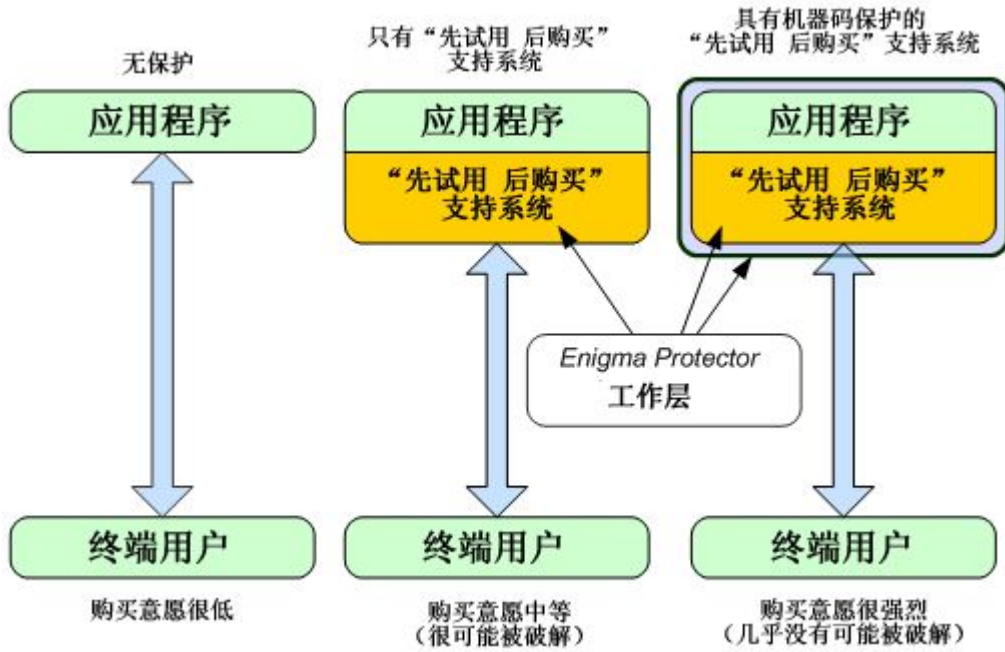


Enigma Protector

Enigma Protector

- | Win32 (* .exe);
- | Windows (* .scr);
- | (* .dll);
- | 32 ActiveX (* .ocx);
- | .NET (* .exe).

" " " "



Enigma

" " " "

Enigma Protector

512 RSA

" "

Enigma Protector

|
|

Enigma Protector

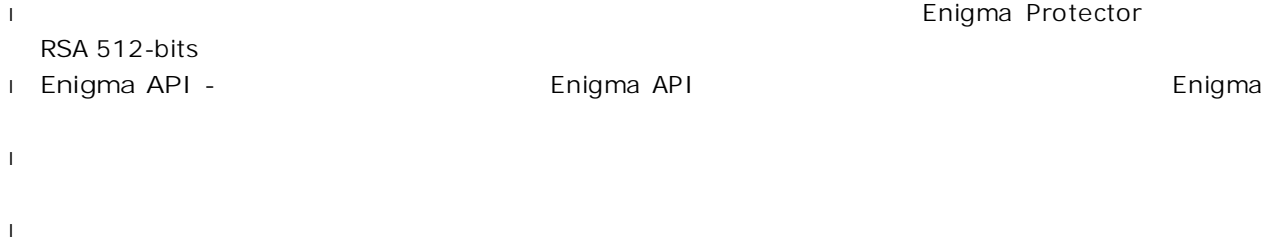
| /
|

Email

Enigma

Enigma Protector

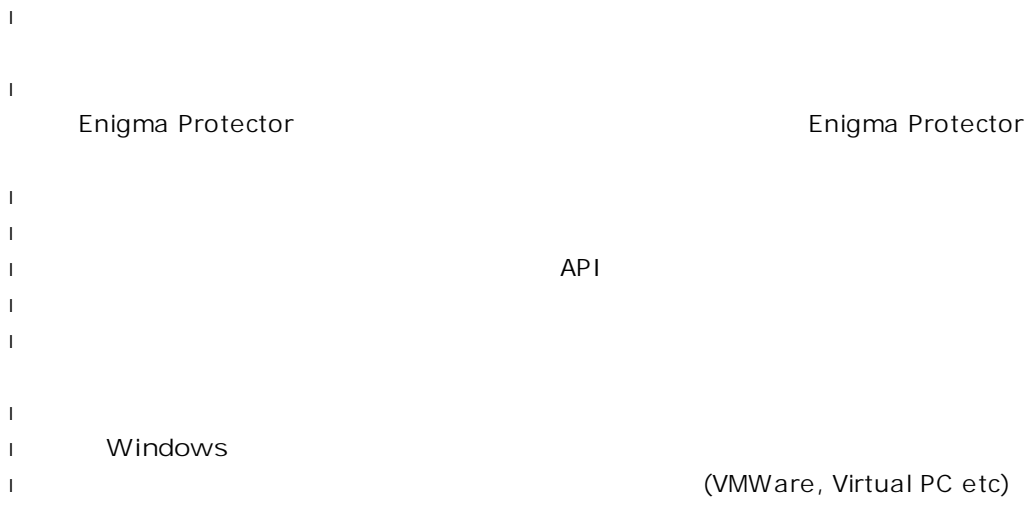
Enigma Protector



Enigma Protector



Enigma Protector



Enigma Protector



EnigmaProtector

www.enigmaprotector.com

Enigma Protector	149 USD	299 USD

9

Shareit (www.shareit.com)

Western Union (www.westernunion.com)

WebMoney (www.wmtransfer.com)

