SoftProtect-AntiDebug - Help Index

SoftProtect-AntiDebug - Description

Anti-debug protection component for Windev® applications

Author : Francis MOREL francis.morel@gmail.com

Révision 4.00A dated 03.23.2017

Description

SoftProtect-AntiDebug is the Windev component primarily intended to secure your applications and sensitive data.

With a debugger or a specialized tool available as free download on the web (eg **HFPRT**), it is very easy to intercept all functions of the Windev dll and to view all of their parameters.
Although improvements have been made in Windev® in the launcher (the .exe), it is still very easy to visualize the assembly code of Windev® applications.
Especially, with such tools, all passwords, encryption strings and more generally all the characters strings of the application can be viewed very easily.
To avoid this it is essential to use debuggers blocking tool.

SoftProtect-AntiDebug is the anti-debug Windev component for your applications.

Included and activated in an application Windev®, **SoftProtect-AntiDebug** detects the presence of debuggers (or considered as such) and thus prevents the curious or malicious users, view sensitive information in your application or your files (internal passwords , encrypted data, protected files ...)

This detection can be activated on demand by calling a specific function (before a sensitive operation for example) or made automatic by using a timed loop.

To further improve the protection of your applications, it also includes a password generator that lets you create for your files and encryption, dynamic application and/or machine specific strings.
This detection can be activated on demand by calling a specific function (before a sensitive operation for example) or made automatic by using a timed loop.

SoftProtect-AntiDebug is designed to prevent the joint use of a protected application and current debugging tools available in Windows. Its architecture also restricts the available information in memory when using a debugger AFTER the use of the protected application.It is written to consume only limited CPU - the use of the component functions typically lasts less than 400 ms (1.7GHz processor) for maximum protection, or 35 ms for standard protection.

SoftProtect-AntiDebug required, by its complexity, many hours of development and focus. To date, intrusion tests, performed by the debug-piracy experts have not been able to crack this anti-debug protection.

Like any protection, **SoftProtect-AntiDebug** however, is not inviolable.
Therefore, to minimize the risk of piracy for your sensitive data, it is strongly recommended to use together :
- the highest level during the use of sensitive functions,
- the automatic protection loop, continually testing the potential vulnerability of the application,
- the internal functions for generating dynamic passwords and
- a deliberately complex code (for sensitive part)(avoid the simple "IF" statements, the "SWITCH"... Prefer instead the comparison intervals, the "IF _AND_ " and even here the

"GOTO" rarely used most of the time)

Activating the self-test loop provides an additional security by preventing (or limiting) the presence of debuggers between the programmed control phases.

Dynamic Passwords can be either specific to the application, the user or the server on which the application is running...

With this last choice, a database file and the associated application can be copied to another server, but are rendered unusable by the dynamic passwords.

The only risk of piracy remaining will be linked to the robustness of the database internal encryption tools.

For more information about SoftProtect-AntiDebug and other tools visit

http://www.SoftProtect.fr

SoftProtect-AntiDebug is guaranteed by the author free of any virus, "spyware" or advertising.

At the first use of each day, test mode only, **SoftProtect-AntiDebug** tests for the presence of an update available on the Internet.
If update available, a message is displayed to suggest visiting the dedicated website.
This feature can be disabled or modified by a specific setting.

SoftProtect-AntiDebug in its current version is available for free download for Windev® 11 (see note) or later.

It can be copied and distributed without restriction in any way whatsoever.
Its use in an executable, is subject to prior licensing, available conditions on the dedicated website.
To be fully used, this component must be registered by a validation code to request the author during the first few uses.

It has been tested in 32 and 64 bit **Windows XP SP3**, **Windows 7**, **Windows 8.1** and **Windows 10** with **Windev® 17** to **Windev® 22** (01F2220052j).
To improve functionality and correct any errors, it is desirable to inform the author of any malfunction or desired improvement.

To be informed about developments SoftProtect-AntiDebug, even apart from its use in a project, you can subscribe to the **NEWSLETTER** associated blog.

All users actively contributing to developments **SoftProtect-AntiDebug** will have in life, all future versions for free.

An updated FAQ, gathering the main issues and tips is available on the same site.

Note :
From the fact that Windev® supports backward compatibility of components, directly with the calling dll project **SoftProtect-AntiDebug** can be used with all versions of Windev® 17 and following, without needing other dll as those used by your application.

Windev® is a registered trademark PCSoft.

SoftProtect-AntiDebug - Component Install

Installation

To ensure maximum compatibility with different versions of Windev, **SoftProtect AntiDebug** is distributed in version Windev 11.
From the fact that Windev® supports backward compatibility of components, directly with the calling dll project, **SoftProtect-AntiDebug** can be used with all versions of Windev® 11 and following, without needing other dll as those used by your application .

SoftProtect-AntiDebug available since version 2.01 for 32 and 64 bit applications.
SoftProtect-AntiDebug is distributed with a complete sample test (reusable sources included).

No installation procedure is provided with the component.

Using the component in the project

It can be used from the files "SoftProtect-AntiDebug.wdk" and "SoftProtect-AntiDebug.wdi" present in the ".exe" subfolder of the example.
However to ensure maximum compatibility, the updates semi-automatic and use in multiple environments, it is desirable to copy these files in one of two installation Windev records.
For example in the sub folder "C:\WinDev 17\Composants\Composants SoftProtect".
The help files "SoftProtect-AntiDebug.chm" and "SoftProtect-AntiDebug.pdf" (also available in ".exe" subfolder of the example) can also be copied to the same folder for easy reference.

To use the component in a project, you need from the editor of your Windev project :

- import the component in the application
  Menu : **Workshop > Component > Import Component in the project > From File**
  Select "**SoftProtect-AntiDebug.wdi**" file (the folder where the copy was made), and validate
- copy in the project initialization (also available in the component description) :
  `pgProtect.SetFréquence()`
  or another function of pgProtect
- launch the project in test mode,
- configure protection using the settings wizard included in the component (it is not necessary at this stage to register the component)
- generate code protection and copy it into a text editor (Notepad for example),
- copy protection code before each call critical functions to secure your application.

