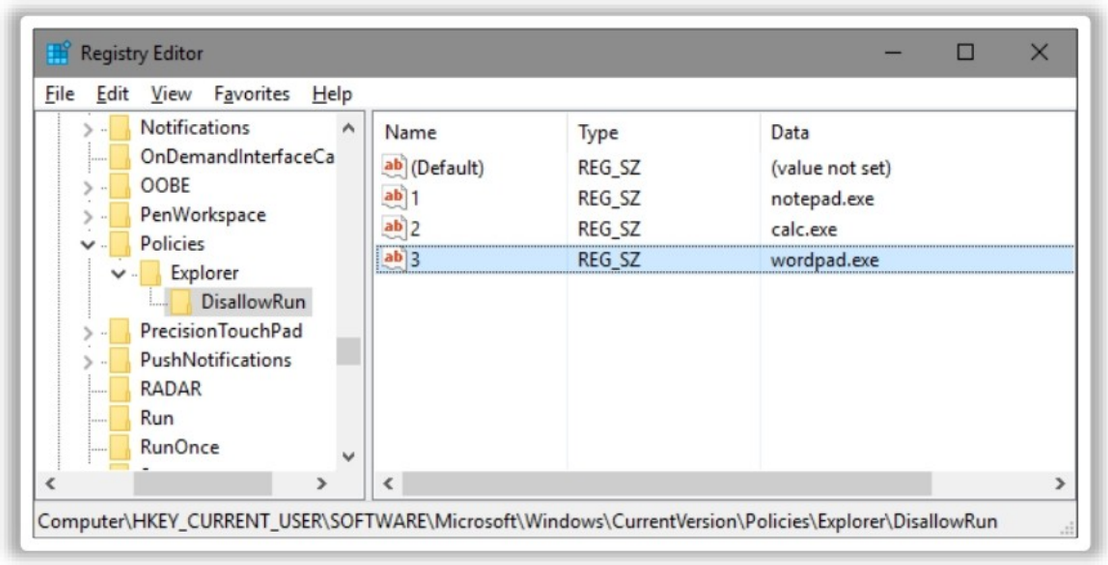


Figure 9.40: Creating 2 and 3 Strings and Setting their Value data

- Restart the system. Next, when the user attempts to execute a blocked application, they will see a Restrictions pop-up notifying that the app cannot be executed.

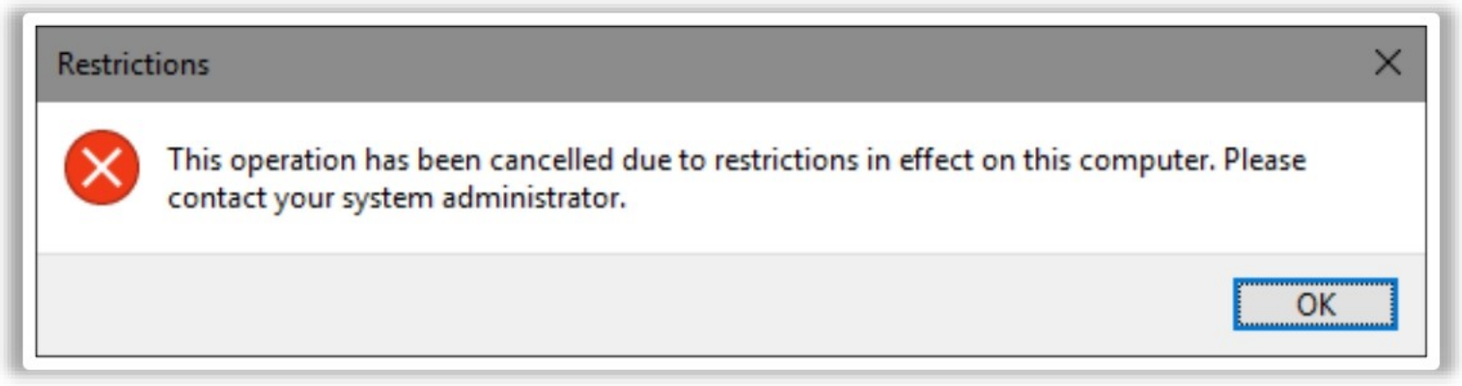


Figure 9.41: Blocking Application

Additional Application Whitelisting Tools

There are many paid and free application whitelisting tools to secure a system. The following is a list of some application whitelisting tools.

Airlock Digital

Source: www.airlockdigital.com

Airlock Digital is a paid application whitelisting software recognized by the Australian Signals Directorate (ASD) as one of the most reliable.

Digital Guardian

Source: https://info.digitalguardian.com

Digital Guardian (DG) provides an application whitelisting functionality that is easy to deploy and transparent to existing operations. It is the most secure option for retail point-of-sale (POS) systems and industrial control systems.

Ivanti Application Control

Source: https://www.ivanti.com

Ivanti Application Control clubs dynamic whitelisting and privilege management to avoid unauthorized code execution. It accomplishes this without the need to manage extensive lists manually and without constraining users. Ivanti Application Control is powered by AppSense and has the following features:

- **Dynamic whitelisting** to create flexible and preventive policies to ensure only trusted applications can be executed on a system

- **Privilege management** to balance access and security by removing full admin rights but providing granular access to the apps the user needs

Delinea

Source: https://delinea.com

Delinea is an enterprise-grade PAM solutions that put privileged access at the center of cybersecurity strategies. Its key solution Delinea's Server PAM controls privileged access to servers in both on-premise, and allows humans and machines to seamlessly authenticate with passwordless login, enforcing least privilege with just-in-time privilege elevation, preventing lateral movement, increasing accountability, and reducing administrative access risk.

Gatekeeper

Source: https://gkaccess.com

Gatekeeper is a security feature of Apple's macOS that reduces the likelihood of the malware attacks by verifying downloaded applications. Gatekeeper verifies the downloaded applications by enforcing code signing.

Kaspersky Whitelist

Source: https://whitelist.kaspersky.com

Kaspersky Whitelist boosts antivirus performance by avoiding the regular checking of an application on whitelisting and improves the efficiency of corporate resource usage by checking applications against the local whitelist database created by the company's system administrators.

PolicyPak

Source: https://www.policypak.com

PolicyPak Application Manager can be used to configure and lockdown various applications such as like Firefox, Java, Flash, Internet Explorer, and Adobe products. It can be integrated with Windows Server and other OS editions. PolicyPak comes in three editions.

- The Group Edition has a BYOD feature.
- The Cloud Edition is suitable for small organizations.
- The mobile device management (MDM) Edition is suitable for existing mobile device management services (Workspace One, Intune, etc.).

PowerBroker

Source: https://www.beyondtrust.com

PowerBroker is an application whitelisting tool that supports Windows, Linux, and Mac. Its key features include activity logging and privilege management. Its mobile policy features help solve issues associated with the BYOD policy.

PowerBroker for Windows uses the native OS security model to default-deny inappropriate user actions while elevating application and task permissions. Thus, it enables the implementation of

least-privilege best practices. Further, it enables users to work with a few rules only, instead of managing a complex whitelist with numerous application signatures.

Faronics Anti-Executable Cloud

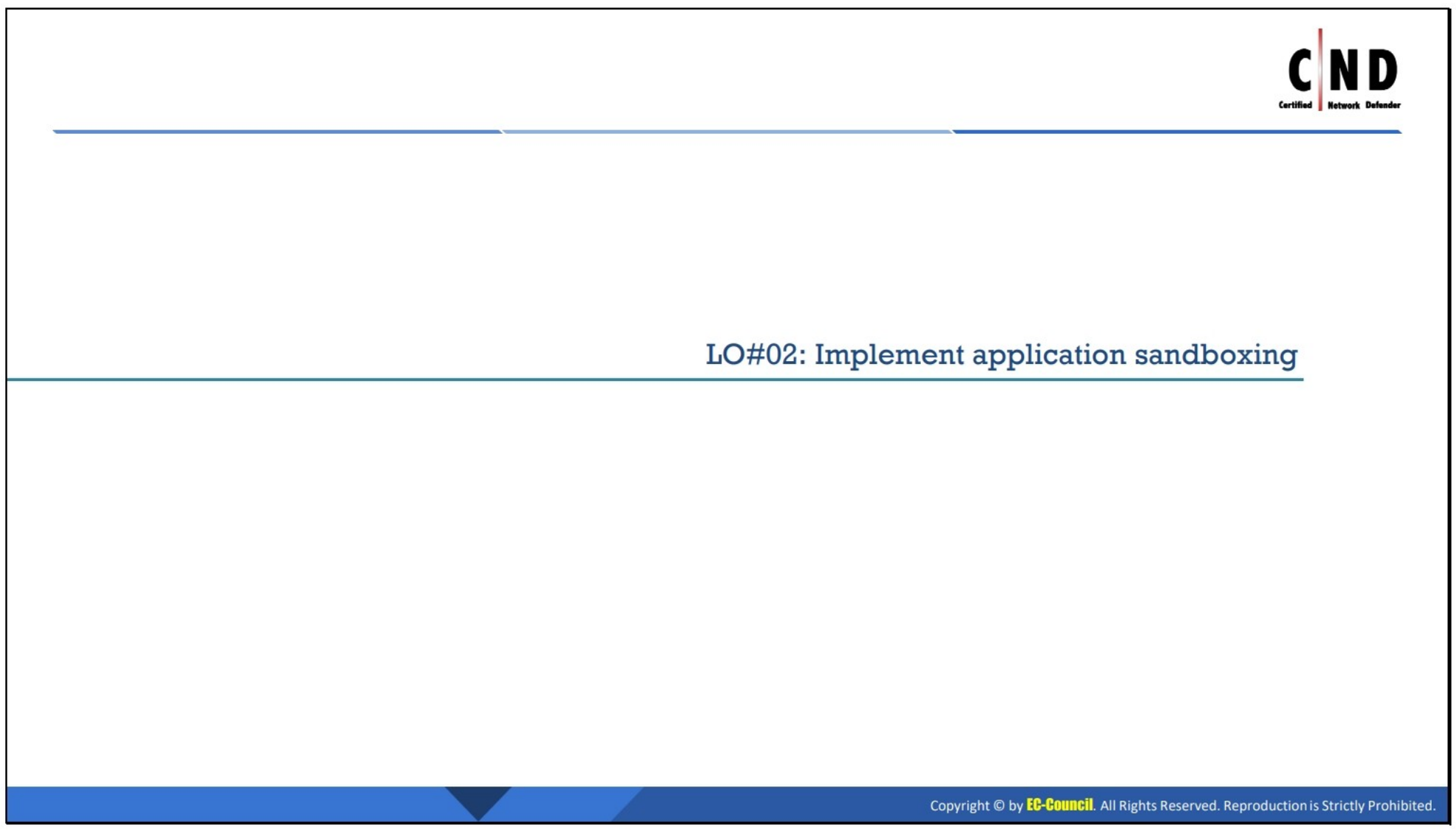
Source: https://www.faronics.com

The Faronics Anti-Executable Security Suite features artificial intelligence that provides machine-learning-assisted application whitelisting for "dirty environments," where this tool learns which files are safe/unsafe.

McAfee Application Control

Source: https://www.websecurityworks.com/

McAfee Application Control prevents zero-day and advanced persistent threat (APT) attacks by blocking the execution of unauthorized applications. It displays the list of applications in a hierarchical format by classifying them as well-known, unknown, and known-bad applications. Attacks from unknown malware can be prevented by implementing whitelisting. Its key features include default Deny whitelisting, Detect and Deny whitelisting, and Verify and Deny whitelisting.



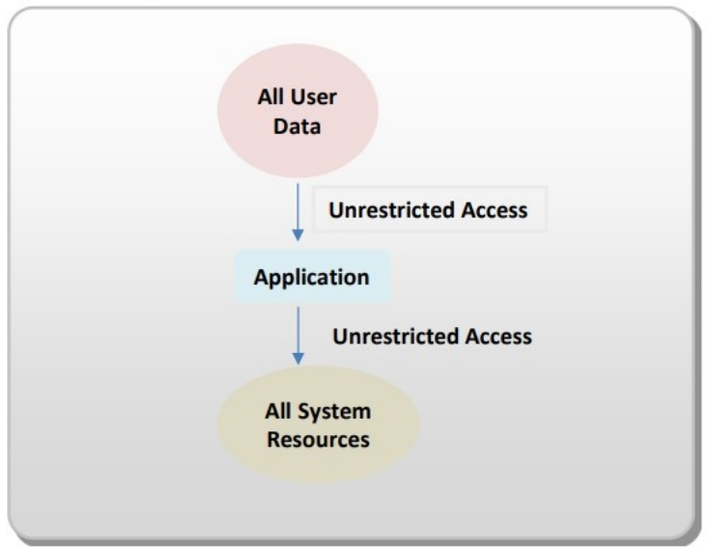
LO#02: Implement Application Sandboxing

Network defenders implement application sandboxes to protect "live" servers and their data. The objective of this section is to impart an understanding of the use of application sandboxing. The section also discusses the use of different Windows and Linux Sandbox tools.

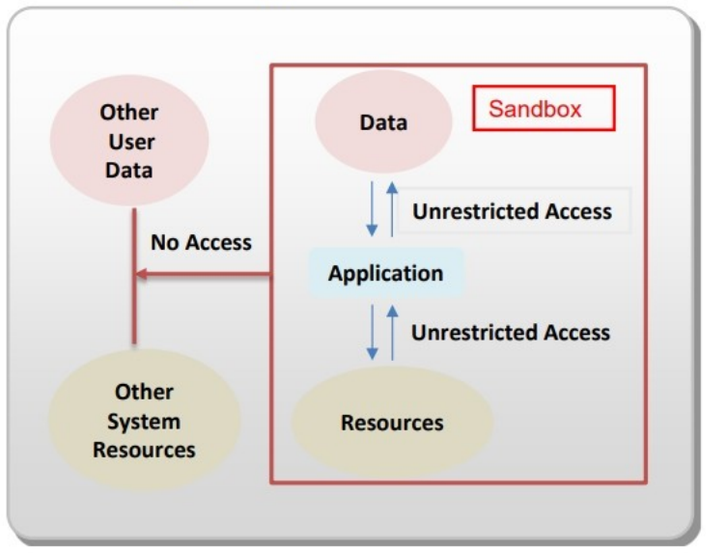
Application Sandboxing

- Sandboxing runs applications in a **sealed container** (sandbox) such that they cannot access critical system resources or other programs
- Sandboxing is used to execute untrusted or untested programs from third parties
- It provides an extra layer of security and protects apps and the system from malicious apps
- Network defenders can test their tasks in a sandbox without affecting the system

Running an Application Without a Sandbox

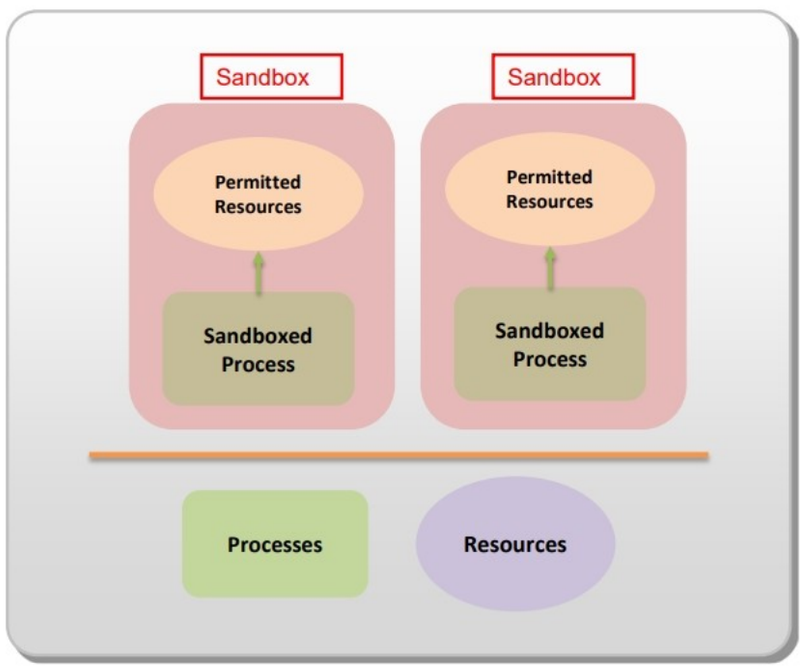


Running an Application With a Sandbox

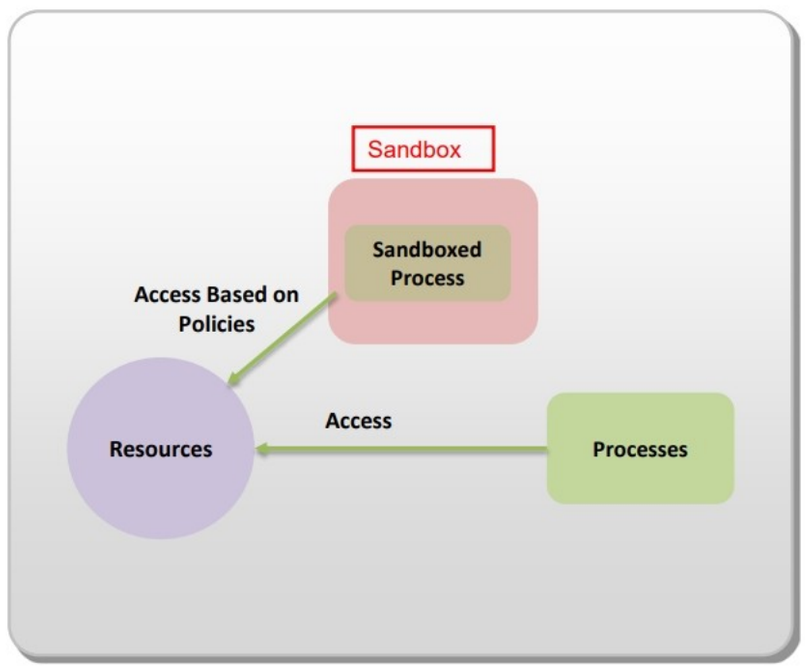


Application Sandboxing (Cont'd)

Isolation-based Sandbox



Rule-based Sandbox



Application Sandboxing

Application sandboxing is the process of running applications in a sealed container (sandbox) so that the applications cannot access critical system resources and other programs. It provides an extra layer of security and protects apps and the system from malicious apps. It is often used to execute untrusted or untested programs or code from untrusted or unverified third parties

without risking the host system or OS. The protection provided by the sandbox is not sufficiently robust against advanced malware that target the OS kernel.

When an application is executed without a sandbox, it has unrestricted access to system resources and all user data. In contrast, an application executed within a sandbox has restricted access to the system resources and data outside the sandbox.

Installing a sandboxed app in a system creates a specific directory (sandboxed directory). By default, the app has unlimited read and write access to the directory. However, apps within the directory are not allowed to read or write the files outside the directory or access other system resources, unless authorized.

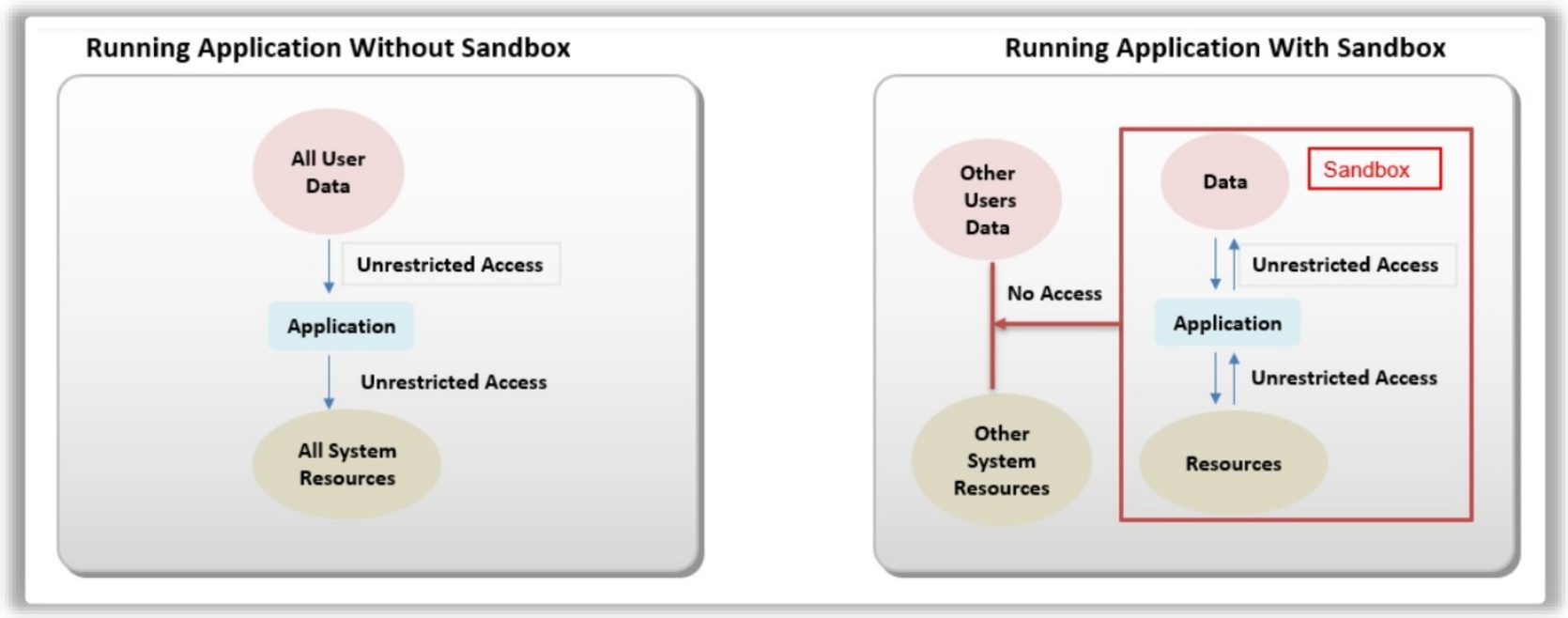


Figure 9.42: Execution of an Application with and without a Sandbox

The following approaches can be used to implement an application sandbox.