*
*

support@enigmaprotector.com

EnigmaProtector

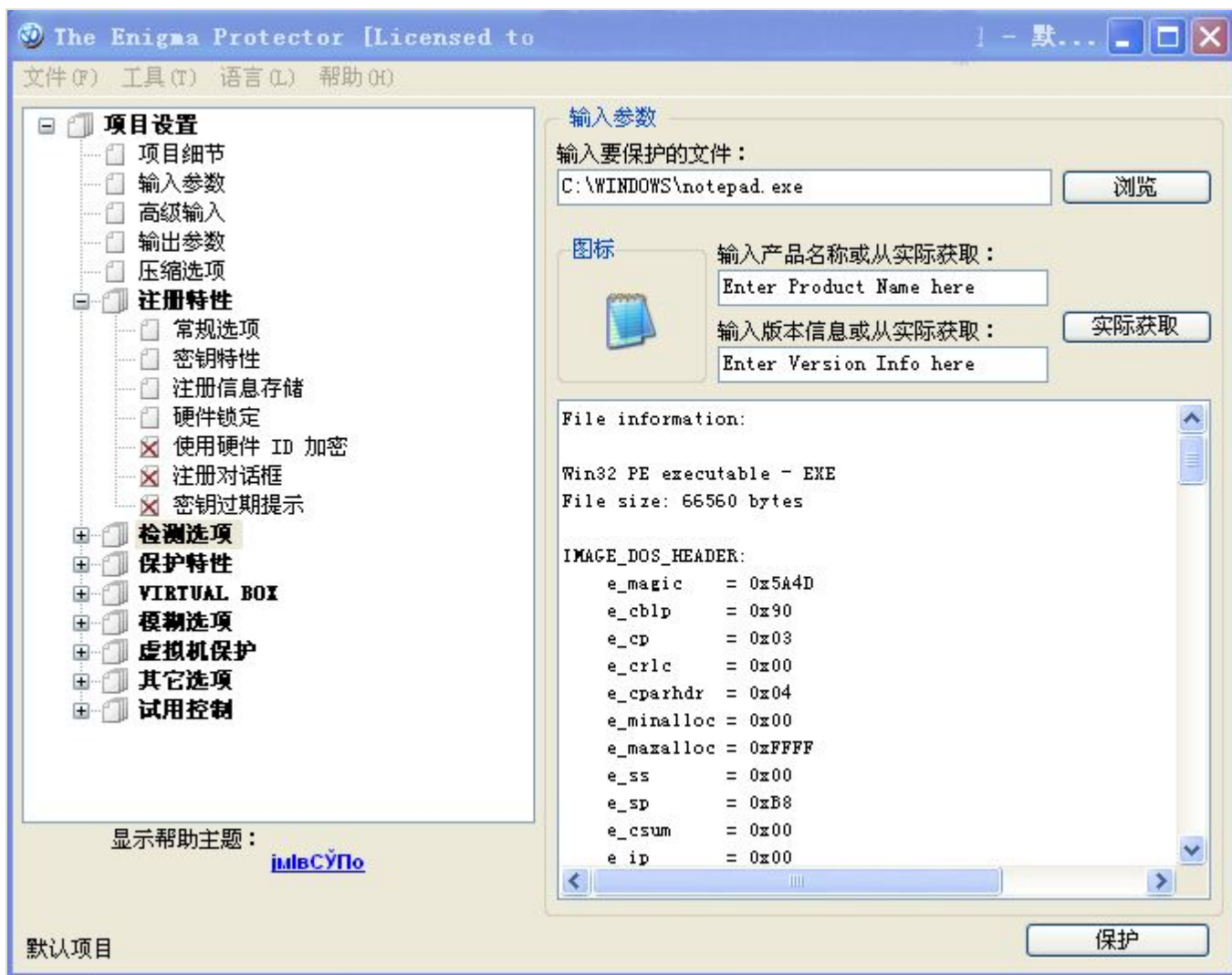
wang@enigmaprotector.com

EnigmaProtector

www.enigmaprotector.com

Enigma Protector

" "

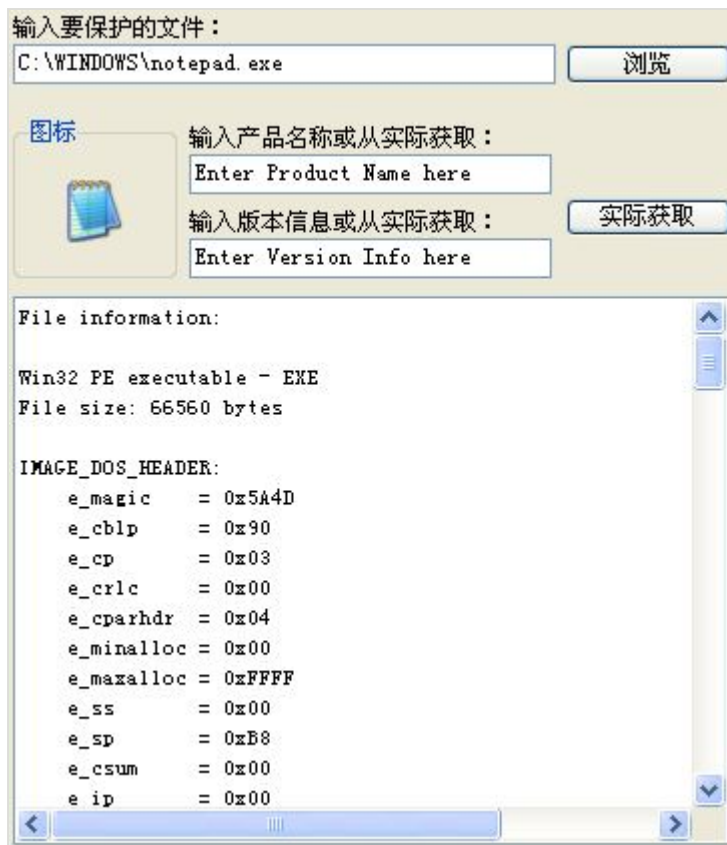


项目名称：

Default project

项目描述：

Description of the default project



-
-
-
-
-

" "

选择其他需要保护的文件

#	 输入文件	输出文件

在主程序加密之后保护其他文件

添加文件



与输入文件相同

输入保护后的文件名称：

不需要创建备份文件

- " "

-

.bak - EnigmaProtector

|
|
|
|
| ID
|
|
|



```

UNICODE      -      UNICODE
| Keys Generator      unicode
| Key Generator of License Manager      unicode
|                                     unicode      Enigma API
| CGI Key Generators      unicode
|       keygen.dll      unicode

|
| EnigmaProtector
| ansi

-

-                                     EnigmaProtector      KEY
                                         SDK

"       "       "       "

/       /

~ RSA 512/768/1024/2048/3072/4096

:       /       /

:

```

- 只允许有硬件锁定的密钥
- 只允许有过期日期的密钥
- 只允许在日期之后需注册的密钥
- 只允许在日期之前需注册的密钥
- 只允许运行次数限制的密钥
- 只允许运行日期限制的密钥
- 只允许运行时间限制的密钥
- 只允许有总时间限制的密钥
- 只允许有国家锁定的密钥

"

"

-

-

-

-

-

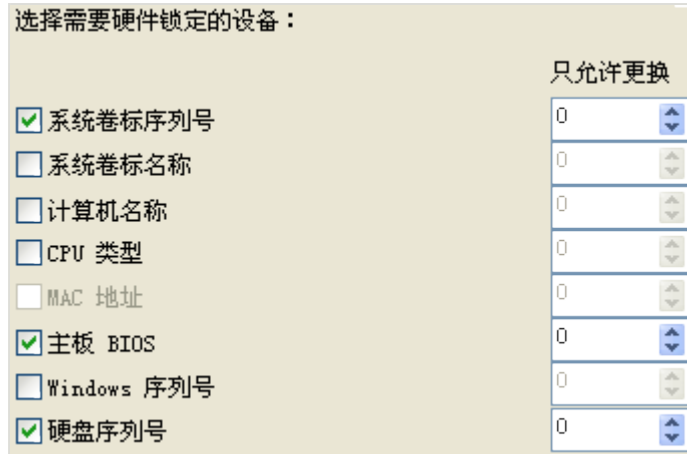
-

-

-

-

ID



Enigma Protector
EP_RegHardwareID

Enigma API

```

|           -
|           -
|           -
| CPU      - CPU
|           - BIOS
| Windows  - Windows
  
```

	(*1)		(*4)	
	(*1)	(*2)		
		(*3)		
CPU (*5)				
BIOS(*5)				
Windows				

(*1)

(*2)

(*3)

(*4)

(*5) Windows NT (Windows 2000, Windows XP, Windows Vista)

-

/

选择注册信息存储位置：

使用注册表存储
 使用文件存储
 两者都用

注册表根目录：
HKEY_CURRENT_USER

注册表相对路径（例如：\SOFTWARE\YourCompany）：
\SOFTWARE\EnigmaDevelopers\

文件存储根目录：
%My Documents FOLDER%

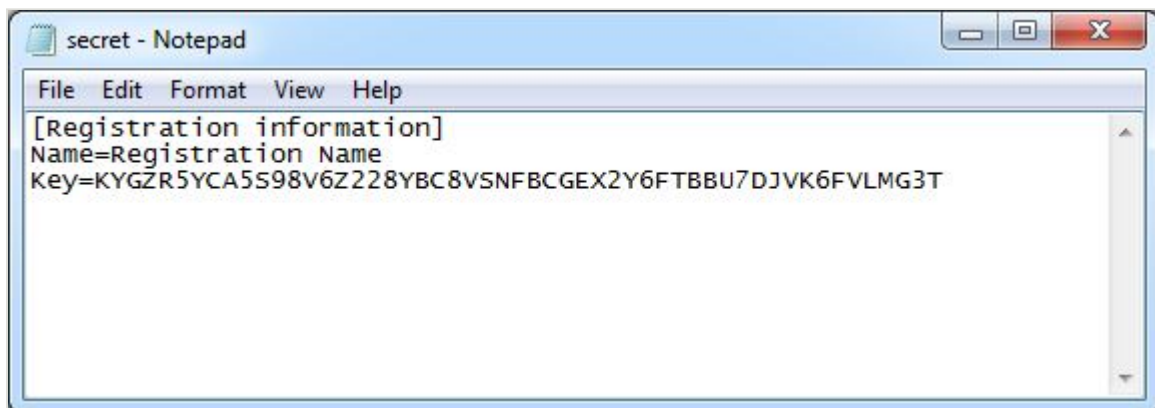
文件存储相对路径（例如：\YourCompany\key.dat）：
\notepad\secret.dat

设置文件属性 Archive, Read only

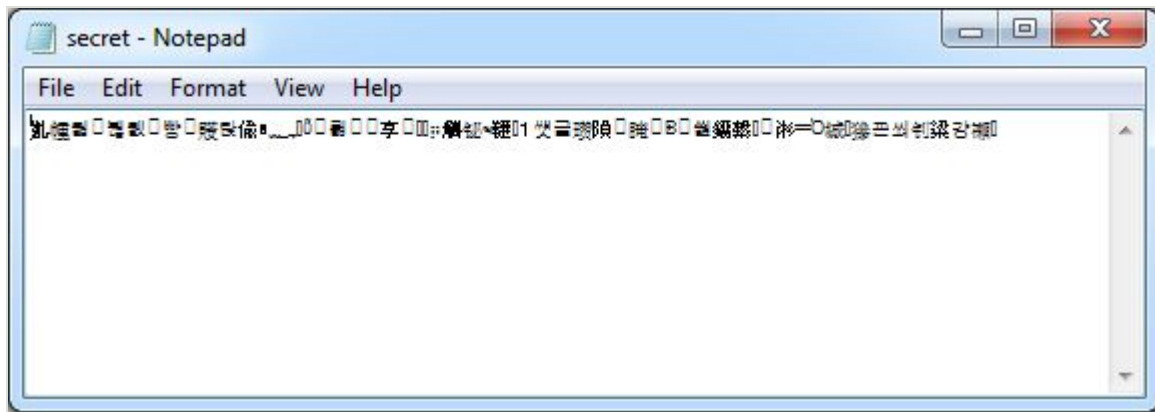
The Enigma Protector

Encrypt Registration Information - allows to encrypt registration information. If the option is disabled, registration information (name and key) will be stored in registry/file as they are.

For example, the file with the registration information, without Encrypt Registration Information feature, will be looking:



with Encrypt Registration Information feature



Disable copying of the registration information to another PC (the registration information will be encrypted with the user's Hardware ID) - this option allows to encrypt registration information also with the user's hardware id. If the registration information will be encrypted with hardware id, the registry item or with with registration information will be working and valid only on the PC where it was created. If user will attempt to copy this registry item or file to another PC, registration will fail on this PC.

