Hard_Configurator - Manual
Version 6.1.1.1

Copyright: Andrzej Pluta, @Andy Ful

Developer Web Page: https://github.com/AndyFul/Hard_Configurator/

Malwaretips forum thread:
https://malwaretips.com/threads/hard_configurator-windows-hardening-configurator.66416/

<u>Distribution</u>
This software may be freely distributed as long as no modification is made to it.

<u>Disclaimer of Warranty</u>
THIS SOFTWARE IS DISTRIBUTED "AS IS". NO WARRANTY OF ANY KIND IS EXPRESSED OR IMPLIED. YOU USE IT AT YOUR OWN RISK. THE AUTHOR WILL NOT BE LIABLE FOR DATA LOSS, DAMAGES, LOSS OF PROFITS OR ANY OTHER KIND OF LOSS WHILE USING THIS SOFTWARE.

TABLE OF CONTENTS

INTRODUCTION

Hard_Configurator is currently a flexible and quite complex piece of software. It can apply many security configurations. Some of them can highly restrict Windows and lock down the system - others can only restrict concrete Windows features or Administrative tools.

Hard_Configurator works on Windows Vista and higher versions. It is intended for users, who want to apply some extended features of Windows built-in security. This program can manage Windows built-in Software Restriction Policies (SRP), forced SmartScreen (on Windows 8+), advanced Microsoft Defender configuration (on Windows 10+), and some well-known Windows Policies. It can be used to harden MS Office, Adobe Acrobat Reader, LOL-Bins (via SRP or Windows Firewall), popular archivers and email clients.

The Hard_Configurator **Recommended Settings** are prepared to keep the balance between usability and security in the home environment. They can be slightly different on different Windows versions because each new Windows version has got stronger protection. So, on Windows 8+ with Recommended Settings, the "Install By Smart-Screen" entry in the right-click Explorer context menu is available. It forces SmartScreen check before running EXE / MSI files and allows the user to bypass safely the default-deny SRP without elevating the application privileges.
The "Install By SmartScreen" entry was implemented in Hard_Configurator ver. 5.0.0.1 and replaced the "Run As SmartScreen" entry.

1. The actual status of all restrictions is visible in 2 panels (VALUE columns), on the left and the right side of the GUI window.
2. The green **<Recommended Settings>** button can be used to recover the Hard_Configurator Recommended Settings. This will delete all previous settings except entries added by the user to the Whitelist. You can adjust SRP settings when pressing buttons in the left panel, and non-SRP settings by pressing buttons in the right panel.
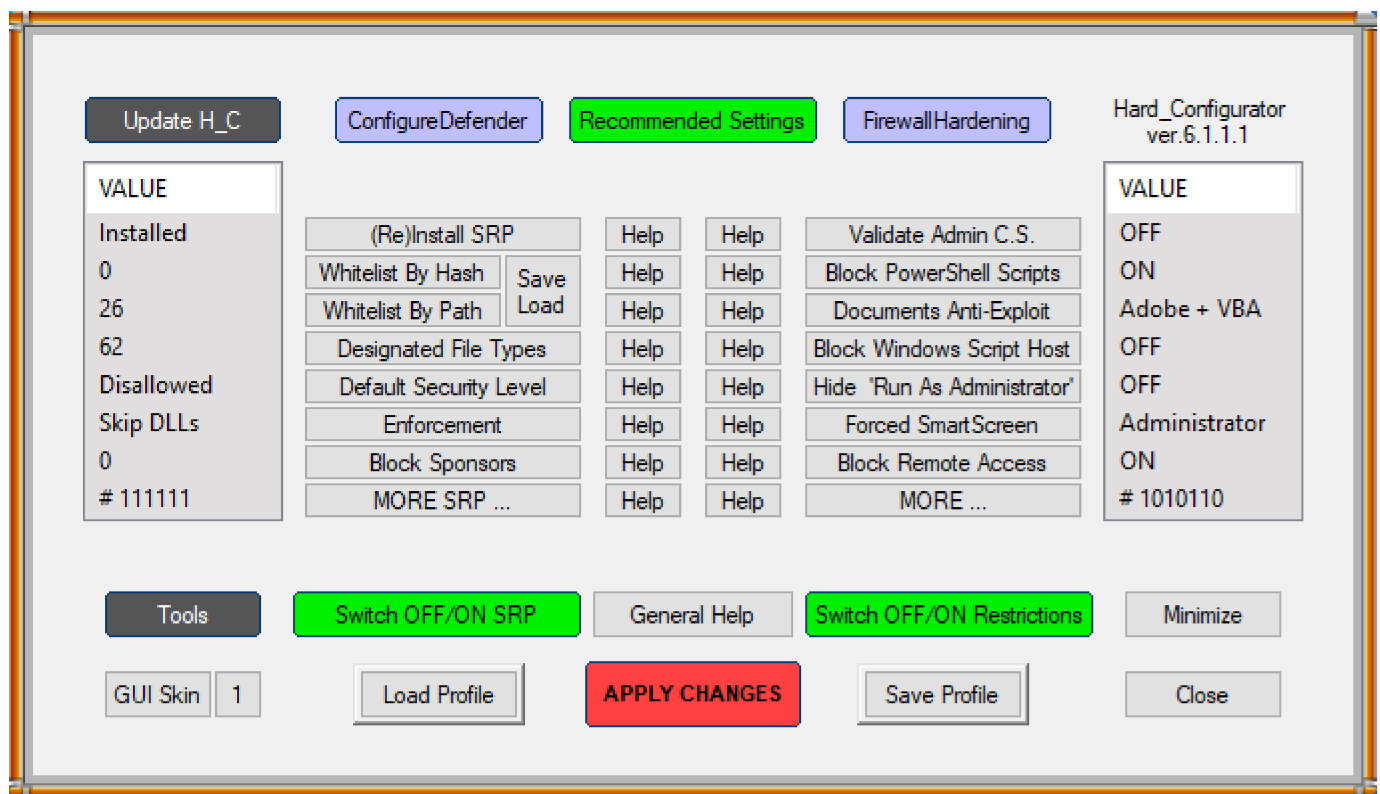
3. Two violet buttons: **<ConfigureDefender>** and **<FirewallHardening>** can be used to run the external tools. They can configure the advanced settings of Microsoft Defender and Windows Firewall. When using the Recommended Settings, it is advisable to set ConfigureDefender "High Protection Level" and apply FirewallHardening "Recommended H_C" rules.

4. There are two important switch buttons <Switch OFF/ON SRP> and <Switch OFF/ON Restrictions>. In the ON state, the switch button is green. When switched OFF, the button changes its background color to blue, and the settings prior to switching are remembered. The ON state can be restored when pressing the switch blue button (the button background color will change to green). The OFF state is also cleared when applying <Recommended Settings> or when loading any setting profile.
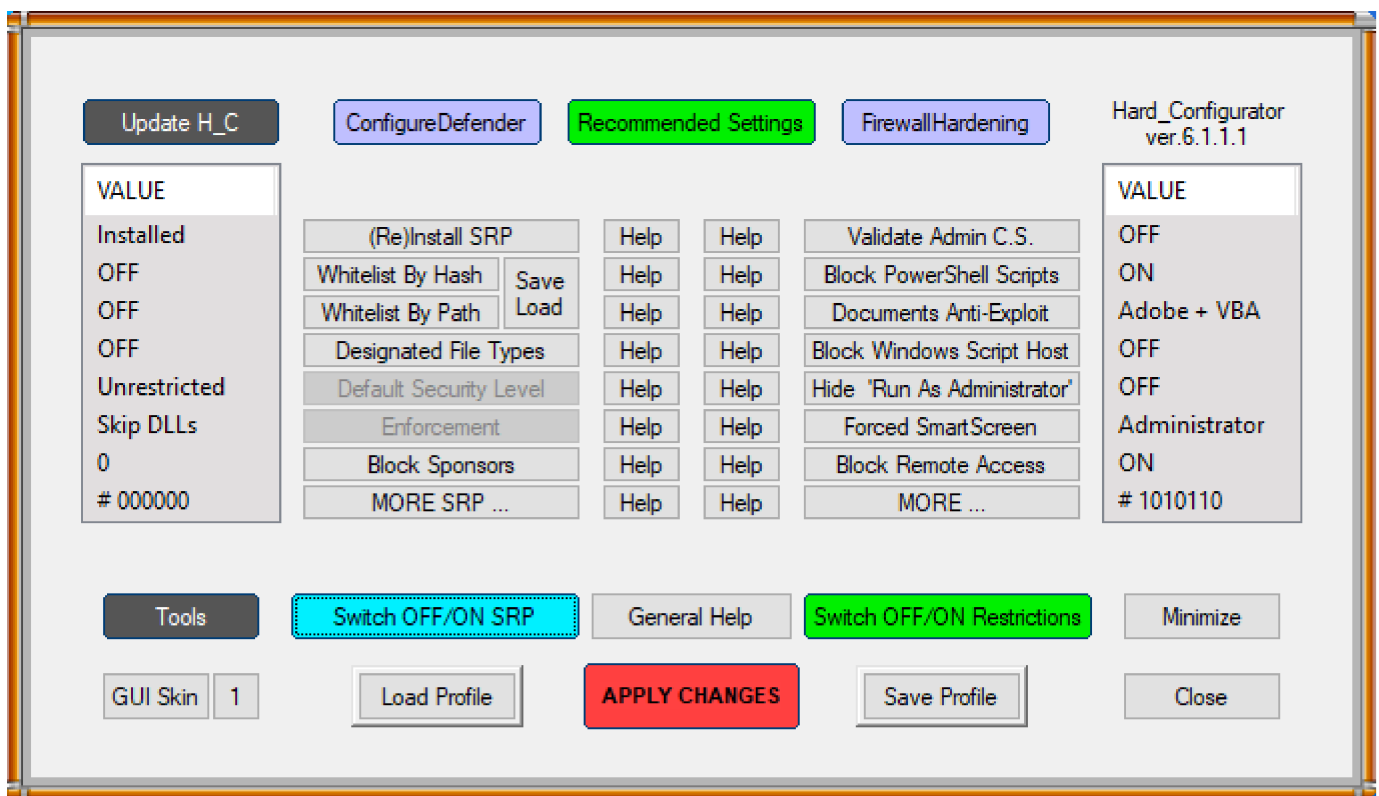
5. The red button <mark>**<APPLY CHANGES>**</mark> works as follows:
   - The "RESTART COMPUTER" alert is displayed, when changes related to WIndows services have to be applied.
   - The "LOG OFF" alert is displayed when SRP <Enforcement>, or <Hide 'Run As Administrator'> settings have been changed.
   - The splash window "Refreshing SRP Rules" is displayed, when only SRP rules have been updated.
   - Otherwise, the splash window "No need to refresh settings." is displayed. This does not mean that no changes have been applied, because many Windows Policies are applied immediately after writing the values into the Registry.
6. If some buttons are grayed out, it means that those options are not supported by Operating System or actual settings do not allow applying them.
7. When configured on the particular account, the changes apply to all accounts (except some whitelisted entries related to the particular account).
8. For SRP restrictions and <Forced SmartScreen>, it is assumed that User Account Control is not disabled. It is not recommendable to completely disable UAC - in the last resort, UAC notifications can be set to a minimum.

9. Some precautions should be taken when turning on SRP and Restrictions. In some hardware/software configurations, the **autoruns located outside** "Windows" and "Program Files ..." system folders, may be blocked. Hard_Configurator can utilize Sysinternals Autorunsc (command-line tool) and NirSoft FullEventLogView to filter out autoruns, or find problematic items that should be whitelisted (see TROUBLESHOOTING paragraph for more info).

INSTALLATION / UPDATING / UNSTALLATION

1. Install H_C by running the application installer and follow the instructions displayed during the installation.
2. The program will be installed in "%WinDir%\Hard_Configurator" folder. It can be run from the Start Menu or by using a shortcut from the Desktop.
3. The application update can be performed via the **< Update H_C > black button** from the H_C main window.

Updating from the version 6.0.0.1
- Reset the option <Documents Anti-Exploit> by choosing the OFF setting and next the current setting.
- Reset the file extensions: <Designated File Types> <Restore Defaults>. Next, add or remove some extensions if needed.

Updating from the ver. 5.0.0.1 - 6.0.0.0
- Set <MORE SRP ...><Block AppInstaller> = ON (new option).
- Reset the file extensions: <Designated File Types> <Restore Defaults>. Next, add or remove the extensions if needed.
- If the user settings include some blocked Sponsors, then it is recommendable to check if the finger.exe and curl.exe are included among the blocked entries (<Block Sponsors> option).
- Update the rules in FirewallHardening by using the <Load> option to load the file UpdateFirewallHardening2011.fwbl. Some new LOLBins will appear on the BlockList as Inactive - they can be activated by the user if necessary.
- Inspect/add the new ConfigureDefender options (if necessary).

Updating from ver. 5.0.0.0 and previous versions.
It is recommendable (just after update) to apply the <Recommended Settings> and next adjust the restrictions.

The detailed changes are included in the file **What_is_new.txt** from the Hard_Configurator application folder.

QUICK CONFIGURATION (after the fresh installation).

1. Run Hard_Configurator and follow the displayed instructions.
2. Allow Hard_Configurator to make the System Restore Point, whitelist the autoruns, and apply Recommended Settings. The restore point can be skipped when the rollback software was installed in the system.
3. If Microsoft Defender is primary real-time protection, then <ConfigureDefender> option (left violet button) can be used to activate advanced Defender settings. It is recommendable to apply <HIGH> Protection Level.
4. Use <FirewallHardening> option (right violet button) to harden Windows Firewall. It is recommendable to apply "Recommended H_C" rules and turn ON "Start logging events".
5. Please note that the actions from points 2-4 require a **Windows restart**.
6. If you want to use Command Prompt or PowerShell with Administrative rights, then the option <Hide 'Run As Administrator'> should be set to OFF (not recommended on the computers of children).
7. Please update your archiver application and email client. In the Recommended Settings the below applications are supported:
   Archivers: Windows built-in Zip archiver, 7-Zip, ALZip, Bandizip, B1 Free Archiver, Explzh, ExpressZip, IZArc, PeaZip, PKZip, PowerArchiver, WinRar, WinZip.
   EmailClients: Mail for Windows 10 (Windows app), Outlook, Claws-mail, eM Client, Foxmail, Hiri, Mailspring, PostBox, Spike, Thunderbird, and any online email client.
8. Please read the help files to get info about Hard_Configurator options.

FULL UNINSTALLATION.

1. Run Hard_Configurator (close ConfigureDefender, SwitchDefaultDeny DocumentAntiExploit, and other instances of Hard_Configurator).
2. Press <Tools> button and next <Uninstall Hard_Configurator> button.
3. Follow the displayed instructions.

REMARKS
After Hard_Configurator uninstallation:
1. The registry values tweaked by Hard_Configurator are set to Windows defaults!!!
2. The System Restore is turned ON. It is good to keep this setting ON, when installing security programs. If not required, it can be manually turned OFF, by using the Control Panel or running the Windows tool:
    SystemPropertiesProtection.exe .
3. The DocumentsAntiExploit tool is copied to the Public Desktop, and it is available to manage the MS Office and Adobe Acrobat Reader XI/DC settings on the particular account. Do not delete it, until you are sure that its protection is turned OFF, on all user accounts.

SETUP_OVERKILL

It is worth remembering that:
1. The Hard_Configurator Recommended Settings are already very strong (if coupled with antivirus).
2. The user should rather learn how to live with them, than add more protection.
3. Adding more advanced features is usually not necessary and often ends with overkill, incompatibilities, and disappointment.
4. When using Recommended Settings, the user should not think about more advanced Hard_Configurator features, but rather about improving the protection of the web browser and router.
5. If the system/software is not properly updated or the computer is used in the vulnerable environment, then using Standard User Account and adding some advanced Hard_Configurator restrictions would be justified.

SwitchDefaultDeny Tool
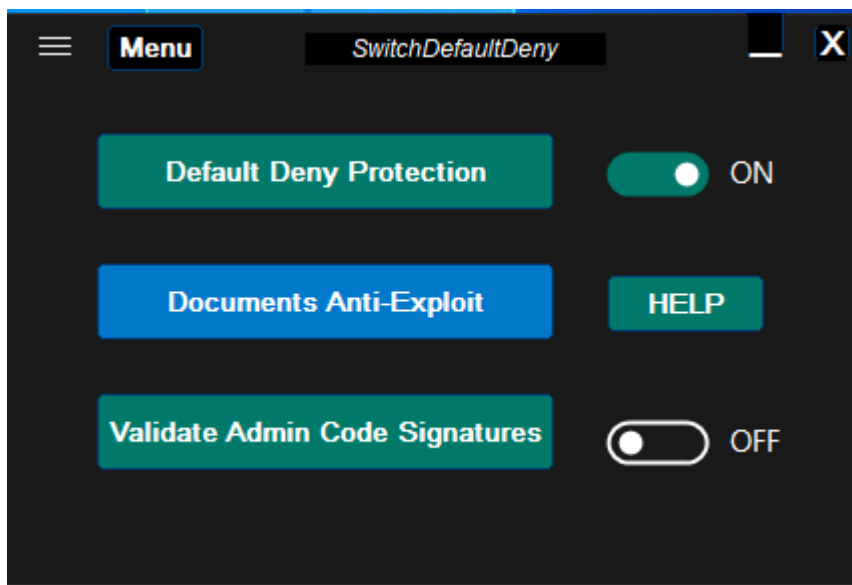
SwitchDefaultDeny is a companion tool to Hard_Configurator. It works only when Software Restrictions Policies are set to default-deny in Hard_Configurator (<Default Security Level> = Disallowed and <Enforcement> = Skip DLLs). It can help to solve the problems with application installations / updates, without running Hard_Configurator.

On Windows 7 (Vista) the Recommended Settings can block most new installations/updates.
When using the Recommended Settings on Windows 8+, similar problems can happen if the software installation / update is not performed via the standalone installer, but from CD / DVD source, CD / DVD image, etc.

In such a case, the user can use SwitchDefaultDeny to:
1. Switch OFF the Default Deny Protection.
2. Install the application normally.
3. Switch ON the Default Deny Protection (SRP rules are automatically restored and activated - no need to Log OFF).

★ When switching OFF the Default Deny Protection, this utility remembers & removes any SRP restriction and adds the autorun entry to start with Windows. **However, switching OFF the Default Deny Protection does not remove the non-SRP restrictions.**

★ When switching ON the Default Deny Protection, it restores the SRP restrictions and removes the autorun entry from the Registry.

On Windows 7 (Vista) the SmartScreen Application Reputation is not supported. So, the user must be cautious when installing new applications after switching OFF the Default Deny Protection.

Similar problems can happen with activated option <Validate Admin Code Signatures> which prevents the elevation of unsigned applications. The user can switch it OFF if necessary, install the application, and switch it ON.

The option 'Documents Anti-Exploit' runs the DocumentsAntiExploit tool (external application). It can be used to harden MS Office and Adobe Acrobat Reader XI/DC applications. On the contrary to the Hard_Configurator <Documents Anti-Exploit> system-wide feature, the DocumentsAntiExploit tool can also configure the settings limited to the current account. So, the user can apply different restrictions on different accounts.
DocumentsAntiExploit tool is copied to the Public Desktop when uninstalling Hard_Configurator, so the user can still use it to harden MS Office and Adobe Acrobat Reader XI/DC applications on the chosen user account. The info about DocumentsAntiExploit tool is included in the PDF document "**DocumentsAntiExploit tool - Manual**"

There are also some additional options available from the menu:
- Help - shows help.
- About - shows info about SwitchDefaultDeny tool.
- Do not start with Windows - removes the autorun entry, so SwitchDefaultDeny does not start automatically with Windows.
- Exit - exits the utility.