- **Isolation-based approach:** In this approach, a program running in the sandbox is isolated from the system resources and programs running outside the sandbox.

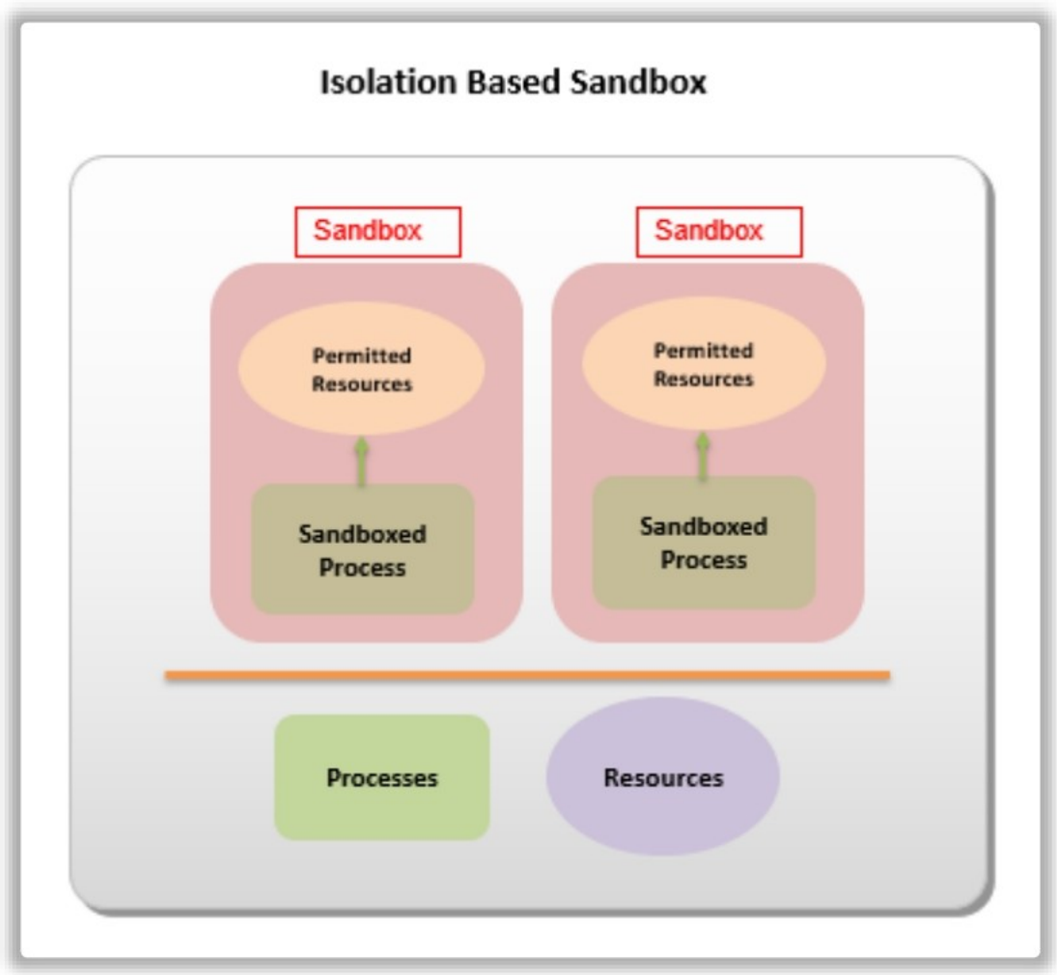


Figure 9.43: Isolation-Based Sandbox

- **Rule-based approach:** In this approach, the sandbox controls what each application can do and permits applications to share resources based on the set rules.

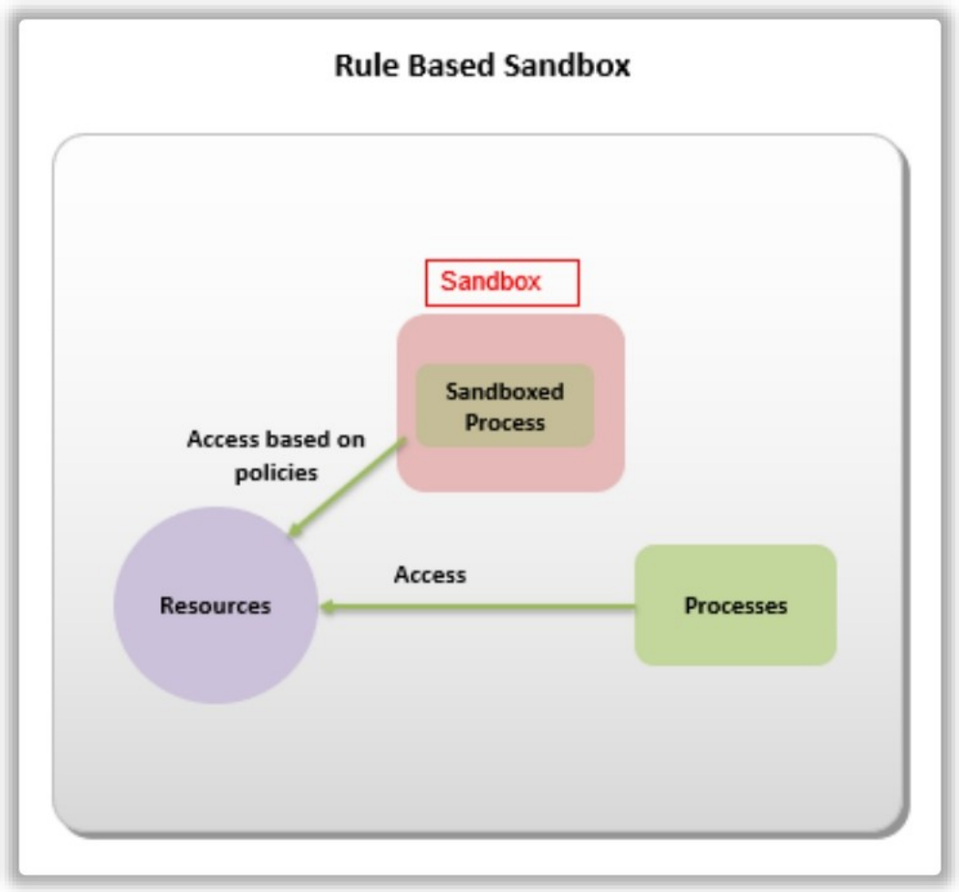
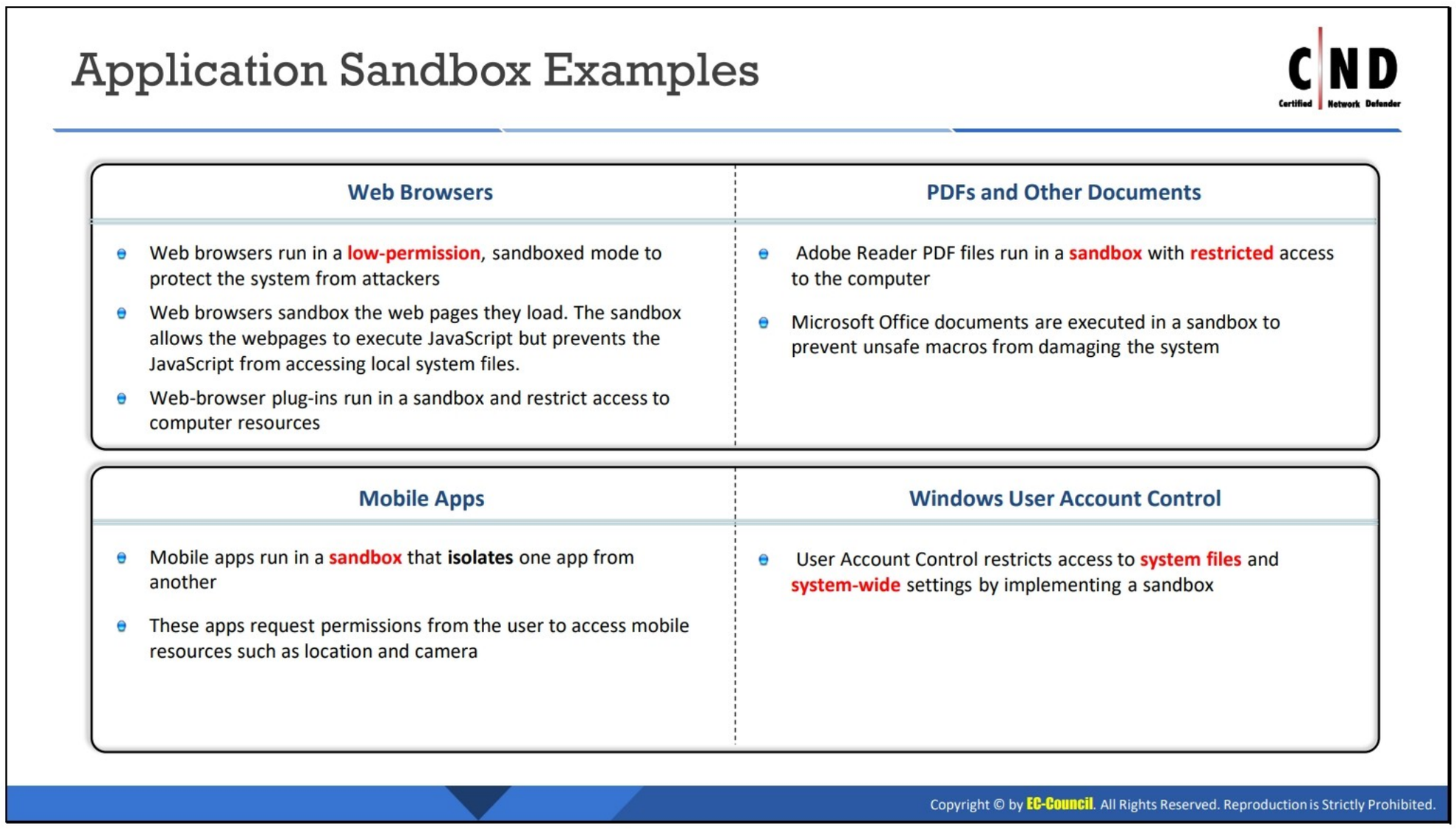


Figure 9.44: Rule-Based Sandbox



Application Sandbox Examples

Many applications are sandboxed for protection in systems. The following are some application sandbox examples:

- **Web browsers**
  - Web browsers run in a low-permission, sandboxed mode to protect the system from attackers.
  - A web browser sandboxes the web pages it loads. The sandbox allows the webpages to execute JavaScript but prevents the JavaScript from accessing local system files.
  - Web-browser plugins run in a sandbox with restricted access to the computer resources.

Steps to enable "Strict-Origin-Isolation" in Google Chrome

The Strict-Origin-Isolation feature of Chrome enables it to load each site in a dedicated process and provide limited access to the website. It blocks the process from receiving certain types of sensitive documents from other sites.

There are two methods to enable Strict-Origin-Isolation in Chrome.

- **Using a Chrome flag**
  - Enter **chrome://flags** in the address bar.

- Find "**Strict-Origin-Isolation**" and select **Enabled**.

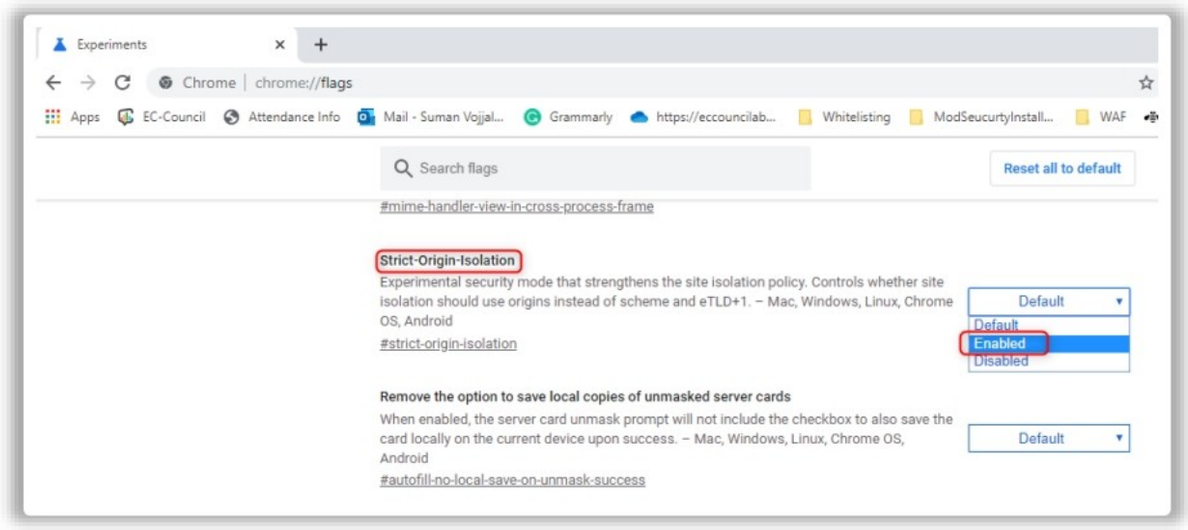


Figure 9.45: Enabling "Strict-Origin-Isolation" in Chrome

- Alternatively, copy and access the URL **chrome://flags/#enable-site-per-process** to go to the Site Isolation flag directly.
- Restart the Chrome browser.

- **Using a command-line flag**
  - Right-click the Chrome icon and select **Properties**.
  - Select the **Shortcut** tab.
  - In the **Target** field, add the text **"--site-per-process"** at the end of the shortcut path and click **OK**.

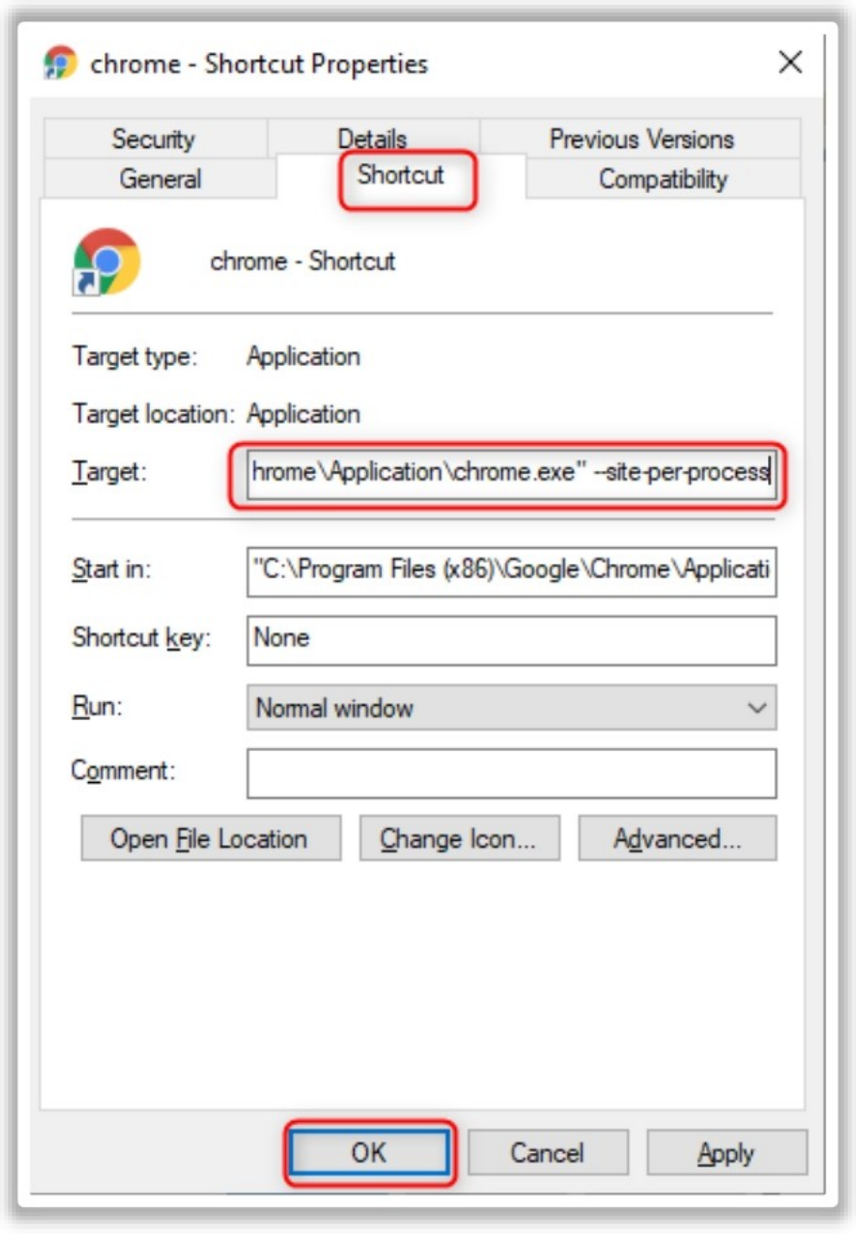


Figure 9.46: Setting a Target to Use Command-Line Flag

Steps to check Mozilla Firefox's sandbox level

Mozilla Firefox shows the sandbox levels used by the browser in two locations in the interface.

Location 1:

- Enter **about:support** in the address bar.
- Scroll down to the **Sandbox** listing.
- Check **Content Process Sandbox Level** to determine the sandbox level.

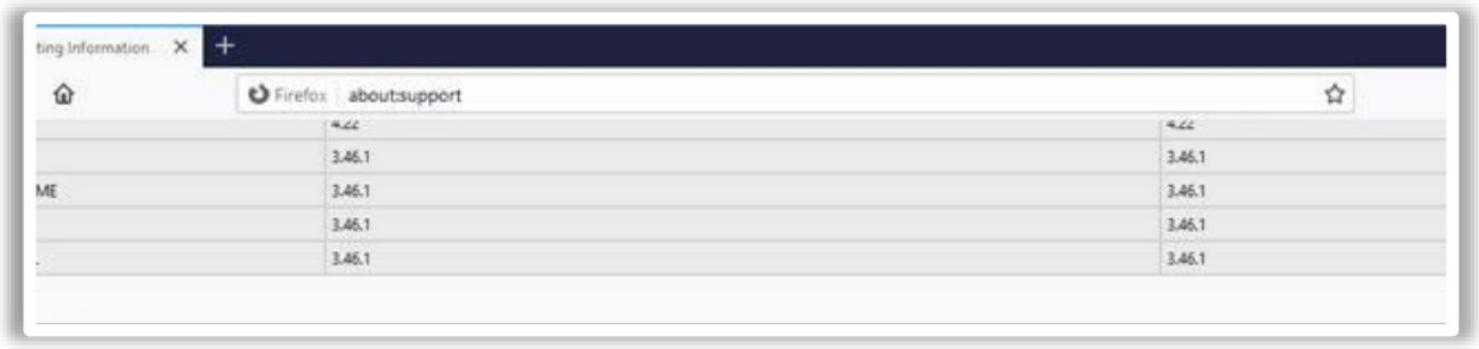


Figure 9.47: Checking Mozilla Firefox's Sandbox Level at Location 1

Location 2:

- Enter **about:config** in the address bar.
- Search for the parameter **security.sandbox.content.level**. The value that is returned is the current content level of the Firefox sandbox.
- To change this value, double-click it and change the value as required.