Please note:

- | This feature is incompatible with the "Allow Hardware Changes" from [Hardware Lock](#) panel. This means that if hardware id will be changed on the user's PC, and even if the Allow Hardware Changes is enabled, the registry item or file with registration information will become invalid and application will require registration again;
- | Do not use this feature to lock registration information to particular PC. To lock license to particular PC you have to use [Hardware Lock](#) panel and generate the registration keys with user's hardware id.

- | HKEY_CURRENT_USER
- | HKEY_LOCAL_MACHINE

" " HKEY_CURRENT_USER" " \SOFTWARE\YourCompany\"

Registration name - HKEY_CURRENT_USER\SOFTWARE\YourCompany\Name
 Registration key - HKEY_CURRENT_USER\SOFTWARE\YourCompany\Key

- | %DEFAULT FOLDER% -
- | %SYSTEM FOLDER% - Windows C:\
- | %WINDOWS FOLDER% - Windows C:\WINDOWS\SYSTEM\
- | %My Documents FOLDER% - " "
- | %My Pictures FOLDER% - " "
- | %Program Files FOLDER% - " "
- | %Program Files\Common FOLDER% - " Program Files\Common folder"
- | %AllUsers\Documents FOLDER% - " All Users\Documents folder"
- | %History FOLDER% - " History "
- | %Cookies FOLDER% - " Cookies"
- | %InternetCache FOLDER% - " InternetCache "

" secret.dat" " \notepad\secret.dat" " notpad" *.ini *.txt

-

“ / / / ”

-

ID

使用指定硬件 ID 加密
输入用于加密的硬件 ID：

ID /

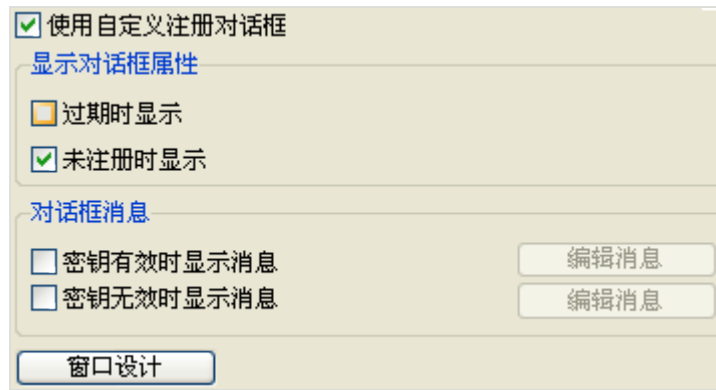
ID /

ID

ID

ID /

ID



mean Enigma API function [EP_RegShowDialog](#).

Note, this Registration Dialog can be also called by

-

-

-

-

-

-

" "

" "

Align Property

Determines how the control aligns within its container (parent control).

Description

Use `Align` to align a control to the top, bottom, left, or right of a form or panel and have it remain there even if the size of the form, panel, or component that contains the control changes. When the parent is resized, an aligned control also resizes so that it continues to span the top, bottom, left, or right edge of the parent.

For example, to use a panel component with various controls on it as a tool palette, change the panel's `Align` value to `alLeft`. The value of `alLeft` for the `Align` property of the panel guarantees that the tool palette remains on the left side of the form and always equals the client height of the form.

The default value of `Align` is `alNone`, which means a control remains where it is positioned on a form or panel.

Tip: If `Align` is set to `alClient`, the control fills the entire client area so that it is impossible to select the parent form by clicking on it. In this case, select the parent by selecting the control on the form and pressing `Esc`, or by using the Object Inspector.

Any number of child components within a single parent can have the same `Align` value, in which case they stack up along the edge of the parent. To adjust the order in which the controls stack up, drag the controls into their desired positions.

Note: To cause a control to maintain a specified relationship with an edge of its parent, but not necessarily lie along one edge of the parent, use the `Anchors` property instead.

Value	Meaning
<code>alNone</code>	The control remains where it was placed. This is the default value
<code>alTop</code>	The control moves to the top of its parent and resizes to fill the width of its parent. The height of the control is not affected
<code>alBottom</code>	The control moves to the bottom of its parent and resizes to fill the width of its parent. The height of the control is not affected
<code>alLeft</code>	The control moves to the left side of its parent and resizes to fill the height of its parent. The width of the control is not affected
<code>alRight</code>	The control moves to the right side of its parent and resizes to fill the height of its parent. The width of the control is not affected
<code>alClient</code>	The control resizes to fill the client area of its parent. If another control already occupies part of the client area, the control resizes to fit within the remaining client area
<code>alCustom</code>	The control's positioning is determined by calls to its parent's <code>CustomAlignInsertBefore</code> and <code>CustomAlignPosition</code> methods

Alignment Property

Controls the horizontal placement of the text within the label.

Description

Set Alignment to specify how the text of the label is justified within the ClientRect of the label control.

The effect of the Alignment property is more obvious if the WordWrap property is true and the label includes more than one line of text.

AlphaBlend Property

Specifies whether the form is translucent.

Description

Set `AlphaBlend` to specify that the form represents a layered window that allows a translucent color. The `AlphaBlendValue` property specifies the degree of translucency.

AlphaBlendValue Property

Specifies the degree of translucency on a translucent form.

Description

Set AlphaBlendValue to a value between 0 and 255 to indicate the degree of translucency when the AlphaBlend property is true. A value of 0 indicates a completely transparent window. A value of 255 indicates complete opacity.

Note: AlphaBlendValue only has an effect when the AlphaBlend property is true.

Anchors Property

Specifies how the control is anchored to its parent.

Description

Use Anchors to ensure that a control maintains its current position relative to an edge of its parent, even if the parent is resized. When its parent is resized, the control holds its position relative to the edges to which it is anchored.

If a control is anchored to opposite edges of its parent, the control stretches when its parent is resized. For example, if a control has its Anchors property set to [akLeft, akRight], the control stretches when the width of its parent changes.

Anchors is enforced only when the parent is resized. Thus, for example, if a control is anchored to opposite edges of a form at design time and the form is created in a maximized state, the control is not stretched because the form is not resized after the control is created.