To update a project using an older version of the component (in Windev 11) it is advisable to :

- remove the old SoftProtect-AntiDebug component from the project
- import the component in the application
  Menu : **Workshop > Component > Import Component in the project > From File**
  Select "**SoftProtect-AntiDebug.wdi**" file (the folder where the copy was made), and validate
- lancer l'exécution du projet en mode test (pour enregistrer physiquement le composant),

SoftProtect-AntiDebug - The Inputs / Outputs Component

The **SoftProtect-AntiDebug** component provides a single set of procedures :

Procedure Collection '**pgProtect**' with 4 exported global procedures

- **iSecure**
- **SetModeProtect**
- **sGetPassword**
- **SetFrequence**

iSecure procedure

`[ <Résultat> = ] iSecure ( <iMode> [, <bChrono>])`

Returns an integer specifying a level of risk with respect to potential debugging

iMode [in/out] : Protection mode inputting, return code output
bChrono [in] : True to activate the timer (for testing) [faux]

Result Integer

100 : The current application running under debugger - maximum risk
95 : iSecure error (probably attempted diversion of flow)
90 : Probable risk of application running debugger
80 : Excessive execution time, potential flow interception
10 : A parent process of the current application is in the list of debuggers
6 : A Windows windows corresponds to a known debugger
5 : One of the current process of the session is in the list of debuggers
1 : The application was launched by an unknown process other than Windows
0 : A priori no identified risk

This procedure is the main, it ensures, in his appeal, the application security to protect.
It is highly recommended to maximize safety to use, where possible the mode of maximum protection.
The stopwatch does not normally have to be used other than to check the response time in the development. When it is activated, it displays a trace window with the execution time of the function.

NEW 1.14 To improve protection and reduce the risk of diversion of functions present in iSecure() it is essential to jointly test the return value, the parameter value "iMode" and the global variable "gi" of the component.
If these three values are inconsistent, it is likely that a hacking attempt of the application is in progress.
You will in that case consider the risk as a maximum. See the instructions for use for more details.

WARNING

NEW 1.14 Since Version 1.14a the "iMode" parameter is no longer optional, it is used and modified at the call iSecure.
See the documentation included in the component for details

SetModeProtect procedure

[SetModeProtect ([<iMode>])

NEW 1.20 Used in test mode, this function is now launching the opening of the setting screen.

NEW 1.20 iMode [in] : In run mode defines the type of protection to apply, In test mode opens the setup screen [0]

0 - Standard mode, processing time of the order of 40 ms on a 1.7 GHz processor

8 - Full mode, longer processing time ~ 300 to 400 ms on a 1.7 GHz processor

sGetPassword procedure

[<Résultat> =] sGetPassword ([<iNbChar> [, <iMode> [, <sGen> [, <bChrono>]]]])

Returns a dynamic password

iNbChar [in] : Number of password characters (from 7-32) [13]

iMode [in] : Password generation mode (combination of values below) [0x0F07]

0 - totally random
1 - specific to the computer that runs the application
2 - specific username
4 - DD specific number on which the application is installed
8 - Windows-specific DD number
0x40 - spécifique à l'application courante (mot de passe statique si seul)
0x80 - complete with test prior iSecure
0x0100 - in capital letters
0x0200 - in tiny characters
0x0400 - in numbers
0x0800 - in the special characters
0x1000 - avoid ambiguous characters (1 and l, 0 and O ...)

sGen [in] : Generating string password (NULL or empty string for default) [Null]

bChrono [in] : True to enable a stopwatch (for testing) [Faux]

SetFrequence procedure

[SetFrequence([<idt>])

NEW 3.00 Changes the interval of the automatic test loop.

idt [in] : Scan rate in seconde [0]

SoftProtect-AntiDebug - Usage Tips

By design of Windev DLL, all functions of the components can easily be intercepted by a debugging session or equivalent. Although improvements have been made since Windev 12, it is always easy to fool the pitcher (.exe) to hide a possible debugger.
During such a session, function settings are visible in the clear in the debugger.

Therefore it is essential to secure your data and your application to call the procedure '`iSecure()`' before each critical function, for example (not exhaustive):

`HPass()` and all HF functions using passwords,

`Crypt()` and all encryption functions / decrypt strings and files,

`gpwLogin()`, `gpwOpenConnection()`

`FTPConnect()`, `NetConnect()`, `NNetOpenRemoteAccess()`

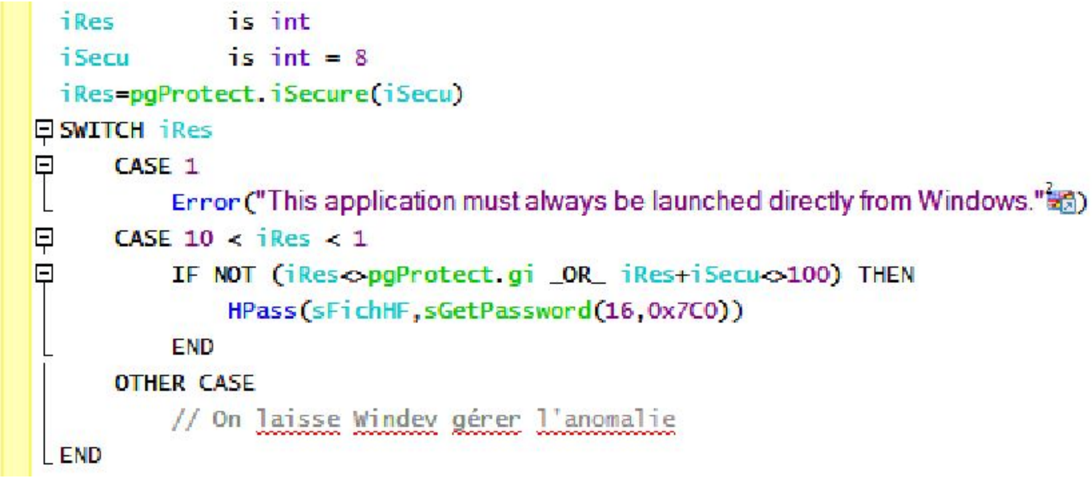
`Hasp()`

`zipPassword...`

You can also (recommended), to complicate the task of a possible debugging, enable automatic test loop.

To improve security, it is strongly advised to use the highest security mode.
The standard mode, faster processing, can be used in repetitive treatments eg dice the second passage in a loop.

A typical example of combined use of '`iSecure()`' and '`sGetPassword()`'



```
iRes        is int
iSecu       is int = 8
iRes=pgProtect.iSecure(iSecu)
SWITCH iRes
    CASE 1
        Error("This application must always be launched directly from Windows.")
    CASE 10 < iRes < 1
        IF NOT (iRes<>pgProtect.gi _OR_ iRes+iSecu<>100) THEN
            HPass(sFichHF,sGetPassword(16,0x7C0))
        END
    OTHER CASE
        // On laisse Windev gérer l'anomalie
END
```

Other such codes are also provided in the associated example and in the window setting.

SoftProtect-AntiDebug also offers dynamic password generation.

These passwords can be generated from the characteristics of the server and / or user.
Thus an encrypted file with such password is no longer available on another computer or another user.
It is highly recommended to use these passwords whenever possible.
However, for the application can be transferred to another server, or to another user, will require provide a procedure for exporting / importing data to change the password.

This technique can also be used when data files are to be delivered with the application.
However, it will either not to use particular parameters specific to the installation to generate the password (static password) or carry on first launch or when installing a file password change delivered (a "WD ChangeMotDePasse" component is available with LST64 or HChangePassword (function) available natively since Windev 18).