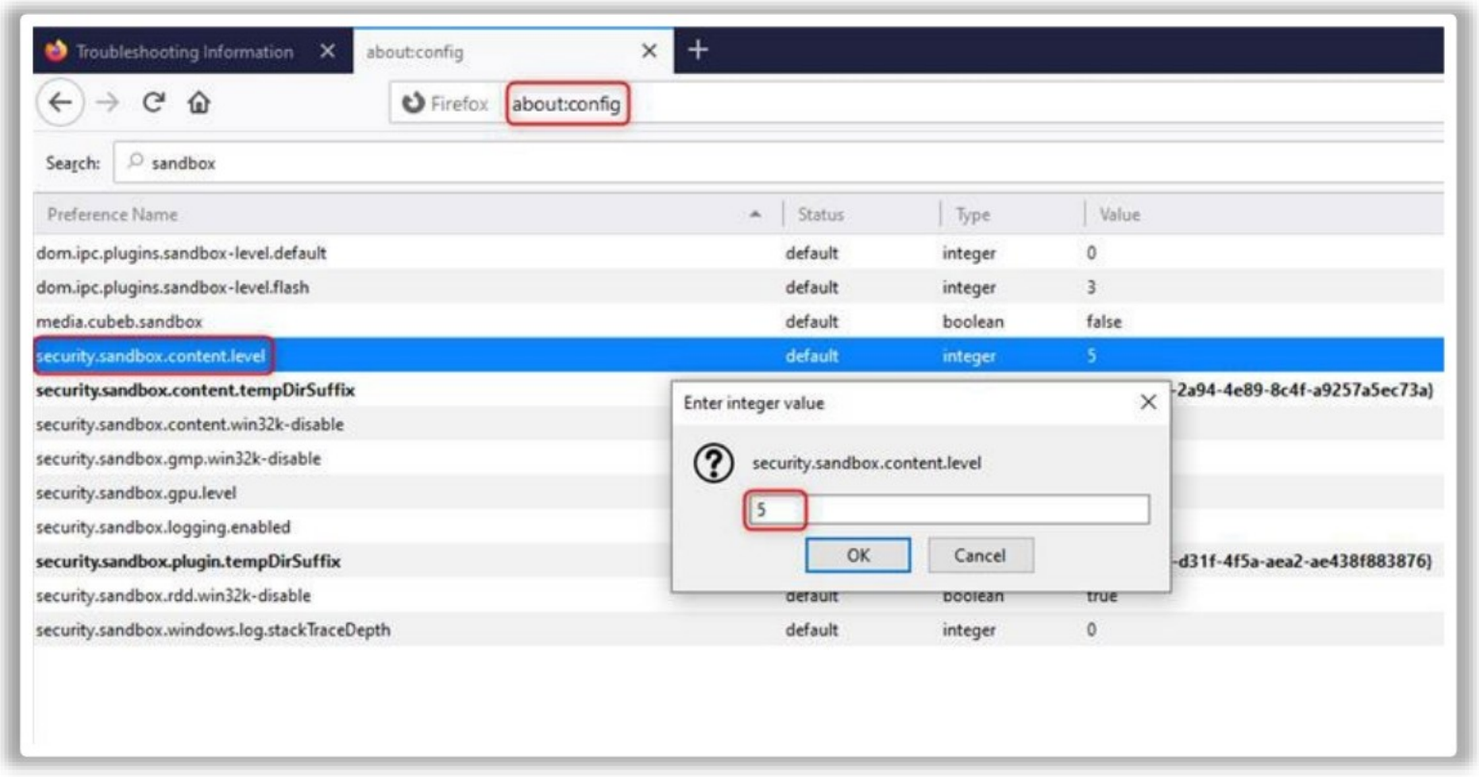


Figure 9.48: Checking Mozilla Firefox's Sandbox Level at Location 2

- **PDFs and other documents**
  - Adobe Reader PDF files run in a sandbox with restricted access to the computer.
  - Microsoft Office documents are executed in a sandbox to prevent unsafe macros from damaging the system.

Steps to configure a sandbox in Acrobat Reader :

- Go to **Edit->Preferences->Security (Enhanced)->Sandbox protections**.
- Toggle the feature controls as needed.

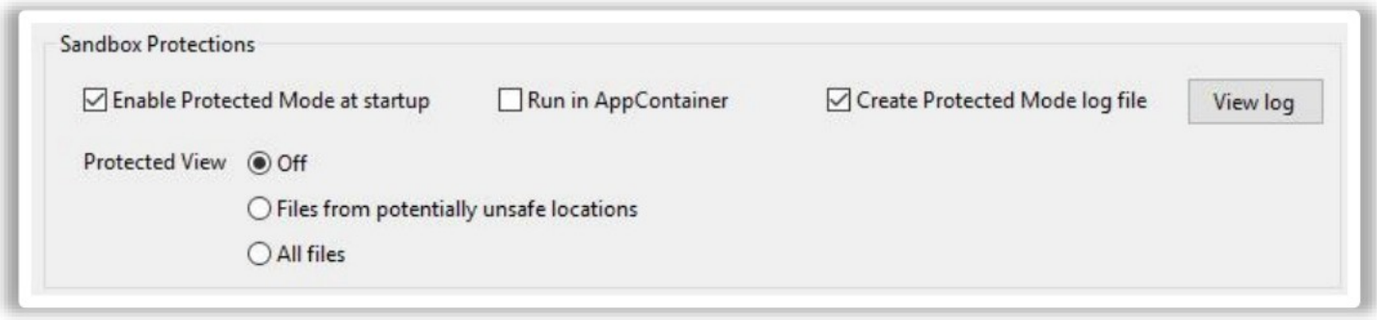
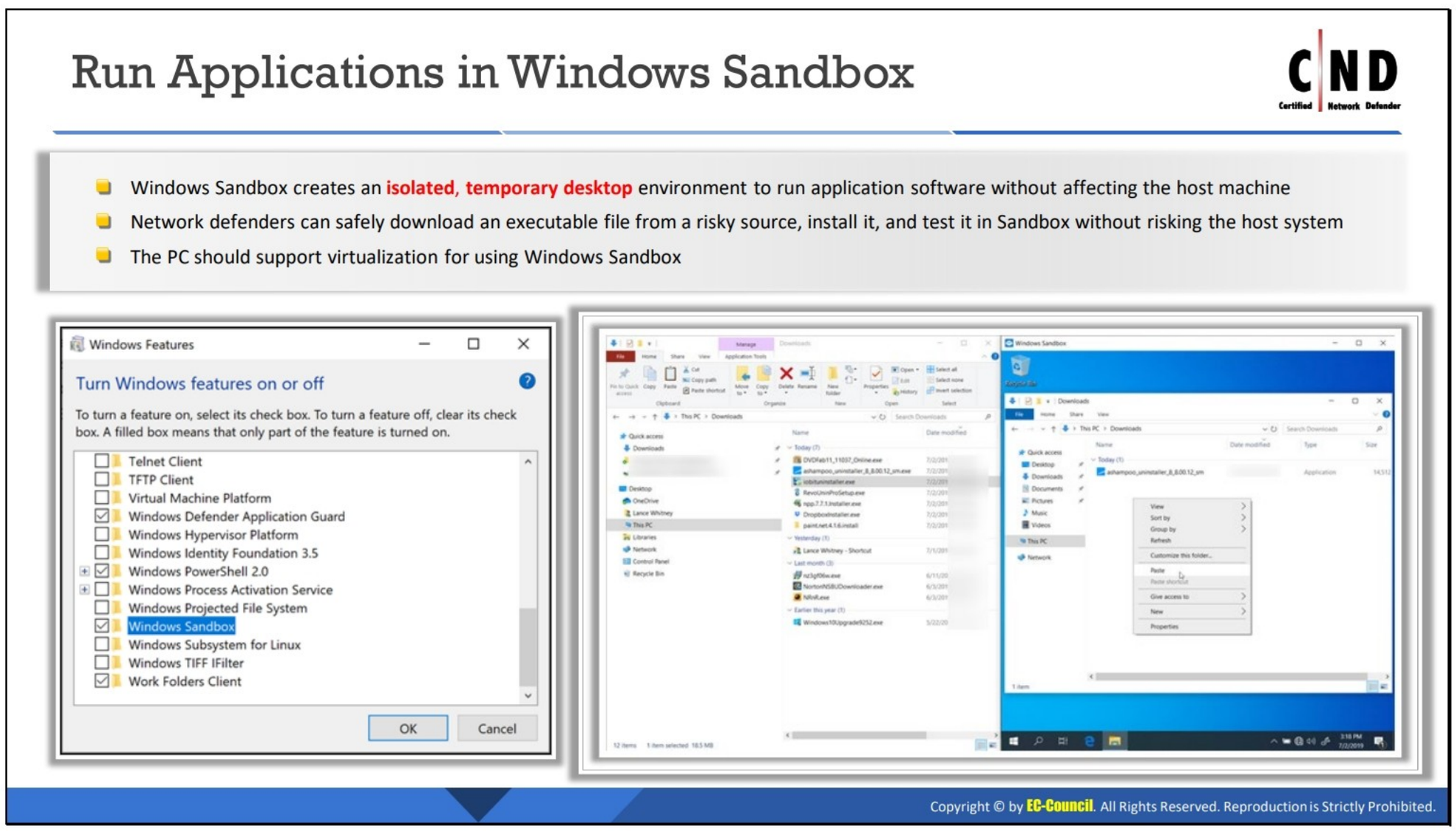


Figure 9.49: Configuring Sandbox in Acrobat Reader

- **Mobile apps**
  - Mobile apps run in a sandbox and isolate one app from another.
  - These apps request permissions from the user to access mobile resources such as location and camera.
- **Windows User Account Control**
  - Windows User Account Control (UAC) restricts access to system files and system-wide settings by implementing a sandbox. Microsoft Edge runs in a sandbox. Its "Protected Mode" feature uses UAC to run with a "low" integrity level (a standard user token has an integrity level of "medium," while an elevated (administrator) token has an integrity level of "high").



Run Applications in Windows Sandbox

Windows Sandbox creates an isolated, temporary desktop environment to run application software without affecting the host machine. Administrators can safely download an executable file from a risky source, install it, and test it in Sandbox without risking the host system.

Note: The PC should support virtualization for using Windows Sandbox.

The following are the steps to run applications in Windows Sandbox:

- To ensure that virtualization is enabled, right-click the **Taskbar** and select **Task Manager**.
- Click the **Performance** tab and ensure that the entry for **Virtualization** is **Enabled**.

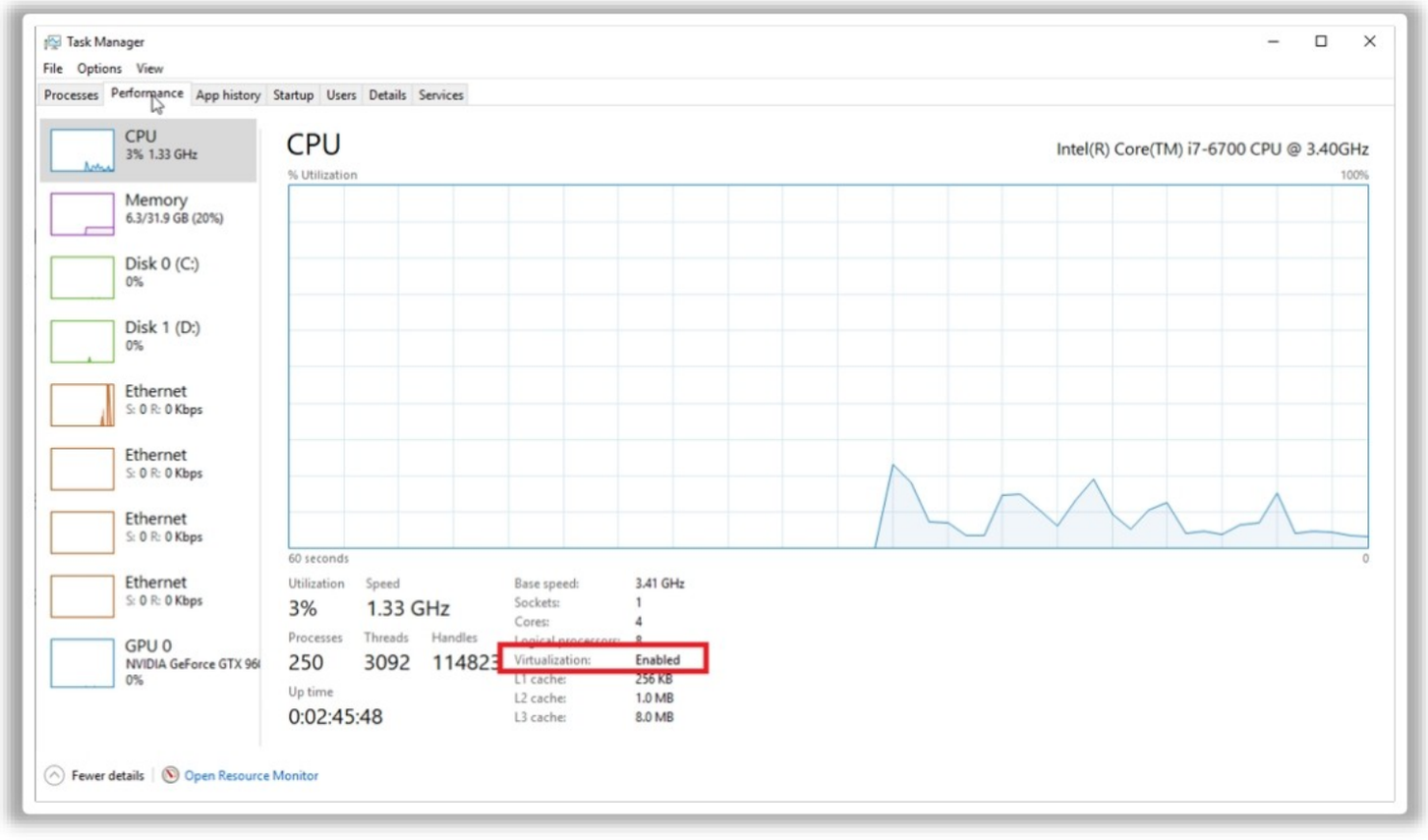


Figure 9.50: Checking whether Virtualization is Enabled

- Search for **Windows Features** in the **Start** menu and check **Windows Sandbox**.

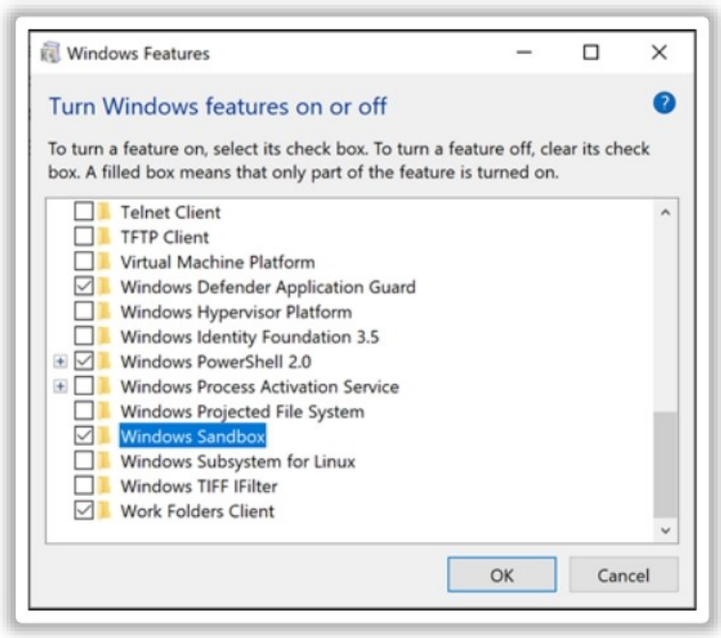


Figure 9.51: Checking for Windows Sandbox feature

- Restart the system after installing Windows Sandbox.
- Go to the **Start** menu, right-click **Windows Sandbox**, and click **Run as administrator**.
- Only the built-in Windows apps (OneDrive, Mail, Edge, Microsoft Store, and Photos) are installed in Windows Sandbox when it is opened.

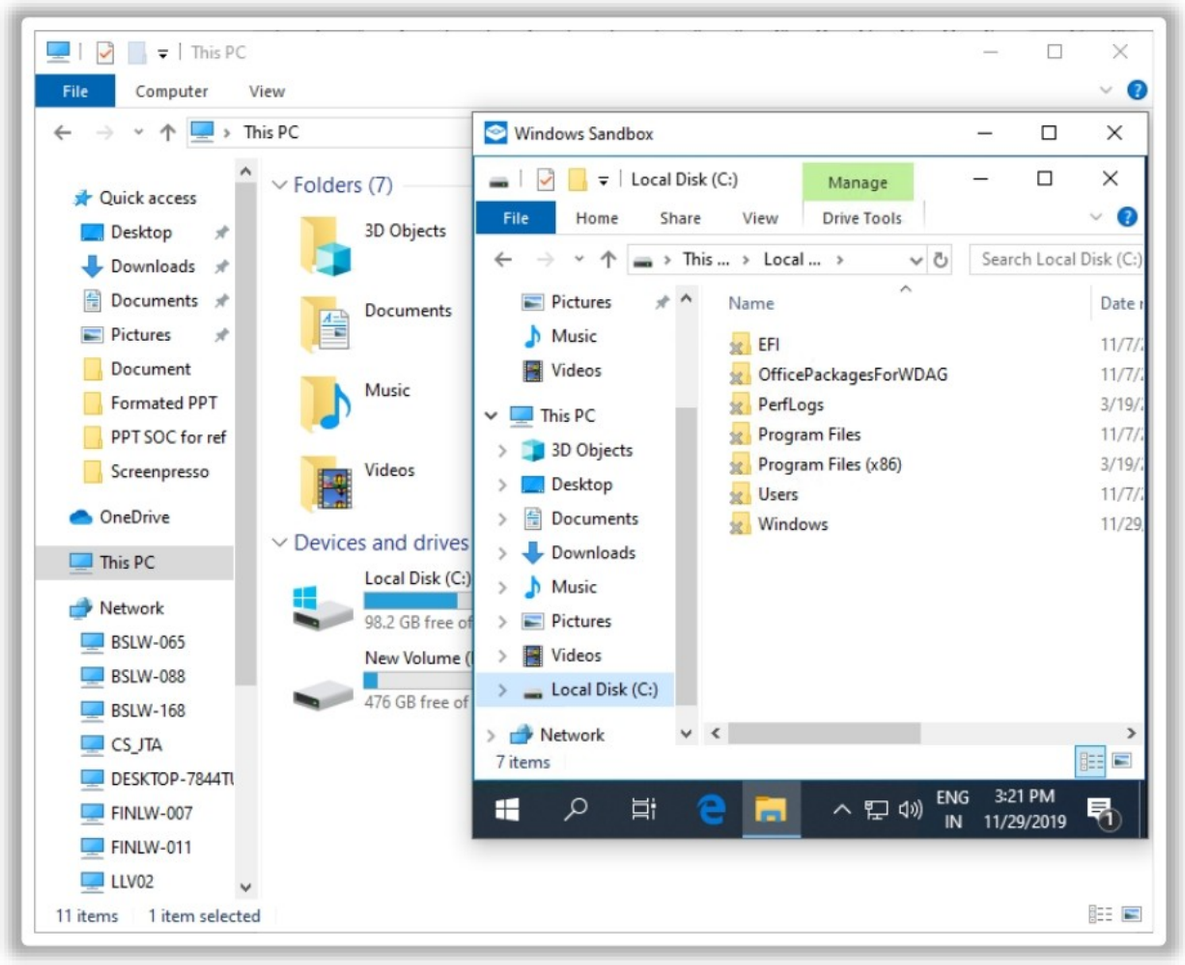
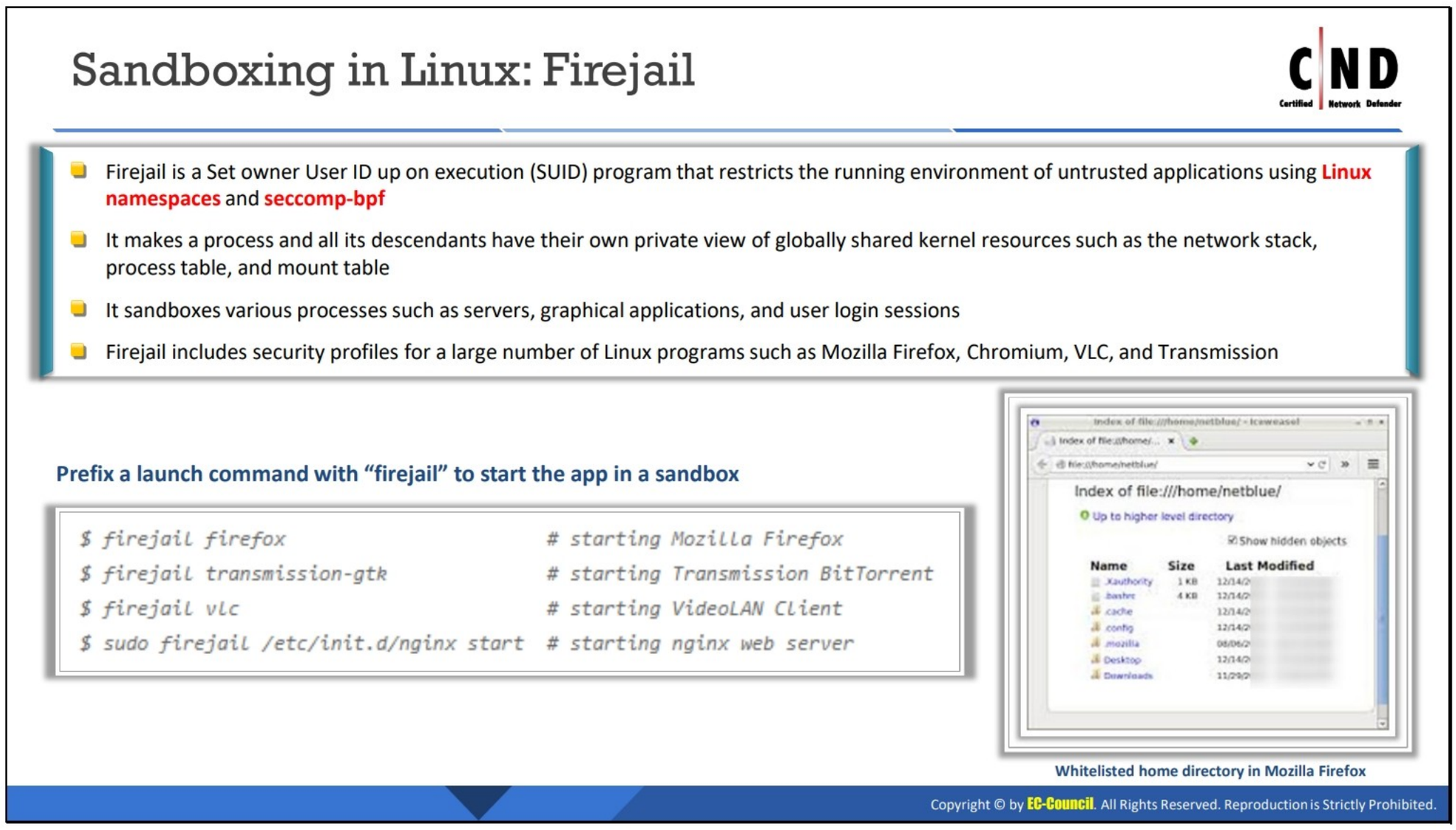


Figure 9.52: Ensuring Built-in Windows Apps are Installed in Sandbox

- To make a particular program available in Windows Sandbox, download it from **Edge** in the Sandbox, or drag and drop the downloaded file to Sandbox from the system.
- Install, run, and use the program.
- Instead of restarting or shutting down the sandbox session, close Sandbox to allow the changes to take effect.