Note: If a control should maintain contact with three edges of its parent (hugging one side of the parent and stretching the length of that side), use the Align property instead. Unlike Anchors, Align allows controls to adjust to changes in the size of other aligned sibling controls as well as changes to the parent's size.

Value	Meaning
akTop	The control's position is fixed with respect to the top edge of its parent
akLeft	The control's position is fixed with respect to the left edge of its parent
akRight	The control's position is fixed with respect to the right edge of its parent
akBottom	The control's position is fixed with respect to the bottom edge of its parent

AutoSize Property

Specifies whether the control sizes itself automatically to accommodate its contents.

Description

Use `AutoSize` to specify whether the control sizes itself automatically. When `AutoSize` is true, the control resizes automatically when its contents change.

BevelEdges Property

Specifies which edges of the control are beveled.

Description

The BevelInner, BevelOuter, and BevelKind properties determine the appearance of the specified edges.

Value	Meaning
beLeft	The left edge is beveled.
beTop	The top edge is beveled.
beRight	The right edge is beveled.
beBottom	The bottom edge is beveled.

BevelInner Property

Specifies the cut of the inner bevel.

Description

Use `BevelInner` to specify whether the inner bevel has a raised, lowered, or flat look.

The inner bevel appears immediately inside the outer bevel. If there is no outer bevel (`BevelOuter` is `bvNone`), the inner bevel appears immediately inside the border.

Value	Meaning
<code>bvNone</code>	The bevel does not exist.
<code>bvLowered</code>	The bevel appears lowered.
<code>bvRaised</code>	The bevel appears raised.
<code>bvSpace</code>	The bevel appears as a space if its kind is not <code>bkTile</code> . Otherwise, the bevel appears raised.

BevelKind Property

Specifies the control's bevel style.

Description

Use BevelKind to modify the appearance of a bevel. BevelKind influences how sharply the bevel stands out.

BevelKind, in combination with BevelWidth and the cut of the bevel specified by BevelInner or BevelOuter, can create a variety of effects. Experiment with various combinations to get the look you want.

Value	Meaning
bkNone	No bevel is added.
bkTile	The bevel appears sharply defined.
bkSoft	The bevel uses softer contrasts than bkTile.
bkFlat	The bevel has a broad, flat appearance.

BevelOuter Property

Specifies the cut of the outer bevel.

Description

Use `BevelInner` to specify whether the outer bevel has a raised, lowered, or flat look.

The outer bevel appears immediately inside the border and outside the inner bevel.

Value	Meaning
<code>bvNone</code>	The bevel does not exist.
<code>bvLowered</code>	The bevel appears lowered.
<code>bvRaised</code>	The bevel appears raised.
<code>bvSpace</code>	The bevel appears as a space if its kind is not <code>bkTile</code> . Otherwise, the bevel appears raised.

BevelWidth Property

Determines the width, in pixels, of both the inner and outer bevels of a panel.

Description

Use `BevelWidth` to specify how wide the inner or outer bevel should be. Do not confuse `BevelWidth`, which is the width of the bevels, with `BorderWidth`, which is the space between the bevels.

If both the `BevelInner` and `BevelOuter` properties are `bvNone`, `BevelWidth` has no effect. To remove both bevels, set the `BevelInner` and `BevelOuter` properties to `bvNone`, rather than setting the `BevelWidth` to 0, as this involves less overhead when painting.

BiDiMode Property

Specifies the bi-directional mode for the control.

Description

The bi-directional mode controls the reading order for the text, the placement of the vertical scroll bar, and whether the alignment is changed.

Value	Meaning
bdLeftToRight	Reading order is left to right. Alignment is not changed. The vertical scroll bar appears on the right edge of the control.
bdRightToLeft	Reading order is right to left. Alignment is changed. The vertical scroll bar appears on the left edge of the control.
bdRightToLeftNoAlign	Reading order is right to left. Alignment is not changed. The vertical scroll bar appears on the left edge of the control.
bdRightToLeftReadingOnly	Reading order is right to left. Alignment and scroll bar are not changed.

BorderStyle Property

Specifies which icons appear on the title bar of the form.

Description

May contain the following values:

Value	Meaning
bsSystemMenu	The form has a Control menu (also known as a System menu).
bsMinimize	The form has a Minimize button
bsMaximize	The form has a Maximize button
bsHelp	If BorderStyle is bsDialog or bsMinimize and bsMaximize are excluded, a question mark appears in the form's title bar and when clicked, the cursor changes to crHelp; otherwise, no question mark appears.

BorderStyle Property

Determines whether the edit control has a single line border around the client area.

Description

Use `BorderStyle` to affect the sharpness with which the client area of the edit control stands out. `BorderStyle` can have a value of either `bsSingle` or `bsNone`. If `BorderStyle` is `bsSingle`, the edit control has a single-line border around the client area. If `BorderStyle` is `bsNone`, there will be no border.

Value	Meaning
<code>bsNone</code>	No visible border.
<code>bsSingle</code>	Single-line border.

BorderWidth Property

Specifies the distance, in pixels, between the outer and inner bevels.

Description

Use `BorderWidth` to specify how wide the border around the panel should be. A value of 0 (zero) means no border should appear.

Brush Property

Specifies the color and pattern used for filling the shape control.

Cancel Property

Determines whether the button's OnClick event handler executes when the Escape key is pressed.

Description

If Cancel is true, the button's OnClick event handler executes when the user presses Esc. Although an application can have more than one Cancel button, the form calls the OnClick event handler only for the first visible button in the tab order.

Caption Property

Specifies a text string that identifies the control to the user.

Description

Use Caption to specify the text string that labels the control.

To underline a character in a Caption that labels a component, include an ampersand (&) before the character. This type of character is called an accelerator character. The user can then select the component by pressing Alt while typing the underlined character. To display an ampersand character in the caption, use two ampersands (&&).

Note: Controls that display text use either the Caption property or the Text property to specify the text value. Which property is used depends on the type of control. In general, Caption is used for text that appears as a window title or label, while Text is used for text that appears as the content of a control.

Center Property

Indicates whether the image is centered in the image control.

Description

When the image does not fit perfectly within the image control, use Center to specify how the image is positioned. When Center is true, the image is centered in the control. When Center is false, the upper left corner of the image is positioned at the upper left corner of the control.