FONT RESCALING ISSUE
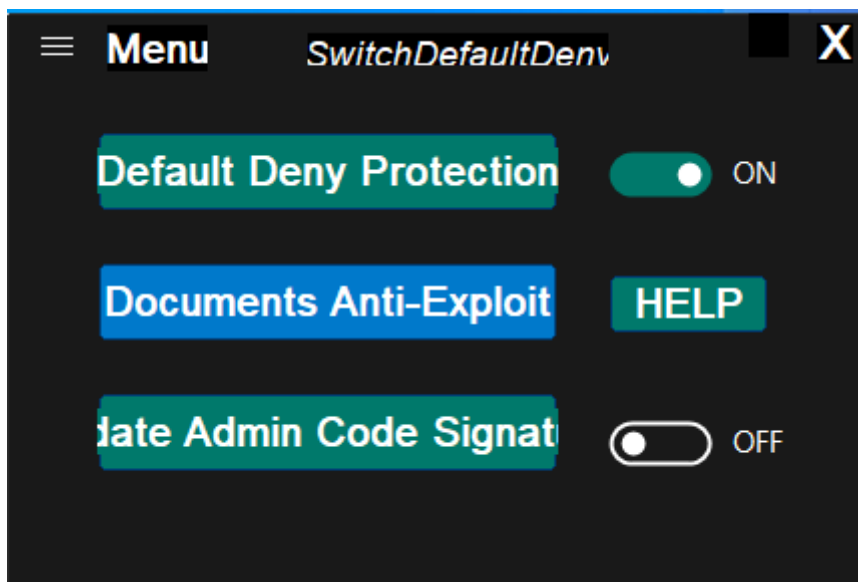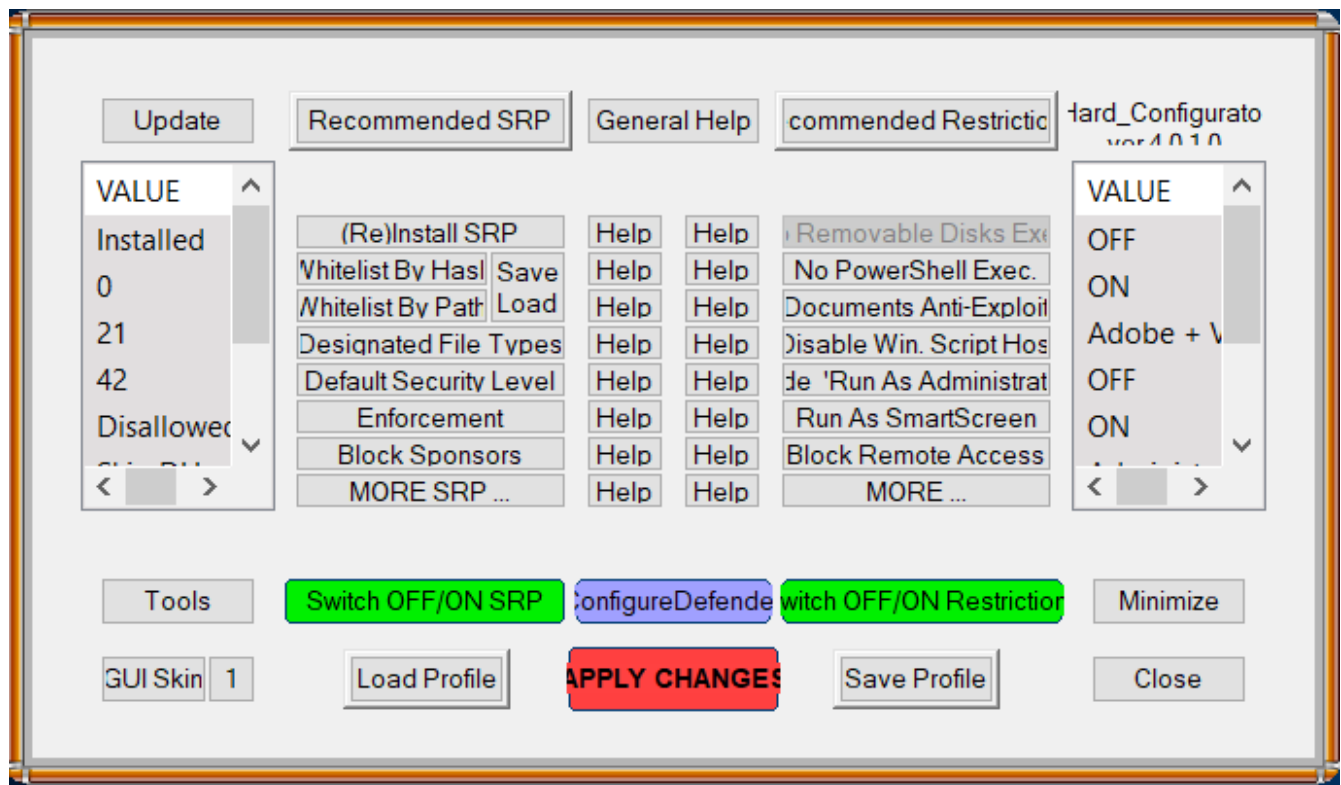
Some display drivers and DPI settings are not fully compatible with AutoIt when the fonts are rescaled by the user. The fonts are usually rescaled but not the buttons and controls. This issue should disappear after using the default font scale (100%) or default DPI settings.

In many cases, the rescaling issue can be solved by adjusting DPI settings for the concrete executable.

As an example we can take the Hard_Configurator executable:
Right-click on H_C shortcut > Properties > Compatibility > Change high DPI settings. Tick "Override high DPI scaling behavior".

Choosing the "Application" (or another) option can often solve the rescaling issues (**important: Log off the account** is required to apply the new DPI settings).
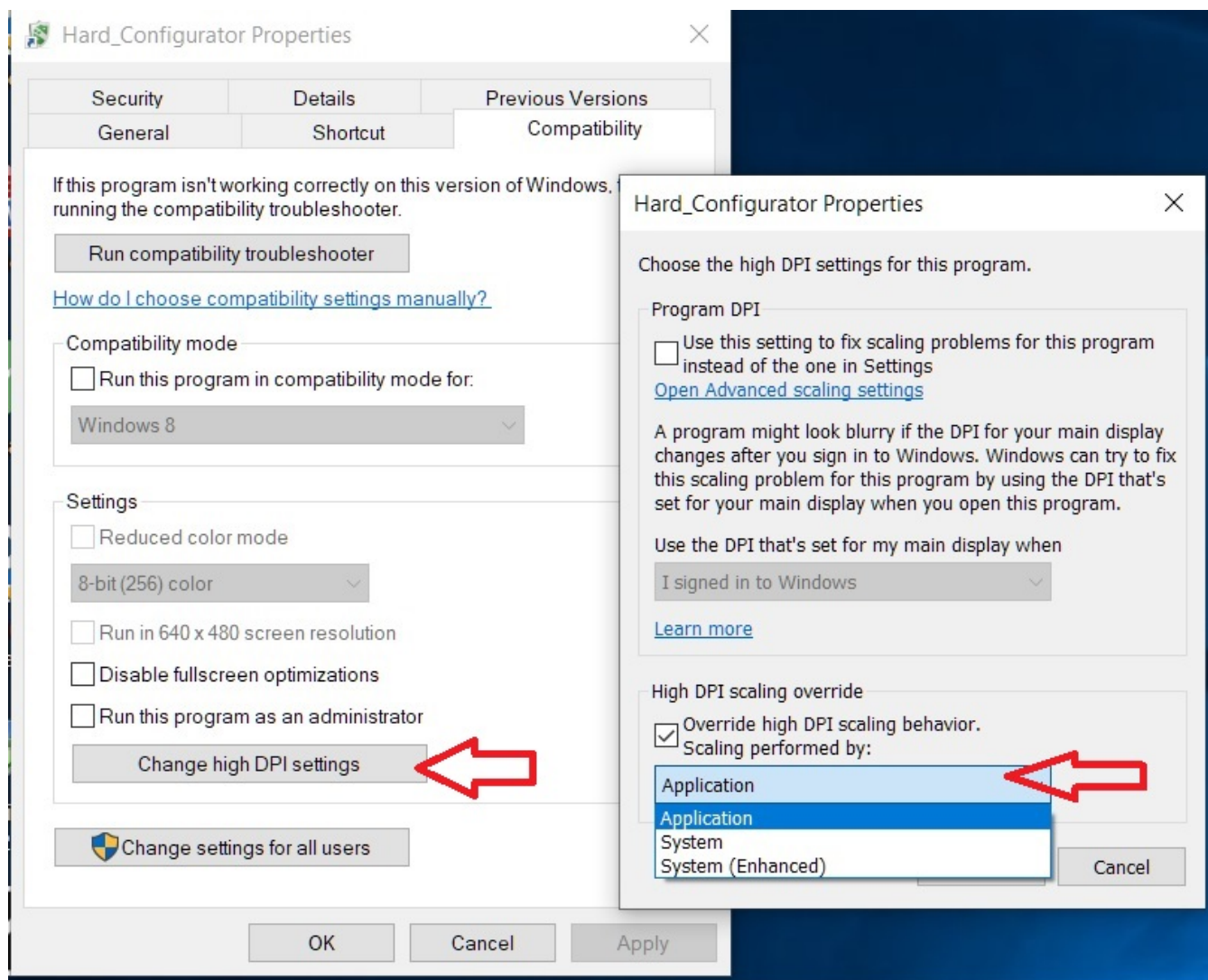
RECOMMENDED SETTINGS

Recommended Settings on Windows 8+

How do they work?

1. Users can run / auto-update the already installed applications in System-Space, ProgramData, and user AppData folders. Other UserSpace locations are blocked, except when whitelisted in SRP by path or by hash.
2. Any new file directly run by the user (executable, script, shortcut, file with unsafe extension) is blocked in UserSpace.
3. There are a few exceptions to point 2:
   a) The new files dropped into ProgramData and user AppData, could be in theory executed by the user, but these folders are normally hidden in the File Explorer.
   b) The users can run shortcuts from Desktop, Start Menu (except Startup folder), Power Menu, Task Bar, and Quick Launch.
   c) The standalone application installers (EXE or MSI files) can be run on-demand. Hard_Configurator adds "Install by SmartScreen" entry to the right-click Explorer context menu. It allows installing applications with a forced SmartScreen check.
4. Already running processes can run EXE (TMP) and MSI files in Program-Data and user AppData folders (other locations in UserSpace are blocked.). As an exception, the user cannot execute files from the archiver applications and email clients (supported by H_C), that can use AppData subfolders for temporary (unpacked) files.
5. The applications/processes running with standard rights cannot run files with unsafe extensions in UserSpace, except the events when the command lines with Sponsors are used. If Hard_Configurator is run with the -p switch, then this protection can be also applied to applications/processes running with Administrator privileges.

Can such a setup be usable?

Yes, it can.

- The already installed applications can auto-update without turning off the protection (point 1).

- Users can install most applications without turning off the protection (point 3c) and usually do not need to whitelist the new-installed applications (points 1, 3a, 3b).
- The common non-executable files like media, photos, documents, etc. can be opened without problems. For example, when clicking on the media file, Windows triggers the already installed application in "Program Files" or in the user AppData folder (point 1), so the file is opened by that application.

Is it safe?
Yes, it is much safer than antivirus protection alone. For example:
- If the user tries to run something new (executable, script, shortcut, file with unsafe extension) by a mouse-click or pressing the Enter key, then it will be blocked (point 1).
- When the user wants to install an application, the "Install by SmartScreen" entry in the Explorer context menu has to be applied. But then, the file is checked by SmartScreen and blocked if not recognized as safe (point 3c).
- If the user opens a weaponized document in MS Office, then VBA macros and anything that needs VBA Interpreter will be blocked. If he/she clicks on the embedded malicious OLE object (non-PE file), then it will be blocked, too (point 5). MS Office can be even more hardened on the concrete account by using the external DocumentsAntiExploit tool (via SwitchDefaultDeny) to block other active components. Many MS Office exploits can be prevented by configuring other Hard_Configurator features (FirewallHardening, Block Sponsors, ConfigureDefender, etc.).
- If the user tries to run an attachment directly from the email client or archiver application, then it will be blocked (points 4 and 5).
- If the user gets an exploit that tries to download/execute a payload (from disk or memory), then it will be prevented in most cases by SRP, Windows Firewall policies or PowerShell restrictions. This protection can be independently configured by several Hard_Configurator features (FirewallHardening, Block Sponsors, ConfigureDefender, etc.).
- If the user gets an exploit which abuses Windows remote feature, then this will fail, because remote features are disabled by Hard_Configurator.
- When the user runs something from the web browser, then it will be blocked (in the Downloads folder) or checked by SmartScreen AppRep.

Can such protection be bypassed?
Yes, but this would require exploiting the Windows system or one of the installed applications. Even then, in most cases the attack would be neutralized - this is true also for fileless attacks and for many exploits with privilege escalation (due to blocking PowerShell scripts, disabling remote features, FirewallHardening, and ConfigureDefender settings).

Can the user feel a difference as compared to the setup without Hard_Configurator?
In daily work it can be hardly visible, except when installing applications from CD/DVD drives, CD/DVD images, and similar non-standalone installation packages. In such cases using "Install By SmartScreen" will fail.
Such installations have to be performed after switching off the Hard_Configurator default-deny protection (via SwitchDefaultDeny tool).
The possibility of running the downloaded executables from the web browser will be blocked by SRP and (or) by SmartScreen AppRep. The files downloaded by the user and blocked by SRP, can be run from the Downloads folder by using "Install By SmartScreen" entry in the right-click Explorer context menu.

Recommended Settings on Windows 11.

When Smart App Control (SAC) is in Evaluate or OFF mode, the Recommended Settings are the same as on Windows 10.

When SAC is ON, users can keep the Recommended Settings like in Evaluate and OFF more, or apply the predefined setting profile:
Windows_11_SAC_ON_Recommended_Settings.
In this profile, the EXE, and MSI files are allowed in SRP because they are protected by SAC. This setup allows users to install applications without using "Install By SmartScreen" via the Explorer context menu.

Basic_Recommended_Settings on Windows 8+.

This is a predefined setting profile that allows EXE (TMP) and MSI files globally. The scripts, shortcuts and other files with unsafe extensions are still blocked by default in UserSpace. On the contrary to the Recommended Settings, shortcuts are allowed in the Startup folder.

This profile can harden Windows 8+ while maintaining maximum functionality and compatibility. **It could be probably called Recommended Settings for cautious users.**

The "Run By SmartScreen" entry in the Explorer context menu can be used to check the standalone application installers (EXE and MSI) by SmartScreen Application Reputation service. This entry should be also used for unsafe executables listed below:

1. Files downloaded from the Internet, especially email attachments and executables from the archives (7-zip, Zip, Arj, Rar, etc.).
2. Executables shared with other people via USB drives, Memory cards, etc.

The users can install/execute/update applications via EXE and MSI files. The only exceptions are EXE and MSI files executed directly from an archive or email client. In such cases, the archive has to be first unpacked and email attachment has to be downloaded to hard disk. Next, it is recommendable to use "Run By SmartScreen" to execute those files via SmartScreen.

It is also recommendable to use this profile with ConfigureDefender HIGH Protection Level (if WD is the main antivirus) and "Recommended H_C" firewall outbound block rules (see <FirewallHardening> option). The profile can be used with any antivirus which can apply strong proactive detection.

Is it safe?

It is as safe as the H_C Recommended Settings if the user is cautious enough to use the "Run By SmartScreen" entry in the Explorer context menu. If not then EXE and MSI files will be covered only by the Antivirus.

PLEASE NOTE: *This profile will be not enough for children. They will be better protected by the H_C Recommended Settings and SmartScreen set to Block, with occasional help from more experienced users.*

Recommended Settings on Windows 7 (Vista).

The SmartScreen integration with Explorer is not supported on Windows 7 (Vista). If the installed antivirus does not have special protection for installers downloaded from the Internet, then it is not recommendable to whitelist the EXE files in AppData and Program data folders (<More SRP ...> <Update Mode> = OFF).

The user has to switch off the Hard_Configurator protection temporarily (recommended) or use "Run as administrator" entry from the Explorer context menu (not recommended on SUA), to install/update applications. Both ways should be used with caution (especially the second), after checking the executable to be sure that it is safe. Furthermore, the applications installed in AppData or ProgramData subfolders will require whitelisting.

The Recommended Settings on Windows 7 are more restrictive and not so convenient as compared to Windows 8+. They will require more attention and skills when installing/updating applications.

SETTING PROFILES FOR AVAST ANTIVIRUS.

There are some predefined Hard_Configurator settings available with Avast:
1. **Recommended Settings** (on Windows 8+).
2. **Windows_*_Avast_Hardened_Mode_Aggressive**
   where the `*` denominates the Windows version.
3. **Windows_7_Avast_CyberCapture**.

Recommended Settings

On Windows 8+, the H_C Recommended Settings can support Avast, because they make use of the <Update Mode> feature. The user has to apply the "Install By SmartScreen" entry in the right-click Explorer context menu to install applications via standalone EXE/MSI installers. The MOTW is added to the file and this triggers both the SmartScreen check and Avast CyberCapture check (detonation in the cloud sandbox).

Windows_*_Avast_Hardened_Mode_Aggressive

These profiles are prepared for any Windows version and are intended to work with Avast Hardened Mode Aggressive.

If the Avast Hardened Mode Aggressive is not available via modern GUI, then the old GIU can be accessed by using:

Menu > Settings > Troubleshooting > Open old settings.

This profile works as follows:

1. The EXE files are allowed to run, but they are protected by Avast Hardened Mode Aggressive (file reputation cloud service).
2. Other new files directly run by users (executables, scripts, shortcuts, files with unsafe extensions) are blocked in UserSpace. This also works when the file is run from the archive without unpacking the archive.
3. As an exception to point 2, the shortcuts (LNK files) can be run by users from Desktop, Start Menu, Power Menu, Task Bar, and Quick Launch.
4. As an exception to point 2, the standalone MSI application installers can be run by users on-demand - Hard_Configurator adds the right-click Explorer context menu entry "Install by SmartScreen" (Windows 8+) or "Install application" (Windows Vista and Windows 7).
   On Windows 8+, the forced SmartScreen check is triggered. But on Windows 7 (Vista) only the standard Avast protection will be applied.
5. Already running processes can run MSI files in ProgramData or user AppData folders.
6. The applications/processes running with standard rights cannot run unsafe files (executables, scripts, files with unsafe extensions) in UserSpace with exceptions for EXE and MSI files, mentioned in points 1 and 5, and except some events when the command line can be accessed (some command lines with Sponsors).

Can it be usable?

Yes, it can.

- Most applications can be installed/run/updated, without turning off the protection (points 1, 3, 4, 5). This is true also for most installations from CD/DVD drives, CD/DVD images, and similar non-standalone installation packages (started by EXE files).

- The common non-executable files like media, photos, documents, etc. can be opened without problems. For example, when clicking on the media file, Windows triggers the already installed application (EXE file), and the file is opened by that application (point 1).
- The only limitation for running applications by a mouse-click or pressing the Enter key, will be Avast file reputation service. But, Avast allows easily to whitelist the blocked files.

Is it safe?