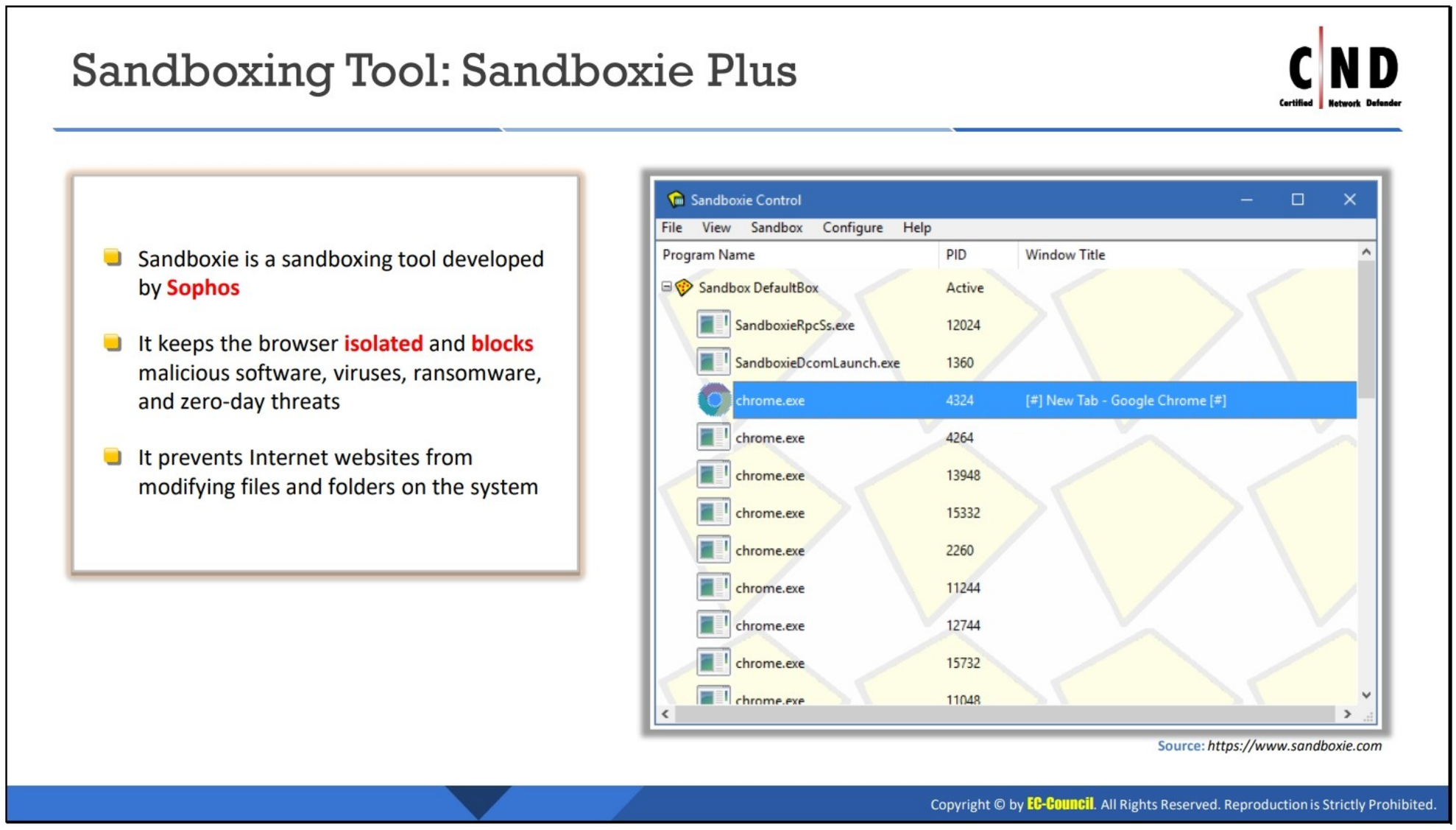
Sandboxing in Linux: Firejail

Firejail is a Set owner User ID up on execution (SUID) program that restricts the running environment of untrusted applications using Linux namespaces and **seccomp-bpf.** It makes a process and all its descendants to have their own private view of globally shared kernel resources such as the network stack, process table, and mount table.

It sandboxes various processes such as servers, graphical applications, and user login sessions. Firejail includes security profiles for many Linux programs: Mozilla Firefox, Chromium, VLC, Transmission, etc.

Prefix a launch command with **firejail** to start the app in a sandbox. Examples are given below:

```
$ firejail firefox  # starting Mozilla Firefox
$ firejail transmission-gtk  # starting Transmission BitTorrent
$ firejail vlc  # starting VideoLAN Client
$ sudo firejail /etc/init.d/nginx start  # starting nginx web server
```



Sandboxing Tool: Sandboxie Plus

Sandboxie is a sandboxing tool developed by Sophos. It keeps the browser isolated and blocks malicious software, viruses, ransomware, and zero-day threats. It prevents websites from modifying files and folders on the system.

The following are the steps to allow already installed programs (e.g., a browser) in Sandboxie:

- Select **Sandbox->Default Box->Run Sandboxed->Run Web browser**.
- Select **Run Any Program** to allow any other application.

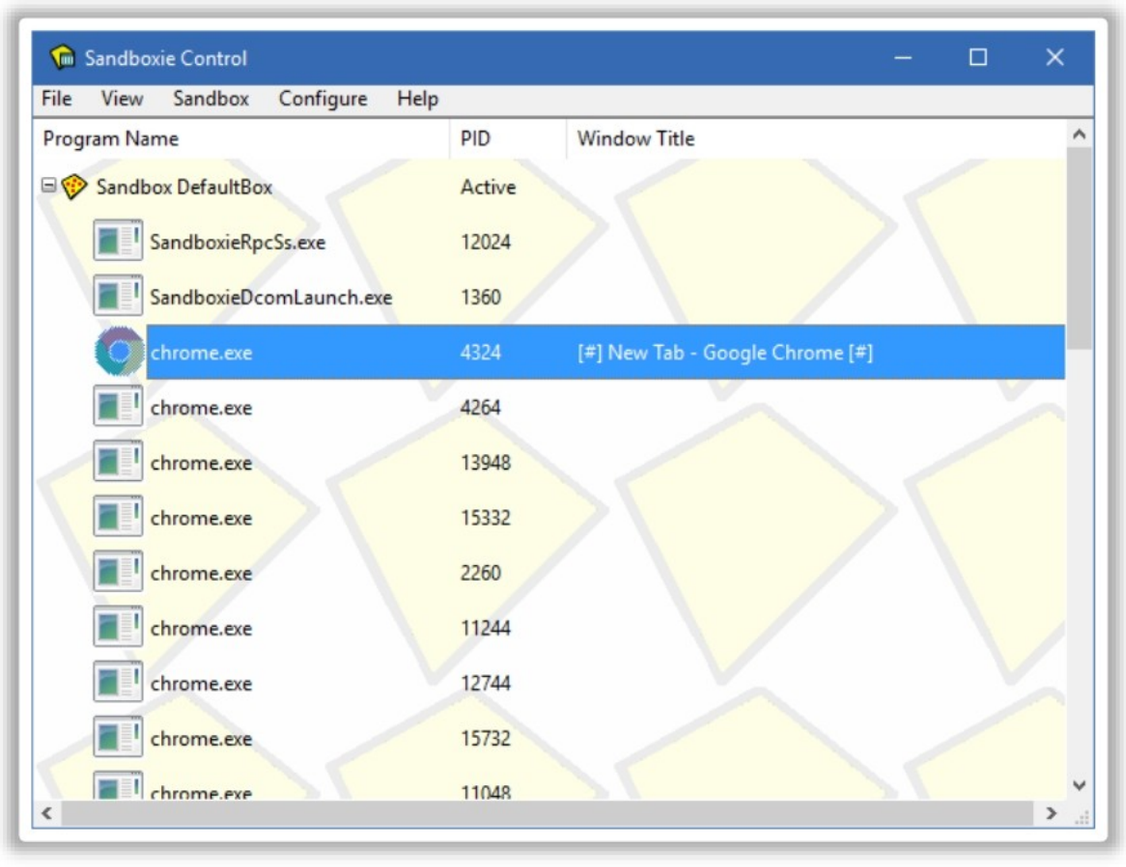
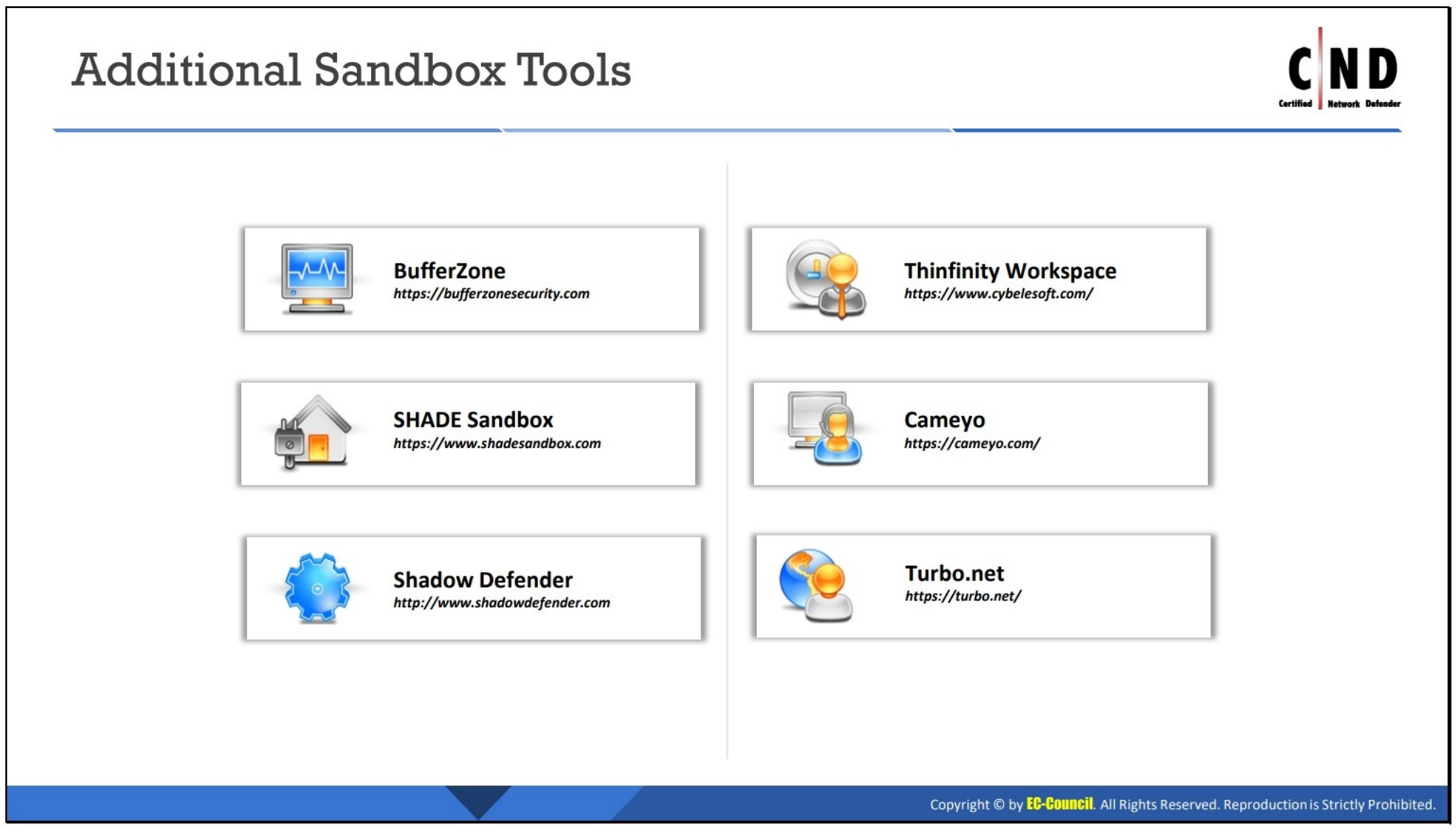


Figure 9.53: Working of Sandboxie Control



Additional Sandbox Tools

BufferZone

Source: https://bufferzonesecurity.com

BufferZone keeps activities in a secure virtual zone to prevent web-based malicious software from entering the system because everything that passes through the zone becomes read-only.

Thinfinity Workspace

Source: https://www.cybelesoft.com/

Use Thinfinity Workspace to connect to Windows, Linux, and Mac Desktops and access applications from any device with a web browser. Implement role-based permissions to define access profiles and regulate user privileges, streamlining security management across organization. Transform individual applications from locally installed products into centrally managed services.

SHADE Sandbox

Source: https://www.shadesandbox.com

SHADE Sandbox provides a simple and beginner-friendly user interface. It allows the dragging and dropping of applications into a sandbox window, and the applications are sandboxed on launch. The sandbox isolates the browsing history, temporary files, cookies, Windows Registry, and system files. The files downloaded while using the sandbox are stored in the Virtual Downloads folder.

Cameyo

Source: https://www.cameyo.com

Cameo allows simple and secure deliver of business-critical apps to any device, from the browser, so people can stay productive from anywhere. Architected with a Zero Trust security model, it can eliminate the need for VPNs or open firewall or server ports.

Shadow Defender

Source: http://www.shadowdefender.com

Shadow Defender prompts to virtualize the system drive and other drives (based on choice) when installing it. After the system is virtualized, rebooting the system discards the changes that are made to it. Clicking the Commit Now button in the Shadow Mode allows the user to save a downloaded file or commit a system change.

Windows Sandboxing

Source: https://learn.microsoft.com

Windows Sandbox is a feature in Windows operating systems that provides a lightweight desktop environment for running applications separately from the host machine. It ensures a secure and isolated execution environment, referred to as "sandboxed," where software installed within the sandbox remains separate from the host system.

Docker

Source: https://www.docker.com

Docker is a cloud-based platform that simplifies the development, deployment, and operation of applications. It uses containerization technology to isolate applications from their underlying infrastructure, making it easier to package and deliver software consistently across different environments.

Firejail

Source: https://firejail.wordpress.com/

Firejail is a Set-User-ID (SUID) sandbox program designed for Linux systems. It enhances security by restricting the runtime environment of untrusted applications using Linux namespaces, seccomp-bpf, and Linux capabilities. Firejail isolates a process and all its descendants, providing them with their private views of the globally shared kernel resources like the network stack, process table, and mount table. It can also work within SELinux or AppArmor security frameworks and is integrated with Linux Control Groups.

Cuckoo Sandboxing

Source: https://cuckoosandbox.org/

Cuckoo Sandbox is an open-source automated malware analysis tool. It allows researchers and analysts to run suspicious files and programs in a controlled and isolated environment, or "sandbox," to analyze their behavior and identify potential threats. Cuckoo Sandbox provides a safe way to study and understand the actions of malware without compromising the security of the host system.

Shade Sandboxing

Source: https://www.shadesandbox.com/

Shade Sandbox is a Windows-based alternative for sandboxing. When subjected to testing within the Shade Sandbox, malware is automatically detected, and any malicious code can be executed within a secure and isolated environment. This allows for the observation of code behavior and its output.

DeepArmor Sandboxing

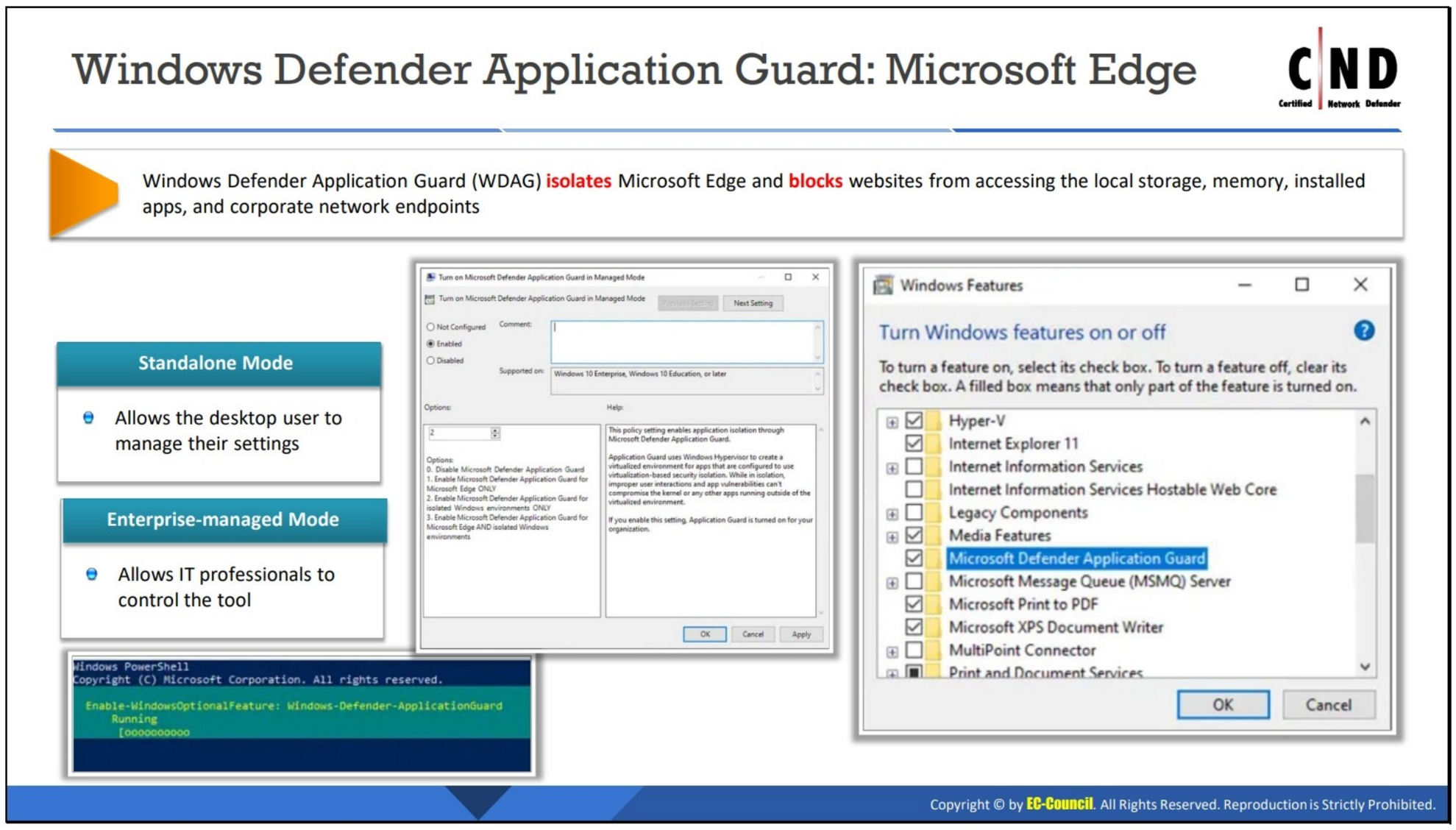
Source: https://www.deeparmor.com/

This tool is used by security operations teams (SOLs) and security providers to identify and mitigate threats, such as ransomware, malware, and viruses, as well as other IT security threats. It leverages AI and machine learning to proactively thwart file-based attacks, fileless attacks, and in-memory attacks, while also conducting in-depth analysis of attackers to enhance future protection measures.

Turbo.net

Source: https://turbo.net/

The Turbo.net Hub offers instant access to thousands of web and native Windows. applications. It allows deploying modern and legacy applications to any endpoint. Easy-to-use tools allow combining of applications and components to personalize workspaces or deploy custom applications.



Windows Defender Application Guard: Microsoft Edge

Windows Defender Application Guard (WDAG) isolates Microsoft Edge and blocks websites from accessing the local storage, memory, installed apps, and corporate network endpoints.

Table 9.1 lists the prerequisites to install WDAG.

Hardware requirements	Software requirements
▪ 64-bit computer with a minimum of 4 cores ▪ CPU virtualization extensions (Second Level Address Translation (SLAT) and VT-x(Intel) or AMD-V) ▪ Minimum of 8 GB RAM ▪ 5 GB free space in a solid-state drive (SSD) ▪ Input/Output Memory Management Unit (IOMMU) support	▪ Windows 10 Enterprise edition, version 1709 or higher ▪ Windows 10 Professional edition, version 1803 or higher ▪ Windows 10 Professional for Workstations edition, version 1803 or higher ▪ Windows 10 Professional Education edition, version 1803 or higher ▪ Windows 10 Education edition, version 1903 or higher ▪ Professional editions are only supported for non-managed devices ▪ Browsers: Edge and Internet Explorer ▪ Management system for managed devices: Intune/SCCM/Group Policy/Third party MDM ▪ Standalone mode: Windows 11 Enterprise, Education, or Pro editions ▪ Enterprise-managed mode: Windows 11 Enterprise or Education editions

Table 9.1: System Requirements for Windows Defender Application Guard

By default, WDAG is turned off and can be installed on user devices through the Control Panel, PowerShell, or a mobile device management (MDM) solution.