CharCase Property

Determines the case of the text within the edit control.

Description

Value	Meaning
ecLowerCase	The text is converted to lowercase.
ecNormal	The text appears in mixed case. It is not forced into any case.
ecUpperCase	The text is converted to uppercase.

ClickAction Property

Determines the OnClick event handler.

Description

ClickAction may have following values.

Value	Action
caRegister	Perform register. This way the application reads name and key value from edit controls and tries to register.
caContinue	Continue execution
caExit	Exit application
caNone	Nothing to do

ClientHeight Property

Specifies the height (in pixels) of the form's client area.

Description

Use `ClientHeight` to determine the height (in pixels) of the form's client area. The client area is the usable area inside the form's border, excluding the title bar, scroll bars, and so on.

Set `ClientHeight` to change the height of the form's window based on the desired client area. To change the height of the form's window based on the total size of the window (including the border, menu, status bar and so on), use the `Height` property instead.

ClientWidth Property

Specifies the width (in pixels) of the form's client area.

Description

Use ClientWidth to determine the width (in pixels) of the form's client area. The client area is the usable area inside the form's border. Set ClientWidth to change the width of the form's window based on the desired client area. To change the width of the form's window based on the total size of the window (including the border, status bar and so on), use the Width property instead.

CloseAction Property

Determines the OnClose event handler.

Description

CloseAction may have following values.

Value	Action
caRegister	Perform register. This way the application reads name and key value from edit controls and tries to register.
caContinue	Continue execution
caExit	Exit application
caNone	Nothing to do

Color Property

Specifies the background color of the control.

Description

Use `Color` to read or change the background color of the control.

Constraints Property

Specifies the size constraints for the control.

Description

Use Constraints to specify the minimum and maximum width and height of the control. When Constraints contains maximum or minimum values, the control cannot be resized to violate those constraints.

Warning: Do not set up constraints that conflict with the value of the Align or Anchors property. When these properties conflict, the response of the control to resize attempts is not well-defined.

Content Property

Determines the content of the edit control. This information is used when the registration information is verifying and the current hardware id should be shown.

Description

Content may have following values.

Value	Action
coRegistratioName	The edit will contain a registration name. Users will have to enter registration name into this control.
coRegistratioKey	The edit will contain a registration key. Users will have to enter registration key into this control.
coHardwareID	This edit will be filled out with the Hardware ID.
coCustom	The edit contains custom text.

Ctl3D Property





















Determines whether a control has a three-dimensional (3-D) or two-dimensional look.

Cursor Property

Specifies the image used to represent the mouse pointer when it passes into the region covered by the control.

Description

Change the value of Cursor to provide feedback to the user when the mouse pointer enters the control.

Cursor	Image
crDefault	Whatever cursor is the default for the window class (usually crArrow)
crNone	
crArrow	
crCross	
crIBeam	
crSizeNESW	
crSizeNS	
crSizeNWSE	
crSizeWE	
crUpArrow	
crHourGlass	
crDrag	
crNoDrop	
crHSplit	
crVSplit	
crMultiDrag	
crSQLWait	
crNo	
crAppStart	
crHelp	
crHandPoint	

crSize	
crSizeAll	

Default Property

Determines whether the button's OnClick event handler executes when the Enter key is pressed.

Description

If Default is true, the button's OnClick event handler executes when the user presses Enter.

Although an application can have more than one Default button, the form calls the OnClick event handler only for the first visible button in the tab order. Moreover, any button that has focus becomes the Default button temporarily; hence, if the user selects another button before pressing Enter, the selected button's OnClick event handler executes instead.

Enabled Property

Controls whether the control responds to mouse and keyboard.

Description

Use Enabled to change the availability of the control to the user. To disable a control, set Enabled to false. Disabled controls appear dimmed. If Enabled is false, the control ignores mouse and keyboard.

Flat Property

Determines whether the button has a 3D border that provides a raised or lowered look.

Description

Set Flat to true to remove the raised border when the button is unselected and the lowered border when the button is clicked or selected. When Flat is true, use separate bitmaps for the different button states to provide visual feedback to the user about the button state.

Font Property

Controls the attributes of text written on or in the control.

FontHovered Property

Controls the attributes of text of the TLink when the mouse pointer is moving over the control.

FormStyle Property

Determines the form's style.

Description

FormStyle consists of the following values.

Value	Meaning
fsNormal	The form is neither an MDI parent window nor an MDI child window.
fsMDIChild	The form is an MDI child window.
fsMDIForm	The form is an MDI parent window.
fsStayOnTop	This form remains on top of the desktop and of other forms in the project, except any others that also have FormStyle set to fsStayOnTop. If one fsStayOnTop form launches another, neither form will consistently remain on top.

Glyph Property

Specifies the bitmap that appears on the speed button.

Description

Set Glyph to a bitmap object that contains the image that should appear on the face of the button. Bring up the Open dialog box from the Object Inspector to choose a bitmap file (with a .BMP extension).

Glyph can provide up to four images within a single bitmap. All images must be the same size and next to each other in a horizontal row. TSpeedButton displays one of these images depending on the state of the button.

Image position	Button state	Description
First	Up	This image appears when the button is unselected. If no other images exist in the bitmap, this image is used for all states.
Second	Disabled	This image usually appears dimmed to indicate that the button can't be selected.
Third	Clicked	This image appears when the button is clicked. If GroupIndex is 0, the Up image reappears when the user releases the mouse button.
Fourth	Down	This image appears when the button stays down indicating that it remains selected.

If only one image is present, TSpeedButton attempts to represent the other states by altering the image slightly for each state, although the Down state is always the same as the Up state.

If the bitmap contains multiple images, specify the number of images in the bitmap with the NumGlyphs property.

Note: The lower left pixel of the bitmap is reserved for the "transparent" color. Any pixel in the bitmap that matches the lower left pixel will be transparent.

Height Property

Specifies the vertical size of the control in pixels.

Hint Property

Contains the text string that can appear when the user moves the mouse over the control.

Description

Use the Hint property to provide a string of help text either as a Help Hint, or as help text on a particular location such as a status bar.

