Hard_Configurator FAQ

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)
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, new privileges. Process elevation is controlled by User Account Control (UAC). If the elevated process is initiated on AA (with standard rights), then process creation and elevation take place on AA, and the process continues to run on AA (account change not required). If it is initiated on SUA, then process creation and elevation take place on AA, and the process no longer runs on SUA (account change SUA ---> AA, admin password required).

H_C smart default-deny setup
Selected Windows built-in security features can restrict Windows, 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.

PLEASE NOTE: Most configuration changes are activated immediately, some requires Log OFF from the user account, and a few require rebooting the computer. Use the <APPLY> red button from the H_C main panel to finish the configuration.

SystemSpace
The following file locations (folders and subfolders) are defined as SystemSpace and are whitelisted by default in H_C:
C:\Windows
C:\Program Files
C:\Program Files (x86) - only on Windows 64-bit
C:\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 writable by processes running with standard rights. All executables in the UserSpace are blocked by default with H_C's default-deny setup, except when whitelisted or initiated by the user via "Run As SmartScreen" (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 SmartScreen". The user who wants to access the elevated shell must first accept the UAC prompt. As long as the applications are initiated from the elevated shell, SRP and UAC will ignore them (i.e., no UAC alerts or SRP restrictions). This can be useful when doing administrative tasks on the computer.

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, e.g. by certificate or path rules. 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 computer more usable, while maintaining a high level of protection in the home environment. Hard_Configurator includes three smart features:

1. Forced SmartScreen (replaces "Run as Administrator"), which can be activated by the setting <Run As SmartScreen> = Administrator. Forced SmartScreen is supported on Windows 8, 8.1, and 10.
2. SRP set to allow executables initiated with Administrator rights.

3. SystemSpace folders/subfolders whitelisted by default. Some files in C:\Windows may be blacklisted by the user when using <Block Sponsors> settings.

These features allow installing most applications without whitelisting or turning OFF the protection. Furthermore, Windows Updates and system 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 are very safe in the home environment, against malware in the wild. Smart features can be bypassed in Enterprises because of targeted attacks. 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?
Occasionally. Some applications download the updater and run it from the Temp folder in user profile with standard rights. In this case, the update will be blocked by the H_C default-deny settings.

How to update applications on *Administrator account* with H_C's default-deny settings.
If the update is blocked, then the application or updater should be run with Administrator rights by using "Run As SmartScreen" (on Windows 8, 8.1, 10) or "Run as administrator" (on Windows Vista or Windows 7).

How to update applications on *SUA* with H_C's default-deny settings.
There may be a problem if the application is installed in the **user profile**, because then an update should not be performed with Administrator rights. Why? If it is run with Administrator rights, then it will usually search the application files in the administrator profile and not in the SUA profile. *The update will thus fail, or will be installed in the wrong user profile*.
H_C users should check as follows:

1. If the application *is not installed in the user profile*, then the update can be done on Administrator account as described above.
2. If the application *is installed in the user profile*(e.g. in the folder C:\Users\Alice when the user name is Alice), then the user must:
    - turn OFF protection temporarily using "Switch Default-Deny";
    - run the application normally and let it autoupdate;
    - if you use an external updater, then you can use “Run By SmartScreen” from the Explorer context menu to run updater with SmartScreen check.
    - turn ON the protection using "Switch Default-Deny".

Is it safe to whitelist SystemSpace?
Generally, it is safe in 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. These applications have to be whitelisted manually. For users who frequently install such applications, default-deny protection may be inconvenient.

What is the difference between an AA and SUA?
Processes initiated by the user **cannot run** with Administrator rights on SUA. If a process running on SUA requires Administrator rights, then the UAC prompt appears, and the user must provide an Administrator password to log on to the AA. After accepting the UAC, the process is no longer running on SUA, but on AA (*user account is switched for that process only:* SUA ---> AA).

This behavior is quite different when a process is initiated on AA, because the user is not obliged to provide the Administrator password. Instead, the UAC prompt asks for a simple "Yes" or "No". After accepting the UAC prompt, the process continues running on the same AA (user account is not switched for that process).

Is SUA more secure than AA?
Yes, most definitely. On SUA, unelevated processes (running with standard rights or lower) do not share the same user account as elevated processes. This is not true on AA. It is much easier to exploit something when both unelevated and elevated processes are running on the same AA account. Malware or exploits cannot run with Administrator rights on SUA - they must first escape to an 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 (running with standard rights), so SUA is an ideal companion to 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 which includes H_C is a dead end for malware/exploits in the home environment.

Does Forced SmartScreen work well on SUA?
Yes, if the application installs in SystemSpace (usually in C:\Program Files). There can be a problem if it installs in user profile, which lies in UserSpace. Why? Because with H_C smart default-deny, Forced SmartScreen uses Administrator rights. Applications which are intended to install in SUA profile, are installed in Administrator profile - even when the

installation is initiated from SUA. The user on SUA cannot run applications from Administrator profile, since Windows isolates user profiles from one another. In this case, the user must disable default-deny protection temporarily, and install the application without using "Run As SmartScreen".

How to install applications on SUA.

1. Run the application installer by using "Run As SmartScreen" option from the Explorer right-click context menu.
2. Check the default installation folder.
3. If it is in the Administrator profile, then cancel the installation and continue with Steps #4-7. If not, then continue with the installation and skip Steps #4-7.
4. Use “Switch Default-Deny“ to turn OFF the protection temporarily.
5. Install the application normally (by left mouse-click or pressing the Enter key).
6. Whitelist the application in the UserSpace.
7. Use “Switch Default-Deny“ to turn ON the protection.

Why Recommended H_C settings are best as a starting 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 in the system.
Advanced H_C settings can mitigate the malware or an exploit which is already running in the system. When using well-patched software on updated Windows 10, advanced settings are not required.

Will 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). This will enable the Recommended Settings, and some well known Sponsors will be blocked (including Script Interpreters).

PLEASE NOTE: It is not advisable to use 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 the SystemSpace (usually from C:\Windows), that can be used by an attacker to bypass default-deny protection. Sponsors include also many LOLBins (Living Off The Land Binaries). 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 H_C's 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, scrcons.exe) can be blocked in H_C by <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.1928327467-092837\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.**

Those 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