The following are the steps to turn on WDAG using the Control Panel:

- Navigate to **Control Panel->Programs** and click **Turn Windows features on or off**.
- Check **Windows Defender Application Guard** and click **OK**.

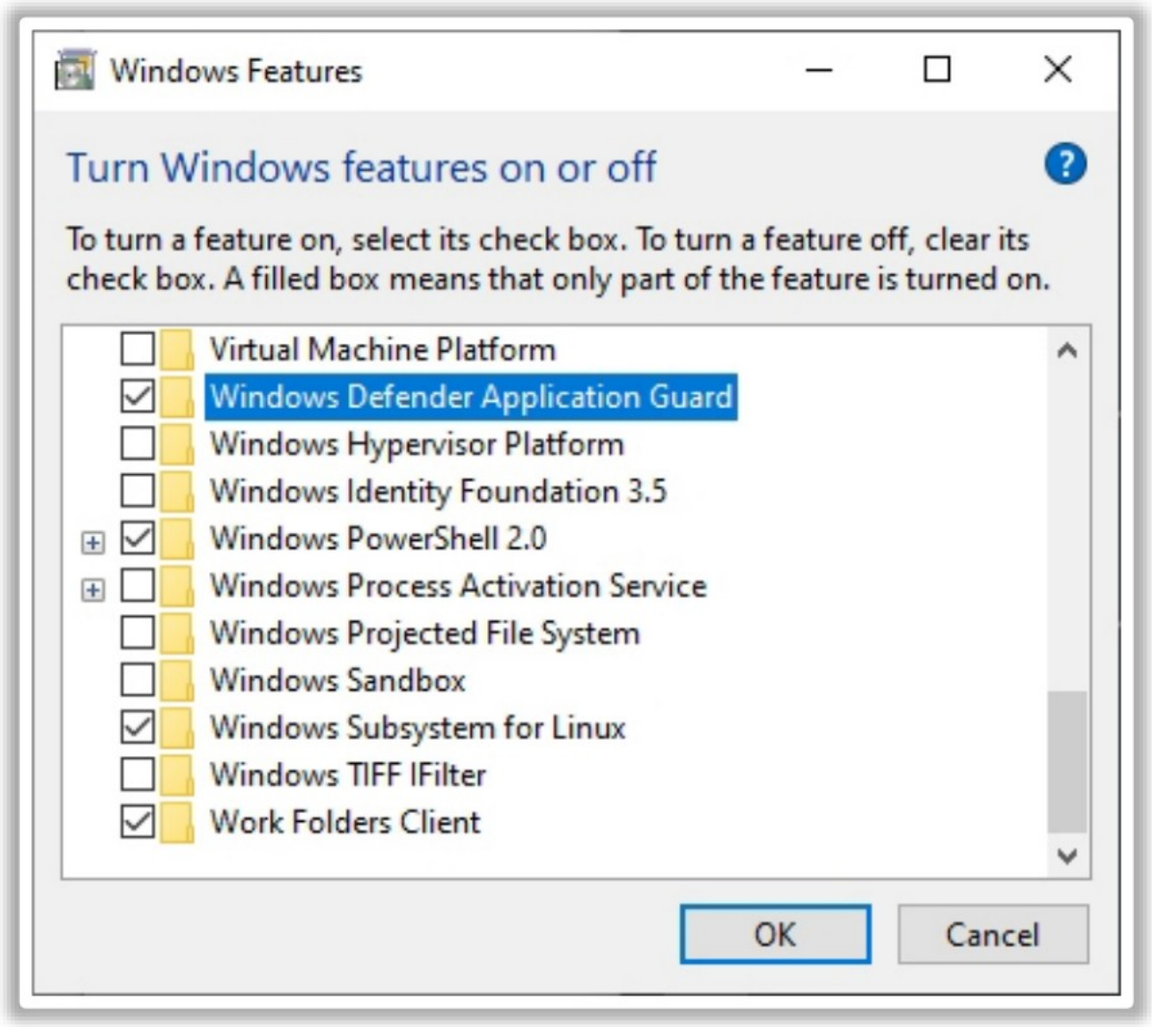


Figure 9.54: Checking Windows Defender Application Guard

- Restart the device after installing WDAG.

The following are the steps to turn on WDAG using PowerShell, which is recommended for enterprise-managed scenarios only:

- Run **PowerShell** as an administrator.
- Enter the following command:

```
Enable-WindowsOptionalFeature -online -FeatureName Windows-Defender-ApplicationGuard
```

- Restart the system.
- WDAG is now installed.

Before installing and using WDAG, choose any one of the following methods based on the requirement.

- **The standalone mode** allows the desktop user to manage settings without any administrator or management policy configuration. To use this mode, first install WDAG and then start Edge in Application Guard when browsing untrusted sites.

Steps to test WDAG in the standalone mode

- After installing WDAG, start **Edge** and click **New Application Guard window** from the **menu**.

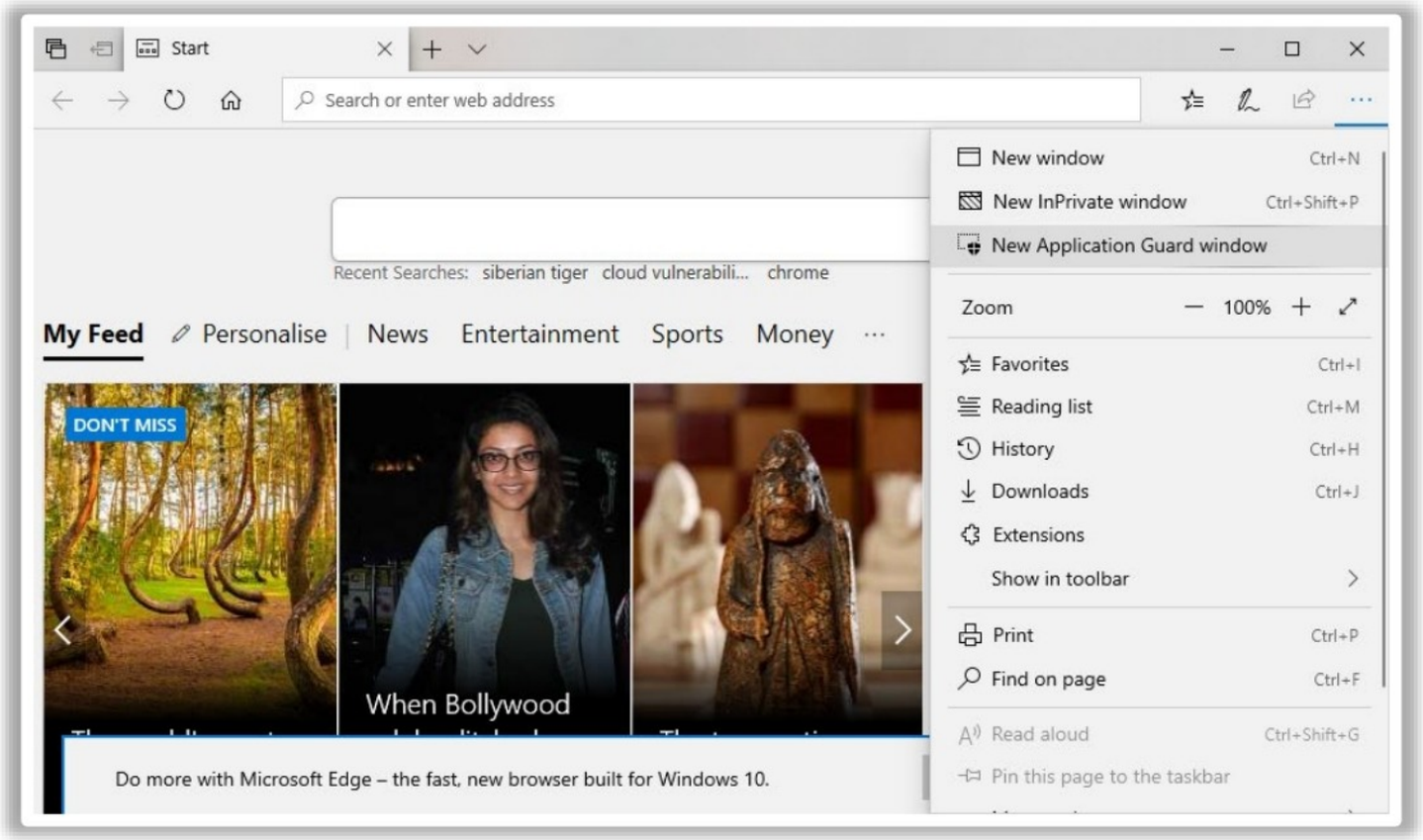


Figure 9.55: Clicking for New Application Guard window

- Wait for WDAG to set up the isolated environment.

Figure 9.56: Setup of the Isolated Envirnonment

- Go to an untrusted but safe URL (in this example, msn.com). View the new Microsoft Edge window, and ensure that the WDAG visual cues are present.

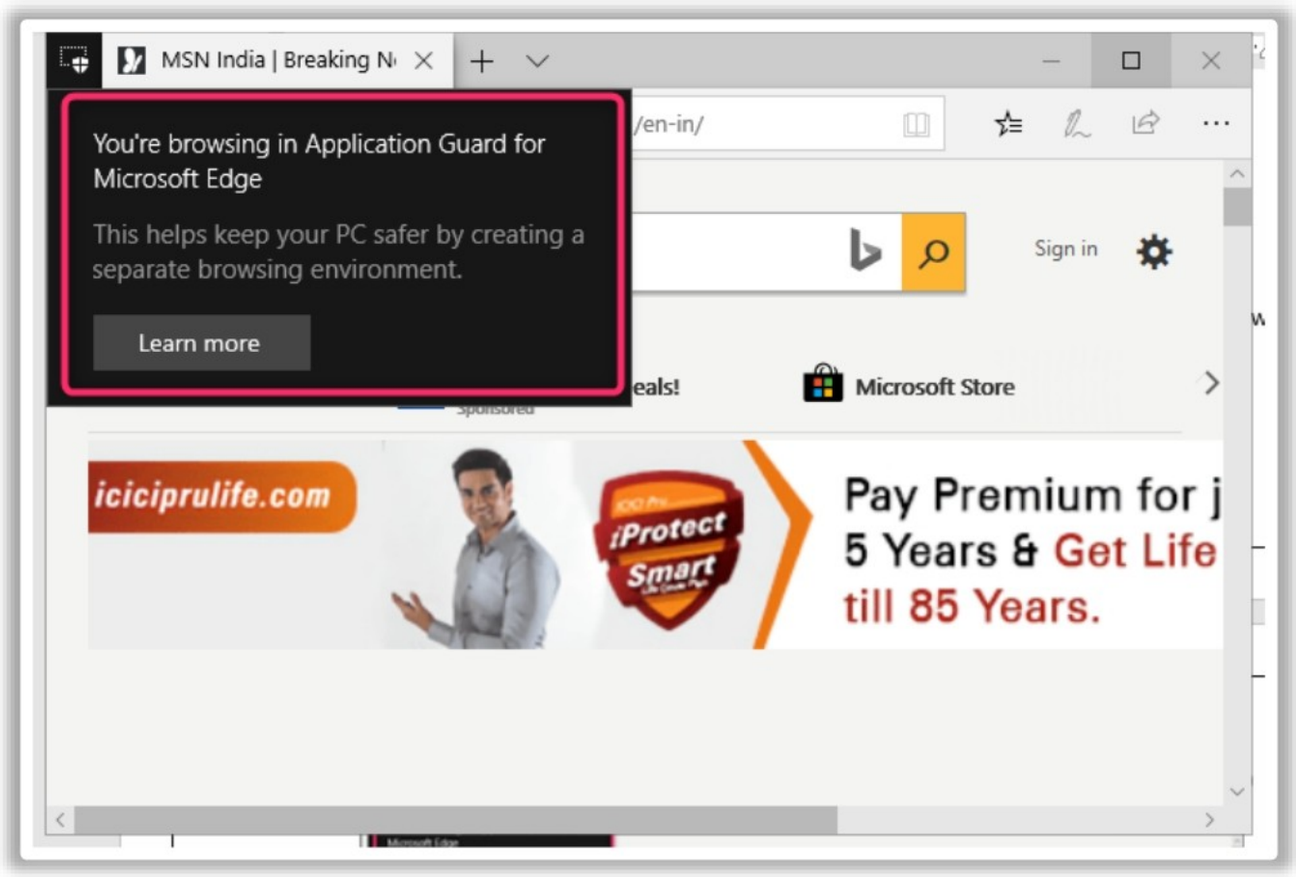


Figure 9.57: Ensuring WDAG Visual Cues are Present

- **The enterprise-managed mode** allows IT professionals to control the tool by adding trusted domains and customizing the experience to meet the needs for employee systems. This mode also redirects browser requests to add non-enterprise domain/domains in the container automatically.

  Steps to test WDAG in the enterprise-managed mode

  To use WDAG in the enterprise mode, use Group Policy to set up the required settings.

  - After installing WDAG, restart the device and start Microsoft Edge.
  - Set up the Network Isolation settings in Group Policy:
    - Go to the **Group Policy/ Edit Group Policy/Administrative Templates/Network/Network Isolation/Enterprise resource domains hosted in the cloud** setting.

- Enter *.microsoft.com* in the **Enterprise cloud resources** box.

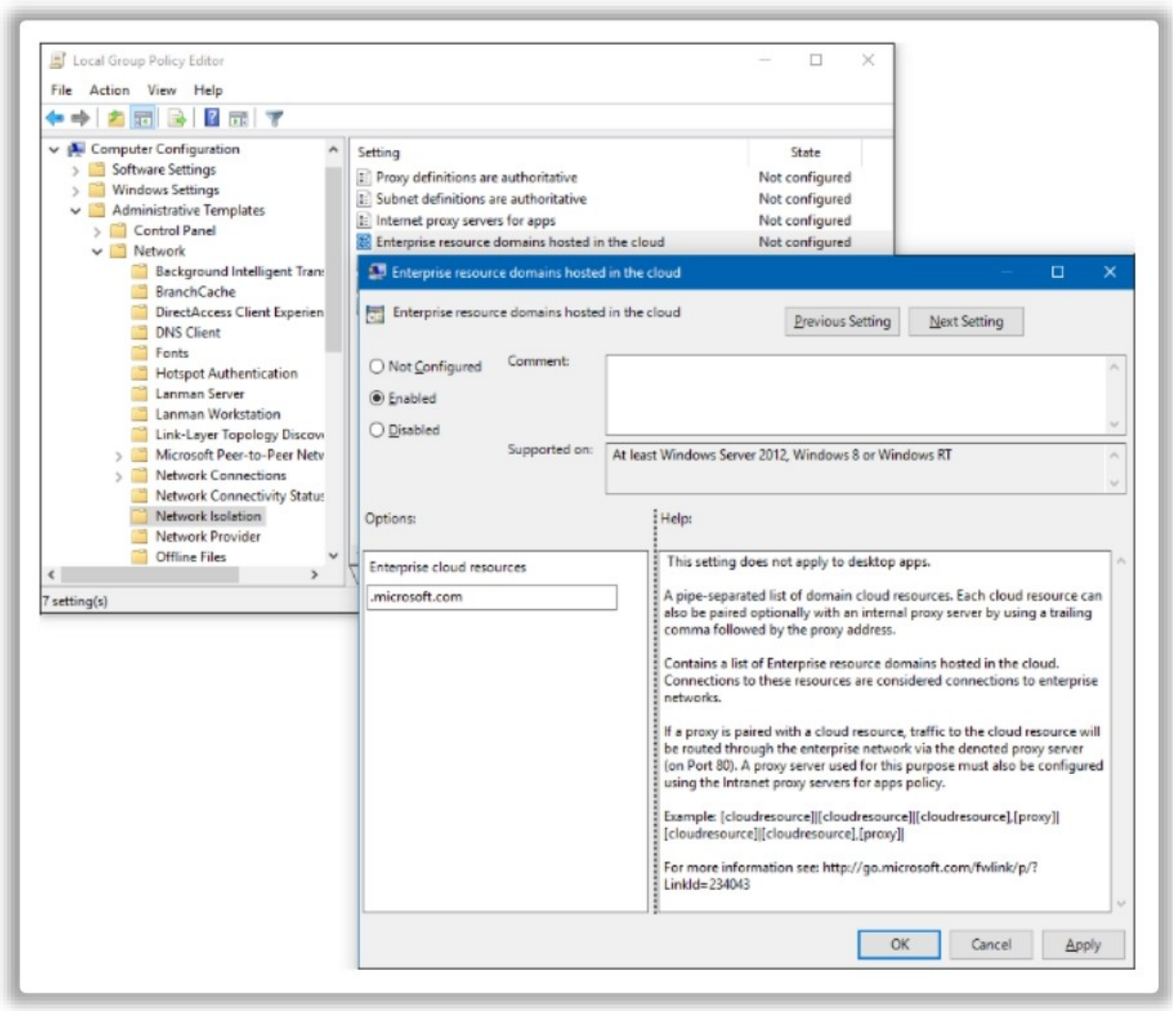


Figure 9.58: Enabling "Enterprise resource domains hosted in the cloud" Policy Setting

- Go to the **Administrative Templates\Network\Network Isolation\Domains categorized as both work and personal** setting.
- Enter *bing.com* in the **Neutral resources** box.

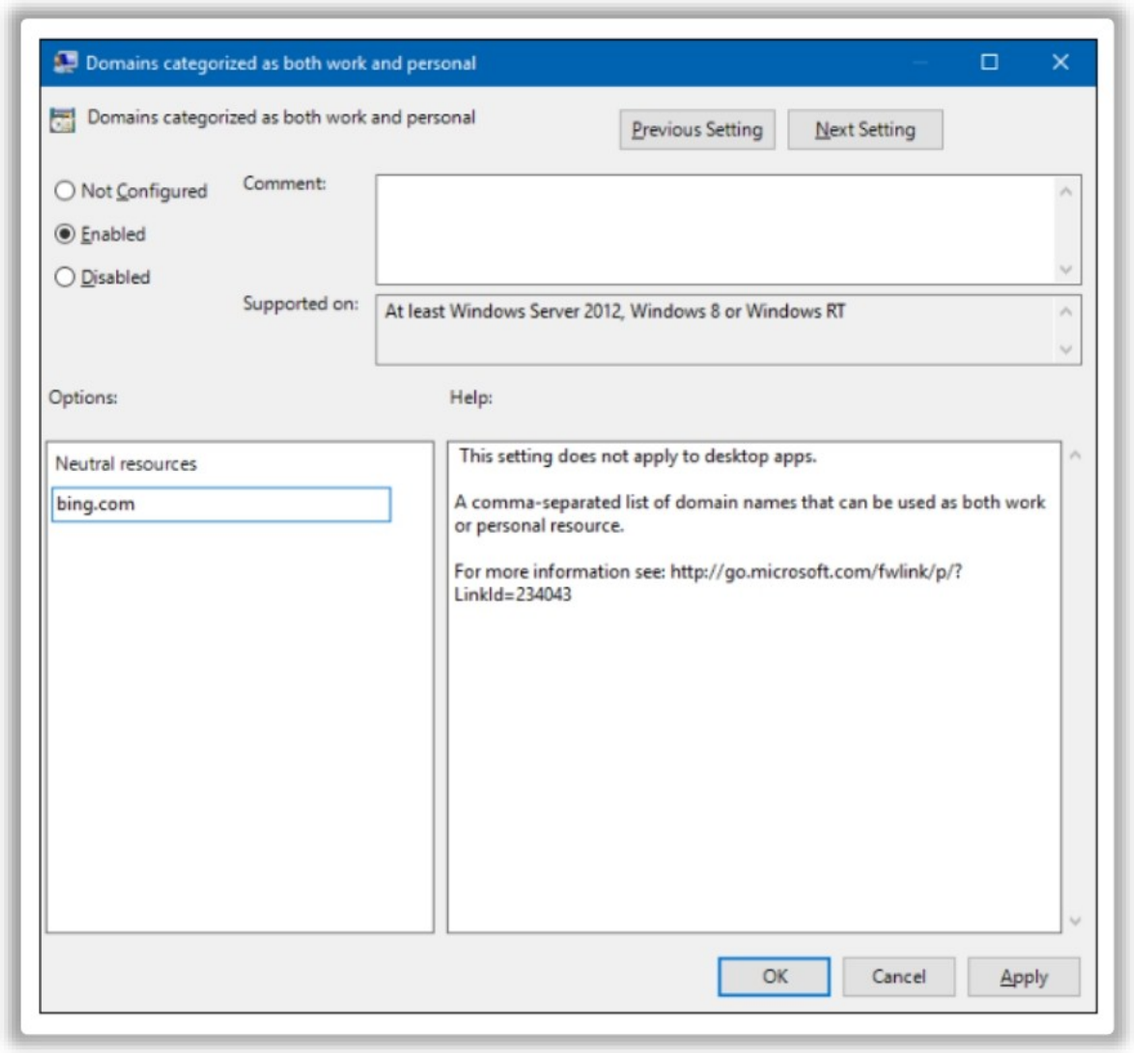


Figure 9.59: Enabling "Domains categorized as both work and personal" Policy Setting

- Go to the **Computer Configuration\Administrative Templates\Windows Components\Windows Defender Application Guard\Turn on Windows Defender Application Guard in Enterprise Mode** setting.
- Click **Enabled**, choose Option **1**, and click **OK**.