A Help Hint is a box containing help text that appears for a control when the user moves the mouse pointer over the control and pauses momentarily. To set up Help Hints:

- Specify the Hint property of each control for which a Help Hint should appear
- Set the ShowHint property of each appropriate control to true and set the ShowHint property of the form to true.

Note: If the application's ShowHint property is false, the Help Hint does not appear.

HorzScrollBar Property

Represents the horizontal scroll bar for the scrolling windowed control.

Layout Property

Specifies the vertical placement of the text within the label.

Description

Set `Layout` to specify how the text of the label is placed within the `ClientRect` of the label control. `Layout` is the vertical analog to the `Alignment` property.

Left Property

Specifies the horizontal coordinate of the left edge of a component relative to its parent.

Lines Property

Contains the individual lines of text in the memo control.

Description

Use Lines to manipulate text in an memo control on a line-by-line basis.

Margin Property

Specifies the number of pixels between the edge of the button and the image or caption drawn on its surface.

Description

Use Margin to specify the indentation of the image specified by the Glyph property or the text specified by the Caption property. The edges that Margin separates depends on the Layout property. If Layout is blGlyphLeft, the margin appears between the left edge of the image or caption and the left edge of the button. If Layout is blGlyphRight, the margin separates the right edges. If Layout is blGlyphTop, the margin separates the top edges, and if Layout is blGlyphBottom, the margin separates the bottom edges.

If Margin is - 1, the image or text is centered on the button.

MaxLength Property

Specifies the maximum number of characters the user can enter into the edit control.

Description

Use `MaxLength` to limit the number of characters that can be entered into the edit control. A value of `-1` indicates that there is no application-defined limit on the length.

Use `MaxLength` to limit the length of the text in an edit control if that text will be copied into a fixed-length buffer.

Note: Setting `MaxLength` to a value less than the number of characters currently in the edit control causes the edit control to truncate its text to `MaxLength` characters.

Name Property

Specifies the name of the control.

Description

Use the Name property to assign a new name to the control or to find out what the name of the control is. Property is informative, it does not affect on workability.

NumGlyphs Property

Specifies the number of images included in the Glyph property.

Description

Set NumGlyphs to the number of images provided by the bitmap assigned to the Glyph property. All images must be the same size and next to each other in a row. The Glyph property can provide up to four images.

PasswordChar Property

Indicates the character, if any, to display in place of the actual characters typed in the control.

Description

Use the PasswordChar property to create an edit control that displays a special character in place of any entered text. If PasswordChar is set to the null character (ANSI character zero), the edit control displays its text normally. If PasswordChar is any other character, the edit control displays PasswordChar in place of each character typed. PasswordChar affects the appearance of the edit control only. The value of the Text property reflects the actual characters that are typed.

Pen Property

Specifies the pen used to outline the shape control.

Picture Property

Specifies the image that appears on the image control.

Position Property

Represents the size and placement of the form.

Proportional Property

Indicates whether the image should be changed, without distortion, so that it fits the bounds of the image control.

Description

Set Proportional to true to ensure that the image can be fully displayed in the image control without any distortion such as occurs with the Stretch property. When Proportional is true, images that are too large to fit in the image control are scaled down (while maintaining the same aspect ratio) until they fit in the image control. Images that are too small are displayed normally. That is, Proportional can reduce the magnification of the image, but does not increase it.

When the image control resizes, the image resizes also.

To resize the image so that it fits exactly in the image control, even if that causes distortion, use the Stretch property instead.

To resize the control to the image rather than resizing the image to the control, use the AutoSize property instead.

The default value for Proportional is false.

Note: Proportional has no effect if the Picture property contains an icon.

ReadOnly Property

Determines whether the user can change the text of the edit control.

Description

To restrict the edit control to display only, set the ReadOnly property to true. Set ReadOnly to false to allow the contents of the edit control to be edited.

ScrollBars Property

Determines whether the control has scroll bars.

Description

ScrollBars can take one of the following values:

Value	Meaning
ssNone	The control has no scroll bars.
ssHorizontal	The control has a single scroll bar on the bottom edge.
ssVertical	The control has a single scroll bar on the right edge.
ssBoth	Horizontal scrollbar that appears as-needed.

Shape Property

Determines the shape of the bevel.

Description

Set Shape to specify whether the bevel appears as a line, box, frame, or space. For shapes that can appear either raised or lowered, the Style property indicates which effect is used.

The default value for Shape is bsBox.

Value	Meaning
bsBox	The entire client area appears raised or lowered.
bsFrame	The client area is outlined by a raised or lowered frame.
bsTopLine	The bevel displays a line at the top of the client area.
bsBottomLine	The bevel displays a line at the bottom of the client area.
bsLeftLine	The bevel displays a line at the left side of the client area.
bsRightLine	The bevel displays a line at the right side of the client area.
bsSpacer	The bevel is an empty space.

ShowCopyButton Property

Allows to show a Copy to Clipboard button near with the edit control and associated with it.

Description

This property could be, for example, used to copy shown Hardware ID from control to clipboard.

ShowHint Property

Determines whether the control displays a Help Hint when the mouse pointer rests momentarily on the control.

Description

The Help Hint is the value of the Hint property, which is displayed in a box just beneath the control. Use ShowHint to determine whether a Help Hint appears for the control.

To enable Help Hint for a particular control, the application ShowHint property must be true.

ShowPasteButton Property

Allows to show a Paste from Clipboard button near with the edit control and associated with it.

Description

This property could be, for example, used to paste registration name or key to the control from clipboard.

Spacing Property

Determines where the image and text appear on a speed button.

Description

Set Spacing to the number of pixels that should appear between the image specified in the Glyph property and the text specified in the Caption property.

If Spacing is a positive number, its value is the number of pixels between the image and text. If Spacing is 0, the image and text appear flush with each other. If Spacing is -1, the text appears centered between the image and the button edge.

Stretch Property

Indicates whether the image should be changed so that it exactly fits the bounds of the image control.

Description

Set `Stretch` to true to cause the image to assume the size and shape of the image control. When the image control resizes, the image resizes also. `Stretch` resizes the height and width of the image independently. Thus, unlike a simple change in magnification, `Stretch` can distort the image if the image control is not the same shape as the image.