WARNING: If you request a dynamic password, built on a disk (Windows or of the application), a hardware change (repair, disk change ...) could make it impossible to access files protected by this password. It will be the same for passwords constructed on the names of the user or the computer in case of change. It is recommended, in these cases, to have a fallback procedure, for recovering the protected files, without however affect the security of informations.

For example, proposing a complementary application for the reconstruction of the original password from information known to a user only, or previously encrypted and stored on a secure medium.

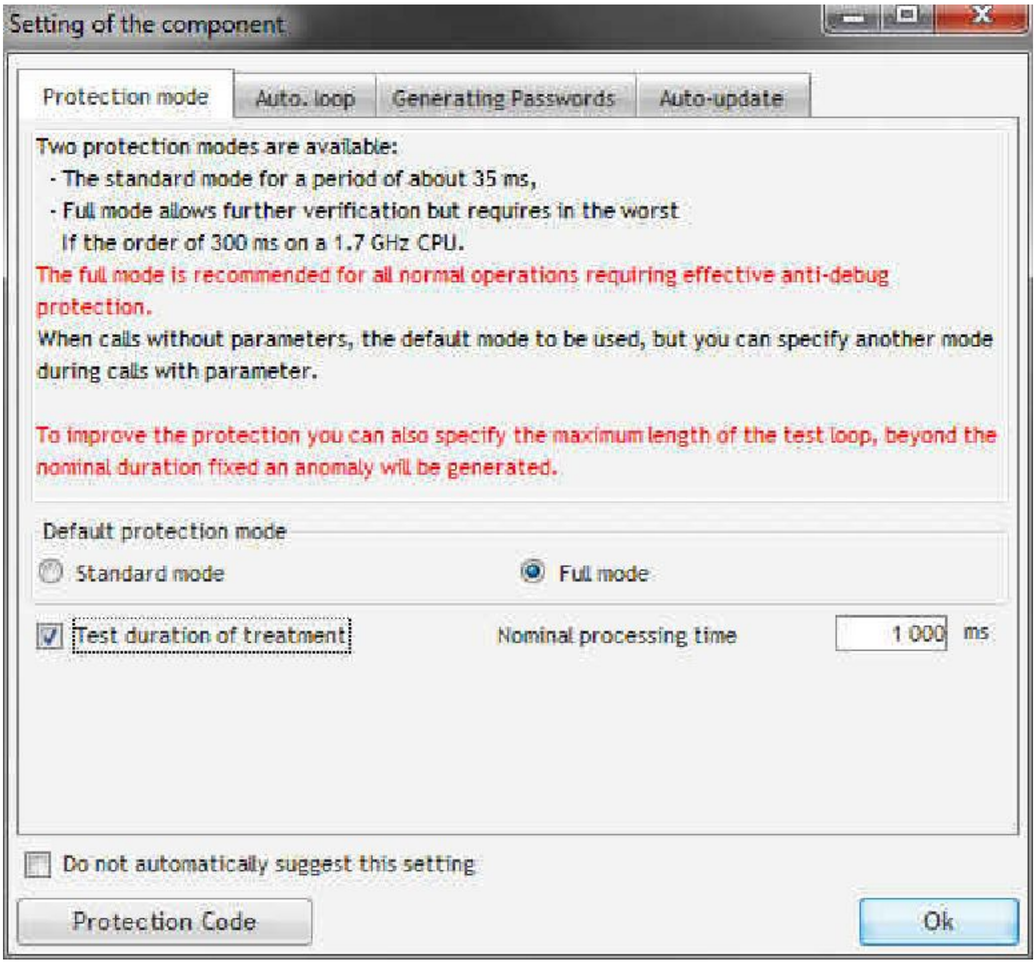
SoftProtect-AntiDebug - Setting

The component setting is, since version 1.20, achieved with a specific screen, accessible only in test mode.
Since version 2.00 this setting screen is brought automatically when using a function of the component from the test mode.
It may also be invoked by code using the tool "SetModeProtect()".
To show this setting during subsequent launches check "Do not automatically suggest this setting"

All these settings can of course not be defined executable mode, only in test mode.

4 tabs are available for this setting

The first is the main protection mode and allows to define the major parameters



Two protection modes are available:
- The standard mode for a period of about 35 ms,
- Full mode allows further verification but requires in the worst
  if the order of 300 ms on a 1.7 GHz CPU.

The full mode is recommended for all normal operations requiring effective anti-debug protection.
When calls without parameters, the default mode to be used, but you can specify another mode during calls with parameter.

To improve the protection you can also specify the maximum length of the test loop, beyond the nominal duration fixed an anomaly will be generated.

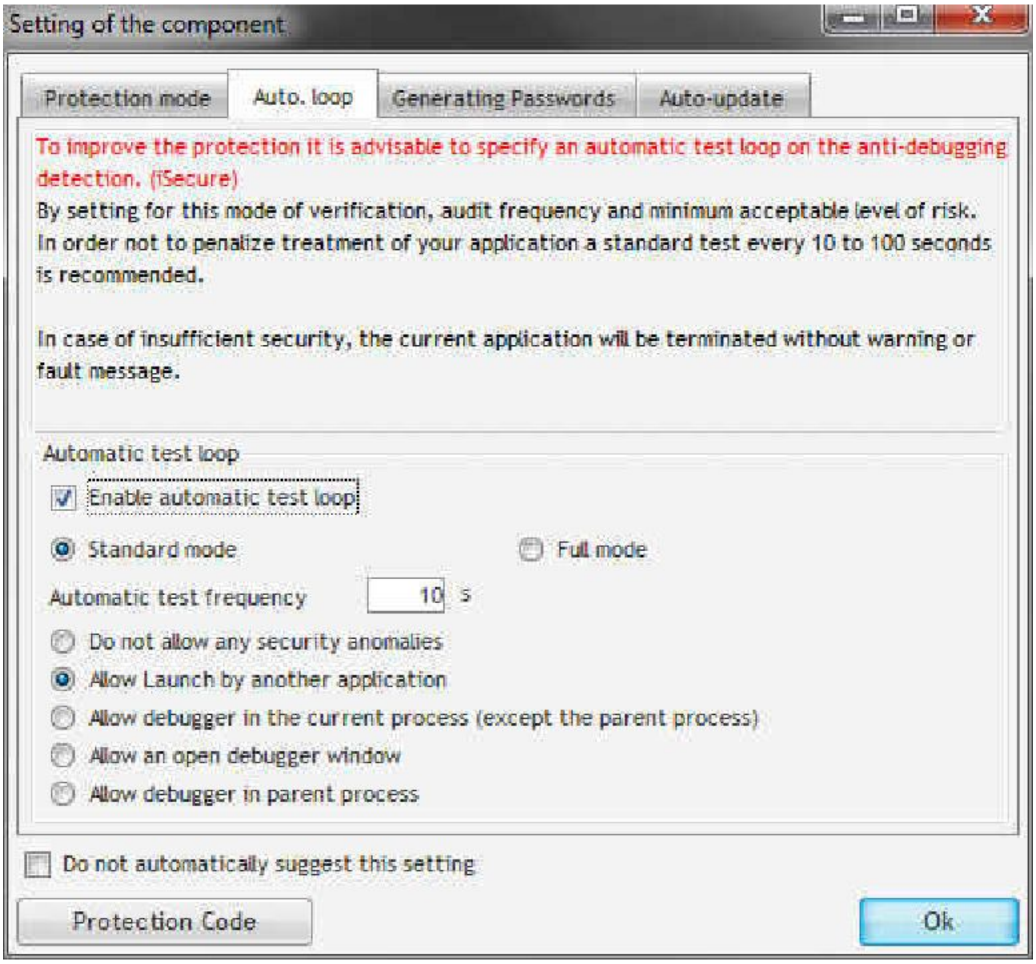
Default protection mode
○ Standard mode ● Full mode