Yes, it is much safer than antivirus protection alone. For example:

- If the user tries to run something new (executable, script, shortcut, file with unsafe extension) by a mouse-click or by pressing the Enter key, then it will be blocked (point 2) or checked by Avast file reputation service (point 1).
- When the user wants to install the application, then it is possible for EXE installers by a mouse-click or pressing the Enter key (point 1) - the application installer will be checked by Avast file reputation service.
- The MSI installers can be run via the "Install by SmartScreen" or "Install application" entry in the Explorer context menu. **On Windows 7 (Vista) the MSI installers are not protected by SmartScreen (point 4), so they will be checked only by standard Avast features .**
- If the user opens a weaponized document in MS Office, then macros and anything that needs VBA Interpreter will be blocked. If he/she clicks on the embedded malicious OLE object, then it will be blocked, too (point 6). Anyway, Avast has not got special protection against weaponized documents as compared to Microsoft Defender with ConfigureDefender High Protection Level. The user should harden MS Office applications by using the external DocumentsAntiExploit tool (via SwitchDefaultDeny) to block other active components. Many MS Office exploits can be also prevented by configuring other Hard_Configurator features (FirewallHardening, Block Sponsors, etc.).
- If the user gets the exploit that tries to download/execute the payload (from disk or memory), then it will be prevented in most cases by SRP, Avast file reputation service, Windows Firewall policies or PowerShell restrictions. This protection can be independently configured by several Hard_Configurator features (FirewallHardening, Block Sponsors, etc.).

- If the user gets the exploit which tries to abuse Windows remote features, then this will fail, because remote features are disabled by Hard_Configurator.
- When the user runs something from the web browser, then it will be blocked, or checked by SmartScreen, or checked by the Avast file reputation service. There is only one exception. On Windows 7 (Vista) some web browsers can run files from AppData folders - **in such case the MSI files will be checked only by the standard Avast features**.

Can such protection be bypassed?
It is possible by exploiting the Windows system or one of the installed applications. But, in most cases the attack will be neutralized - this is true also for fileless attacks and many exploits with privilege escalation (due to blocking PowerShell scripts, disabling remote features, FirewallHardening, etc.).
The second possibility can happen on Windows 7 when the user installs an application with a malicious MSI installer (not detected by the standard Avast features).

Can the user feel the difference as compared to the setup without Hard_Configurator?
It will be hardly visible, except when Avast will block the execution of an EXE file with a poor reputation.

Windows_7_Avast_CyberCapture

This setting profile works similarly to the Recommended Settings on Windows 8+ with disabled SmartScreen for Explorer. So, the EXE and MSI installers are not checked by SmartScreen. Anyway, the EXE installers are still checked by Avast CyberCapture, which is strong enough. But, the MSI installers are checked only by the standard Avast features. Such setup will produce fewer false positives than Recommended Settings on Windows 8+.

SOFTWARE INCOMPATIBILITIES

Windows built-in SRP cannot work with **AppLocker** (introduced via GPO or MDM WMI Bridge). In such a case, Hard_Configurator shows an alert, and the options from the left H_C panel (related to SRP) will be deactivated.

From the year 2022, AppLocker (GPO) policies can work on Windows 10/11 Home and Pro. AppLocker is activated by default on Windows 11 ver. 22H2 or later (also on Windows Home), so SRP is disabled in the default configuration.

Hard_Configurator ver. 6.1.1.1 can enable SRP on Windows 11, and SRP can also work with enabled Smart App Control (SAC).
Users can apply the H_C Recommended settings or load another predefined profile: **Windows_11_SAC_ON_Recommended_Settings**. In this profile, the EXE and MSI files are whitelisted (already protected by SAC).

Windows built-in SRP is incompatible with **Child Account** activated on Windows 10+ via Microsoft Family Safety. Child Account adds some AppLocker rules (via MDM), so SRP cannot work. Unfortunately, after removing Child Account, the AppLocker Policy files are not removed (unpleasant bug)! These policy files have to be removed manually to recover the SRP functionality.

Hard_Configurator settings are not compatible with **SRP introduced via Group Policies Object** (GPO) available in Windows Pro, Education, and Enterprise editions. GPO refresh feature will overwrite the Hard_Configurator settings. So, before installing Hard_Configurator, the SRP has to be removed from GPO.

Hard_Configurator will also conflict with **any software which uses SRP**, but such applications are rare (CryptoPrevent, SBGuard, AskAdmin, Ultra Virus Killer). Before installing Hard_Configurator it will be necessary to uninstall the conflicting application. Next, use in Hard_Configurator the option: <Tools> <Restore Windows Defaults>, and next apply the Hard_Configurator settings.

SOFTWARE RESTRICTION POLICIES (SRP)

From the technet.microsoft.com :

"Software Restriction Policies (SRP) is Group Policy-based feature that identifies software programs running on computers in a domain, and controls the ability of those programs to run. Software restriction policies are part of the Microsoft security and management strategy to assist enterprises in increasing the reliability, integrity, and manageability of their computers.
You can also use software restriction policies to create a highly restricted configuration for computers, in which you allow only specifically identified applications to run. Software restriction policies are integrated with Microsoft Active Directory and Group Policy. You can also create software restriction policies on stand-alone computers. Software restriction policies are trust policies, which are regulations set by an administrator to restrict scripts and other code that is not fully trusted from running."

https://technet.microsoft.com/en-us/library/hh831534(v=ws.11).aspx

SRP is available via Group Policy in Windows Pro, Windows Enterprise, Windows Server, and Windows Education. Well-configured SRP is known in enterprise case studies as a proven protection against virus infections.
A few years ago, the development of SRP was stopped because, in enterprises, Microsoft recommends Applocker and Defender Application Control. Yet, SRP is still fully functional on all Windows versions.
Hard_Configurator can apply several SRP settings in Windows Home, too. Some settings were skipped because they are not needed for home users (for example Zone and Certificate rules).

<(Re)Install SRP> button makes changes in the Registry to install Windows SRP. The SRP parameters can be changed when using the buttons:
<Whitelist By Hash>, <Whitelist By Path>, <Designated File Types>,
<Default Security Level>, <Enforcement>.

Two SRP options: "Ignore certificate rules" and "All users except local administrators" are hardcoded, and set to ON. Some Disallowed (Blacklist) rules are applied by <Protect Windows Folder> , <Update Mode>, <Protect Shortcuts> (in <More SRP ...>), and <Block Sponsors> options. There are no other options in Hard_Configurator to customize Disallowed rules.
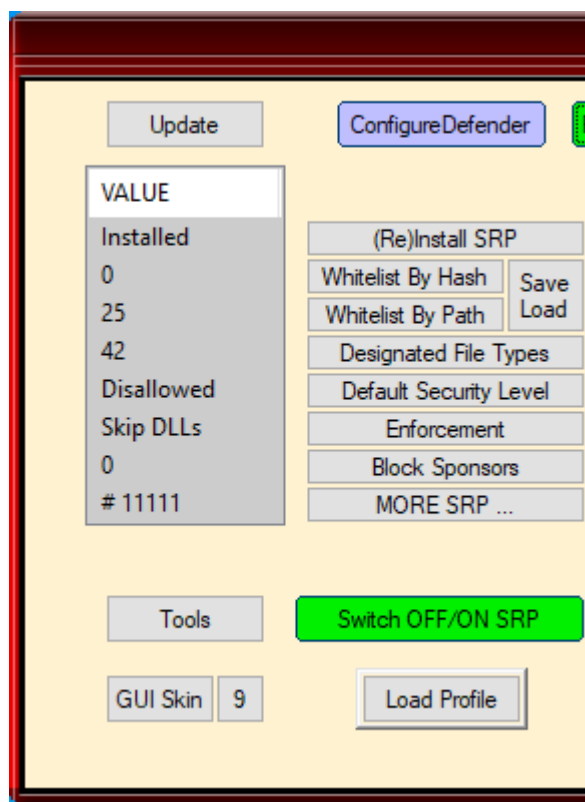
In this program, Windows built-in SRP can apply some default-deny and whitelisting/blacklisting features. Executables can be run without SRP restrictions in **SystemSpace**, that contains UAC protected system folders: 'Windows' and 'Program Files' (also 'Program Files (x86)' in 64-bit versions). Outside of those folders (= **UserSpace**), executable files will be blocked by default (see <Update Mode> for some exceptions), when running in a standard way: by mouse clicking, pressing the ENTER key or using 'Open'/'Open With ...' from the Explorer context menu. The list of protected file extensions (Designated File Types) can be accessed by pressing <Designated File Types> button.

There is a group of privileged file types, that can be blocked by SRP, even if they are not on the 'Designated File Types' list (see **'How SRP can control file execution/opening'**). This type of execution control relates to API functions: CreateProcess, and LoadLibrary, which can call into SRP. Also, the 'Privileged Objects' like: 'Windows CMD', 'Windows Script Host', and 'Windows Installer' have such ability.
Executables from UserSpace can be run in a standard way, only if they are whitelisted by hash or by path.

REMARKS
SRP restrictions can be bypassed, when using "Run as administrator" entry in the right-click Explorer context menu. But, running the new files with Administrative Rights can be dangerous for many users, so Hard_Configurator can replace "Run as administrator" entry in the Explorer context menu, with the safer "Install By SmartScreen" (see <Forced SmartScreen> section).

SRP in Hard_Configurator can be completely deactivated by using the button sequence \<Switch OFF/ON SRP\> \<APPLY CHANGES\> .

How SRP can control file execution/opening.

This section is for the users who want to understand SRP on a deeper level. It is not necessary for using Hard_Configurator.

SRP can know what should be monitored from:
• Designated File Types list.
• Enforcement settings.
See also the TABLE (1) and (2).

File monitoring by calling into SRP.

1. ShellExecute API function.
   It calls into SRP while opening files with extensions included in the SRP 'Designated File Types' list (the list of protected file extensions). SRP will apply when Windows Explorer, Edge or Internet Explorer is used to open the files from the local disk. If the file extension is on this list, then the file access will be controlled by SRP, while double-clicking, pressing ENTER key or choosing "Open"/"Open With ..." from Explorer context menu. If the file is blocked by SRP, then the program (Sponsor), that can manage the extension (for example regedit.exe for the REG file) is not invoked at all. Yet, the file can still be opened from within this program (in Regedit the REG file can be imported) or indirectly by the command line, when using the Sponsor (for example: '**regedit.exe path_to_file.reg**').

2. Privileged Objects.
   There are some objects, which can call into SRP (extended protection): **Windows Command Shell (Windows CMD), Windows Script Host,** and **Windows Installer**. They can host the file types: **BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH,** and **MSI**. This is much safer than blocking files by extension (see point 1.). The file cannot be run, both: from Explorer (Edge or IE) and by command line with the Sponsor (for example: 'cmd /c path_to_malicious.bat').

3. CreateProcess API function (extended protection).
   It calls into SRP while executing **COM/EXE/SCR** files, and SRP is applied directly to those **COM/EXE/SCR** files. The **COM** and **SCR** files can be protected, by both ShellExecute and CreateProcess API functions, if those extensions are added to the 'Designated File Types' list.

4. LoadLibrary API function (extended protection).
   It calls into SRP, while loading libraries **DLL/OCX**, and SRP is applied directly to those **DLL/OCX** files.

TABLE (1) - Enforcement settings and file monitoring.

No Enforcement	Skip DLLs	All Files
Windows CMD Windows Script Host Windows Installer	Designated File Types Windows CMD Windows Script Host Windows Installer ShellExecute() CreateProcess()	Designated File Types Windows CMD Windows Script Host Windows Installer ShellExecute() CreateProcess() LoadLibrary()

When **'No Enforcement'** setting is applied, only **Windows CMD, Windows Script Host,** and **Windows Installer** can call into SRP, so only **BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH,** and **MSI** files can be monitored. The files with other extensions are not monitored, even when they are on 'Designated File Types' list.

We can translate TABLE (1) to see explicitly, which file types are monitored by SRP according to Enforcement settings.

TABLE (2) - Monitored file types

	No Enforcement	Skip DLLs	All Files
Blocking by Extension controlled by ShellExecute	*****	Designated File Types	Designated File Types
Windows CMD **Windows Script Host** **Windows Installer** **CreateProcess** **LoadLibrary**	BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH, MSI ***** *****	BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH, MSI, COM, EXE, SCR *****	BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH, MSI, COM, EXE, SCR, DLL, OCX,

Hard_Configurator default 'Designated File Types' in Windows 7+:
ACCDA, ACCDU, ADE, ADP, BAS, BAT, CHM, CMD, COM, CPL, CRT, CSV, DLL, DQY, ECF, ELF, EXE, HLP, HTA, INF, INS, IQY, ISP, JAR, JS, JSE, LNK, MDA, MDB, MDE, MSC, MSP, MST, OCX, OQY, PA, PCD, PIF, PPA, PPAM, REG, RQY, RTF, SCR, SCT, SETTINGCONTENT-MS, SHS, SLK, URL, VB, VBE, VBS, WLL, WS, WSC, WSF, WSH, WWL, XLA, XLAM, XLL, XLM.

EXAMPLES

From the above we can see, that with **'No Enforcement'** setting the below Disallowed file path rules :

c:\Program Files*.reg
c:\Program Files*.scr
c:\Program Files*.ocx
c:\Program Files*.bat
c:\Program Files*.vbs
are only valid for **BAT** and **VBS** files, and they will be applied because Windows CMD and Windows Script Host can call into SRP. 'Designated File Types' list is skipped. The rules for REG, **SCR, OCX** files will be ignored (not monitored by SRP) with 'No Enforcement' setting.

With Hard_Configurator default settings: **<Enforcement> = 'Skip DLLs'**, all the above rules, are valid (and monitored by SRP).

SRP can know which monitored files should be blocked from:
1. 'Default Security Level' settings.
2. Unrestricted/Disallowed rules (by path, by hash, wildcards supported). (<Whitelist By Hash>, <Whitelist By Path>, <Protect Windows Folder>, <Protect Shortcuts> buttons in Hard_Configurator).

Recommended Settings in Hard_Configurator, assume whitelisting by path the **SystemSpace** = 'Windows' + 'Program Files' (and 'Program Files (x86) in 64-bit systems).
The below table shows, what files are blocked by SRP in **UserSpace** (= everything on local drives outside of **SystemSpace**).

TABLE (3).
Files blocked by default in UserSpace. Enforcement settings are in the first row.
Default Security Level settings are in the first column.

	No Enforcement	Skip DLLs	All Files
Unrestricted (Windows Vista+)	all files allowed	all files allowed	all files allowed
Basic User (Windows 7+)	MSI	MSI, COM, EXE, SCR Designated File Types	MSI, COM, EXE, SCR, DLL, OCX, Designated File Types
Basic User (Windows Vista)	MSI	MSI Designated File Types	MSI, DLL, OCX, Designated File Types
Disallowed (Windows Vista+)	BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH, MSI,	BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH, MSI, COM, EXE, SCR, Designated File Types	BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH, MSI, COM, EXE, SCR, DLL, OCX, Designated File Types

We can see that some files can be monitored by SRP, but not blocked by default. For example, script files when 'Basic User' + 'No Enforcement' settings are applied. Yet, they can be blocked when using Disallowed rules (do not confuse Disallowed rules with Disallowed setting of Default Security Level).

It is worth mentioning, that any Unrestricted/Disallowed rule can override 'Default Security Level' settings.

So, all file types included in the TABLE (3) are not blocked by default in SystemSpace, because of Unrestricted folder path rules for system folders: Windows , Program Files , Program Files (x86)

Useful links:

https://technet.microsoft.com/en-us/library/cc786941(v=ws.10).aspx

https://technet.microsoft.com/en-us/library/bb457006.aspx

https://malwaretips.com/threads/windows-pro-owner-use-software-restriction-policies.61871/

http://www.wilderssecurity.com/threads/maximising-windows-7-security-with-srp-under-lua-whatever-the-win7-version.262686/

http://www.bleepingcomputer.com/tutorials/create-an-application-whitelist-policy-in-windows/

Registry changes:

HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers

WHITELISTING BY HASH

<Whitelist By Hash> button opens ADD / REMOVE / EDIT window to manage file whitelisting by hash. It can be useful in the locked setup for running programs located in UserSpace (outside of the system folders: Windows, Program Files, and Program Files (x86)). The UserSpace is not protected by UAC, so the file can be silently modified by the virus infection. Yet, this also changes the file hash, and then SRP will block file execution. Such a situation can happen when using the vulnerable system/software or due to running the malware in the system. If the Hard_Configurator Recommended Settings are applied on the well updated system with updated software, then whitelisting by hash is not necessary.

Managing file hashes is not comfortable. Use this function only if you have to. The program tries to extract some info about the file to make hash entries more readable.
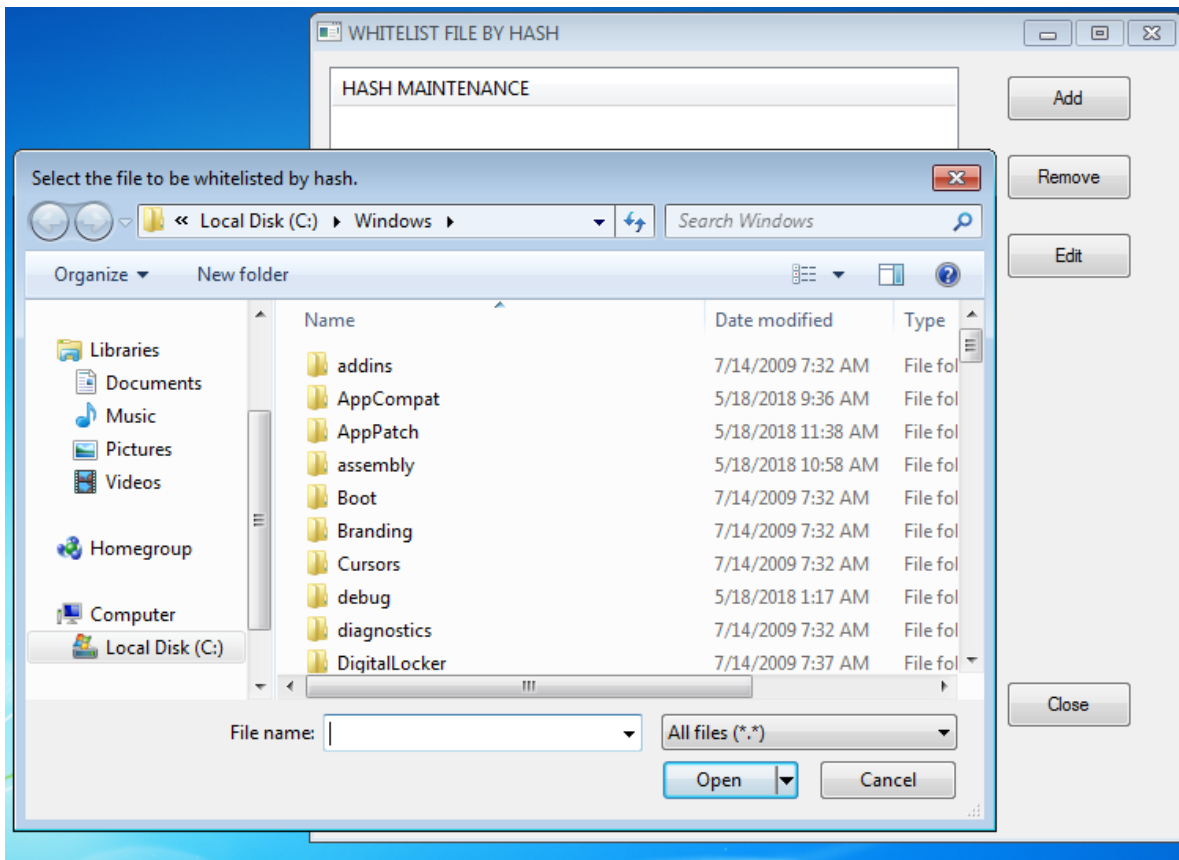
REMARKS

Sometimes programs are wrapped and have to use TEMP folder to execute (most frequently it is '%UserProfile%\AppData\Local\Temp').