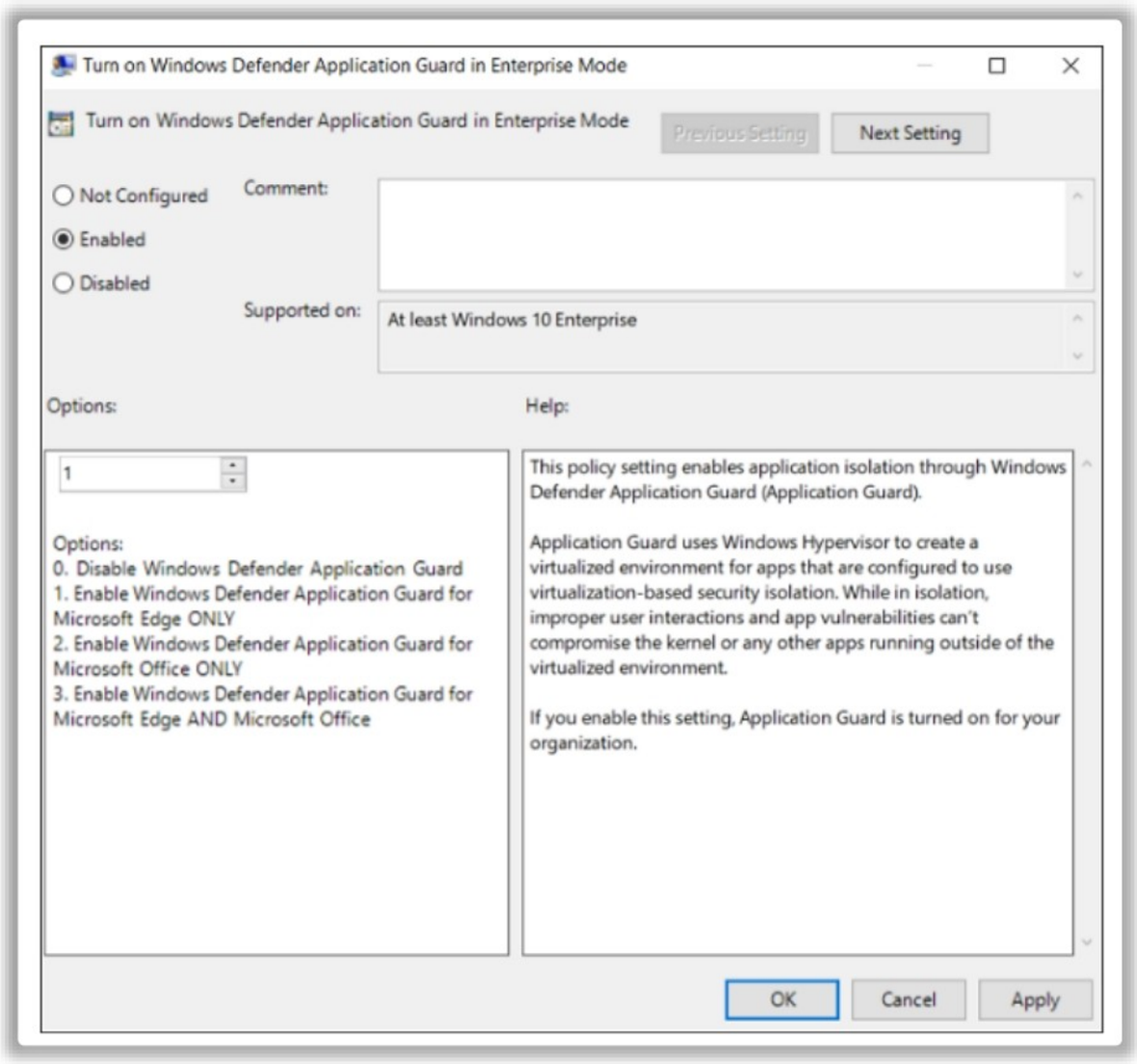
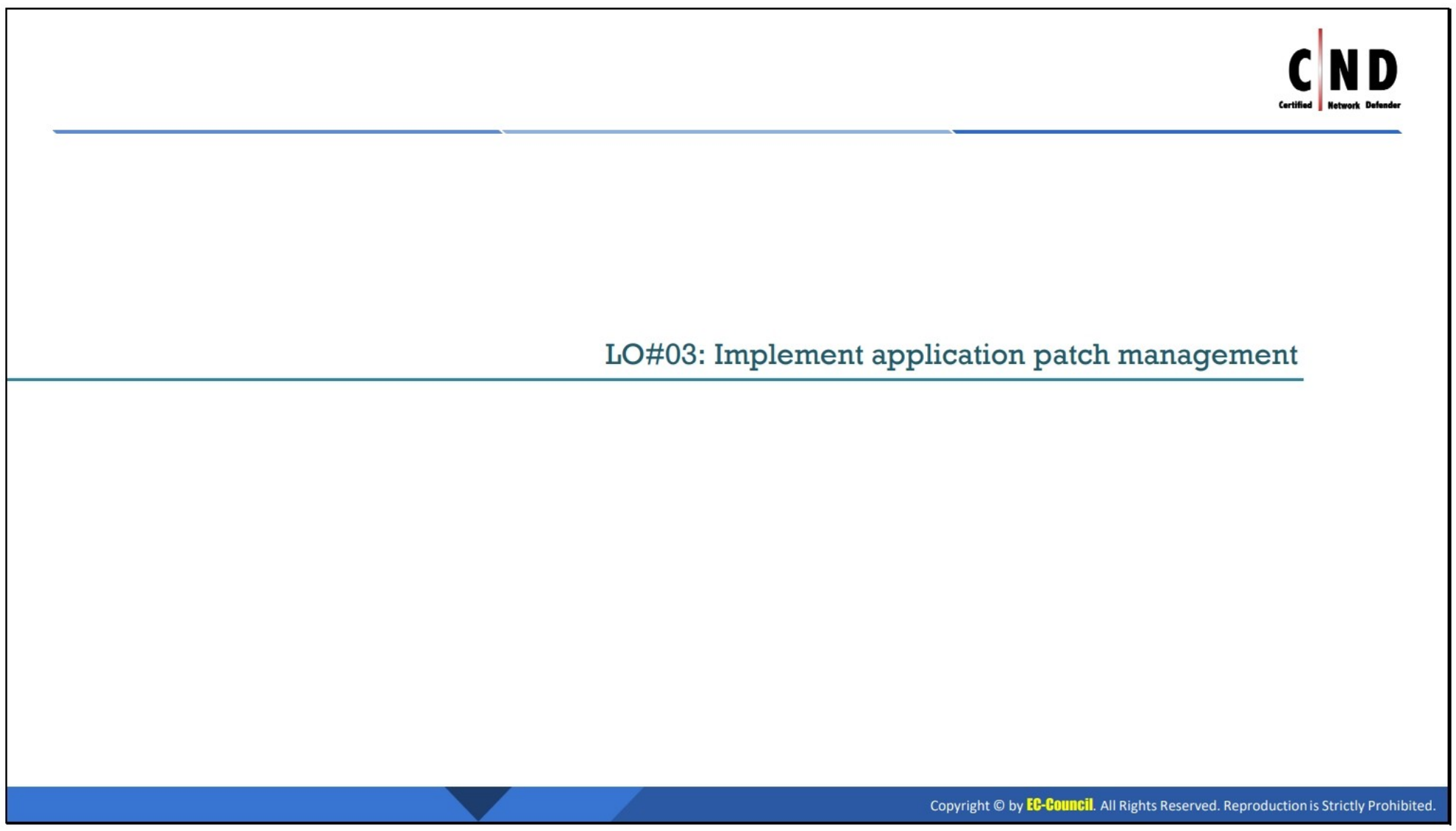


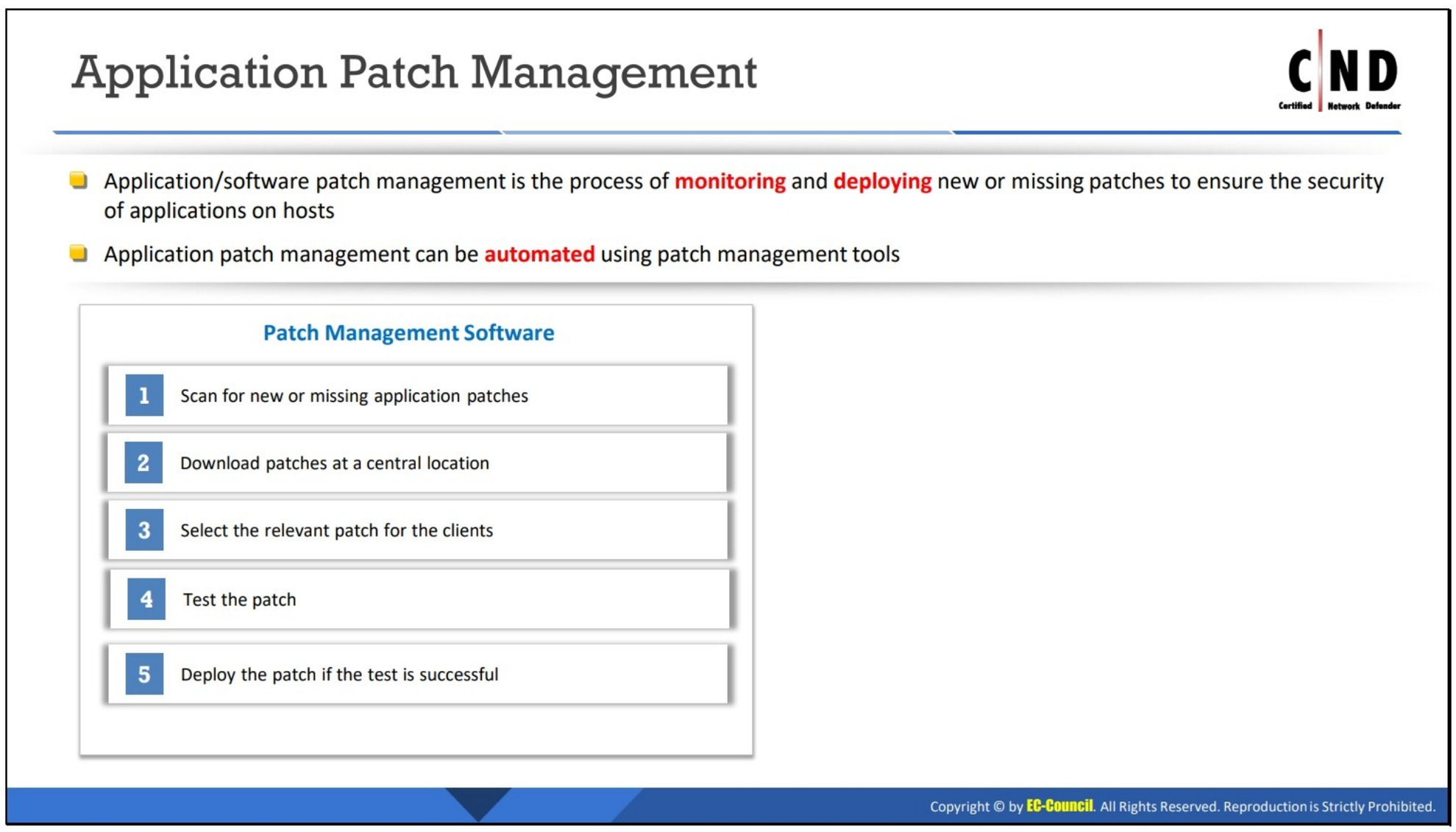
Figure 9.60: Enabling "Turn on Windows Defender Application Guard in Enterprise Mode" Policy Setting

- Start Microsoft Edge and enter *www.microsoft.com* in the address bar. After submitting the URL, WDAG decides that the URL is trusted because it uses the domain marked as trusted and shows the site directly on the host PC instead of in WDAG.
- Type any URL that is not part of the trusted or neutral site lists in the same instance of Microsoft Edge. After submitting the URL, WDAG decides that the URL is untrusted and redirects the request to a hardware-isolated environment.



LO#03: Implement Application Patch Management

Identifying missing patches and installing the correct patches regularly enables a network defender to keep applications up to date. The objective of this section is to impart an understanding of the importance of application patch management. The section discusses various application patch management tools.



Application Patch Management

Application patch management is the process of ensuring the security of applications on hosts by regularly deploying new or missing patches.

Operating unpatched applications places organizations at risk of serious security breaches. Therefore, application patch management is one of the most important tasks in organizations.

Secured application patch management improves the productivity of an organization by reducing application-related failures and saves the costs associated with poor application patch management.

Patch Management Software

Hackers use the information on the vulnerabilities fixed by software patches to create exploits. Attackers use these exploits to compromise unpatched applications. The most important factor here is applying software patches as quickly as possible because attackers can create exploits within hours. The action of applying patch software quickly reduces the window of vulnerability. In these cases, network defenders should use a patch management software. Using manual patch management, rather than systematic and automatic methods, cause expose systems to vulnerabilities because some software patches may be delayed or missed altogether.

In these cases, patch management software enables network defenders to ensure fully patched applications.

- **Key features of most patch management solutions**

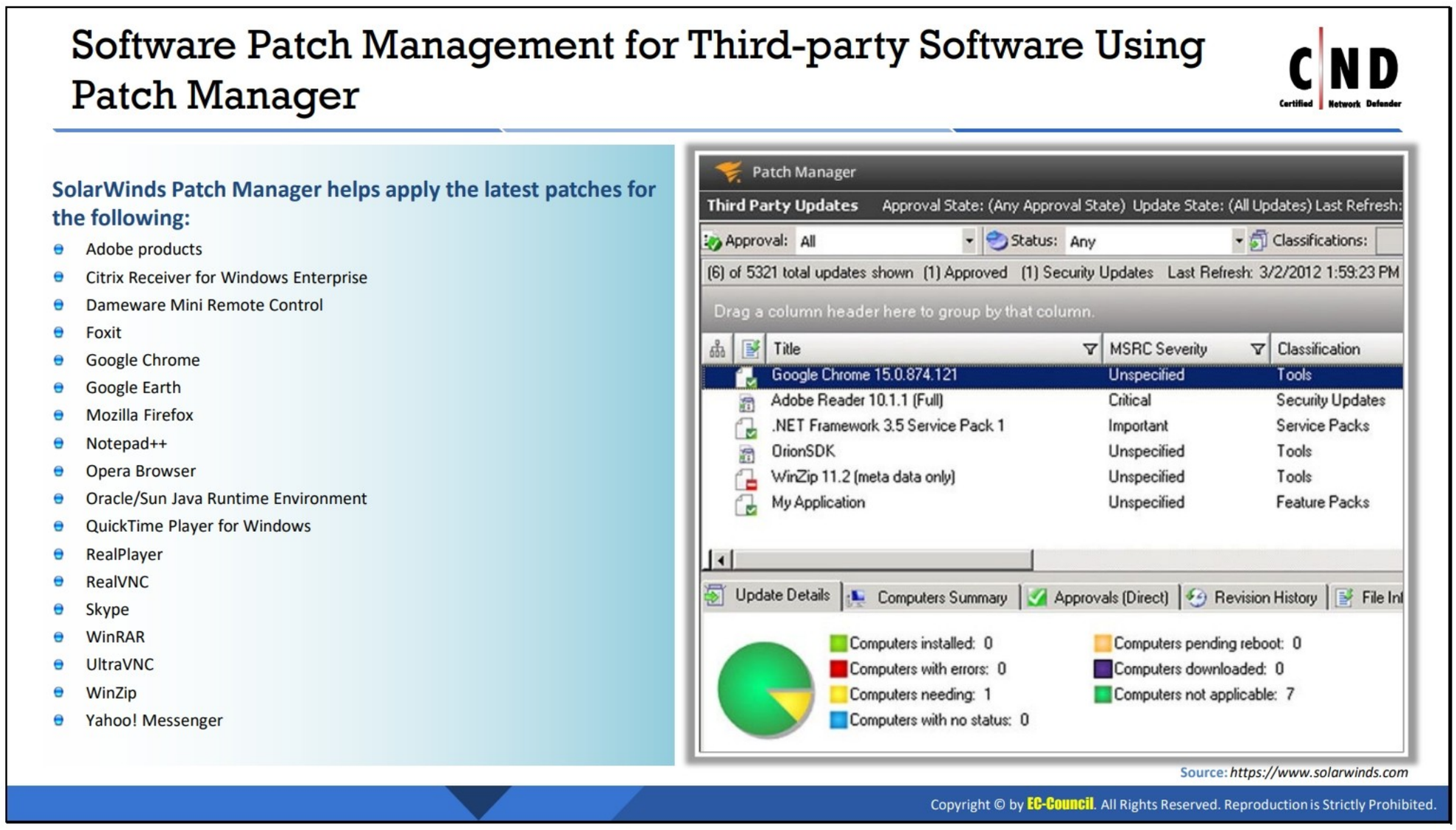
Different patch management solutions offer different levels of functionality. The key features offered by most patch management solutions are as follows.

  - Patch management tools offer the **automation** of application patch management.

- They support **centralized patch management** for patching third-party applications by allowing patch downloads to a central location.
- They **scan** the whole network for connected servers and end users. Later, they detect OSes (any) and other software running on them. A few patch management solutions offer scanning virtual machines and machines running in the cloud.
- The evaluation of the patch status of all applications by scanners (patch status detection) provides network defenders a dashboard that shows the following:
  - Patched and malicious software
  - System flags with unrecognized applications
  - Unknown patch status of applications
  - Software patch application reports to demonstrate compliance

This information enables network defender to determine the order of **testing and application of critical patches**. Network defenders test patches on a few systems before deploying them organization-wide to ensure that patches do not fall through the cracks.

- Patch management tools allow the network defender to **deploy new patches** if a test is successful.
- A few other solutions allow network defenders **to create groups**. This option allows them to apply patches to different groups at different times. This can prevent network congestion.



Software Patch Management for Third-party Software Using Patch Manager

SolarWinds Patch Manager is an automated patch management software for Microsoft servers, workstations, and third-party applications.

Its key features include the following:

- Microsoft Windows Server Update Services (WSUS) server patch management
- Integrations with Microsoft System Center Configuration Manager (SCCM)
- Vulnerability management
- Pre-built/pre-tested packages
- Patch compliance reports
- Patch status dashboard

The SolarWinds Patch Manager features updates for applications to patch the most popular third-party software. They include Adobe, Yahoo! Messenger, Citrix Receiver for Windows Enterprise, Dameware Mini Remote Control, WinZip, Foxit, UltraVNC, Google Chrome, WinRAR, Google Earth, Skype, Mozilla Firefox, RealVNC, Notepad++, Opera Browser, RealPlayer, Oracle/Sun Java, Runtime Environment, and QuickTime Player for Windows.

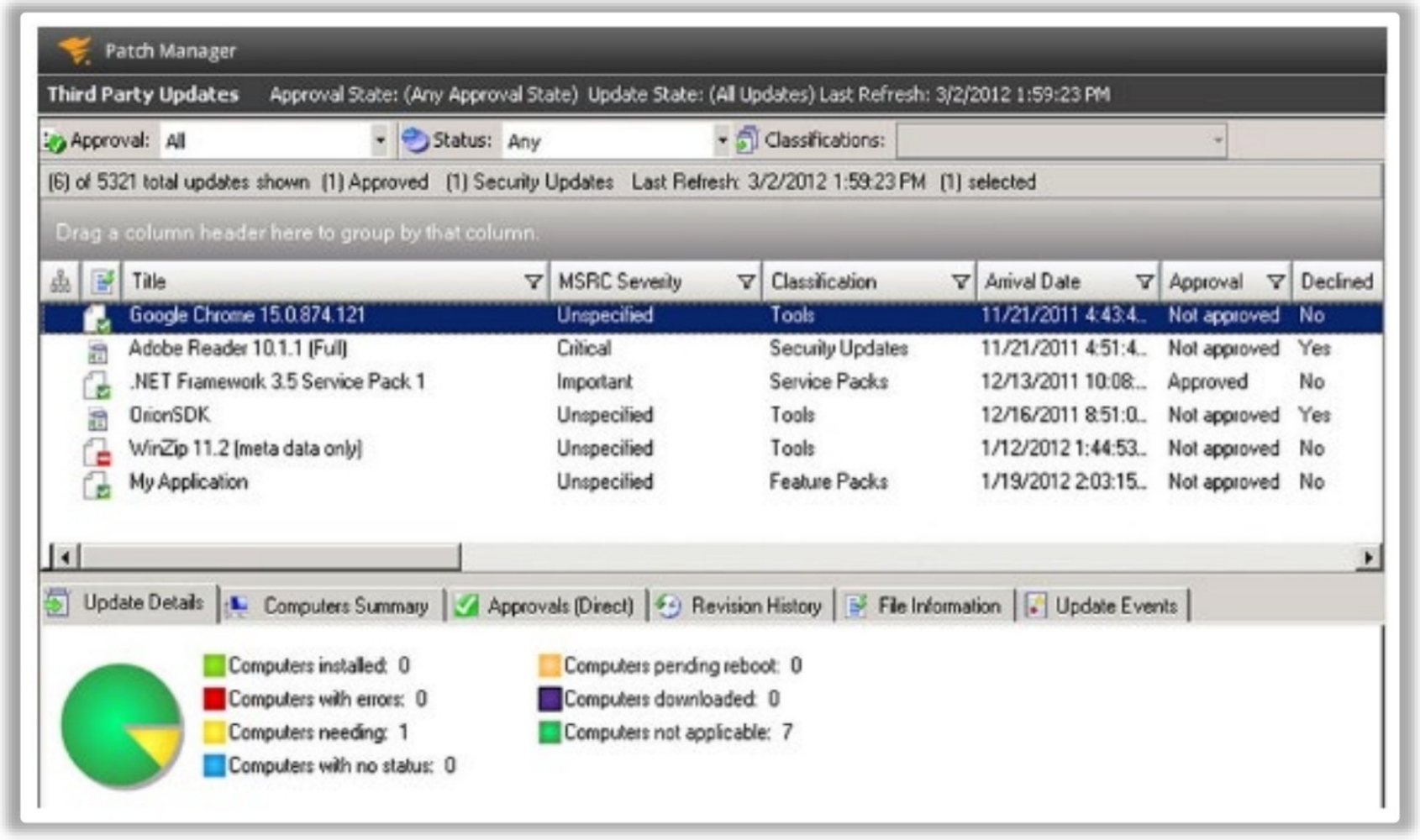


Figure 9.61: SolarWinds Patch Manager

Application Patch Management Solutions and Tools

The following are some application patch management tools.