☑ Test duration of treatment Nominal processing time 1 000 ms

☐ Do not automatically suggest this setting

The full mode is recommended if a treatment time of approximately 300 ms is conceivable
This mode can be, upon request, supplemented by a test of the maximum acceptable duration of treatment.

The second defines the parameters of the automatic test loop.



To improve the protection it is advisable to specify an automatic test loop on the anti-debugging detection. (iSecure)
By setting for this mode of verification, audit frequency and minimum acceptable level of risk.
In order not to penalize treatment of your application a standard test every 10 to 100 seconds is recommended.

In case of insufficient security, the current application will be terminated without warning or fault message.

Automatic test loop

☑ Enable automatic test loop

● Standard mode ○ Full mode

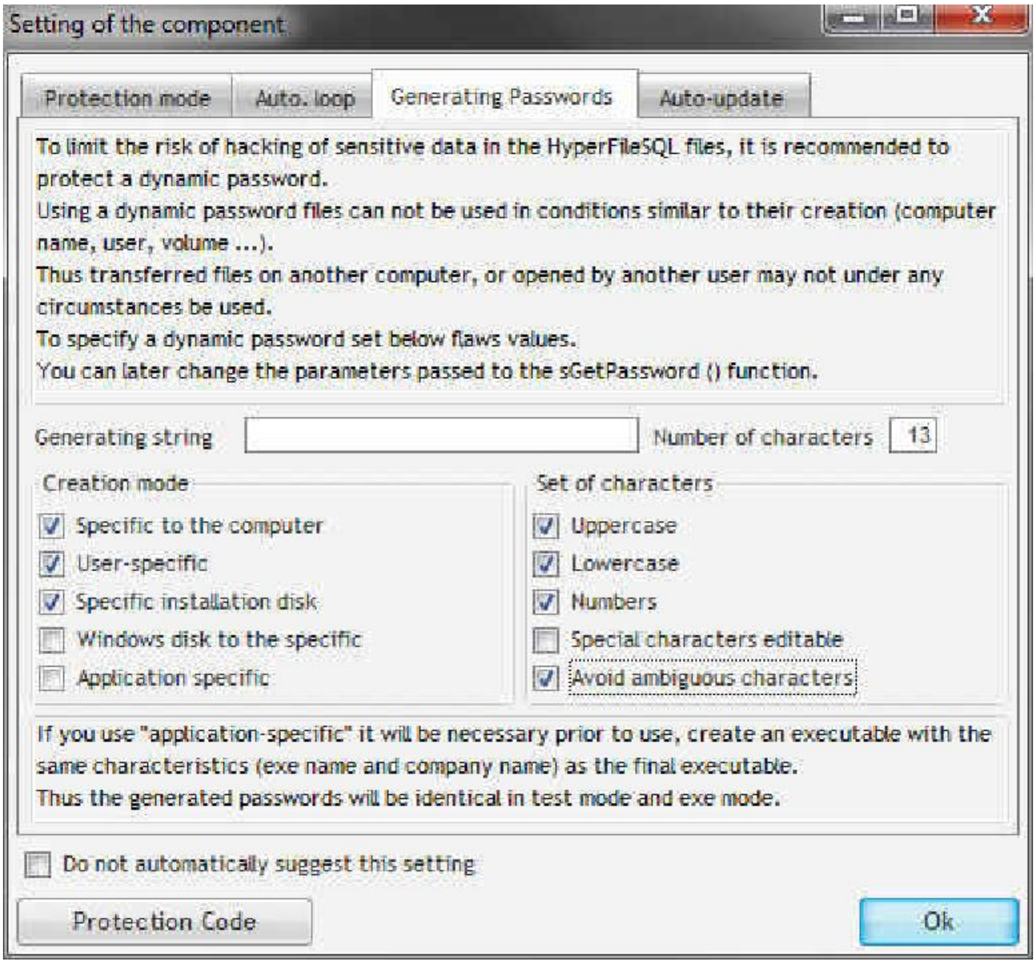
Automatic test frequency 10 s

○ Do not allow any security anomalies
● Allow Launch by another application
○ Allow debugger in the current process (except the parent process)
○ Allow an open debugger window
○ Allow debugger in parent process

☐ Do not automatically suggest this setting

Generally the standard mode with a frequency of the order of 10 to 20 seconds is more than enough.
In addition it is also possible to define the acceptable level of risk for this loop.
The maximum security level is recommended.
If during this test found the risk is higher than permitted, the application is stopped without notice.

The third tab is for setting the automatic password returned by the function "sGetPassword()".



To limit the risk of hacking of sensitive data in the HyperFileSQL files, it is recommended to protect a dynamic password.
Using a dynamic password files can not be used in conditions similar to their creation (computer name, user, volume ...).
Thus transferred files on another computer, or opened by another user may not under any circumstances be used.
To specify a dynamic password set below flaws values.
You can later change the parameters passed to the sGetPassword () function.

Generating string Number of characters 13

Creation mode
☑ Specific to the computer
☑ User-specific
☑ Specific installation disk
☐ Windows disk to the specific
☐ Application specific