Depending on SRP configuration, the file execution in the TEMP folder can be blocked by SRP. If so, then the unwrapped file can be whitelisted by hash (in the TEMP folder this is safer than whitelisting by path). Hard_Configurator has the option: <Blocked Events / Security Logs> in the 'Tools' section. It can use NirSoft FullEventLogView utility to filter/view SRP blocked events and find out which file in the TEMP folder should be whitelisted. This utility is already included in the Hard_Configurator package as an external tool.

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers\262144\Hashes

WHITELISTING BY PATH

If something is blocked, then it is recommendable to use first the option: <Tools><Blocked Events / Security Logs>. It can help to identify the file path that can be used to whitelist the file. More info about this option can be found in the section TROUBLESHOOTING.
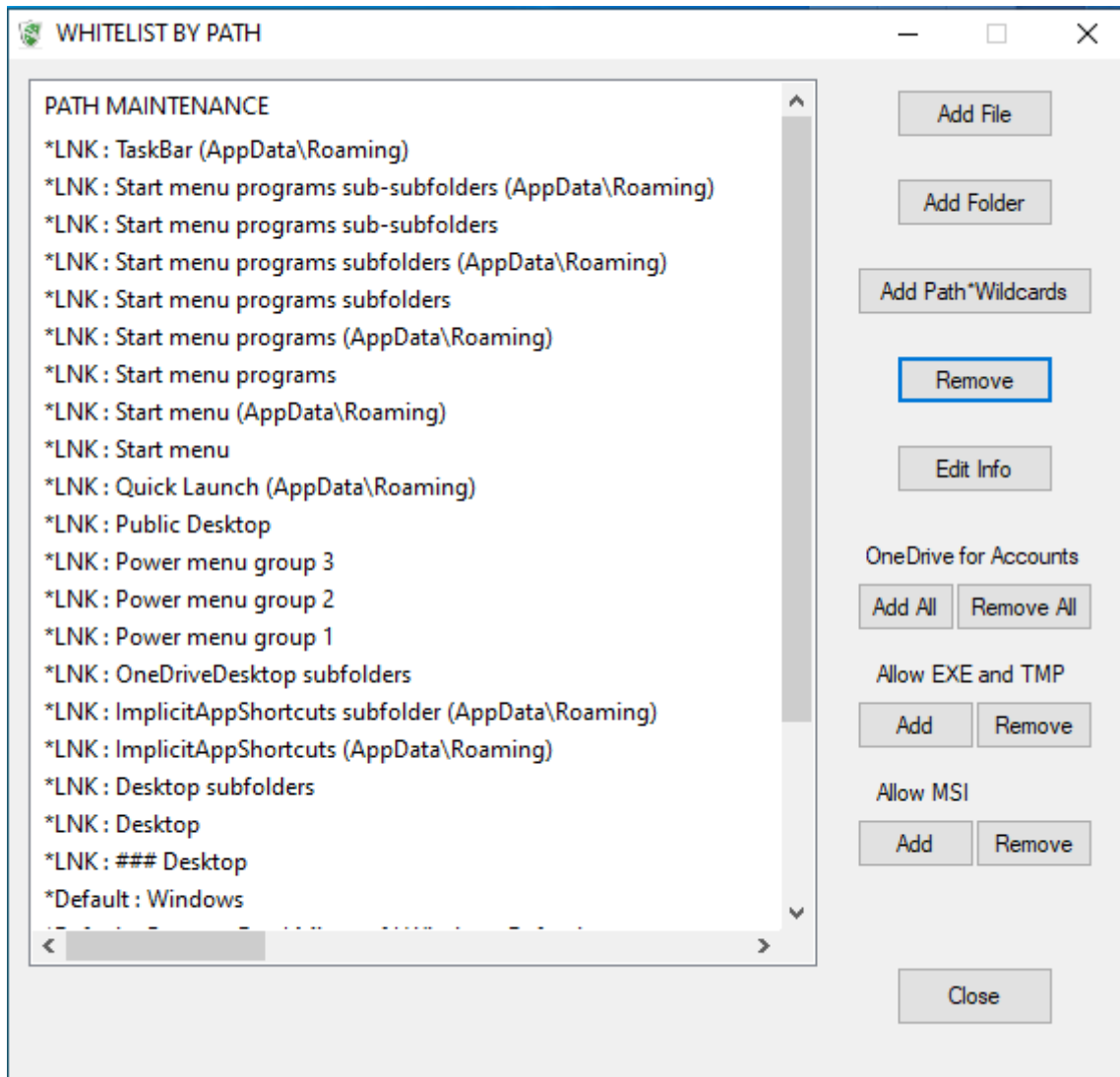
<Whitelist By Path> button opens a window to manage file/folder whitelisting by path. It is very useful when running programs located in UserSpace (outside of the system folders: 'Windows', 'Program Files ...'). Whitelisting has to be done with cautious, because SRP will not block the malware running from the whitelisted path.
Whitelisting by path a shortcut (LNK file) or a path with wildcards, is only

possible when using <Add Path*Wildcards> option. **This option does not support the paths with environment variables or quotation marks.**

The Whitelist can be saved into the file, using <Save Load> button on the right side of the Whitelist buttons.
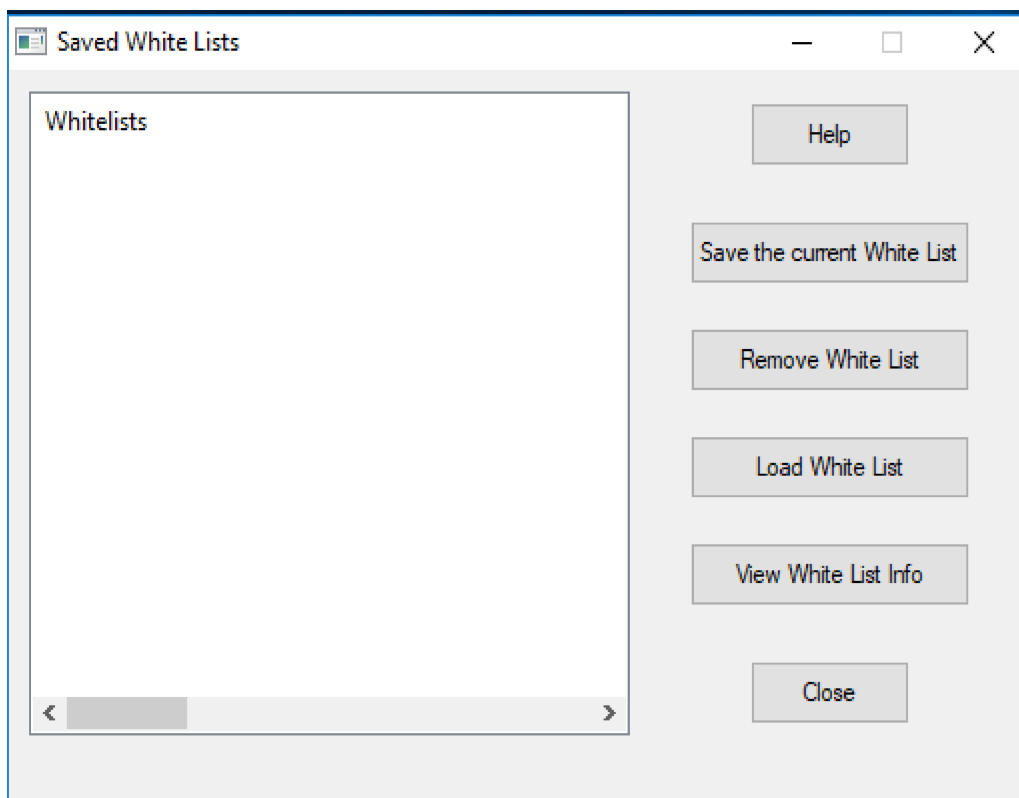
WARNING!!!

It is forbidden to adopt environment variables when using <Add Path*Wildcards> option to whitelist the paths!

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers\262144\Paths

WHITELIST PROFILES

<Save Load> button from Hard_Configurator main menu opens the window to manage the user made White List Profiles.

\<Save the current White List\> option saves the current, active White List to the White List Profile Base. The base is placed in the Windows Registry. Each White List Profile contains: White List entries, the name of the White List, and the short info. So, while saving the profile, the user first has to write the name for the current, active White List, and next is asked to put some info about the profile (for example the creation date/time, and the short White List characteristics). The names of the saved White List Profiles are visible in the left panel. If the profile with the same name is already in the base, the user is asked if it should be overwritten.

\<Remove White List\> option removes the chosen White List Profile from the Profile Base.

\<Load White List\> option loads the White List from the chosen White List Profile. The loaded White List overwrites the current, active White List. Before loading the profile, it is recommendable to view info about the profile using \<View White List Info\> option. Please, do not forget to \<APPLY CHANGES\> after loading the White List.

\<View White List Info\> option allows viewing the info about the chosen profile, which was written by the user while saving the White List. The info usually contains some useful information for example, the creation date/time, and the short White List characteristics.

The White List Profile Base can be also exported together with Setting Profiles Profiles into the one compressed backup file. This can be done in Hard_Configurator via \<Tools\> \<Manage Profiles Backup\> option .

DESIGNATED FILE TYPES

<Designated File Types> button opens ADD/REMOVE window with the list of actually protected extensions.

Default extensions in Hard_Configurator (Windows 7+):
ACCDA, ACCDU, ADE, ADP, BAS, BAT, CHM, CMD, COM, CPL, CRT, CSV, DLL, DQY, ECF, ELF, EXE, HLP, HTA, INF, INS, IQY, ISP, JAR, JS, JSE, LNK, MDA, MDB, MDE, MSC, MSP, MST, OCX, OQY, PA, PCD, PIF, PPA, PPAM, REG, RQY, RTF, SCR, SCT, SETTINGCONTENT-MS, SHS, SLK, URL, VB, VBE, VBS, WLL, WS, WSC, WSF, WSH, WWL, XLA, XLAM, XLL, XLM.
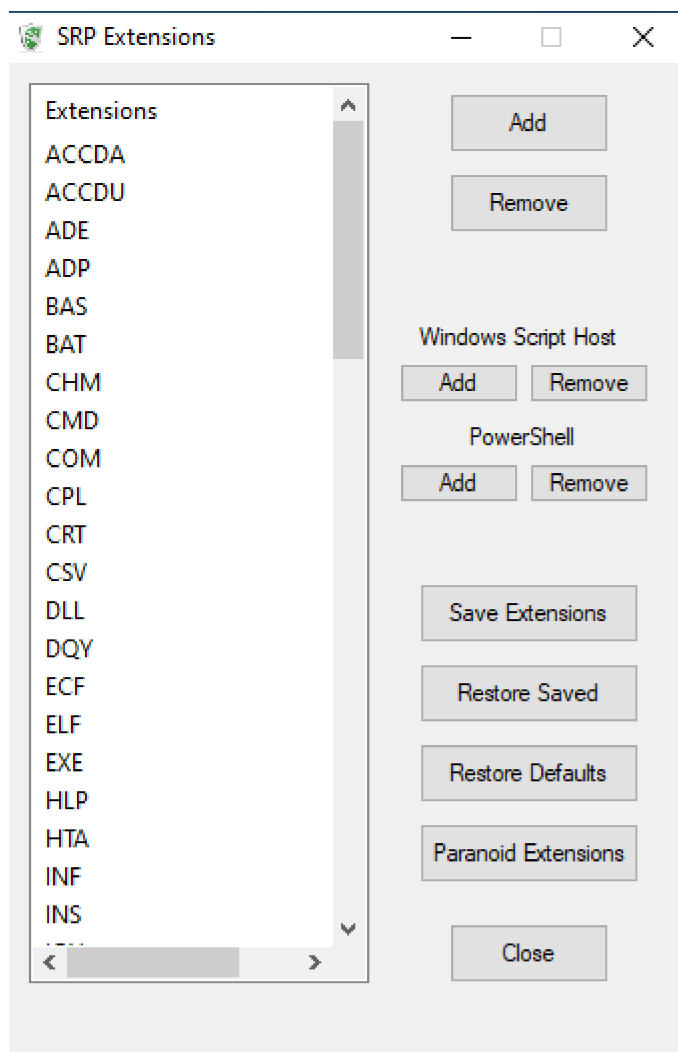
In Windows Vista, some PowerShell extensions are added by default: PS1, PS2, PSC1, PSC2, PS1XML, PS2XML, because the option <Block Power-Shell Scripts> is not supported.

The PowerShell script extensions were removed for Windows 7+, because Hard_Configurator has <Block PowerShell Scripts> option to deal with them. Also, the MSI extension was removed to work with <Forced SmartScreen> option (SRP can still protect MSI files, even if they are not on the extension list).

Paranoid Extensions include extended number of potentially dangerous file extensions (over 300 entries), which were abused in the wild to exploit Windows or MS Office.

You can customize the list of extensions via <Add> and <Remove> buttons. When using a custom list, it is good to save it (<Save Extensions>). The list can be restored by using <Restore Saved> button.

The protection by the file extension restrictions can have some limitations:
1. If the file type is opened by default via **Universal Windows Platform app**, then usually the restriction is bypassed.
2. Some file types are opened by Windows built-in handlers that does not support SRP. The example can be disk images (**IMG, ISO, VHDX**). Such files can be protected by SRP when using 3rd party applications like WinISO, Deamon Tools, etc.

In some cases, the **below extensions** can be skipped:
- **MSI** extension if <Forced SmartScreen> is set to 'ON'.
- **PS1, PS2, PSC1, PSC2, PS1XML,** and **PS2XML** extensions, if <Block PowerShell Scripts> is set to 'ON'.

REMARKS
Windows Script Host protection depends also on the below registry value:
HKLM\SOFTWARE\Microsoft\Windows Script Host\Settings
UseWINSAFER = 1 (Windows default value)

and on 64-bit system (for 32-bit programs), the same in the key:
HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows Script Host\Settings
UseWINSAFER = 1 (Windows default value)

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers!ExecutableTypes

DEFAULT SECURITY LEVELS

<Default Security Level> button changes the security level between:
'Basic User' ---> 'Unrestricted' ---> 'Disallowed'

'Disallowed' setting blocks by default all monitored files (default-deny), except those that match the winning Unrestricted/Disallowed rules.
With <Enforcement> option set to 'Skip DLLs', it can apply in UserSpace:
★ protection to all files included in 'Designated File Types' list
★ extended security for Windows native executables (COM, EXE, SCR), scripts (BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH), and MSI installers.

'Basic User' (in Windows 7+) is very similar to 'Disallowed', except when handling LNK, MSI, or script files. In Windows Vista, 'Basic User' works differently, for example, it allows running EXE files even from UserSpace.

'Unrestricted' setting corresponds to Default-Allow SRP. It allows the execution/opening of all files, except those that are monitored by SRP and match the winning Disallowed rules. So, the SRP rules related to <Block Sponsors>, <Protect Windows Folder>, <Protect Shortcuts>, <Harden Archivers>, <Harden Email Clients>, and <Block AppInstaller> can still work.

See also: **How SRP can control file execution/opening.**

If you want to run the executable file in UserSpace, with SRP set to 'Basic User' or 'Disallowed', then it can be done with "Run as administrator" entry in Explorer context menu. Hard_Configurator provides a safer option by replacing "Run as administrator" with "Install By SmartScreen" (only EXE and MSI files). If you want to use frequently, any application that is located in UserSpace, then consider to whitelist it by path (hash).

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers!DefaultLevel
Value (Dword)
0   'Disallowed'
131072   'Basic User' (131072 = 20000 hex)
262144   'Unrestricted' (262144 = 40000 hex)

ENFORCEMENT

<Enforcement> button changes the SRP Enforcement settings between:
'Skip DLLs' and 'No Enforcement'
These settings tell SRP which files should be monitored. For file blocking, SRP uses additional rules and 'Default Security Level' settings.

'Skip DLLs' can control file execution by the file extension (Designated File Types). It also provides extended protection for scripts (BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH), MSI installers, and native Windows executables (COM, EXE, SCR). This is a default setting in Hard_Configurator, because it is most usable for the average users.

'No Enforcement' option can control only scripts (BAT, CMD, JS, JSE, VBE, VBS, WSF, WSH) and MSI files due to 'Windows CMD', 'Windows Script Host', and 'Windows Installer'. The other files are not monitored (for example COM, EXE, SCR, etc.). File blocking (if monitored) can be applied by the combined 'Disallowed\Unrestricted' path rules.

There is also the Enforcement 'All Files' setting available in SRP. But, this setting is no longer supported in Hard_Configurator due to possible incompatibilities with 3rd party security applications.

See also: **How SRP can control file execution/opening.**

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers!TransparentEnabled
Value(Dword)
0 - No Enforcement
1 - Skip DLLs

ENFORCEMENT FOR "ALL USERS" (experimental feature)

This enforcement can cause problems because it can have an impact on Windows administrative processes.

The enforcement for "All users" means that also users from the Administrator group (using high privileges) will be prevented from bypassing SRP restrictions. Normally, the Hard_Configurator settings allow the users from the Administrator group to bypass SRP to avoid problems with blocking administrative tasks in UserSpace.

The enforcement for "All users" is sometimes used in Enterprises to prevent the malware introduced by elevated processes.

In the Home environment, such vectors of attack are usually negligible. Furthermore, one can use the Standard User Account (SUA) to prevent privilege escalation. This is usually a more comprehensive solution as compared to enforcement for **"All users"**.

This enforcement is also incompatible with Strict_Recommended_Settings on Windows 8+ or Recommended_Settings on Windows 7 (Vista). These setting profiles block execution in the whole UserSpace, so some actions related to software installation or Administrative tasks with high privileges can be blocked in ProgramData or User AppData folders. For example, the Windows built-in **Disk Cleanup tool** (cleanmgr.exe) will not work properly to clean system files - it uses dismhost.exe, which will be blocked in the Appdata\Local\Temp folder. Similar problems can happen sometimes for other Administrative tasks, depending on users' settings and installed software.

In the Home environment on Admin account, the expert users can apply the enforcement for **"All users"** (including Administrators) in some situations:

1. Extreme hardening (computer LockDown).
2. Support for older Windows versions.
3. Support for the H_C default-allow setup with some blocked Sponsors (LOLBins).

When using SUA, the enforcement for **"All users"** is not necessary (even for points 1, 2, and 3).

How to apply the enforcement for "All users".

It can be applied by running Hard_Configurator (SwitchDefaultDeny) with the -p switch, for example:

Hard_Configurator(x64).exe -p

SwitchDefaultDeny(x64).exe -p

This can be understood as follows:

1. By using the -p switch, Hard_Configurator (SwitchDefaultDeny) is executed in **"All users"** mode**.** This switch must be used consequently every time the Hard_Configurator (SwitchDefaultDeny) is executed. All restrictions are now applied also for local Administrators.