PDQ Deploy

Source**:** https://www.pdq.com

Update third-party software, deploy custom scripts, and make impactful configuration changes in minutes with PDQ Deploy.

LANDESK Patch Manager

Source: http://www.quantum.com.hk

LANDESK Patch Manager automatically applies patches to applications to simplify administration.

Shavlik Protect

Source: https://www.ivanti.com

Shavlik Protect encompasses Microsoft SCCM, Mac OS, virtual infrastructure, and third-party application patching.

Kaseya Patch Management Software

Source: https://www.kaseya.com

Kaseya Patch Management Software enables the consistent, timely installation of the latest security patches and software updates to keep servers, workstations, and third-party applications up to date.

SOFTWARE VULNERABILITY MANAGEMENT

Source: https://www.flexera.com

Flexera Software Vulnerability Manager can identify vulnerable applications and apply security patches.

HP Touchpoint Manager

Source: https://www8.hp.com

The HP Touchpoint Manager Patch Management service enable the user to track software updates and patches for multiple devices. It gives a complete picture of the status of all software patches for the user's devices. HP Touchpoint Manager can be used to install software updates including third-party patches for applications such as Adobe Acrobat Reader, Java, and iTunes.

Patch for Endpoint Manager

Source: https://www.ivanti.com

Ivanti Patch automatically adds patch management to the endpoint manager environment to evaluate, test, and apply OS and app patches organization-wide. Its key features include the patching of third-party apps, extensive platform support, distributed and remote patching, patch lifecycle management, automated updates, patch whenever and wherever, and a unified endpoint management add-on.

Syxsense

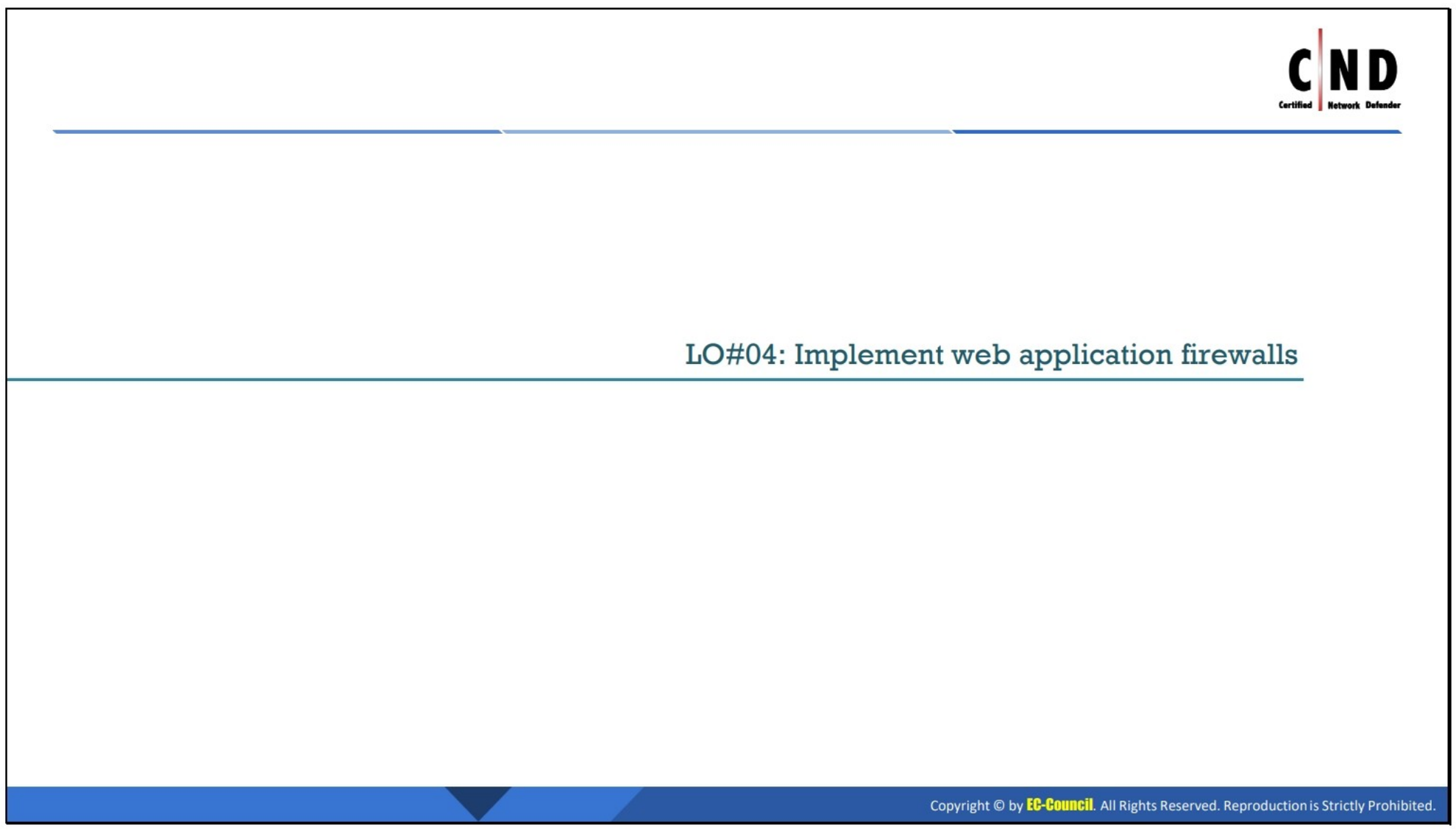
Source: https://www.syxsense.com/

Syxsense gives IT and Security teams visibility and control over their environment: from keeping endpoints up to date with the latest patches to detecting and resolving vulnerabilities before they can be exploited. It enables organizations to easily oversee their environment, stay ahead of potential threats, and prove compliance with real-time data and reporting.

Itarian

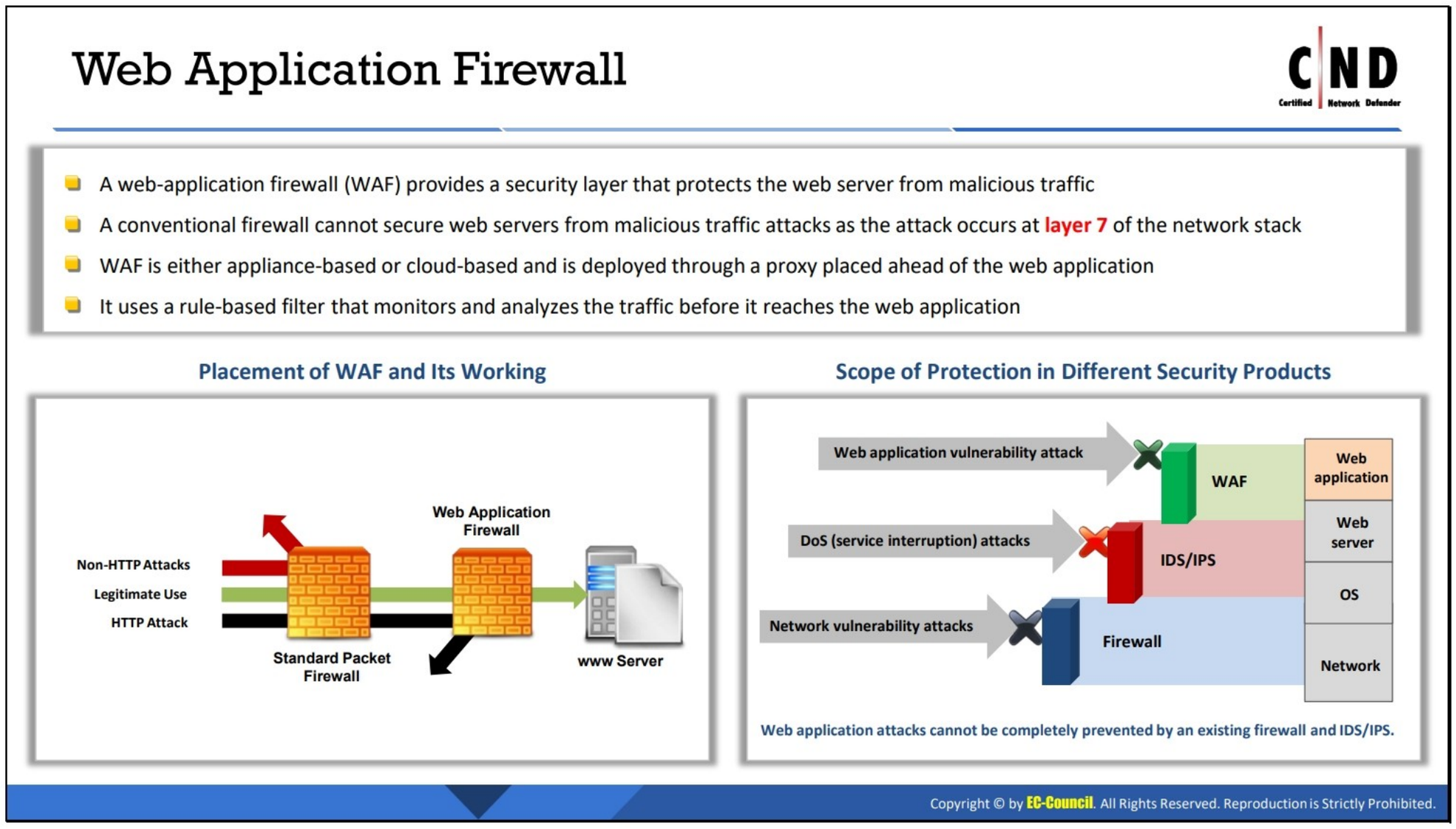
Source: https://www.itarian.com/

Itarian software allows to identify which endpoints contain vulnerabilities and need to be patched, create policies to automatically apply updates to groups on a schedule, and remotely deploy operating system updates.



LO#04: Implement Web Application Firewalls

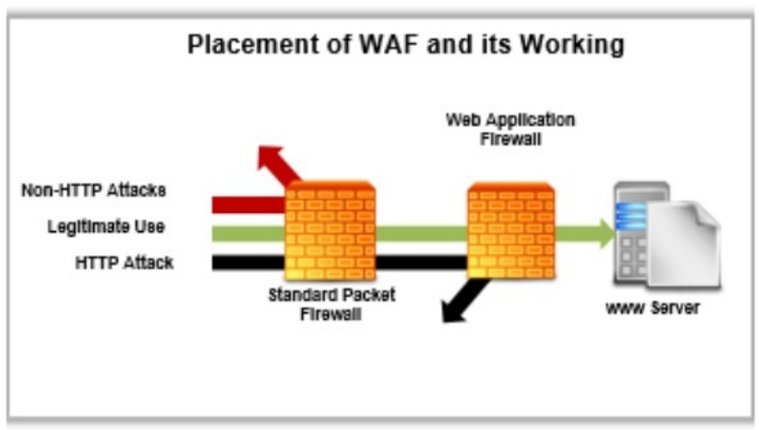
Websites, web applications, and web servers are the main cyber-attack targets. Implementing web-application firewalls (WAFs) protects web servers from various attacks. The objective of this section is to impart an understanding of the advantages and limitations of WAF.



Web Application Firewall

WAF provides a security layer that protects a web server from malicious traffic. A conventional firewall cannot secure web servers from a malicious traffic attack as the attack occurs at layer 7 of the network stack.

WAF is either appliance-based or cloud-based and is deployed through a proxy placed ahead of the web application. It uses a rule-based filter that monitors and analyzes the traffic before it reaches the web application.



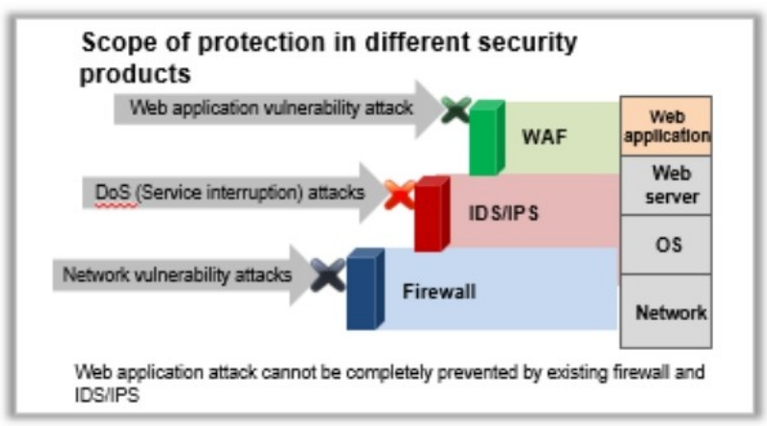
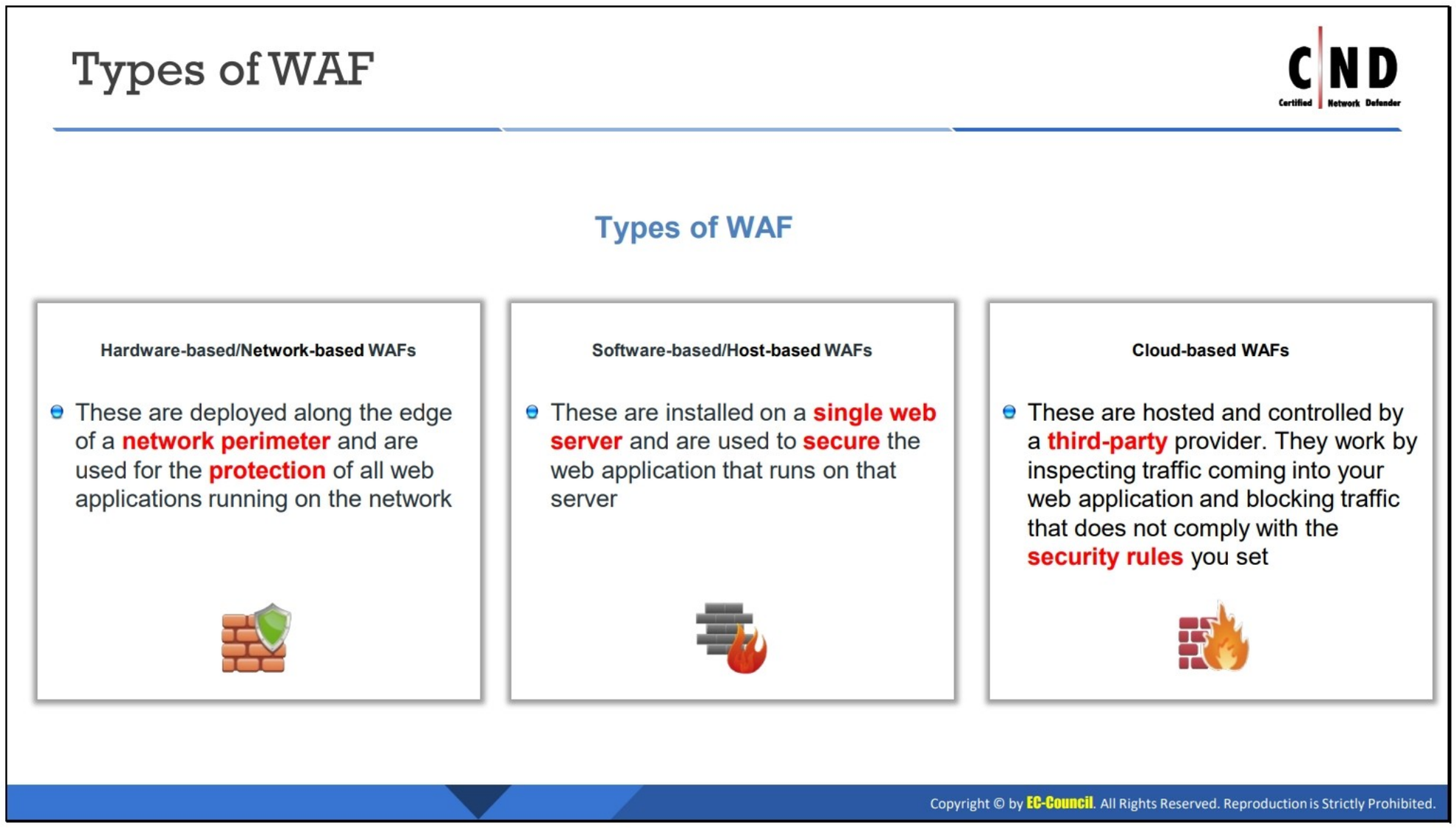


Figure 9.63: Working and Features of Web-Application Firewalls



Types of WAF

There are three types of WAF:

- **Network-based/hardware-based WAFs** are installed at the edge of a network and protect all web applications running on a network. They do this by monitoring incoming traffic to the network and blocking traffic that does not comply with the network's security policies. It can be installed on hardware devices or run on a dedicated server as a software solution.
    - **Advantages of network-based WAF**
        - It protects all web applications on a network
        - It is used to protect against a wide range of threats, including network-based threats
        - It blocks traffic based on a variety of criteria, such as IP addresses and port numbers
    - **Disadvantages of network-based WAF**
        - It requires a dedicated hardware or software solution
        - It requires a significant investment in hardware and maintenance
        - It did not provide as granular a level of control as host-based WAFs
- **Host-based/software-based WAF** is installed on a single web server and is responsible for protecting the web application that runs on that server. It does this by inspecting the traffic that comes into the web application and by blocking any traffic that does not

comply with the security rules that are set up for that web server. It is usually deployed as software solutions running on the web server.

- **Advantages of host-based WAF**
  - It gives the user more control over the protection of their web application.
  - It can be set up on any kind of web server.
  - It does not need special hardware.
- **Disadvantages of host-based WAF**
  - It protects only the web application that is running on the server that it is deployed on.
  - It may need additional resources to operate and maintain.

- **Cloud-hosted WAFs** are hosted and operated by a third-party provider. These WAFs are designed to detect and block any traffic entering a web application that does not adhere to the configured security protocols. Generally, WAFs are deployed as a service and are managed by the WAF provider, who is responsible for managing the necessary hardware and software infrastructure to operate the WAF.
  - **Advantages of cloud-hosted WAFs**
    - It eliminates the need to purchase and maintain hardware and software infrastructure.
    - It is easy to scale up or down as needed.
    - It can be utilized to safeguard web applications hosted on a variety of server types.
  - **Disadvantages of cloud-hosted WAFs**
    - It requires a third-party subscription.
    - It does not offer as much control over WAF setup as an in-house solution.
    - It may not provide the same level of protection as an on-premises WAF, depending on the provider and the specific service being used.

WAF Deployment Options

A WAF comprises various architectures and operating mechanisms that vary in terms of the ease of WAF deployment and resulting WAF functionality.

- **Reverse proxy**

  In the reverse proxy mode, a WAF works as a proxy to the application server. Therefore, the device traffic goes directly to the WAF. The encrypted connections terminated at layer 7 let the WAF decrypt and analyze web traffic. This gives the WAF full control over the traffic in terms of rewriting content based on security mechanisms.

- **Layer-2 bridge**

  In the layer-2 bridge mode, a WAF stays in-line and acts as a layer-2 switch. The WAF monitors incoming requests, performs passive Secure Sockets Layer (SSL) decryption, and blocks traffic by simply dropping packets. This mechanism provides more performance than reverse proxy without the need for many network changes. However, it does not support rewriting content based on security mechanisms. The layer-2 bridge mode is architecturally very similar to the reverse proxy mode.