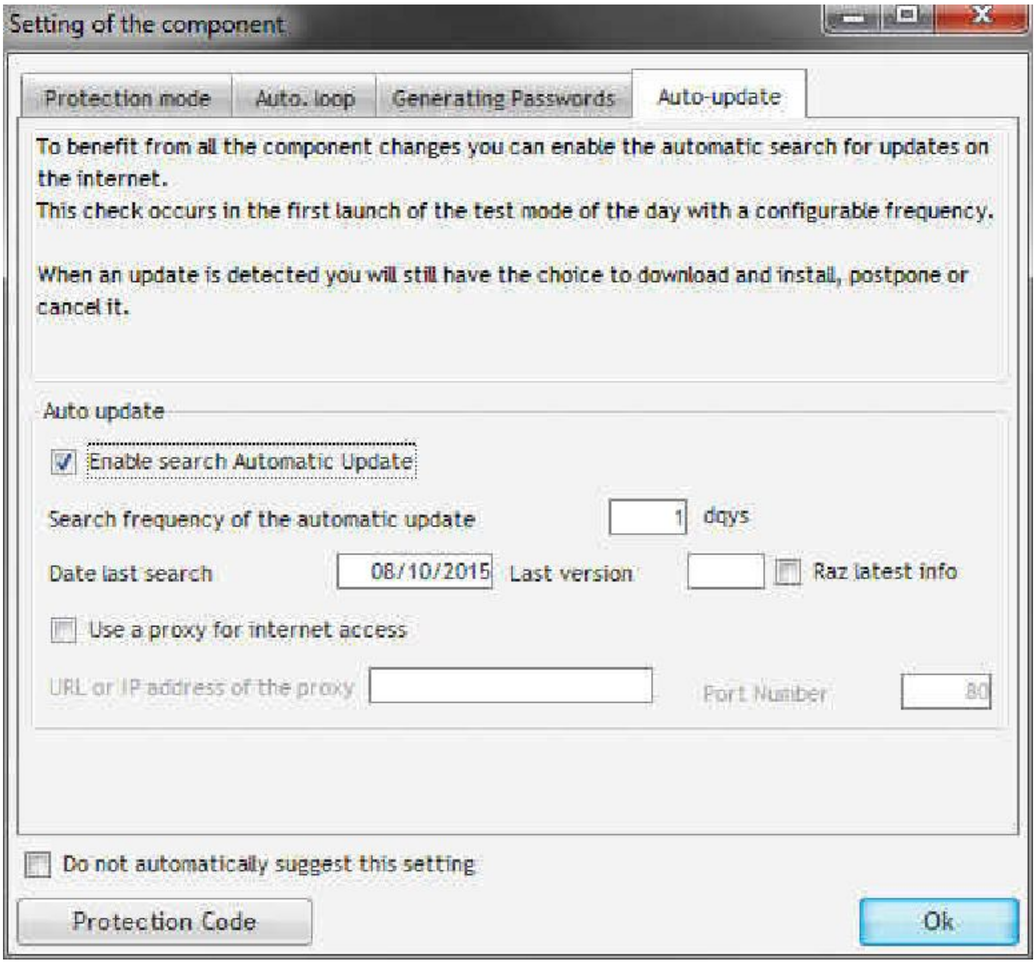
Set of characters
☑ Uppercase
☑ Lowercase
☑ Numbers
☐ Special characters editable
☑ Avoid ambiguous characters

If you use "application-specific" it will be necessary prior to use, create an executable with the same characteristics (exe name and company name) as the final executable.
Thus the generated passwords will be identical in test mode and exe mode.

☐ Do not automatically suggest this setting

If no check mark in the "creation mode" passwords are completely random and therefore different for each call.
If the passwords generated by **SoftProtect-AntiDebug** should be used to protect the HyperFileSQL files, it is desirable to use more than 16 characters.

The fourth tab is for setting the automatic check for updates.



To benefit from all the component changes you can enable the automatic search for updates on the internet.
This check occurs in the first launch of the test mode of the day with a configurable frequency.

When an update is detected you will still have the choice to download and install, postpone or cancel it.

Auto update

☑ Enable search Automatic Update

Search frequency of the automatic update: 1 dqys

Date last search: 08/10/2015 Last version: ☐ Raz latest info

☐ Use a proxy for internet access

URL or IP address of the proxy: Port Number: 80

☐ Do not automatically suggest this setting

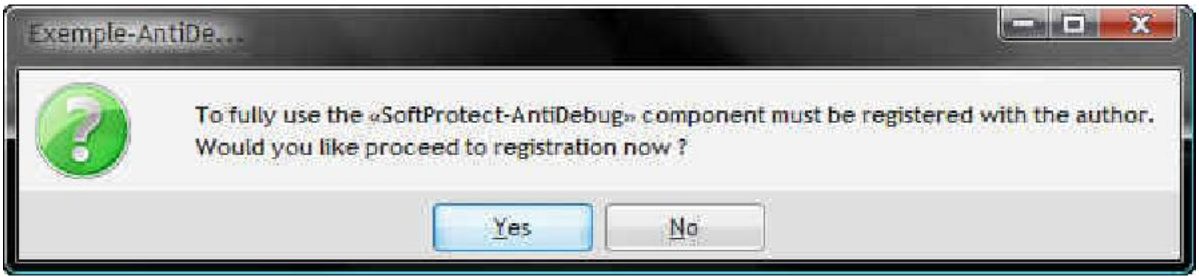
You can especially enable or disable this research and set the frequency.
Finally you can also set the default proxy settings are appropriate (if the connection is active these parameters are checked).
When an update is available, a window will inform you and propose you to open the dedicated website to perform the manual download.

This research is not quite sure ever made in executable mode.

During the development phase of the application to protect all these parameters are saved in the persistent values of the application in the subkey "**SoftProtect-AntiDebug**".

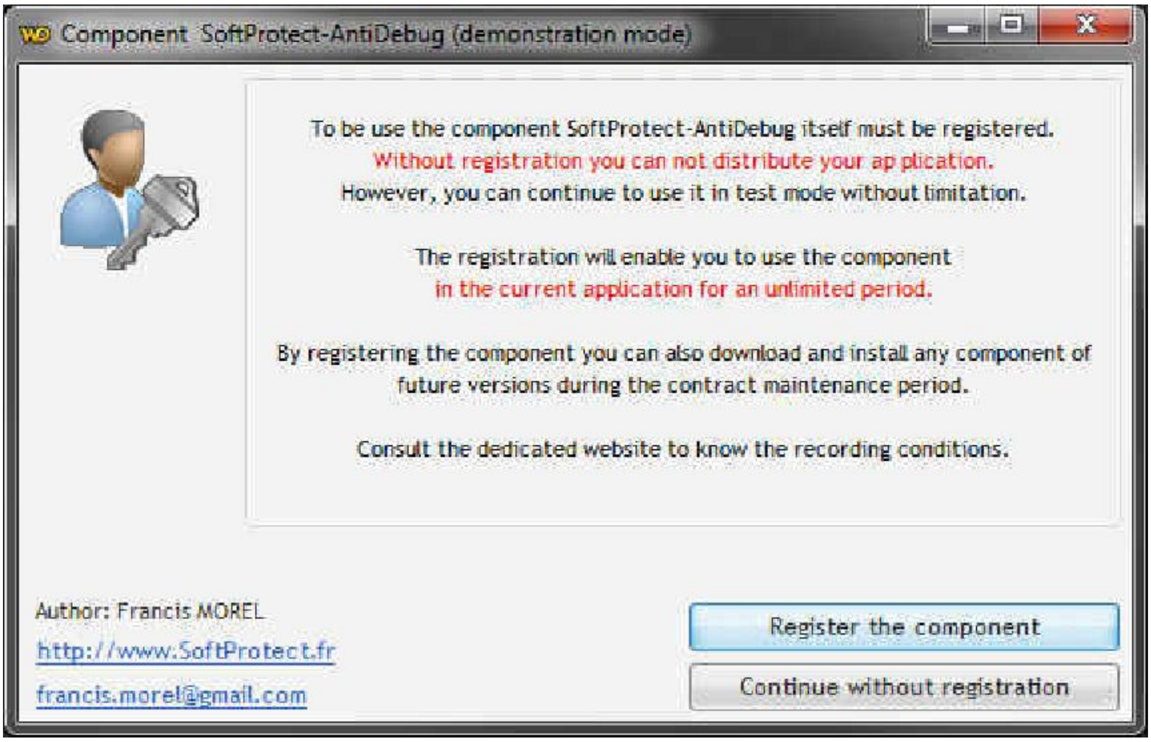
SoftProtect-AntiDebug - Component Registration