2. Running Hard_Configurator (SwitchDefaultDeny) without the -p switch means that the user wants to use/change the enforcement to allow Administrators. If the current enforcement blocks **"All users"** then Hard_Configurator shows an alert:
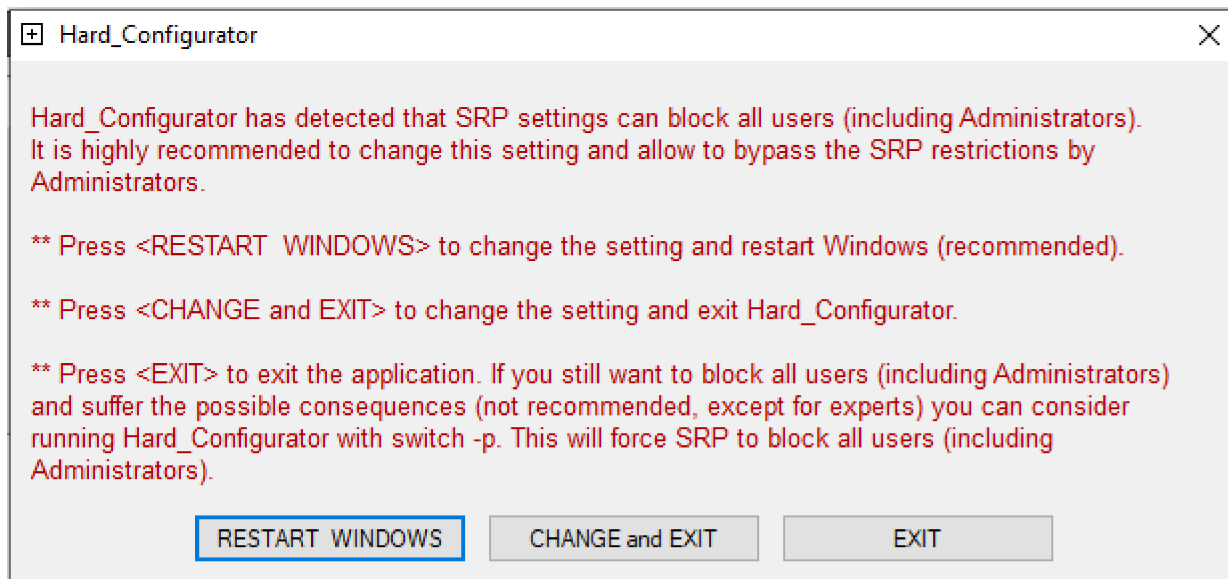
The recommendable choice is <RESTART WINDOWS>. After the restart, the SRP restrictions will not block local Administrators anymore.

If the user chooses <CHANGE and EXIT>, the restrictions will block Administrators in the current Windows session, but Administrators will be allowed after the Windows restart.

If the user chooses <EXIT>, the current restrictions do not change.
This option includes the information that the -p switch must be used to run Hard_Configurator (SwitchDefaultDeny).

When using Hard_Configurator with -p switch, it is necessary to also run SwitchDefaultDeny with this switch. The most convenient way is to edit the commands in the shortcuts by adding the -p switch. After applying the enforcement for **"All users",** the Hard_Configurator (SwitchDefaultDeny) must be consequently executed with the -p switch.

The enforcement for "All users" can be used with SRP default-allow setup or with some default-deny setting profiles, like:
Basic_Recommended_Settings,
Recommended_Settings (on Windows 8+)
MT_Windows_Security_hardening,
Avast_Hardened_Mode_Aggressive.

When applying these setting profiles, the "Install By SmartScreen" ("Run By SmartScreen") can be used in most cases to install applications without switching OFF the SRP protection.

It is not recommendable to apply enforcement for "All users" when using other setting profiles or custom settings.
The common issue will be related to the **"Install By SmartScreen"** feature, which cannot work properly with default-deny setup, when <Update Mode> = OFF. Furthermore, due to blocking processes with high privileges, SRP restrictions cannot be bypassed in UserSpace when using the system **"Run as administrator"** feature.

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers!PolicyScope
Value(Dword)
0 - block all users (including Administrators)
1 - block users except Administrators

BLOCKING SPONSORS

\<Block Sponsors\> button, opens the blacklist of executables that can be used as Sponsors (LOLBins). They can bypass the default deny protection when system files are whitelisted.

The H_C Recommended_Settings allow all Sponsors (except PowerShell in Windows Vista, 7, 8, 8.1), but sometimes users can block Sponsors temporarily. For example, when the computer is connected to the public network. The access to the **chosen executable** is disabled by SRP when the combo box on the **left side** is ticked.
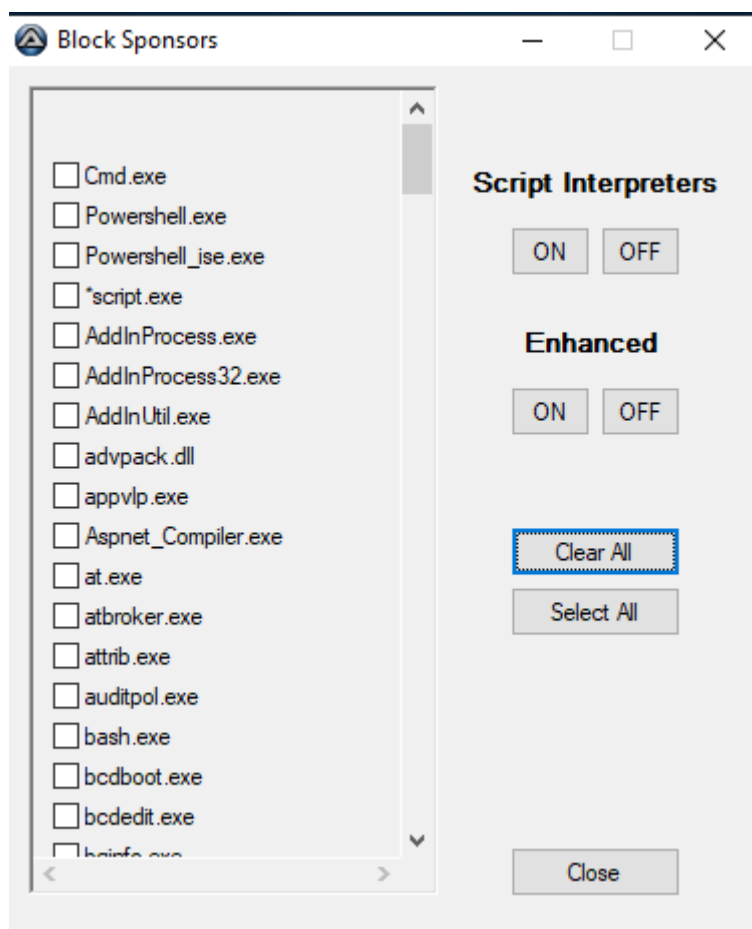
Blocked Sponsors won't run with standard rights, **even from SystemSpace!** Blocking the cmd.exe, automatically blocks the CmdLines with cmd.exe, but the user can still execute the whitelisted BAT and CMD scripts from the Explorer.

POWERSHELL SPONSORS

If PowerShell Sponsors are blocked, then PowerShell scripts cannot run with standard rights when using powershell.exe or powershell_ise.exe, **even from SystemSpace**. Some PowerShell scripts are run by scheduled system tasks, but those tasks operate with Administrative Rights (or higher), so they are not disrupted by SRP.

Blocking PowerShell Sponsors is included in Recommended Settings on Windows Vista, 7, 8, 8.1. It is not included on Windows 10+, because SRP with PowerShell ver. 5.0 (installed by default) can apply Constrained Language mode. The necessary conditions for applying Constrained Language mode are fulfilled, when SRP in Hard_Configurator is set to default-deny (<Default Security Level> = 'Disallowed' or 'Basic User') and PowerShell scripts are not whitelisted in user's TEMP folder.

Constrained Language mode locks down PowerShell to the core elements (no access to: direct .NET scripting, invocation of Win32 APIs via the Add-Type cmdlet, and interaction with COM objects).
When PowerShell is run as administrator, the Language mode changes to FullLanguage. Constrained Language Mode can also be applied in Windows 7 and 8.1, after updating .NET Framework (to the version 4.5.2 or later), and next installing WMF 5.1 (PowerShell 5.1 included).

https://msdn.microsoft.com/en-us/powershell/wmf/5.1/install-configure

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\safer_Hard_Configurator\CodeIdentifiers\Block-Sponsors\
HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\CodeIdentifiers\0\Paths\
{1016bbe0-a716-428b-822e-5E544B6A3100}
{1016bbe0-a716-428b-822e-5E544B6A3101}
...
{1016bbe0-a716-428b-822e-5E544B6A3156}

PROTECTING 'WINDOWS' FOLDER

Setting **<Protect Windows Folder>** to 'ON', denies execution of native Windows executables, Windows CMD, Windows Script Host, and MSI Installer from writable 'Windows' subfolders. So, the execution of EXE, COM, SCR, BAT, CMD, JS, JSE, VBS, VBE, WSF, WSH, and MSI files is blocked, when they are run directly or via command lines with Sponsors: cmd.exe, wscript.exe, cscript.exe, and msiexec.exe.

This protection uses the SRP Disallowed rules, so extended protection is applied to Windows CMD, Windows Script Host, and MSI Installer. It denies the execution even if SRP Default Security Level is set to 'Unrestricted' or these file extensions are not on the SRP Designated File Types list.

To block command lines with Sponsors of other file types (like CHM, HTA, REG, etc.), the Sponsors should be blocked via <Block Sponsors> (hh.exe, mshta.exe, regedit.exe, reg.exe, regedt32.exe, etc.). Still, the execution is allowed, for programs started with Administrative Rights (or higher) independently of SRP restrictions.

The below writable Windows subfolders are added to SRP blacklist:
%SYSTEMROOT%\debug\WIA
%SYSTEMROOT%\Registration\CRMLog
%SYSTEMROOT%\servicing\Packages
%SYSTEMROOT%\servicing\Sessions
%SYSTEMROOT%\System32\catroot2\{F750E6C3-38EE-11D1-85E5-00C04FC295EE}
%SYSTEMROOT%\System32\com\dmp
%SYSTEMROOT%\System32\FxsTmp
%SYSTEMROOT%\System32\Microsoft\Crypto\RSA\MachineKeys
%SYSTEMROOT%\System32\spool\drivers\color
%SYSTEMROOT%\System32\spool\PRINTERS
%SYSTEMROOT%\System32\spool\SERVERS
%SYSTEMROOT%\System32\Tasks
%SYSTEMROOT%\System32\Tasks_Migrated
%SYSTEMROOT%\System32\Tasks\Microsoft\Windows\PLA\System
%SYSTEMROOT%\System32\Tasks\Microsoft\Windows\RemoteApp and Desktop Connections Update
%SYSTEMROOT%\SysWOW64\Com\dmp

%SYSTEMROOT%\SysWOW64\FxsTmp
%SYSTEMROOT%\SysWOW64\Tasks
%SYSTEMROOT%\SysWOW64\Tasks\Microsoft\Windows\PLA\System
%SYSTEMROOT%\SysWOW64\Tasks\Microsoft\Windows\RemoteApp
and Desktop Connections Update
%SYSTEMROOT%\Tasks
%SYSTEMROOT%\Temp
%SYSTEMROOT%\tracing

Some of them are not writable in Windows 10+ (but writable in the prior versions), and a few have not got executable ACL permission.

Registry changes:
[HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\CodeIdentifiers\0\Paths\

Added GUIDs for whitelisted locations:
{1016bbe0-a716-428b-822e-5E544B6A3302}
...
{1016bbe0-a716-428b-822e-5E544B6A3320}

PROTECTING SHORTCUTS

<Protect Shortcuts> button can handle shortcut execution restrictions.
If this option is set to 'ON', then shortcuts can be executed only in 'Windows', 'Program Files' (Program Files (x86)), 'Desktop', 'Power Menu', 'Start Menu', 'Quick Launch', 'Taskbar', and 'Public Desktop' locations.
This restriction is applied because specially crafted shortcuts can bypass Software Restriction Policies.

If <Default Security Level> = 'Disallowed' and <Protect Shortcuts> = 'OFF', then all shortcuts in UserSpace will be blocked, because the LNK shortcut extension is on 'Designated File Type' list.

If <Default Security Level> = 'Basic User' and <Protect Shortcuts> = 'OFF', then shortcuts can run EXE files from any location, even if the LNK shortcut

extension is on 'Designated File Type' list. This allows also running any script by the shortcut when using script Sponsors (cmd.exe, wscript.exe, cscript.exe, hh.exe, mshta.exe).

Registry changes:

Added GUIDs for Unrestricted rules (whitelisted locations):
[HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\CodeIdentifiers\262144\Paths\
{625B53C3-AB48-4EC1-BA1F-A1EF4146FC19}]
...
{625B53C3-AB48-4EC1-BA1F-A1EF4146FC26}]
{99a0fd77-ed0c-4e30-91ff-9d51428d2f21}]
{99a0fd77-ed0c-4e30-91ff-9d51428d2f22}]
{99a0fd77-ed0c-4e30-91ff-9d51428d2f23}]
{B4BFCC3A-DB2C-424C-B029-7FE99A87C641}]
...
{B4BFCC3A-DB2C-424C-B029-7FE99A87C645}]

Added GUIDs for Disallowed rules (also due to ***.LNK*** folder trick):
HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\CodeIdentifiers\0\Paths\
{1016bbe0-a716-428b-822e-5E544B6A3301}
{525B53C3-AB48-4EC1-BA1F-A1EF4146FC19}]
...
{525B53C3-AB48-4EC1-BA1F-A1EF4146FC26}]
{89a0fd77-ed0c-4e30-91ff-9d51428d2f21}]
{89a0fd77-ed0c-4e30-91ff-9d51428d2f22}]
{89a0fd77-ed0c-4e30-91ff-9d51428d2f23}]
{A4BFCC3A-DB2C-424C-B029-7FE99A87C641}]
...
{A4BFCC3A-DB2C-424C-B029-7FE99A87C645}]

UPDATE MODE.

<More SRP ... > <Update Mode> set to ON, allows the execution of EXE (TMP) and MSI files in the hidden folders: ProgramData and user AppData, except the Current Users Startup folder.

<More SRP ... > <Update Mode> set to MSI, allows only MSI files in ProgramData and user AppData folders.

<More SRP ... > <Update Mode> set to OFF, disables the Update Mode in ProgramData and user AppData folders.

PLEASE NOTE: *This feature works only for the default locations of ProgramData and user AppData folders on the system disk, so it will not work if these system folders will be moved to non-standard locations.*

In the default Explorer settings, ProgramData and user AppData folders are not visible for the user, and they are commonly used when installing / updating applications. The <Update Mode> options do not allow other executable files (like scripts, etc.).

<Update Mode> settings can be overwritten by the global whitelisting rules available for EXE (TMP) or MSI files via <Whitelist By Path> options: "Allow EXE and TMP" or "Allow MSI". These rules can allow EXE (TMP) or MSI files in all UserSpace locations.

Lowering the restrictions for the EXE (TMP) and MSI files on Windows 8+ is covered by Windows SmartScreen integrated with Explorer and two Hard_Configurator options: <Harden Archivers> and <Harden Email Clients>. So it does not lower the preventive protection of Hard_Configurator settings on the pre-execution stage. Yet, it does not block the rare cases of primary EXE (TMP) or MSI payloads on the post-exploitation stage. Other types of payloads (like scripts, etc.) are mostly blocked, except when the exploit can get access to the command line to execute some command-lines with LOLBins.

The settings: <Update Mode> = ON, <Harden Archivers> = ON, and <Harden Email Clients> = ON are included in the Hard_Configurator Recommended Settings on Windows 8+ and some other setting profiles. Such setup is well suited for most users, because it avoids much of whitelisting and application updating problems. It is a strong security setup in the home environment on the well updated system with well updated software.

<Update Mode> set to OFF is included in the Recommended Settings on Windows 7 (and Vista), where Windows SmartScreen is not integrated with

Explorer. These settings can block the EXE (TMP) / MSI primary payloads and can be recommendable for users who install / use unpatched and vulnerable software.

Turning OFF the <Update Mode> will require more attention when installing or updating applications. For example, it will allow application auto-updates done via scheduled tasks like in the case of Edge Chromium Dev, Chrome, Firefox or Brave web browsers. But, it will block the standard application auto-updates via downloaded executable updaters like in the case of Opera web browser. Furthermore, on Standard User type of account, installing/updating applications in UserSpace will usually require turning off (temporarily) the Hard_Configurator default-deny protection.

Avast antivirus and <Update Mode> settings.

<Update Mode> = ON is applied in the Hard_Configurator profile prepared for Avast Antivirus (with CyberCapture).

<Update Mode> = MSI is applied in the Hard_Configurator profile prepared for Avast Antivirus set to Hardened Aggressive Mode with EXE files whitelisted globally by another option.

If the Avast Hardened Mode Aggressive is not available via Avast modern GUI, then it can be set in Avast by using:
Menu > Settings > Troubleshooting > Open old settings.

The CyberCapture feature is turned ON by default in Avast. Normally it works only for EXE files originated from the Internet Zone (file must have got MOTW). Furthermore, it does not work for files contained in archives (with some exceptions), flash drives, CD/DVD drives, CD/DVD images, and Memory Cards.

On Windows 8+, the Hard_Configurator Recommended Settings force both SmartScreen Application Reputation and CyberCapture, via the right-click Explorer context menu entry: 'Install By SmartScreen'.

On Windows 7 (and Vista), the integration of SmartScreen Application Reputation with Explorer is not supported. But still, the CyberCapture can be for-

ced via the Hard_Configurator "Install application" entry in the right-click Explorer context menu. Both "Install by SmartScreen" and "Install application" can add the 'Mark Of The Web' to the EXE file before executing it. This triggers AvastCyberCapture also for files contained in archives, flash drives, CD/DVD drives, Memory Cards, and CD/DVD images.

HARDENING ARCHIVERS

<Harden Archivers> option blocks the execution of EXE and MSI files from archiver applications.
This can prevent bypassing the SmartScreen AppRep feature on Windows 8+ or Avast CyberCapture, by executing EXE and MSI files directly from the archiver application. To execute such files, the archive has to be first uncompressed and the user can apply "Install By SmartScreen" or "Install Application" entry from the right-click Explorer context menu.

Pressing this button changes between the settings:
MSI --> ON --> OFF

The 'MSI' setting blocks only the execution of MSI files.
The 'ON' setting blocks the execution of EXE and MSI files.
The 'OFF' setting removes restrictions.

Hard_Configurator supports the below archiver applications:
Windows built-in Zip, 7-Zip, ALZip, Bandizip, B1 Free Archiver, Explzh, ExpressZip, IZArc, PeaZip, PKZip, PowerArchiver, WinRar, WinZip.

REMARKS
You can keep <Harden Archivers> = 'OFF' when applying the Strict_Recommended_Settings because the temporary locations used by archivers are in the user AppData folder that is blocked in this setting profile. The archiver applications hardening uses some SRP rules related to the below folders:

Windows built-in Zip archiver
%USERPROFILE%\AppData\Local\Temp\Temp*_*.zip\

7-Zip
%USERPROFILE%\AppData\Local\Temp\7zO????????\

ALZip
%USERPROFILE%\AppData\Local\Temp_AZTMP*_\

Bandizip
%USERPROFILE%\AppData\Local\Temp\BNZ.??????????????\

B1 Free Archiver
%USERPROFILE%\AppData\Local\Temp\B1FreeArchiver-*-*-*-*-*\

Explzh
%USERPROFILE%\AppData\Local\Temp\?EXTMP??\

ExpressZip
%USERPROFILE%\AppData\Local\Temp\ExpressZip-*-*\

IZArc
%USERPROFILE%\AppData\Local\Temp\$$_????\

PeaZip
%USERPROFILE%\AppData\Local\Temp\.ptmp??????\

PKZip
%USERPROFILE%\AppData\Local\Temp\PK????.tmp\

PowerArchiver
%USERPROFILE%\AppData\Local\Temp_PA*\

WinZip
%USERPROFILE%\AppData\Local\Temp\wz????\

WinRar
%USERPROFILE%\AppData\Local\Temp\Rar$EX*\

HARDENING EMAIL CLIENTS

<Harden Email Clients> option blocks the execution of EXE and MSI files from email client applications. This can prevent bypassing the SmartScreen AppRep feature on Windows 8+ or Avast CyberCapture by executing EXE and MSI attachments directly from the archiver application. To execute such files, the attachments have to be first downloaded to the computer.