- **Out of band**

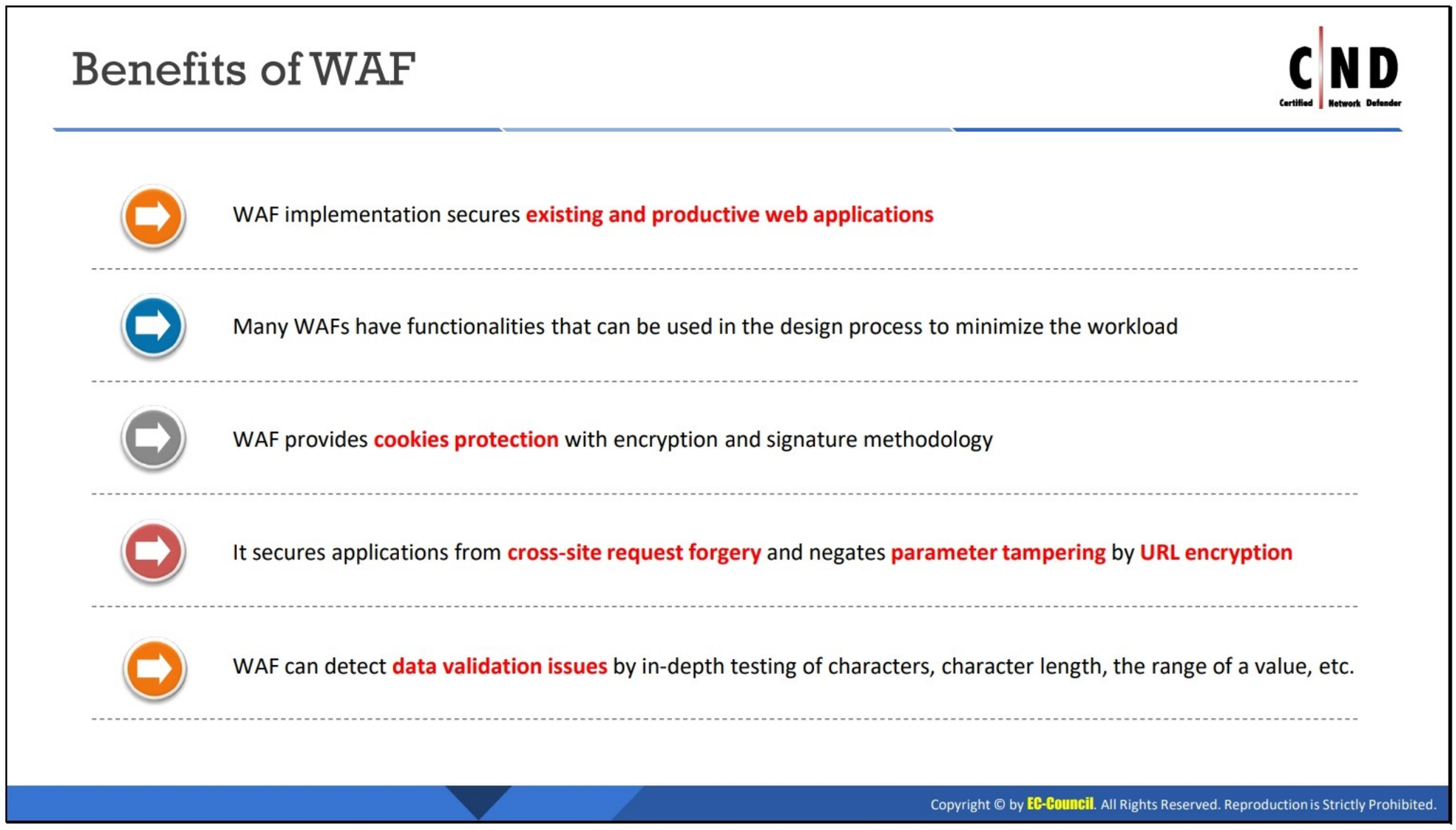
  In the out-of-band method, a WAF does not stay in-line and has the least impact on both applications and the network. The monitoring port in the network sends a copy of incoming traffic to the WAF. Here, the WAF only passively decrypts SSL traffic and transfers Transmission Control Protocol (TCP)-reset packets to block traffic. Here, configuring the WAF to detect malware network traffic prevents the interruption of false-positive traffic, leading to application outages.

- **Server resident**

  A server resident or an embedded WAF is a software installed on the host executing the web server. This can be installed as an application or a server plugin. A server resident WAF creates extra load on the server, and it is not as functional as its network appliance counterparts. Therefore, it is better to check the server utilization resources before installing the WAF.

- **Internet hosted/cloud**

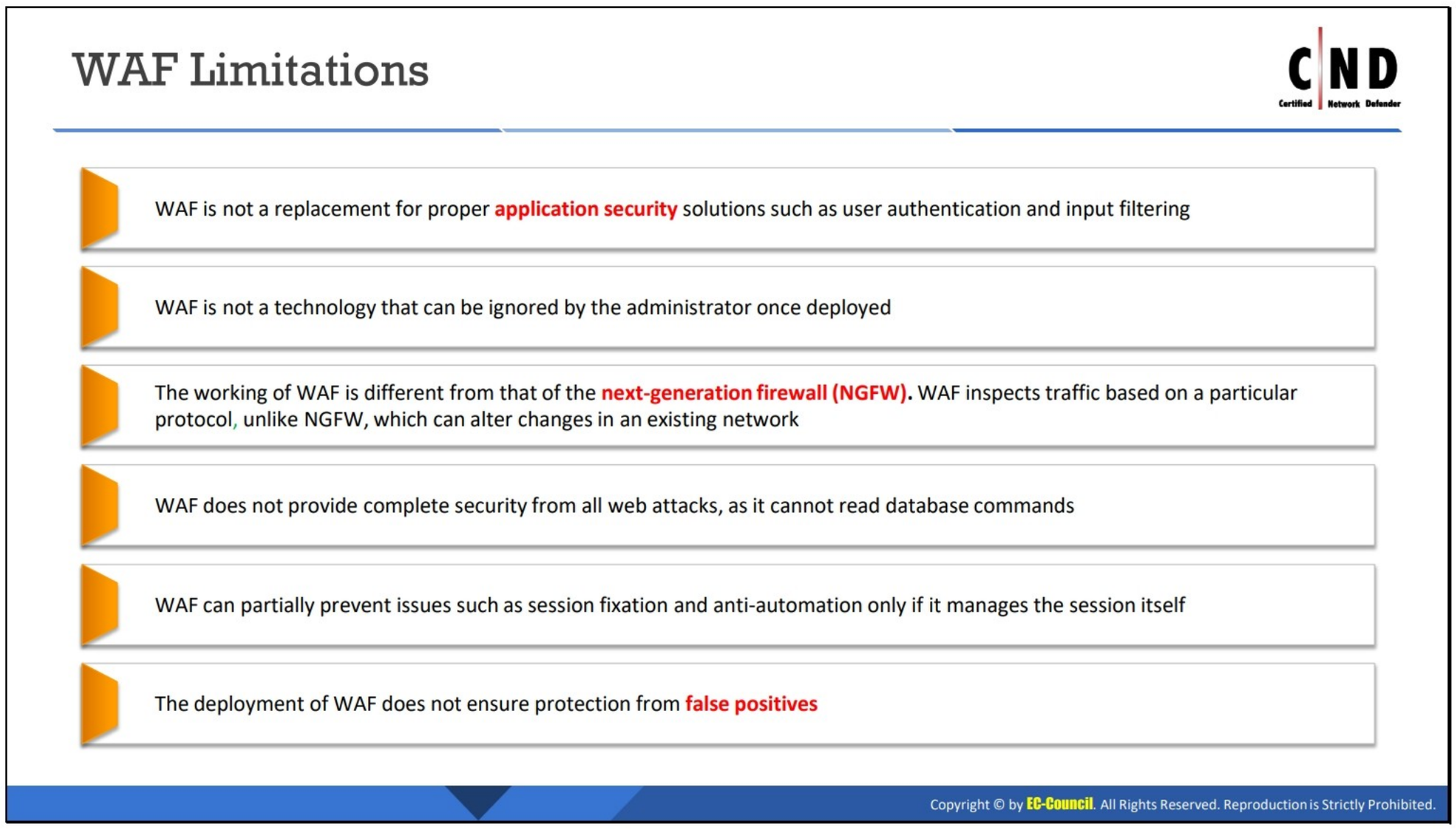
  Using a cloud provider to implement a WAF works like the reverse proxy mode. Here, a Domain Name System (DNS) is configured to a point in the cloud, which creates another connection to the web application. Though it is increasingly a popular option for WAF deployment, it has a drawback. The WAF implementation is not under the control of organizations, necessitating reviews to ensure that compliance requirements are met by the cloud provider.



Benefits of WAF

The benefits of WAF that can help an organization strengthen its web application security from evolving threats include the following:

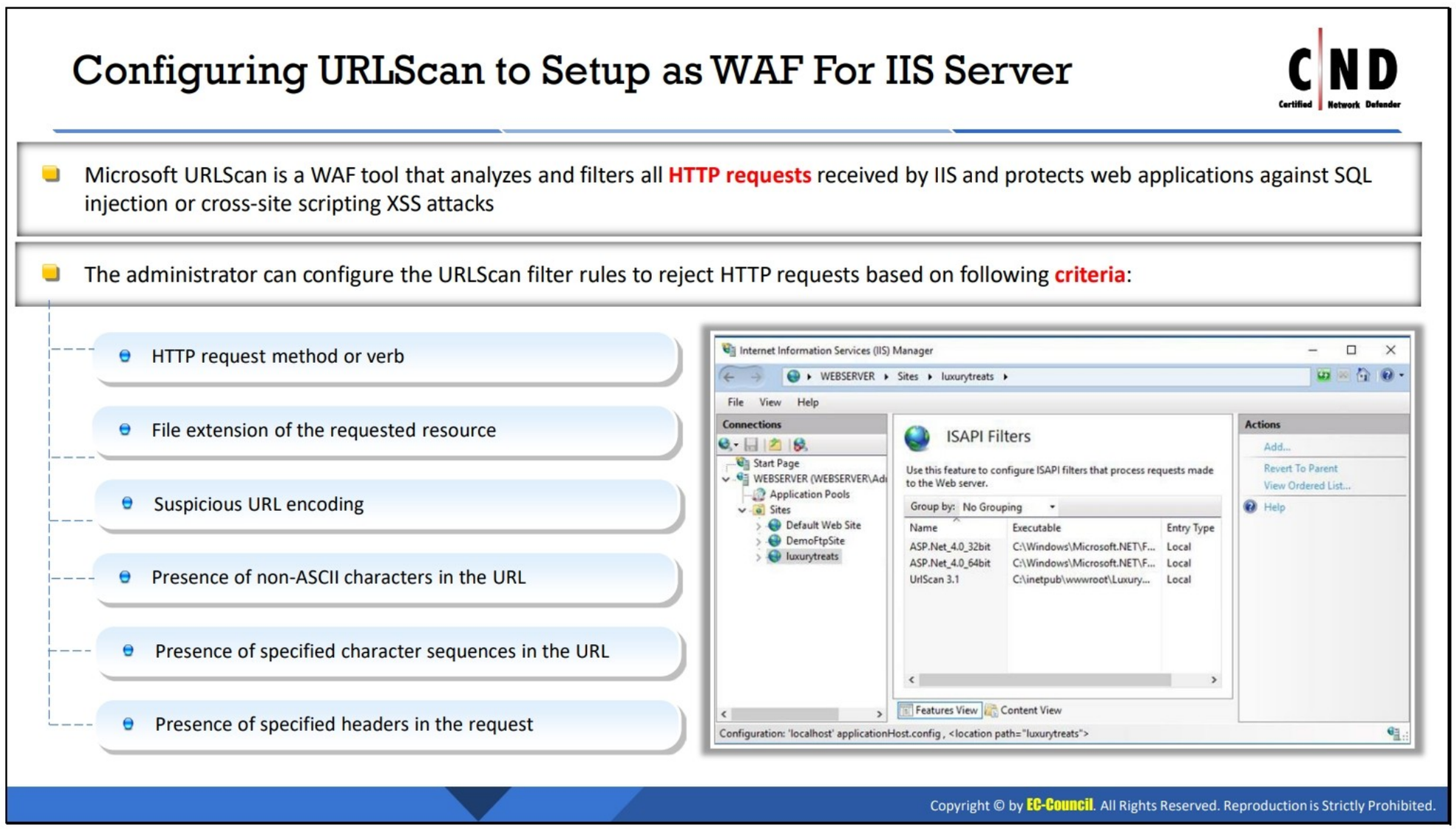
- WAF implementation secures **existing and productive web applications.**
- Many WAFs have functionalities that can be used in the design process to minimize the workload.
- It provides **cookies protection** with encryption and signature methodology.
- It secures applications from **cross-site request forgery** and negates **parameter tampering** by **URL encryption.**
- A WAF can detect **data-validation issues** through the in-depth testing of characters, character length, the range of a value, etc.
- It allows network defender to **illustrate compliance with regulatory standards** such as Payment Card Industry (PCI), Health Insurance Portability and Accountability Act (HIPAA), and General Data Protection Regulation (GDPR).



WAF Limitations

The limitations of WAF include the following:

- WAF is not a replacement for proper **application security** solutions such as user authentication and input filtering.
- WAF is not a technology that can be ignored by the network defender once deployed.
- The working of WAF is different from that of the **next-generation firewall (NGFW).** WAF inspects traffic based on a particular protocol, unlike NGFW, which can make changes in an existing network.
- WAF does not provide complete security from all web attacks, as it cannot read database commands.
- WAF can partially prevent issues such as session fixation and anti-automation only if it manages the session itself.
- The deployment of WAF does not ensure protection from **false positives.**



Configuring URLScan to Setup as WAF For IIS Server

Microsoft URLScan is a WAF tool that analyzes and filters all Hypertext Transfer Protocol (HTTP) requests received by the Internet Information Service (IIS) web service and protects web applications against Structured Query Language (SQL) injection or cross-site scripting (XSS) attacks. It can log requests to allow the diagnosis of attempts to upset a server. If a request is identified as a risk, the script immediately returns an HTTP 404 message to the client. This mechanism protects the script, website, and server.

Its key features include denying rules independently, a global DenyQueryString section for adding deny rules, a global AlwaysAllowedUrls section to specify safe query strings, the use of escape sequences, installation of multiple URLScan instances, propagating configuration change notifications to IIS worker processes, and enhanced W3C formatted logging.

An administrator can configure URLScan filter rules to reject HTTP requests based on the following criteria:

- HTTP request method or verb
- File extension of the requested resource
- Suspicious URL encoding
- Presence of non-ASCII characters in the URL
- Presence of specified character sequences in the URL
- Presence of specified headers in the request

Additional WAF Solutions

The implementation of a WAF is one of the best methods to protect websites from online threats. WAF allows the customization of rules set by identifying and blocking malware traffic. The following are some WAFs that can be useful for web application security.

F5 NGINX App Protect WAF

Source: www.nginx.com

F5 NGINX App App Protect WAF allows enforcing, automating and scaling the app and API security across distributed architectures and hybrid environments with lightweight, high-performance, advanced protection against Layer 7 attacks.

NAXSI

Source: https://github.com

NAXSI is referred to as a positive-model application firewall. It is an open-source, high-performance, and low-rule-maintenance WAF for Nginx. Its key features include good resilience against attacks, the fact that attack signatures need not be updated, and low administration knowledge requirements.

WebKnight

Source: http://www.aqtronix.com

WebKnight WAF is an Internet Server Application Programming Interface (ISAPI) filter that secures web servers by blocking bad requests. It is used for Microsoft IIS protects against SQL injection, buffer overflow, directory transversal, character encoding, etc.

Cloudflare WAF

Source: https://www.cloudflare.com

Cloudflare WAF can be used to protect web applications from malicious attacks. Its dashboard enables users to create rules. Every HTTP request is monitored against the rule engine and the threat intelligence based on various sources. Any suspicious requests can be blocked, challenged, or logged as per the requirement.

Shadow Daemon

Source: https://shadowd.zecure.org

Shadow Daemon is an open-source software that filters requests from malicious parameters to detect, record, and prevent web attacks.

Wallarm

Source: https://www.wallarm.com/

The Wallarm platform continuously analyzes application traffic and mitigates malicious requests in real time. The Wallarm solution protects APIs, microservices, and web applications from OWASP API Top 10 threats, API abuse, and other automated threats. It detects input validation attacks in any requested part including binary files like SVG, JPEG, PNG, GIF, PDF, etc. using the listed tools.

NetScaler App Firewall

Source: https://citrix.com

Citrix Web App Firewall protects web applications and sites from both known and unknown attacks, including application-layer and zero-day threats. Citrix Web App Firewall delivers comprehensive protection without degrading performance or application response time. It can be used as a cloud solution or integrated within the Citrix application delivery controller (ADC) platform.

AppWall

Source: https://www.radware.com

AppWall provides complete protection against web application attacks, web application attacks behind content delivery networks (CDNs), API manipulations, advanced HTTP attacks (Slowloris, dynamic floods, etc.), brute-force attacks on login pages, and so on through the fast, reliable, and secure delivery of mission-critical web applications for corporate networks and in the cloud.

Barracuda WAF

Source: https://www.barracuda.com

Barracuda WAF protects applications, APIs, and mobile app backends against a variety of attacks including Open Web Application Security Project (OWASP) Top 10, zero-day threats, data leakage, and application-layer denial of service (DoS) attacks.

Qualys WAF

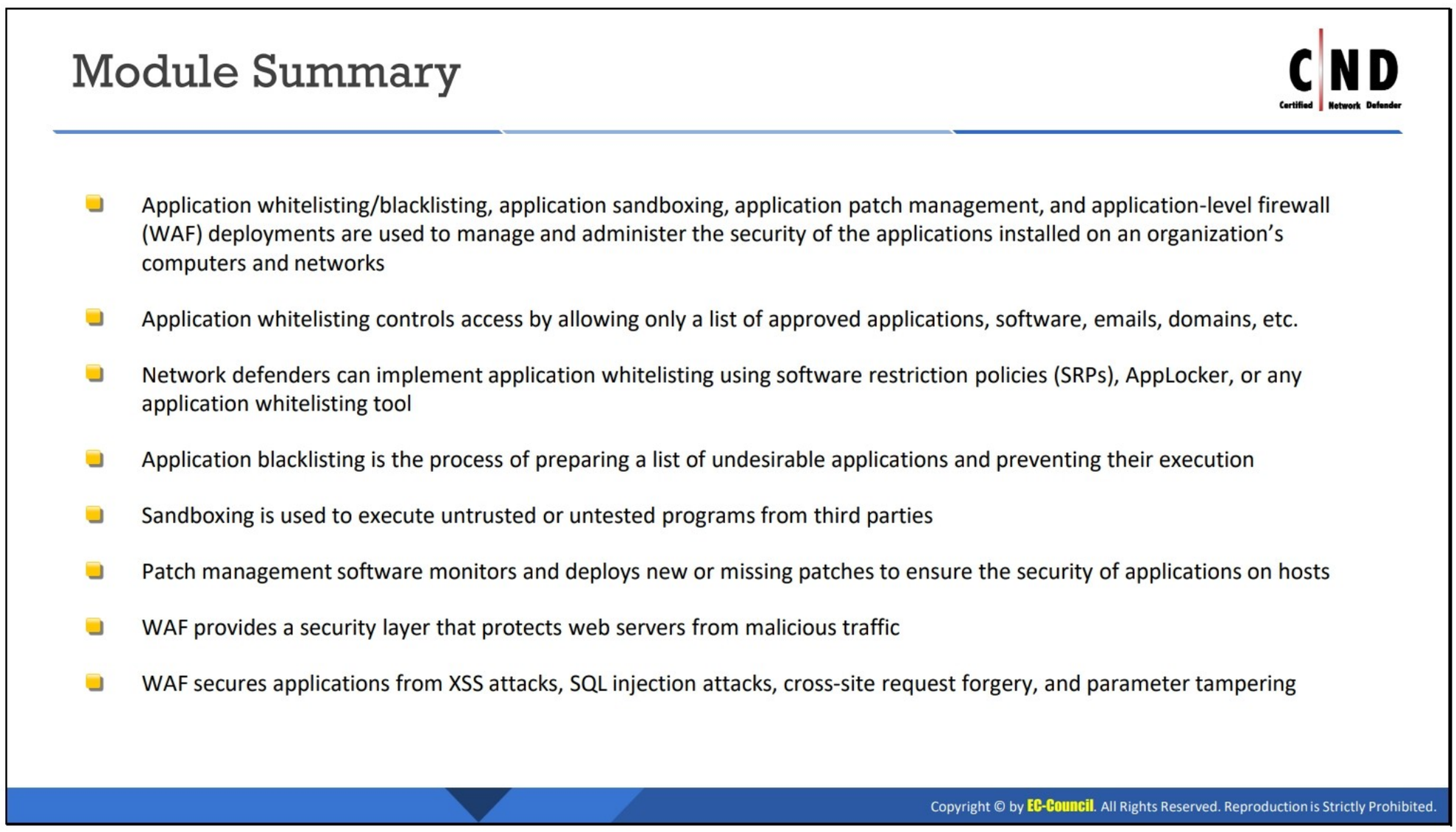
Source: https://www.qualys.com

Qualys WAF, powered by the Qualys Cloud Platform, block attacks and patches web-application vulnerabilities. It allows the user to deploy multiple firewall instances for web applications.

FortiWeb web application firewall (WAF)

Source: https://www.Fortinet.com

FortiWeb WAF and API protection offers machine-learning-enabled protection for business-critical applications from known and unknown vulnerabilities.



Module Summary

This module discussed various administrative activities related to application security. It described solutions such as application whitelisting, application blacklisting, application sandboxing, application patch management, and web-application firewalls (WAFs) implemented to manage and secure installed applications.

The following key points have been highlighted in this module:

- Application whitelisting controls access by allowing only a list of approved applications, software, emails, domains, etc.
- Network defender can implement application whitelisting using software restriction policies (SRPs), AppLocker, or any application whitelisting tool.
- Application blacklisting is the process of preparing a list of undesirable applications and preventing their execution.
- Sandboxing is used to execute untrusted or untested programs from third parties.
- Patch management software monitors and deploys new or missing patches to ensure the security of applications on hosts.
- WAF provides a security layer that protects web servers from malicious traffic.
- WAF secures applications from XSS attacks, SQL injection attacks, cross-site request forgery, and parameter tampering attacks.

This page is intentionally left blank.