Since version 1.15 the SoftProtect-AntiDebug component needs to be fully used to enter a validation code.
At the first launches in test mode the following message is displayed :



To fully use the «SoftProtect-AntiDebug» component must be registered with the author.
Would you like proceed to registration now ?

By clicking on the button <Yes> you can proceed with the registration component.



To be use the component SoftProtect-AntiDebug itself must be registered.
Without registration you can not distribute your ap plication.
However, you can continue to use it in test mode without limitation.

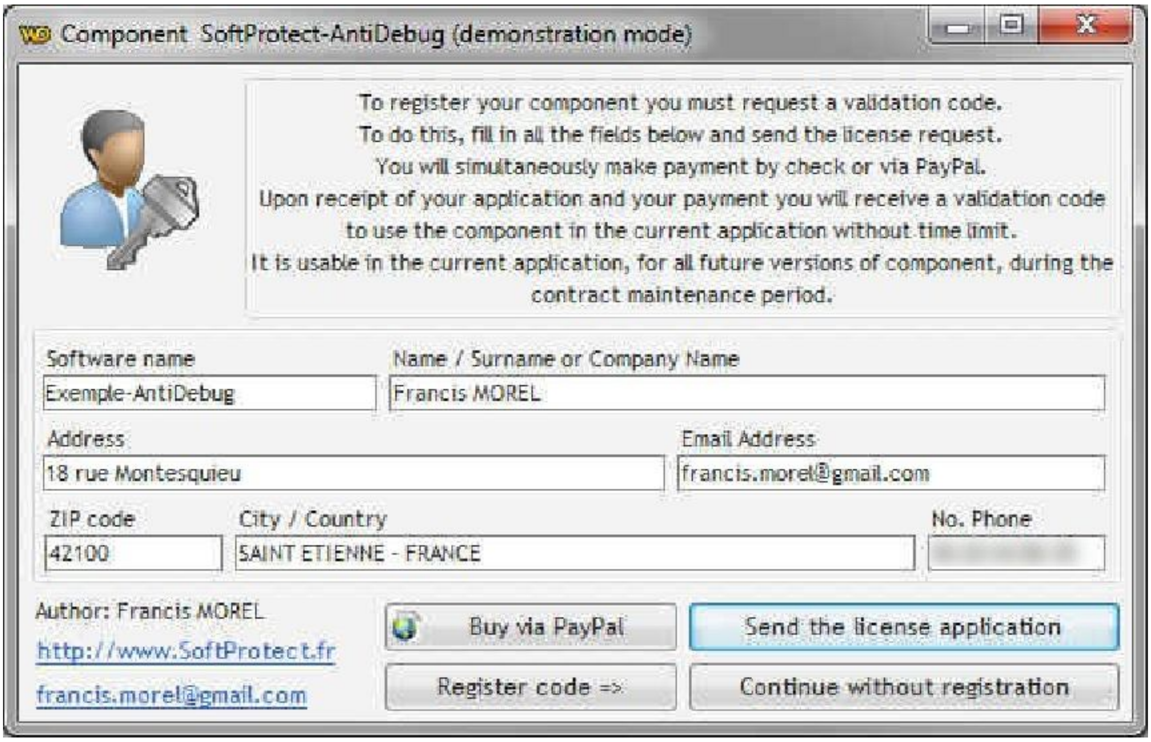
The registration will enable you to use the component
in the current application for an unlimited period.

By registering the component you can also download and install any component of
future versions during the contract maintenance period.

Consult the dedicated website to know the recording conditions.

Author: Francis MOREL
http://www.SoftProtect.fr
francis.morel@gmail.com

The first window tells you the terms of use.
You can either continue without registration or to register by clicking on one of the two available buttons.
The use of component without registration is limited to test mode Windev.
Especially if you choose to register you will get the following window :


Component SoftProtect-AntiDebug (demonstration mode)
Buy via PayPal
Send the license application
Register code =>
Continue without registration

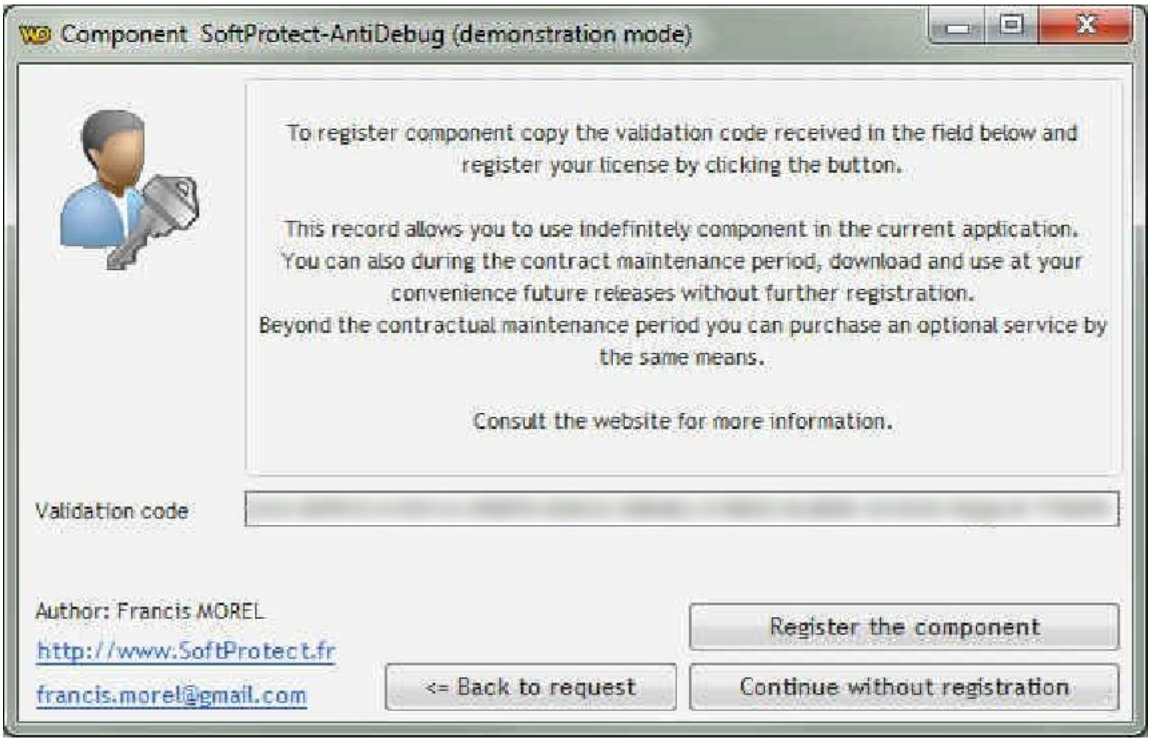
To register your component you must request a validation code.
To do this, fill in all the fields below and send the license request.
You will simultaneously make payment by check or via PayPal.
Upon receipt of your application and your payment you will receive a validation code to use the component in the current application without time limit.
It is usable in the current application, for all future versions of component, during the contract maintenance period.