Pressing this button changes between the settings:
MSI --> ON --> OFF

The 'MSI' setting blocks only the execution of MSI files.
The 'ON' setting blocks the execution of EXE and MSI files.
The 'OFF' setting removes restrictions.
Hard_Configurator supports the below email client applications:
Mail for Windows 10 (Windows app), Outlook, Claws-mail, eM Client, Foxmail, Hiri, Mailspring, PostBox, Spike, Thunderbird.

REMARKS
You can keep <Harden Email Clients> = 'OFF' when applying the Strict_Recommended_Settings because the temporary locations used by the supported email clients are in the user AppData folder that is blocked by default in this setting profile.

The email client applications hardening uses some SRP rules related to the below folders:

Claws-mail
%USERPROFILE%\AppData\Roaming\Claws-mail\mimetmp\

eM Client
%USERPROFILE%\AppData\Local\Temp\eM Client temporary files*\

Foxmail
%USERPROFILE%\AppData\Roaming\Foxmail*\Temp-*\Attach\

Hiri
%USERPROFILE%\AppData\Local\hiri\temp\

Mailspring
%USERPROFILE%\AppData\Roaming\Mailspring\files***\

The other supported email client applications: Mail for Windows 10 (Windows app), Outlook, PostBox, and Thunderbird do not allow attachment execution. The Spike email client can execute email attachments, but the file is downloaded and blocked in the Downloads folder.

BLOCKING APPINSTALLER

Setting the <Block AppInstaller> to ON applies SRP restrictions to the application installations:

1. Universal Windows Platform (UWP) app installations are allowed only via Microsoft Store.
2. Desktop application installations are not allowed via Microsoft Store.

Due to blocking AppInstaller, the executable WindowsPackageManagerServer.exe can be blocked each time when the user opens Microsoft Store. This event is logged, so several such blocks can be visible in the <Blocked Events / Security Logs>.
Software developers must set <Block Appinstaller> = OFF, to use the winget tool and Windows Package Manager to manage the software prepared for Microsoft Store.

After blocking the AppInstaller, installations/updates of UWP app packages (*.appx, *.msix, *.appxbundle, and *.msixbundle) are blocked when performed via a web browser or by opening the package downloaded to disk. The user will be forced to use Microsoft Store to install UWP apps.

Microsoft Store expanded the offer and included the possibility to publish both UWP apps and desktop applications. So when AppInstaller is not blocked, the users can install also desktop applications directly from Microsoft Store. This possibility will be blocked when <Block AppInstaller> is set to ON.

It is recommendable to restrict AppInstaller because installing / updating the UWP apps outside of Microsoft Store is not safe.
If one needs to install UWP apps outside of Microsoft Store, then the <Block AppInstaller> option can be set to OFF, knowing that precautions should be taken when installing / updating such app packages.

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\CodeIdentifiers\0\Paths\
{1016bbe0-a716-428b-822e-5E544B6A3300}

VALIDATE ADMIN CODE SIGNATURES

<Validate Admin Code Signatures> button enables/disables the User Account Control (UAC) setting to enforce cryptographic signatures on any interactive application that requests elevation of privilege. It is available in Hard_Configurator only for Windows 8+, because it may cause problems on unproperly updated Windows Vista and Windows 7.

Enabling it is not recommendable with default-deny setup, because it will block many unsigned application installers which were accepted by forced SmartScreen (via "Install By SmartScreen" entry in the Explorer context menu).
It is recommendable with 'Allow EXE' setup to prevent the user from running application installers or programs which are both unsigned and require Administrative rights. Most malware files are usually unsigned and want to elevate, so this option is a good preventive feature. Yet, it is worth remembering that Validate Admin Code Signatures is the UAC setting, so if the UAC is bypassed then Validate Admin Code Signatures is bypassed too. It is stronger on Standard User type of account (SUA) as compared to default Admin account, because SUA has a stronger design to prevent such bypasses.

Enabling this setting will prevent auto-updates of the unsigned applications which were installed in 'Program Files' or 'Program Files (x86)' system folder. Such applications have to be installed/updated when <Validate Admin Code Signatures> is set to OFF.

It works best when the user installs digitally signed applications, or unsigned applications which do not require Administrative rights.

When the unsigned file is blocked, then the Error message is displayed, which ends with: *"... A referral was returned from the server".*

POWERSHELL SCRIPTS

<Block PowerShell Scripts> button disables/enables PowerShell script execution (not supported on Windows Vista).
.
If this option is ON, then script file execution is blocked by Windows Policy, but the user can still execute PowerShell **commands and cmdlets.** So, **they** are also allowed via Office macros, DDE, etc. Keep this option 'ON', because scripts are the weak point of most antimalware programs.

In Windows 10+ the script protection is strengthened by combining SRP with PowerShell Constrained Language mode. In Windows 7, 8, and 8.1 the Constrained Language mode is normally not supported, so Hard_Configurator in the Recommended Settings, blocks also PowerShell Sponsors (powershell.exe and powershell_ise.exe). It is worth mentioning that Constrained Language can be also introduced to Windows 7 and 8.1 with updated PowerShell to version 5.0 (or higher).

In Windows Vista, both <Block PowerShell Scripts> and Constrained Language mode are not supported, so Hard_Configurator can apply the protection only via SRP by adding PowerShell script extensions to 'Designated File Types' list, and by blocking PowerShell Sponsors: powershell.exe and powershell_ise.exe (<Block Sponsors> button).
See also the info in **Blocking Sponsors ---> PowerShell Sponsors.**

In Windows 64-bit there are two PowerShell Hosts (32-bit and 64-bit), but both are disabled/enabled by the below registry key:

HKLM\Software\Policies\Microsoft\Windows\PowerShell!EnableScripts
Value (Dword)
0 - script execution is disabled   1 - script execution is enabled

How to run PowerShell scripts with H_C?

If one uses PowerShell scripts frequently, then it is better to enable scripts by setting <Block PowerShell Script> = OFF, but block the PowerShell Sponsors (powershell.exe and powershell_ise.exe) via <Block Sponsors>. In this way, the PowerShell Interpreter will be blocked, except when running PowerShell with Admin privileges (via elevated PowerShell console or elevated file explorer). Using the elevated file explorer (like Q-Dir, Total Commander, etc.) is the best way to perform Administrative tasks and running scripts (Powershell, Windows Script Host, or CMD), without turning off the SRP protection.

WINDOWS SCRIPT HOST

<Block Windows Script Host> button disables/enables Windows Script Host.
If this option is ON, then execution of JS, JSE, VBS, VBE, WSF, and WSH scripts is blocked (also as administrator). Keep this option ON, if SRP <Default Security Level> is not set to 'Disallowed', because only 'Disallowed' setting can force extended SRP protection for those scripts. It is important, because scripts are the weak point of most antimalware programs.

Some scripts can be executed at the boot time, for example:
%SYSTEMROOT%\system32\gathernetworkinfo.vbs
%SYSTEMROOT%\syswow64\gathernetworkinfo.vbs
%SYSTEMROOT%\system32\gatherwiredinfo.vbs
%SYSTEMROOT%\syswow64\gatherwiredinfo.vbs

%SYSTEMROOT%\system32\gatherwirelessinfo.vbs
%SYSTEMROOT%\syswow64\gatherwirelessinfo.vbs

The above scripts are not essential for the Windows system in the home environment, so they can be blocked.
In Windows 64-bit there are two Windows Script Hosts (32-bit and 64-bit).

Registry changes:
HKLM\SOFTWARE\Microsoft\Windows Script Host\Settings
Enabled
Value (Dword)
0   script execution is disabled
1   script execution is enabled

HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows Script Host\Settings
Enabled (REG_DWORD)
0   script execution is disabled
1   script execution is enabled

DOCUMENTS ANTI-EXPLOIT

Important remarks.
The home users should avoid installing MS Office or Adobe Acrobat Reader, except when it is necessary. These applications have so many advanced features, that it is hardly possible to fully protect users against all vulnerabilities.
They can be fully protected only in theory, because the more protection, the greater the chances to see some documents unreadable. Most home users do not use those advanced features at all, and on the contrary, the weaponized documents often use them to exploit/infect the system.
Generally, it is recommendable to use online services or Universal Applications from Microsoft store (**if they use AppContainer**), for managing documents (Office Online, Google Drive, Word Mobile, Excel Mobile, Power-Point Mobile, Xodo PDF, Adobe Reader Touch, etc.).

Such popular applications like Libre Office, WPS Office, SoftMaker Office

are also the better choice, but they are not as safe as the above solutions. For compatibility reasons, most of the active content of documents can be still functional in these applications (macros, OLE, scripts, etc.).

Anyway, some users have no choice and are obliged to use Acrobat Reader or MS Office. So, what can they do to provide enhanced security?

MS Office 2007 and newer versions provide not so bad default protection against weaponized office documents, but the user has to avoid allowing the active content (macros, OLE, DDE, ActiveX, etc.). That is hardly possible for inexperienced users, who usually do not understand the security alerts.

The situation is even worse for Adobe Acrobat Reader, because in many cases, the active content embedded in PDF documents, is allowed by default without any alert. Furthermore, there is no possibility to silently block the active content, because Adobe Acrobat Reader shows the 'Yellow Message Bar' with an option to allow the blocked features.

On Windows 8+ it is recommendable to install Adobe Acrobat Reader DC (from the year 2019 at least), because of AppContainer feature which can mitigate many dangerous actions.

What <Documents Anti-Exploit> option can do:
- **This feature works well for the desktop versions of MS Office** - the versions based on the Universal Windows Platform may ignore it. In such a case it is possible to use the DocumentsAntiExploit tool (via SwitchDefaultDeny) and apply the ON2 settings.
- It can support Microsoft Defender (ConfigureDefender HIGH Protection Level) to mitigate the weaponized documents. Otherwise additional hardening is required by using DocumentsAntiExploit tool.
- VBA interpreter in MS Office is disabled, so VBA Macros (in documents, templates, etc.), VBA Add-ins, VBA UserForms are blocked. This may have a direct impact on the proper functioning of OLE Automation, Form/ActiveX/COM controls, etc.
- On Windows 8+, the dangerous features in Adobe Acrobat Reader DC can be blocked with the 'Yellow Message Bar' (Protected View), and if allowed by the user, then silently mitigated in AppContainer. Otherwise (Windows 7, Adobe Acrobat Reader XI, etc.) an additional hardening is required by

using DocumentsAntiExploit tool.

- The restrictions apply as policies for all accounts and override (but not overwrite) applications' native settings in MS Office and Adobe Acrobat Reader.
- The restrictions cannot be modified by the user from within MS Office and Adobe Acrobat Reader.

The available settings.

<Documents Anti-Exploit> = Adobe + VBA
The VBA interpreter is disabled for MS Office XP/2003, and higher versions up to MS Office 2021 (Excel, FrontPage, Outlook, PowerPoint, Publisher, and Word). In Adobe Acrobat Reader XI/DC, the documents are opened in Protected View (highly restricted sandbox).

<Documents Anti-Exploit> = Adobe
The policy restrictions apply only for Adobe Acrobat Reader XI/DC.
If MS Office is installed, then additional protection is required via the DocumentsAntiExploit tool.

<Documents Anti-Exploit> = OFF
Generally, the above system-wide policies can override but do not overwrite the current user settings.
The OFF setting removes policy restrictions for MS Office and Adobe Acrobat Reader XI/DC, so the current user restrictions can apply. This setting is also displayed when both MS Office and Adobe Acrobat Reader are not installed.

<Documents Anti-Exploit> = Partial
This setting is displayed when the system wide policy restrictions were applied via the external program (like DocumentsAntiExploit tool), and do not match predefined Hard_Configurator settings ('Adobe + VBA', 'Adobe', 'OFF'). It is related only to the system-wide policies, so it does not show the restrictions made for the current user via the DocumentsAntiExploit tool or from within MS Office and Adobe Acrobat Reader applications.

If another Antivirus is installed (instead of Microsoft Defender), or ASR rules are not enabled in Microsoft Defender, then this protection can be insufficient. In such a case, it is recommendable to use the DocumentsAntiExploit tool via SwitchDefaultDeny. This tool can apply extended protective features on a particular account (non-system-wide). Applying such extended settings is also recommendable when the user installed the full MS Office version (not free mobile applications) **based on the Universal Windows Platform (the ON2 setting is required).**

The information about DocumentsAntiExploit tool is included in the PDF document "DocumentsAntiExploit tool - Manual".

RUN AS ADMINISTRATOR

<Hide 'Run As Administrator'> button hides/shows "Run as administrator" entry in the right-click Explorer context menu.
When applying the Hard_Configurator Recommended Settings, the user is asked if the entry "Run as administrator" should be visible. The users who want to use Command Prompt or PowerShell with Administrator rights, should keep it visible and Hard_Configurator automatically will use the setting:
<Hide 'Run As Administrator'> = 'OFF'.
The average users do not run Command Prompt or PowerShell with Administrative Rights, so it is recommendable to hide the "Run as administrator" entry on their computers. This entry will be hidden by setting:
<Hide 'Run As Administrator'> = 'ON'
REMARKS
When <Hide 'Run As Administrator'> is set to 'ON', then "Command Prompt (Administrator)" option in Windows Power Menu, and "Run as administrator" option in the Search context menu, are hidden too. Furthermore, with Hard_Configurator Recommended Settings, the user cannot run files with extensions: BAT, CMD, CPL, and MSC, from UserSpace (= outside of 'Windows', 'Program Files', and 'Program Files (x86)' system folders). Normally, files with those extensions can be opened by using the 'Run as administrator' entry from the right-click Explorer context menu.

Registry changes:
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Explorer!HideRunAsVerb

Value (Dword)
0   "Run As Administrator" is not hidden
1   "Run As Administrator" is hidden

FORCED SMARTSCREEN

<Forced SmartScreen> button adds/removes 'Install By SmartScreen' or 'Run By SmartScreen' entry in the right-click Explorer context menu. Both options **force** file execution with **SmartScreen check** for files located in UserSpace. If the file is located in SystemSpace (inside 'Windows', 'Program Files ...' system folders), then SmartScreen works as usual (**not forced**). This works only on Windows 8+, because on Windows 7 (Vista) the SmartScreen integration with Explorer is not supported.

Pressing <Forced SmartScreen> button changes between values:
'Administrator' -> 'Standard User' -> 'OFF'

- The setting 'Administrator' corresponds to "Install By SmartScreen" entry in the right-click Explorer context menu.
- The setting 'Standard User' corresponds to "Run By SmartScreen" entry in the right-click Explorer context menu.
- The setting 'OFF' removes these entries from the Explorer context menu.

'Install By SmartScreen' works as follows:
1. It can check EXE and MSI files (also via shortcuts), and run them if accepted by SmartScreen Application Reputation (bypassing SRP restrictions). All other files are blocked (with notification).
2. If the EXE / MSI restrictions are lowered by the setting:
   <More SRP ...> <Update Mode> = ON
   then, 'Install By SmartScreen' executes the file with standard rights. Still, the executable can ask for elevation via UAC. The file can run with standard rights (or lower). This can be important for some installations, for example, those performed in UserSpace on SUA.

3. If the EXE / MSI restrictions are not lowered:
   <More SRP ...><Update Mode> = OFF
      then, 'Install By SmartScreen' forces the file to ask for elevation (via UAC), even if the file does not require Administrative rights to run.

On the contrary, the "Run By SmartScreen" entry always executes the file with standard rights (it can ask for elevation like in point 2).
'Run By SmartScreen' is not designed to run files from the root 'c:\' location, because in UserSpace the location has to allow write access with standard rights, but root 'c:\' requires Administrative Rights for that.

(A) Keep <Forced SmartScreen> = 'Administrator' when SRP default-deny setup is activated. If so, then the users can safely:
1. Run programs (with a mouse click or pressing ENTER button) that have been already installed in SystemSpace or put on the Whitelist.
2. Open the media files, documents, and other file types, which are not on the 'Designated File Types' list.
3. Install new programs from UserSpace by using 'Install By SmartScreen' entry from Explorer context menu (only EXE and MSI standalone installers).

(B) Keep <Forced SmartScreen> = 'Standard User' when SRP is deactivated. If so, then 'Run By SmartScreen' entry from the Explorer context menu can be used on-demand to safely run / open any file.

REMARKS
The SmartScreen Application Reputation in Windows 8+ allows some vectors of infection if you have got the executable file like (COM, EXE, MSI, SCR) by using:
★ the downloader or torrent application (EagleGet, utorrent, etc.),
★ container file (ZIP, 7Z, ARJ, RAR, ...) except Windows built-in ZIP,
★ CD/DVD/Blue-ray disc or disc image (iso, bin, etc.),
★ non-NTFS USB storage device (FAT32 pen drive, FAT32 USB disk),
★ Memory Card,
so the file does not have the proper Alternate Data Stream attached.

Registry changes:
HKCR*\shell\Install By SmartScreen\
HKCR*\shell\Run By SmartScreen\

The default shortcut flag **IsShortcut** is changed to **NoIsShortcut** under the below Registry keys:
HKCR\Application.Reference
HKCR\IE.AssocFile.URL
HKCR\IE.AssocFile.WEBSITE
HKCR\InternetShortcut
HKCR\piffile
HKCR\Microsoft.Website
HKCR\WSHFile

RUN BY SMARTSCREEN ENTRY IN EXPLORER CONTEXT MENU.

"Run By SmartScreen" entry is added to the right-click Explorer context menu when Hard_Configurator <Forced Smartscreen> feature is set to **'Standard User'**. So, when 'Run By SmartScreen' entry is used, the files are opened/run **with standard** rights or lower (can ask to elevate).
The feature "Run By SmartScreen" works as follows:
1. Executables (COM, EXE, MSI, SCR) which are located in SystemSpace ('Windows', 'Program Files', 'Program Files (x86)' system folders) are opened normally, without SmartScreen check.
2. Executables located in UserSpace (= outside 'Windows', 'Program Files', 'Program Files (x86)' system folders) are checked by SmartScreen before running.
3. Files from UserSpace with potentially dangerous extensions (scripts, most MS Office files, etc.), are not allowed to open, and the program shows an alert.
4. Shortcut (*.lnk) to any file (target file) is managed as the target file.
5. Shortcuts with a command line in the 'Target' area are always blocked, and the program shows an alert.
6. Compressed archives not supported by Windows build-in unpacker (7Z, ARJ, BZ, BZIP, BZ2, BZIP2, FAT, LHA, LZH, RAR, R00-09, REV, XZ, ZIPX) are not opened - only the short instruction is displayed.

7. Popular file formats related to MS Office and Adobe Reader:
   DOC, DOCX, XLS, XLSX, PUB, PPT, PPTX, ACCDB, PDF
   are opened with the warning instruction and the MOTW is added to the file. If Protected View is not disabled, then these files are always opened in Protected View.
8. Other files (ZIP archives, media, photos, etc.) are opened normally.
9. The disk image files (IMG, ISO NTFS, VHD, VHDX) are not opened - only the short instruction is displayed.

The program has a hardcoded list of unsafe file extensions:
ACCDA, ACCDE, ACCDR, ACCDT, ACCDU, ACM, AD, ADE, ADN, ADP, AIR, APP, APPLICATION, APPREF-MS, APPX, APPXBUNDLE, ARC, ASA, ASP, ASPX, ASX, AX, BAS, BAT, CAB, CDB, CER, CFG, CHI, CHM, CLA, CLASS, CLB, CMD, CNT, CNV, COMMAND, CPL, CPX, CRAZY, CRT, CRX, CSH, CSV, DB, DCR, DER, DESKLINK, DESKTOP, DIAGCAB, DIF, DIR, DLL, DMG, DOCB, DOCM, DOT, DOTM, DOTX, DQY, DRV, ECF, ELF, FON, FXP, GADGET, GLK, GRP, GZ, HEX, HLP, HPJ, HQX, HTA, HTC, HTM, HTT, IE, IME, INF, INI, INS, IQY, ISP, ITS, JAR, JNLP, JOB, JS, JSE, KSH, LACCDB, LDB, LIBRARY-MS, LOCAL, MAD, MAF, MAG, MAM, MANI-FEST, MAPIMAIL, MAQ, MAR, MAS, MAT, MAU, MAV, MAW, MAY, MCF, MDA, MDB, MDE, MDF, MDN, MDT, MDW, MDZ, MHT, MHTML, MMC, MOF, MSC, MSH, MSH1, MSH1XML, MSH2, MSH2XML, MSHXML, MSIX, MSIXBUNDLE, MSP, MST, MSU, MUI, MYDOCS, NLS, NSH, OCX, ODS, OPS, OQY, OSD, PCD, PERL, PA, PI, PIF, PKG, PL, PLG, POT, POTM, POTX, PPA, PPAM, PPS, PPSM, PPSX, PPTM, PRF, PRG, PRINTEREXPORT, PRN, PS1, PS1XML, PS2, PS2XML, PSC1, PSC2, PSD1, PSDM1, PST, PSTREG, PXD, PY, PY3, PYC, PYD, PYDE, PYI, PYO, PYP, PYT, PYW, PYWZ, PYX, PYZ, PYZW, RB, REG, RPY, RQY, RTF, SCT, SEA, SEARCH-MS, SEAR-CHCONNECTOR-MS, SETTINGCONTENT-MS, SHB, SHS, SIT, SLDM, SLDX, SLK, SPL, STM, SWF, SYS, TAR, TAZ, TBZ, TERM, TERMINAL, TGZ, THEME, TLB, TMP, TOOL, TPZ, TSP, TXZ, TZ, URL, VB, VBE, VBP, VBS, VSMACROS, VSS, VST, VSW, VXD, WAS, WBK, WEBLOC, WEBPNP, WEBSITE, WIZ, WLL, WS, WSC, WSF, WSH, WWL, XBAP, XLA, XLAM, XLB, XLC, XLD, XLL, XLM, XLSB, XLSM, XLT, XLTM, XLTX, XLW, XML, XNK, XPI, XPS, XSL, Z, ZFSENDTOTARGET, ZLO, ZOO.

The above list is based on SRP, Outlook Web Access, Gmail, and Adobe Acrobat Reader file extension blacklists.

REMOTE ACCESS

If <Block Remote Access> is set to ON, then the below remote features are disabled:
* Remote Assistance (solicited and unsolicited)
* Remote Desktop
* Remote Shell Access
* Remote Registry Access
The user cannot enable those remote features via System Properties and Control Panel.

If <Block Remote Access> set to OFF, then the user can enable or disable the remote features via System Properties and Control Panel.

It is recommendable for home users to keep <Block Remote Access> set to 'ON'. Remote connections are frequently exploited by malware and hackers.

REMARKS
Changing this option (either set to 'ON' or to 'OFF') always stops 'Remote Registry' service, if it was started. The potential problems may occur when disabling Remote Acess:

"Note that print spooler and directory services replication require access through the remote registry service for certain functions to work properly. Other custom applications may also depend on remote registry access."
http://www.blackviper.com/windows-services/remote-registry/

Registry changes for <Block Remote Access> set to ON:

HKLM\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services
"fAllowUnsolicited" = dword:00000000,   "fAllowToGetHelp" = dword:00000000
"fDenyTSConnections" = dword:00000001

HKLM\SOFTWARE\Policies\Microsoft\Windows\WinRM\Service\WinRS
"AllowRemoteShellAccess"=dword:00000000

[HKLM\SYSTEM\CurrentControlSet\Services\RemoteRegistry]
"Start"=dword:00000004

If "Block Remote Access" is set to OFF the below values are deleted:
fAllowUnsolicited, fAllowToGetHelp, fDenyTSConnections, AllowRemoteShellAccess,

Remote Registry setting does not change on Windows 8+ ("Start=dword:00000004"), but on Windows 7 and Vista it will be changed to "Start=dword:00000003".

16-BIT APPLICATIONS

If **<DISABLE 16-BITS>** = 'ON', then 16-bit applications are disabled.
The 32-bit applications that rely on 16-bit components will not run properly with the setting <Disable 16-bits> = 'ON'.
Windows 64-bit has not got NTVDM subsystem, so 16-bit applications cannot run (yet, there are 64-bit NTVDM alternatives available on GitHub).

Registry changes:
HKLM\SOFTWARE\Policies\Microsoft\Windows\AppCompat!VDMDisallowed
Value (REG_DWORD)   00000001   Disable access to 16-bits
                              00000000   Enable access to 16-bits

SECURING SHELL EXTENSIONS

<Shell Extension Security>
If this option is set to 'ON', Windows is directed to run only those shell extensions, that have been approved by an administrator. Any approved shell extension must be an entry in the Registry key:
'HKLM\Software\Microsoft\Windows\CurrentVersion\Shell Extensions\Approved'

Securing shell extension blocks the well-known path, that malware can exploit for persistence. This option is not included in Recommended Settings, because some applications may have problems with context menus, etc. But, it can be used by advanced users, who knows how to overcome problems with shell integration. See also the possible bypass:

http://oalabs.openanalysis.net/2015/06/04/malware-persistence-hkey_current_user-shell-extension-handlers/

Registry changes:
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Explorer
EnforceShellExtensionSecurity
Value (REG_DWORD)
00000001   Enforce Shell Extension Security
00000000   Do not Enforce Shell Extension Security

PROGRAMS ELEVATION ON SUA

\<Disable Elevation on SUA\>
If this option is set to 'ON', then any operation that requires the elevation of privileges (higher than medium integrity level) will fail on Standard User Account (SUA). The 'User Account Control' alerts are not displayed on SUA, when this setting is 'ON'. The user can see only the alert, that file execution was blocked by Administrator.

When used with SRP, this setting locks down any 'Standard User Account', so the users cannot install/run new programs on SUA. The already installed applications can still run, except if they want to elevate.

There are no problems with Windows Updates, scheduled system tasks, and installing/updating Universal Applications from Windows Store. But, the new installations/updates of desktop applications, have to be made via scheduled tasks with high privileges, or manually after logging on 'Administrator Account'.

If the application uses %UserProfile% for updating with standard rights, then the concrete update/application folder in UserSpace (but not all %UserProfile%) should be whitelisted, and then the updating can be performed on SUA. Whitelisting can be done either from Administrator account, or from the concrete SUA (with temporarily disabled protection).

This option is not included in Recommended Settings, because many users do not like such highly restricted configuration.

It should be mentioned, that the above two account configuration is very hard to exploit and very secure, even when not using third-party security software (anti-exe, anti-exploit, HIPS).

Registry changes:
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System!Consent-PromptBehaviorUser

Value (REG_DWORD)
00000000   Automatically deny elevation requests
00000001   Prompt for credentials on the secure desktop
00000003   Prompt for credentials

ELEVATION OF MSI FILES

<MSI Elevation> button, adds/removes 'Run as administrator' entry in the Explorer context menu, for MSI files.
This entry is visible in the right-click Explorer context menu only when <Hide 'Run As Administrator'> option is set to 'OFF'. Normally, 'Run as administrator' is combined with COM, EXE, BAT files, but not with MSI files.

The setting <MSI Elevation> = ON, can be useful when SRP is activated (MSI files are blocked by default) and the option <More SRP ..> <Update Mode> = OFF. Then, one can bypass SRP by choosing 'Run as administrator' entry from the right-click Explorer context menu.

<MSI Elevation> = ON is included in the Recommended Settings on Windows 7 (Vista).
In the Recommended Settings on Windows 8+ the <Update Mode> = ON, so there is no need to elevate MSI files to bypass SRP.

Registry changes:
HKEY_CLASSES_ROOT\Msi.Package\shell\runas\command
Value (REG_EXPAND_SZ)
"%SystemRoot%\System32\msiexec.exe" /i "%1" %*

<u>DISABLING SMB PROTOCOLS 1.0, 2.0, 3.0</u>

<Disable SMB> button disables/enables Windows SMB Protocols 1.0, 2.0, 3.0. This option requires restarting the computer.

Possible options:
ON123   - SMB 1.0, 2.0, 3.0 disabled
OFF   - SMB 1.0, 2.0, 3.0 not disabled
ON1   - only SMB 1.0 disabled

Disabling SMB 1.0, 2.0, 3.0 does not mean that these features are uninstalled from Windows. The 'OFF' setting is available only when SMB 1.0 is installed.

SMB 1.0 can be installed/uninstalled on Windows 8.1 and later versions via:
Programs and Features > Turn Windows Features On or Off >'SMB 1.0/CIFS File Sharing Support'
or using the Windows system tool: OptionalFeatures.exe.

Disabling SMB in Enterprises requires a thorough investigation, because many important sharing network solutions use this protocol.

<Disable SMB> option is not included in Recommended Settings, because sometimes (rarely), it can be required in the home network for sharing folders/files/printers.
Anyway, in the home networks one should try disabling SMB 1.0, because it is most vulnerable, and sharing devices (network printers, NAS) mostly use SMB 2.0 or 3.0. Home users who do not use local network devices, and sharing services in a home local network, can probably disable all SMB protocols, without any issues. In public networks, one can temporarily disable SMB to harden the system against 0-day remote exploits (like EternalBlue).

https://www.pdq.com/blog/disable-smbv1-considerations-execution/

Registry changes:
HKLM\SYSTEM\CurrentControlSet\Services\mrxsmb10!Start
HKLM\SYSTEM\CurrentControlSet\Services\mrxsmb20!Start
HKLM\SYSTEM\CurrentControlSet\Services\LanmanWorkstation!DependOnService
HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters!SMB1
HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters!SMB2

CACHED LOGONS

<Disable Cached Logons> setting is related to Active Directory Domain (ADD) credential caching. The default Windows configuration caches the last logon credentials for users who log on interactively to ADD. Caching the credentials, let users log on to the domain when no domain controllers are available or when the machine is disconnected from the network. Normally, home networks don't use Active Directory, but rather HomeGroup to share files and printers (removed on Windows 10 ver 1803+).

Typically, in the home networks (even with Active Directory), the Cached Logons feature can be disabled. Secure caching means that the system Local Security Authority (LSA) stores a hash of the 'password hash' (double hashing) in the system registry.
The cached log-on credentials are stored in the 'HKLM\Security\Cache' registry key, which can be available only with system privileges.

<Disable Cached Logons> = 'ON'   disables storing cached log-on ADD credentials.
<Disable Cached Logons> = 'OFF' enables storing cached log-on ADD credentials.

Registry changes:
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon!CachedLogons-Count'
Value (REG__SZ)
10   default Windows value
0   Cached Logons disabled

ENABLING SECURE CREDENTIAL PROMPTING

<UAC_CTRL_ALT_DEL> setting turns ON/OFF the Secure Attention Sequence (SAS), before User Account Control (UAC) prompt. Instead of being automatically taken to a secure desktop with the UAC elevation prompt, users have to press Ctrl+Alt+Del keystroke combination, before the secure desktop is presented. As the SAS can't be emulated other than by physically pressing Ctrl+Alt+Del, the user can be sure that the secure desktop is genuine (not simulated by the malware).

The SAS is rather inconvenient (not recommended) if application elevation is required on a regular basis, but it offers additional protection against malware programs, that can simulate the behavior of common system applications.

Warning.
This feature sometimes fails to show the SAS prompt, that can have unexpected consequences while updating Windows. It is recommendable to turn it off while making Windows Updates.

Registry changes:

HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\CredUI!EnableSecureCredentialPrompting

Value (REG_DWORD)
1   Secure Credential Prompting enabled
0   Secure Credential Prompting disabled

CONFIGURING WINDOWS DEFENDER

<ConfigureDefender> button opens ConfigureDefender application.
It can be useful for changing the hidden features of Microsoft Defender on Windows 10+. It mostly uses PowerShell cmdlets (with a few exceptions). ConfigureDefender application is described in the document:
ConfigureDefenderHelp.pdf

WINDOWS FIREWALL HARDENING

FirewallHardening tool can apply and manage Outbound Block Rules in Windows Firewall by using Windows policies. ***The restart of Windows is required to apply the configuration changes.*** On Windows 10+, the execution of bitsadmin.exe (LOLBin) can be blocked via Windows built-in Exploit Protection ('Disable win32k system calls').

The paths of blocked executables are displayed as a list. Each entry can be managed by using the buttons located at the bottom of the application GUI. The applied rules may be also viewed when using Windows Firewall Advanced settings, but can be managed only by FirewallHardening tool, or by editing the Registry under the key:
HKLM\SOFTWARE\Policies\Microsoft\WindowsFirewall\FirewallRules

User Custom Rules Controls allow the user to manage the rules on the BlockList. Please, use the mouse-click and CTRL key to highlight several items on the BlockList. The block of rules can be selected by using the mouse-click to highlight the beginning of the block and then the combination of <Shift> <Down arrow> to select the block.

<Add Rule> button allows adding the rule for any executable.

<Deactivate Rule> button can make the highlighted rules 'Inactive' on the BlockList, but it does not remove them. If the user does not want to block a particular executable (LOLBin) then the rule should be deactivated, but not removed. The 'Inactive' rule will not be changed when adding/updating the rules.

<Block Rule> button changes highlighted 'Inactive' rules to 'Block' rules.

<Remove Rule> button removes the highlighted rules from the BlockList (and Windows Firewall settings). It can be used to remove the users' custom rules. It should not be used to unblock predefined FirewallHardening rules.

The user can add/remove some predefined rules: *'LOLBins', 'MS Office', 'Adobe Acrobat Reader', 'Recommended H_C'*. They are visible on the right of the application GUI.

★ *'LOLBins'* rules are related to Living Of The Land executables from system folders, which are known to be commonly abused by malc0ders.

★ *'MS Office'* and *'Adobe Acrobat Reader'* rules are related to Word, Excel, PowerPoint, Equation Editor, and Acrobat Reader applications.

★ *'Recommended H_C'* is a part of *'LOLBins'*. These rules are suited to users who installed the Hard_Configurator Windows hardening application and applied the <Recommended Settings>.

Using the FirewallHardening Log of blocked events.

The user can enable auditing Windows Firewall with Advanced Security in category 'Object Access' and subcategory 'Audit Filtering Platform Connection' (Event Id=5157). This can be done by choosing the radio button 'ON', under 'Start logging events'.

If auditing is enabled, then the blocked events can be filtered from Windows Security Log by pressing the <Blocked Events> button, visible under the OFF/ON radio buttons. The Security Log can store these events for several hours (usually 48 hours).

Please note, that <Blocked Events> output can show only the events related to the current FirewallHardening BlockRules. After deactivating or removing a particular rule, the blocked events caused by this rule are skipped in the <Blocked Events> output. If the FirewallHardening BlockList is empty or all rules are inactive, then the <Blocked Events> output will not include any blocked event.

Loading and saving BlockLists.

'BlockList' <Load> and <Save> buttons allow using several BlockLists from the *.fhbl files. The new FirewallHardening version is usually published with the file 'UpdateFirewallRules.fhbl' that contains the new rules. Each rule starts with terms: Block : , Inactive : , Block!: , Inactive!:

The exclamation mark means that the file path is not checked, so the rule will be added even if the file does not exist on the disk. The rule without exclamation mark is checked, and if the file is not on the disk in the required location, the rule will be skipped. But on Windows 32-bit, rules related to Windows 64-bit are always ignored (not loaded), even rules with the exclamation mark. The rules can include the known Windows environment variables like:

%SystemRoot%
%ProgramData%
%ProgramFiles%
%ProgramFiles(x86)%
%SystemDrive%

FirewallHardening can also accept a few custom variables like:

%System32%   (usually C:\Windows\System32),
%SysWOW64%   (usually C:\Windows\SysWOW64),
%Framework%   (usually C:\Windows\Microsoft.NET\Framework),
%Framework64% (usually C:\Windows\Microsoft.NET\Framework64),
%PowerShell%   (usually C:\Windows\System32\WindowsPowerShell\v1.0),
%PowerShell64% (usually C:\Windows\SysWOW64\WindowsPowerShell\v1.0).
It may happen that the rule included in the file 'UpdateFirewallRules.fhbl' may be already on the FirewallHardening BlockList. In such a case the new rule will be skipped. Other new rules are initially added at the end of the FirewallHardening BlockList, but after running the tool a second time all rules are sorted, so the 'Block' rules are at the beginning and 'Inactive' rules at the end. The 'Inactive' rules are related to LOLBins that are not included in the 'Recommended H_C' preset.

TROUBLESHOOTING

Hard_Configurator troubleshooting.
1. If the system hangs after reboot (very rarely), then this can be a sign, that SRP or one of the program restrictions has blocked something important from loading at the boot time.
2. The simplest method to solve this is using one of System Restore Points.
3. Another solution is booting into Safe Mode and running Hard_Configurator to whitelist the blocked entry or deactivate restrictions (<Switch OFF/ON SRP> + <Switch OFF/ON Restrictions> + <APPLY CHANGES>).

Using TOOLS.
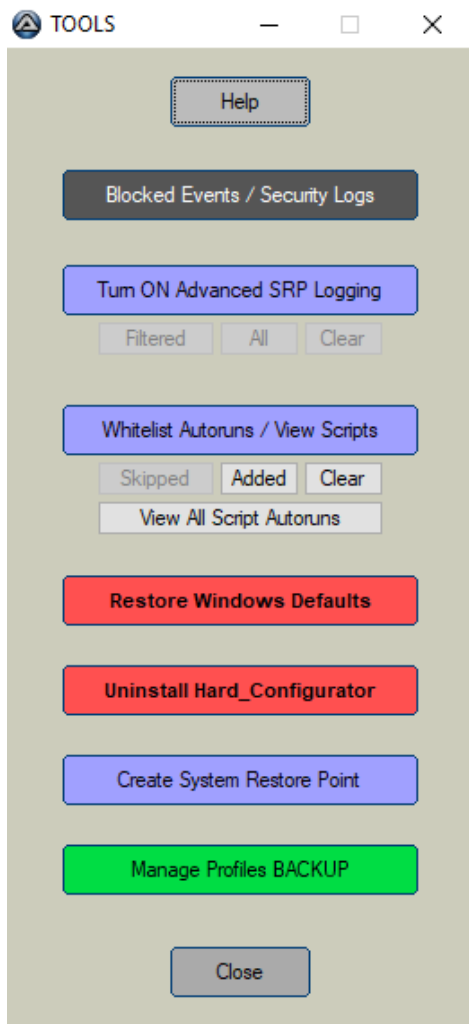
Pressing <Tools> button allows some tools, that can help to prevent blocking important processes in UserSpace, restore Windows defaults, make a System Restore Point, backup and restore predefined profiles, or uninstall Hard_Configurator.