Software name: Exemple-AntiDebug
Name / Surname or Company Name: Francis MOREL
Address: 18 rue Montesquieu
Email Address: francis.morel@gmail.com
ZIP code: 42100
City / Country: SAINT ETIENNE - FRANCE
No. Phone: [redacted]

Author: Francis MOREL
http://www.SoftProtect.fr
francis.morel@gmail.com

Complete the fields and submit the license request by clicking the appropriate button, which will open the default email application with the message to send. If unable or anomaly the validation request message is copied to the clipboard to be sent by the means of your choice.
Simultaneously you will pay the component by clicking the button or by sending a check payable to the author, see the dedicated website
http://www.SoftProtect.fr/SoftProtect-AntiDebug
for more information.

VYou will receive, within a period of about 48 hours (excluding holiday period) a validation code, to be placed in the correct area of the window: :


Component SoftProtect-AntiDebug (demonstration mode)
Register the component
<= Back to request
Continue without registration

To register component copy the validation code received in the field below and register your license by clicking the button.

This record allows you to use indefinitely component in the current application.
You can also during the contract maintenance period, download and use at your convenience future releases without further registration.
Beyond the contractual maintenance period you can purchase an optional service by the same means.

Consult the website for more information.

Validation code: [redacted]

Author: Francis MOREL
http://www.SoftProtect.fr
francis.morel@gmail.com

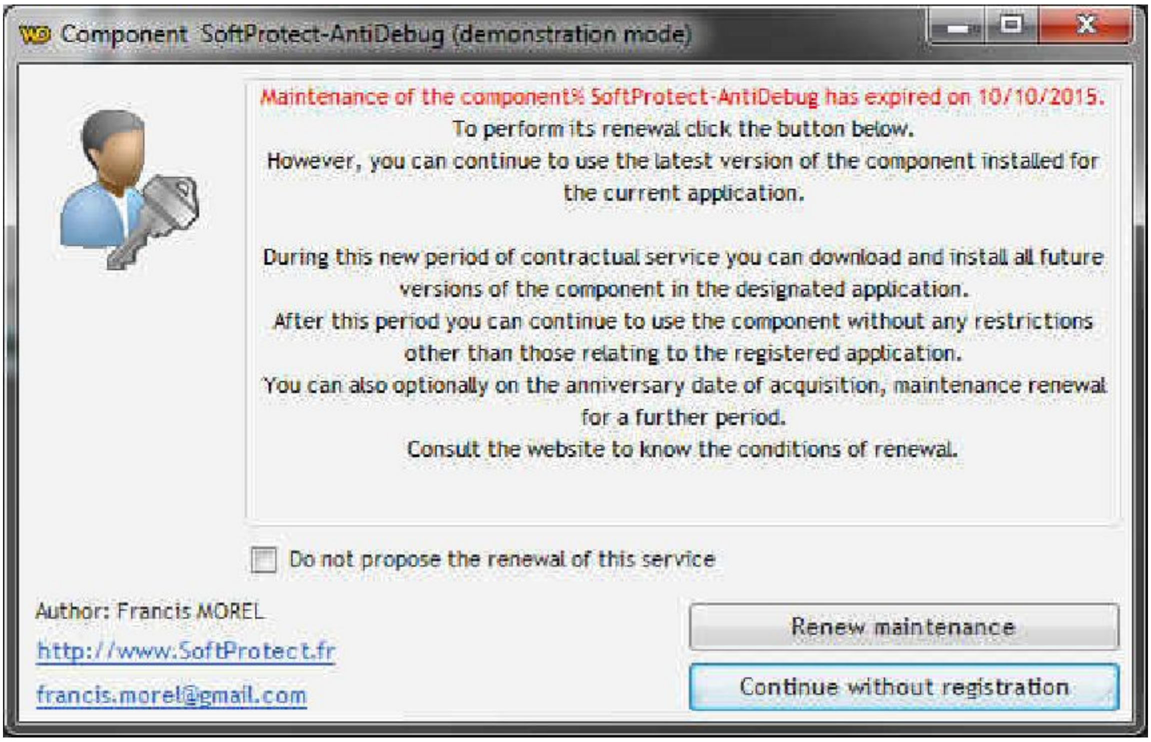
then confirm by clicking <Register the component>.

Pending the receipt of validation code, you can continue to use the component and the protected application in test mode (Go in Windev) without anomalies.
However, the Exe operation mode will be rendered impossible **Only registered components will be operational in the executable**.

The validation code **is valid indefinitely for one designated application** for the current version of **SoftProtect-AntiDebug** component.

Il permet également le téléchargement et l'utilisation, dans les mêmes conditions, de toutes les versions futures du composant **SoftProtect-AntiDebug** during the maintenance period.
However you will, in certain circumstances, perform validation (free during the maintenance period) for each new component update.
Carried out in the same development environment (same application, same PC, even setting) this validation will be automatic. Only a required launch in test mode Windev will be necessary.

Beyond this period, by relaunching the project in test mode, you can subscribe again to the maintenance for a further period with the window :



Component SoftProtect-AntiDebug (demonstration mode)

Maintenance of the component% SoftProtect-AntiDebug has expired on 10/10/2015.
To perform its renewal click the button below.
However, you can continue to use the latest version of the component installed for the current application.

During this new period of contractual service you can download and install all future versions of the component in the designated application.
After this period you can continue to use the component without any restrictions other than those relating to the registered application.
You can also optionally on the anniversary date of acquisition, maintenance renewal for a further period.
Consult the website to know the conditions of renewal.

☐ Do not propose the renewal of this service

Author: Francis MOREL
http://www.SoftProtect.fr
francis.morel@gmail.com

Then proceed similarly as in the first record.

SoftProtect-AntiDebug - History

SoftProtect-AntiDebug 4.00A - 23.03.2017

Update to Windev 17
Improved recognition of new debuggers

SoftProtect-AntiDebug 3.10B - 11.10.2015

HMI anomalies update for Windows 10
Translation in English of component and help (largely thanks to Fabrice HARARI)

SoftProtect-AntiDebug 3.00A - 20.04.2012

Changed the component name
Added semi-automatic validation of component via internet

SoftProtect-Basic 2.01A - 02.04.2011

Compiled and tested for 64-bit Windows

SoftProtect-Basic 2.00A - 21.01.2011

Changed the component name
Completed overhaul of the code to improve security of the component and the protected application
Improved settings
Improved the component registration procedure

DataProtect 1.20A - 16.11.2009

Added complete internal parametrization management
Added automatic anti-debug loop

DataProtect 1.14A - 30.01.2009

Improved iSecure() method functionning by the combined use of several return variables

DataProtect 1.13A - 27.11.2008

Corrected sGetPassword anomaly in special cases
This update requires a change of password for files using passwords associated with the application

DataProtect 1.12A - 26.11.2008

Added test of update availability (only in development mode)

DataProtect 1.11A - 15.11.2008

Improved protection

DataProtect 1.10A - 10.11.2008

First version of the component for Windev 11 and Windev 12

SoftProtect-AntiDebug - FAQ

Is it possible to use a component from a different machine than the one that was used to regis

The registration of a component is independent of the machine where the validation has been done. already registered for a project, the copy on another machine in the same project (executable) is po restriction or further registration validation. If necessary, the saved settings (key to the registry, ini to the project) for the component (SoftProtect-Basic section of the persistent variables of Windev) one machine to another to facilitate the transfer.

I'm working on two machines on the same project using the component, for that, I copied the the second machine. Now when loading the project Windev generates an error because it can component folder. How do I correct this anomaly?

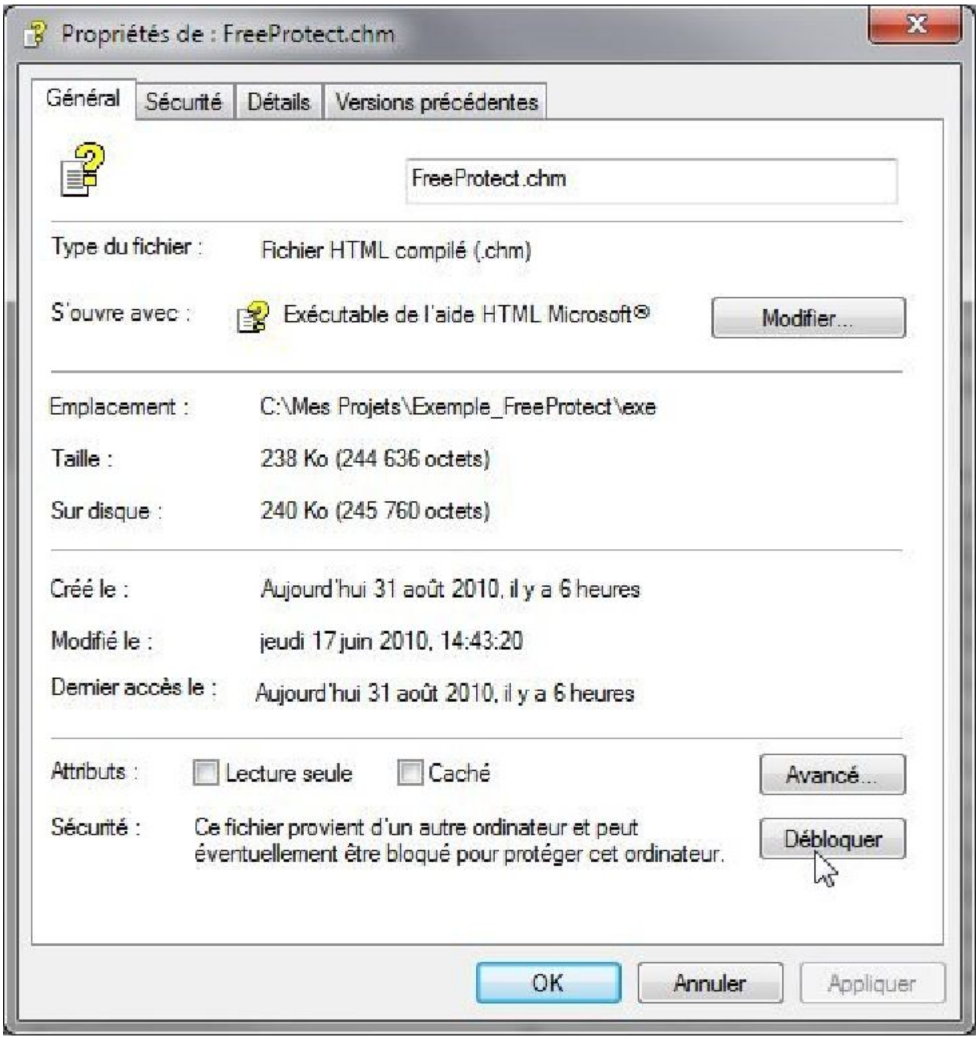
While loading the project WindDev is testing a possible update of the component. Therefore it is nec component description files (SoftProtect Basic.wdk-and-SoftProtect Basic.wdi) in the same folder on example:
"C:\WinDev 17\Components\Components SoftProtect"
and add the component in the project from this folder.

When updating the component, do I need to register it again?

When registering the component the license information is stored in the persistent values of the appl the new version validation will be automatic during the support period.
However, it is essential to relaunch the project in test mode for this automatic validation to occur.
However if for any reason (deletion, hardware change at the same time...) these persistent values w automatic registration you may ask for a new validation code (for the same project) through the inte procedure. A new code will then transmitted to you by return.
In both cases, after validation (or self-validation), it is better to restart Windows before creating th

I am using Windows 7 and I can not see the content of the associated chm help file?

Under Windows 7, chm help files issued from another computer are usually blocked to protect the sy potential vulnerability issues. To fully view these files:
- The file to view should be copied a local drive (not mandatory but preferable)
- In the properties of the local chm file (right click > Properties), "General" tab click <Unblock>.



FreeProtect.chm

Type du fichier : Fichier HTML compilé (.chm)

S'ouvre avec : Exécutable de l'aide HTML Microsoft®

Emplacement : C:\Mes Projets\Exemple_FreeProtect\exe

Taille : 238 Ko (244 636 octets)

Sur disque : 240 Ko (245 760 octets)

Créé le : Aujourd'hui 31 août 2010, il y a 6 heures

Modifié le : jeudi 17 juin 2010, 14:43:20

Dernier accès le : Aujourd'hui 31 août 2010, il y a 6 heures

Attributs : ☐ Lecture seule ☐ Caché

Sécurité : Ce fichier provient d'un autre ordinateur et peut éventuellement être bloqué pour protéger cet ordinateur.

The corresponding chm file is then viewable correctly.