<Blocked Events / Security Logs>

When the program/script is blocked by Hard_Configurator, the information is usually written in the Windows Event Log. This option filters the output of NirSoft tool: FullEventLogView to retrieve information about the blocked events and some security – related events.
The config file uses events ID as follows:

★ **SRP** (provider: Microsoft-Windows-SoftwareRestrictionPolicies)

- 865 --> restricted by policy level
- 866 --> restricted by path rule
- 867 --> restricted by certificate rule
- 868 --> restricted by hash or zone rule
- 882 --> other

★ **SRP** (provider: MsiInstaller)

- 1007 --> installation of MSI file is not permitted by SRP
- 1008 --> installation of MSI file is not permitted due to an error in SRP

★ **Non-SRP related**

 Windows Script Host (provider: Windows Script Host**)**
- 1000 --> Attempt to execute Windows Script Host with Administrative Rights while it is disabled

 PowerShell (provider: Microsoft-Windows-PowerShell**)**
- 4100 --> PowerShell encounters an error (also when script execution is disabled via policy)

Blocked/Audited by ASR rules, Controlled Folder Access, Network Protection (provider Microsoft-Windows-Windows Defender)
- 1121-1128

Windows Defender Antivirus event IDs (provider Microsoft-Windows-Windows Defender**)**
- 1000, 1006, 1007, 1008, 1015, 1116, 1118, 1119

\<Turn ON Advanced SRP logging\> (Verbose trace logging of SRP).

This option activates Verbose trace logging of SRP, by changing the Registry:
HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\CodeIdentifiers!LogFileName
Value (REG_SZ)
%SYSTEMROOT%\Hard_Configurator\SRP.log

It can handle the events when the processes **run with Administrative** privileges. It extends the logging capabilities of \<Blocked Events / Security Logs\> option.

\<Turn ON Advanced SRP logging\> option puts the info about processes to the file SRP.log. Yet, this log has usually many entries from SystemSpace, so some filtering is required. The \<Filtered\> button checks SRP.log and leaves only entries related to scripts or processes which were run from UserSpace.

Let's look at example related to the Hard_Configurator settings with \<Update Mode\> = OFF, which blocks file execution with standard rights also in AppData and ProgramData folders.
We assume that some application 'myapp.exe' is run via "Install By SmartScreen" (uses Admin Rights in this setup) and the entries in the log look as follows:
myapp_setup.exe (PID = 7246) identified C:\Users\Admin\AppData\Local\Temp\is-PPQV9.tmp**myapp_setup.tmp** as Unrestricted using default rule, Guid = {11015445-d282-4f86-96a2-9e485f593302}"

So, we know that myapp.exe is wrapped and uses **myapp_setup.tmp** to execute in the temporary folder:
'C:\Users\USERNAME\AppData\Local\Temp\is-PPQV9.tmp\'.

Now, the **myapp_setup.tmp** file can be whitelisted by path:
'C:\Users\USERNAME\AppData\Local\Temp\is-?????.tmp**myapp_setup.tmp**'
or by hash (if the file **myapp_setup.tmp** was not deleted).

<Whitelist Autoruns / View Scripts>
Some processes can be loaded at the boot time from UserSpace (= outside 'Windows', 'Program Files ...' system folders). They should be whitelisted by path in SRP to load properly. Sysinternals Autorunsc command-line utility allows finding the paths of those processes.
<Whitelist Autoruns / View Scripts> option can filter out all numerous autoruns from SystemSpace leaving only a few entries from UserSpace. They can be seen when pressing <Added> button.
Rarely, the autoruns can have a complicated structure, and the filtering algorithm may give up. Those entries should be checked manually - they can be seen when pressing <Skipped> button.
Pressing <View All Script Autoruns> shows all scripts (from System and User Space) started at the boot time. This option may be helpful when the user wants to disable Windows Script Host, PowerShell or Windows CMD.

<Restore Windows Defaults>
This option allows restoring all system-wide Windows Registry keys that could be changed by Hard_Configurator, to default values (including ConfigureDefender and FirewallHardening). Those values are mostly the same, as before installation of Hard_Configurator program, except when programs that utilize SRP were installed or the user tweaked himself the Registry.
Please note: The System Restore settings and user-dependent settings made by Documents Anti-Exploit from SwitchDefaultDeny tool, are not restored.
System Restore settings can be managed from the Control Panel or by running the Windows tool --> SystemPropertiesProtection.exe.
This option requires rebooting the system.

<Uninstall Hard_Configurator>

Hard_Configurator cannot be uninstalled via Windows uninstall feature, because the Hard_Configurator entry is deleted from the list of installed applications (after first run). This prevents users from uninstalling Hard_Configurator without restoring Windows default settings.

This option performs **<Restore Windows Defaults>** and removes Hard_Configurator files from disk.

<Create System Restore Point>

Makes Windows restore point named Hard_Configurator. If the System Restore feature was turned off, then it will be turned on, and the 1GB of the disk space will be reserved for the restore points.

<Manage Profiles Backup>
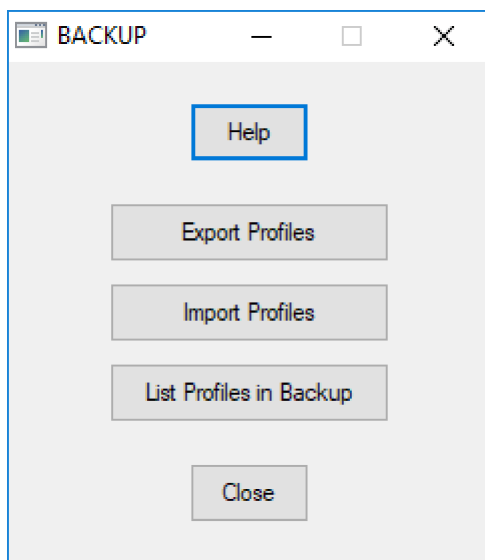
Hard_Configurator can back up its Profile Base (all saved Whitelist Profiles and Setting Profiles) into the one compressed backup file with the '.hbp' extension. It is useful when making a fresh Windows installation, because user Whitelist Profiles are stored in the Registry and Setting Profiles in the folder : '%SYSTEMROOT%\Hard_Configurator\Configuration', and they will be lost when making the fresh Windows installation. So, before the fresh installation, the user has to make a backup, and next, copy the backup file (or the folder with backup files) to the flash drive or another non-system disk.

Hard_Configurator saves by default the backup files in the folder: '%SYSTEMROOT%\Hard_Configurator\Backup'
After the installation, the Profile Base can be recovered from any backup file.

Export Profiles
Makes a backup of the actual Profile Base. All Setting Profiles from '\Hard_Configurator\Configuration' folder and all Whitelist Profiles from the Registry, are exported to password-compressed file.

List Profiles in Backup
Displays the report about profiles in the backup file.

The report shows all profiles contained in the backup and points out which profiles will not be imported (because they have the same names as some profiles in the Profile Base).

In the above example, the backup has not any Whitelist Profile (no *.whl fi-
les) and no Setting Profile can be imported - all Setting Profiles (*.hdc files)
are already in the Profile Base (Duplicated Setting Profiles).

Import Profiles
Imports new profiles from the backup. Importing the profiles do not change
the actual Hard_Configurator settings (SRP settings and Restriction settings),
only Profile Base is updated.
When the user wants to change Hard_Configurator settings, it is possible
from the main window by pressing the option buttons or by loading the profi-
le from the Profile Base (the buttons: <Load Save> for Whitelist Profiles and
<Load Profile> for setting Profiles). Imported profiles do not overwrite the
profiles that were already in the Profile Base. If the profile in the backup has
the same name as the profile in the Profile Base, then it will not be imported.

Frequently Asked Questions

Abbreviations used in this FAQ:

H_C   - Hard_Configurator

AV   - Antivirus application

SRP   - Software Restriction Policies (Windows built-in security feature)

UAC - User Account Control

SUA - Standard User Account

AA   - Administrator Account; not to be confused with 'Built-in Administrator Account' (disabled by default), that can be used to boot Windows to 'Audit mode'.

Basic concepts:

Standard rights (standard user rights, Medium Integrity Level)

These are standard (default) rights granted by the Windows system to processes initiated by the user on AA or SUA. Access to higher rights is controlled by User Account Control (UAC). This feature was introduced with Windows Vista.

An Administrator Account (AA) created during a fresh installation of Windows, or any account created manually by the user (AA or SUA), is limited to standard rights by UAC.

Administrator rights (Administrative rights)

A process initiated by the user on AA or SUA may be elevated to Administrator rights and access important, high privileges. Process elevation is controlled by User Account Control (UAC).

Process elevation cannot be done on SUA. If the process is initiated by the user logged on SUA, then process elevation is redirected to AA (**account change SUA ---> AA, admin password required**). The elevated process runs in fact on AA (not on SUA).

H_C smart default-deny setup

Selected Windows built-in security features can restrict Windows, archiver applications, email client applications, MS Office, and Adobe Acrobat Reader with smart default-deny protection. These features are normally disabled in Windows. H_C allows the user to enable them, make configuration changes, and displays the user's chosen settings. After configuration, real-time protection comes *only* from Windows' built-in security features.

SystemSpace

The following file locations (system folders and subfolders) are defined as SystemSpace and are whitelisted by default in H_C:

'Windows'

'Program Files'

'Program Files (x86)' - only on Windows 64-bit

'ProgramData\Microsoft\Windows Defender'.

UserSpace

All locations on the *user's local* drives (also USB external drives) which are not included in SystemSpace, are defined as UserSpace. *Network locations* are excluded either from UserSpace or SystemSpace. UserSpace locations are usually writable by processes running with standard rights. All executables in UserSpace are blocked by default with H_C's default-deny setup, except when whitelisted or initiated by the user via "Run as administrator" (see also the Elevated Shell).

PLEASE NOTE: *The terms SystemSpace and UserSpace are specific to H_C settings. They should not be confused with the terms 'System Space' and 'User Space', which can have a more general meaning.*

Elevated Shell

Normally, the user on AA or SUA may initiate applications *only with standard rights*. However, this can be changed by accessing an elevated shell: PowerShell (Administrator), Command Prompt (Administrator), etc. An alternative solution is to run Total Commander via "Run as administrator". As long as the applications are initiated from the elevated shell, SRP (configured by H_C) and UAC will ignore them (i.e., no UAC alerts or SRP restrictions). This can be useful when doing administrative tasks on the computer.

Questions and answers:

What is conventional default-deny protection?
It allows all installed applications and system processes but blocks by default all new executables, except those which are whitelisted. Some executables may be whitelisted automatically. Others must first be whitelisted by the user, in order to run. It is the user's responsibility to whitelist clean files.

What are the advantages of H_C's smart default-deny vs conventional default-deny protection?
Smart default-deny makes the security setup more usable, while maintaining a high level of protection in the home environment. Hard_Configurator includes the below smart features:

• Forced SmartScreen, which allows safely to pass by SRP restrictions. Forced SmartScreen is supported on Windows 8+.
• SRP set to allow executables initiated with Administrator rights.
• SystemSpace folders/subfolders whitelisted by default.
• <Update Mode> feature which can whitelist the ProgramData and user AppData (hidden) folders for EXE (TMP) and MSI files.
• Some files in system 'Windows' folder may be blacklisted by the user when using <Block Sponsors> settings. Some folders may be blacklisted by options: <Harden Archivers> and <Harden Email Clients>.

These features allow installing most applications without much whitelisting or turning OFF the protection. Furthermore, Windows Updates, software updates, and scheduled tasks can automatically bypass SRP restrictions.
It is worth mentioning that Forced SmartScreen significantly extends the SmartScreen protection.

Are H_C's smart features safe?
They can be considered as safe in the home environment. Smart features can be bypassed in Enterprises because of targeted attacks and exploits. Also, certain H_C restrictions, e.g. "Block remote access", are not practical in enterprises

Will H_C smart default-deny setup block system processes, Windows Updates, or system scheduled tasks?

No. System processes, Windows Updates, and system scheduled tasks are not started directly by the user. These are initiated with higher than standard rights and automatically bypass SRP restrictions configured with H_C.

Will H_C smart default-deny block updates of user applications?

Not on Windows 8+ in the Recommended Settings with <Update Mode> feature. But, on Windows 7 (Vista) many application updates will be blocked, except when the H_C profile Windows_7_Basic_Recommended_Settings or the profiles for Avast AV are applied.

How to update applications with the H_C Recommended Settings.

On Windows 8+ the applications usually can auto-update without problems. The manual updates with standalone EXE or MSI installers can be done via "Install By SmartScreen" entry in the right-click Explorer context menu.

If the Recommended Settings are applied on Windows 7 (Vista), then the H_C protection should be temporarily turned off to allow software updates. The user should be very cautious to run only safe executables.

How to update applications with H_C's custom default-deny settings.

1. Use the SwitchDefaultDeny tool to switch OFF the Default Deny Protection temporarily.
2. Run & update the application normally.
3. Switch ON the Default Deny Protection.

Is it safe to whitelist SystemSpace?

Generally, it is safe in the smart default-deny setup. SystemSpace locations are usually not writable with standard rights. There are known exceptions, but they are covered by H_C's <Protect Windows Folder> setting.

The exploit or malware cannot silently drop payloads to SystemSpace when running with standard rights.

Are all applications installed in SystemSpace?

Usually they are, and this is recommended by Microsoft. However, some legal applications still install in UserSpace. Most of them are installed in App-

Data or ProgramData folders, which are whitelisted by default on Windows 8+ with H_C's Recommended Settings, and on any Windows version with Windows_*_Basic_Recommended_Settings profile or with profiles prepared for Avast AV.

If the user applied H_C's Recommended Settings on Windows 7 (Vista), then applications installed in UserSpacehave have to be whitelisted manually.

What is the difference between an AA and SUA?

Technically, UAC warns the user when the process wants to get higher than standard privileges, and if allowed, the process runs without changing the user context. Let's assume that we have a default admin account ("King" user) and SUA ("Alice" user).

If the process is initiated with standard rights as "King" and UAC prompt is accepted, then the process will run with high privileges also as "King".

The above is not possible on SUA ("Alice" user), because this account cannot host the processes running with higher privileges. If one will initiate the process as "Alice", then Windows will prompt for Administrator credentials ("King" user credentials). After inserting the credentials, the process will run as "King" with high privileges even when the process window still runs on the "Alice" desktop with standard privileges.

Is SUA more secure than AA?

Yes, most definitely. On SUA, any unelevated processes (running with standard rights or lower) do not share the same user account as elevated processes. Malware or exploits cannot run with Administrator rights on SUA - they must first escape to the Administrator account. This is hardly possible, because Microsoft usually patches any system vulnerabilities which might allow malware to escape from SUA. H_C's smart default-deny setup relies on blocking unelevated programs, so SUA is an ideal environment for the H_C.

When should SUA be used instead of AA?

SUA should be considered a vital part of any security solution when *using a vulnerable* system, or popular & vulnerable software. However, it is not necessary to use SUA with H_C's smart default-deny *when Windows 10+ and all installed software are updated regularly.* A well maintained system that includes H_C is a dead-end for malware/exploits in the home environment.

How to install applications on Windows 8+ with the H_C Recommended Settings.

Forced SmartScreen feature is available only on Windows 8+.

In the Recommended Settings, the Forced SmartScreen feature is integrated with <Update Mode> = ON. So, the "Install By SmartScreen" entry in the right-click Explorer context menu can be used to install applications. This works well for EXE and MSI standalone installers. When <Update Mode> is set to ON, the installation process does not force high privileges and the application always installs in the right user profile.

The "Install By SmartScreen" entry will not work for non-standalone installers, for example when the installation must be done from CD/DVD drives, CD/DVD images, archives containing the installation files copied from CD/DVD, etc. In such cases, the user must disable default-deny protection temporarily with SwitchDefaultDeny tool like for Windows 7 (see below).

How to install applications on Windows 7 (Vista) with the H_C Recommended Settings.

1. Use the SwitchDefaultDeny tool to turn OFF the protection temporarily.
2. Install the application normally (by using left mouse-click or pressing the Enter key).
3. Whitelist the application if necessary.
4. Use the SwitchDefaultDeny tool again to turn ON the protection.

Why Recommended H_C settings are best as a starting default-deny setup?

New users of default-deny protection should be aware that it requires more skill than using an AV alone. Please use *only* the Recommended H_C settings along with your AV, until you are comfortable and familiar with H_C. Prematurely adding advanced H_C settings or more security software to this configuration, may lead to complications and user discouragement with default-deny protection.

Who should consider applying advanced H_C settings?

Recommended H_C settings provide strong preventive protection against running malware/exploits in the system. Advanced H_C settings can mitigate the malware or an exploit that is already running in the system (post-exploitation prevention). When using well-patched software on updated Windows 10+, advanced settings are not required.

Can advanced settings spoil the system?

On most computers, even maximum H_C settings cannot break anything important in the system, but some applications may be not fully functional. Enabling advanced settings will usually require more whitelisting, more researching of logs, etc., and may be annoying for most users. If so, then the user should restore Recommended Settings.

How to restore Recommended Settings.

1. Press <Recommended Settings> green button,
2. Press <APPLY CHANGES> button.

Restoring the Recommended Settings preserves the user's whitelisted entries and blocked file extensions.

How to apply advanced H_C settings.

Advanced settings can be activated by turning ON additional individual H_C options, or by loading the setting profile (<Load Profile> button).
It is advisable to begin with the Recommended_Enhanced profile. This may be done by loading the file: Windows_*_Recommended_Enhanced.hdc, where the asterisk replaces the Windows version (7, 8, or 10).
Recommended_Enhanced profile will enable the Recommended Settings, and some well known Sponsors/LOLBins will be blocked (including Script Interpreters).

PLEASE NOTE: *It is not advisable to apply multiple advanced settings at once. When using advanced settings, the user should occasionally check for blocked entries (<Tools><Blocked Events / Security Logs>). This is because sometimes there is no alert when a process is blocked by Windows policies.*

What is a Sponsor?

A Sponsor is an executable from SystemSpace (usually from system 'Windows' folder), that can run another file.

For example, the executable **powershell.exe** can run PowerShell script files *.ps1. Sponsors can be used by an attacker to bypass default-deny protection (see also LOLBins). They are frequently used in targeted attacks on organizations and businesses, especially via exploits. Blocking some Sponsors in the home environment can be important for people who use a vulnerable system or software.

In the Recommended Settings, Windows Script Host Sponsors (wscript.exe and cscript.exe) are blocked by SRP. Furthermore, PowerShell Sponsors (powershell.exe and powershell_ise.exe) are restricted by Constrained Language mode in Windows 10+ and blocked by SRP in Windows Vista, 7, 8, 8.1. These Sponsors are the most popular Script Interpreters. Some other Interpreters (mshta.exe, hh.exe, wmic.exe, etc.) can be blocked in H_C by using <Block Sponsors> option. Unfortunately, a few of them can be used occasionally by older software, usually those related to peripherals. Applications and web browser plugins may also use Interpreters for some actions, though most applications and plugins do not use them at all.

In H_C, Sponsors are blocked for processes running with standard rights, but allowed for administrative processes running with higher rights.

Can wildcards be used for whitelisting files and folders?

Yes, they can. Here are some examples, where the random characters are replaced by wildcards to whitelist the particular EXE file:

- C:\Users\Alice\Fly2theMoon\App.**1928-0928**\setup_**101989873**.exe
- C:\Users\Alice\Fly2theMoon\App.**????-????**\setup_**?????????**.exe
- C:\Users\Alice\Fly2theMoon\App.*****\setup_**?????????**.exe
- C:\Users\Alice\Fly2theMoon\App.*****\setup_*****.exe
- C:\Users\Alice\Fly2theMoon\App.**********

These rules (except the first) are correct, and the EXE file will be whitelisted even when the random numbers will change after some time. The last rule is most general because it will whitelist many other files and folders, for example: C:\Users\Alice\Fly2theMoon\App.malware\virus.js

@Andy Ful