To resize the control to the image rather than resizing the image to the control, use the `AutoSize` property instead.

Style Property

Determines whether the bevel appears raised or lowered.

Description

Set `Style` to indicate whether the bevel should create a raised or a lowered effect. When the `Shape` property is `bsBox`, the entire client area appears raised or lowered. For all other values of `Shape`, the bevel displays a raised or lowered line along the edge or edges of the client area. The default value of `Style` is `bsLowered`.

Value	Meaning
<code>bsLowered</code>	The bevel is lowered.
<code>bsRaised</code>	The bevel is raised.

TabOrder Property

Indicates the position of the control in its parent's tab order.

Description

TabOrder is the order in which child windows are visited when the user presses the Tab key. The control with the TabOrder value of 0 is the control that has the focus when the form first appears.

Initially, the tab order is always the order in which the controls were added to the form. The first control added to the form has a TabOrder value of 0, the second is 1, the third is 2, and so on. Change this by changing the TabOrder property.

Each control has a unique tab-order value within its parent. If you change the TabOrder property value of one control to be the same as the value of a second control, the TabOrder value for all the other controls changes. For example, suppose a control is sixth in the tab order. If you change the control's TabOrder property value to 3 (making the control fourth in the tab order), the control that was originally fourth in the tab order now becomes fifth, and the control that was fifth becomes sixth.

Assigning TabOrder a value greater than the number of controls contained in the parent control moves the control to the end of the tab order. The control does not take on the assigned value of TabOrder, but instead is given the number that assures the control is the last in the tab order.

Note: TabOrder is meaningful only if the TabStop property is true and if the control has a parent. (The TabOrder property of a form is not used unless the form is the child of another form.) A control with a TabOrder of -1 has no parent, and therefore cannot be reached by pressing the Tab key. To remove a parented control from the Tab order, set its TabStop property to false.

TabStop Property

Determines if the user can tab to a control.

Description

Use the TabStop to allow or disallow access to the control using the Tab key.

If TabStop is true, the control is in the tab order. If TabStop is false, the control is not in the tab order and the user can't press the Tab key to move to the control.

Note: TabStop is not meaningful for a form unless the form assigns another form to be its parent.

Top Property

Specifies the Y coordinate of the top left corner of a control, relative to its parent or containing control in pixels.

Transparent Property

Specifies whether the background of the control obscures objects below the control object.

Description

Set Transparent to true to allow objects behind the control object to show through the background of the control.
Set Transparent to false to make the background of the control opaque.

TransparentColor Property

Specifies whether a color on the form appears transparent.

Description

Use `TransparentColor` to indicate that one of the colors on the form should be treated as transparent, allowing windows behind the form to completely show through. The `TransparentColorValue` property indicates the color that appears completely transparent.

Note: To make the entire form transparent, or to make it translucent rather than transparent, use the `AlphaBlend` and `AlphaBlendValue` properties.

TransparentColorValue Property

Indicates the color on the form that appears transparent when TransparentColor is true.

Description

Use TransparentColorValue to indicate the color that appears transparent when the TransparentColor property is true.

Url Property

Specifies the Url that will be opened on OnClick action.

Description

Url property can open not only the browser window but also a default email client and some other files. For example:

url `http://enigmaprotector.com/` opens a browser window with the current address,

url `mailto:support@enigmaprotector.com` opens a default email client with the offer to create email for `support@enigmaprotector.com`

url `License.txt` opens a Windows notepad with the `License.txt` file.

VertScrollBar Property

Represents the vertical scroll bar for the scrolling windowed control.

Visible Property

Determines whether the component appears onscreen.

WantReturns Property

Determines whether the user can insert return characters into the text.

Description

Set `WantReturns` to `true` to allow users to enter return characters into the text. Set `WantReturns` to `false` to allow the form to handle return characters instead.

For example, in a form with a default button (such as an OK button) and a memo control, if `WantReturns` is `false`, pressing Enter chooses the default button. If `WantReturns` is `true`, pressing Enter inserts a return character in the text.

Note: If `WantReturns` is `false`, users can still enter return characters into the text by pressing `Ctrl+Enter`.

Width Property

Specifies the horizontal size of the control or form in pixels.

WindowState Property

Represents how the form appears on the screen.

Description

WindowState describes the state of a form window. The following table lists the possible values:

Value	Meaning
wsNormal	The form is in its normal state (that is, neither minimized nor maximized).
wsMinimized	The form is minimized.
wsMaximized	The form is maximized.

WordWrap Property

Specifies whether the button text wraps to fit the width of the control.

Description

Set `WordWrap` to `true` to allow the label to display multiple line of text. When `WordWrap` is `true`, text that is too wide for the control wraps at the right margin.

Set `WordWrap` to `false` to limit the label to a single line. When `WordWrap` is `false`, text that is too wide for the label appears truncated.

TBevel



Use TBevel to create beveled boxes, frames, or lines. The bevel can appear raised or lowered.

Properties

- | [Align](#)
- | [Anchors](#)
- | [Constraints](#)
- | [Cursor](#)
- | [Height](#)
- | [Hint](#)
- | [Left](#)
- | [Name](#)
- | [Shape](#)
- | [ShowHint](#)
- | [Style](#)
- | [Top](#)
- | [Visible](#)
- | [Width](#)

TButton

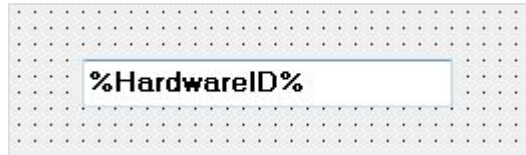


Use TButton to put a standard push button on a form. TButton may process such user actions like register, close application and continue execution.

Properties

- | [Anchors](#)
- | [BiDiMode](#)
- | [Cancel](#)
- | [Caption](#)
- | [ClickAction](#)
- | [Constraints](#)
- | [Cursor](#)
- | [Default](#)
- | [Enabled](#)
- | [Font](#)
- | [Height](#)
- | [Hint](#)
- | [Left](#)
- | [Name](#)
- | [ShowHint](#)
- | [TabOrder](#)
- | [TabStop](#)
- | [Top](#)
- | [Visible](#)
- | [Width](#)
- | [WordWrap](#)

TEdit

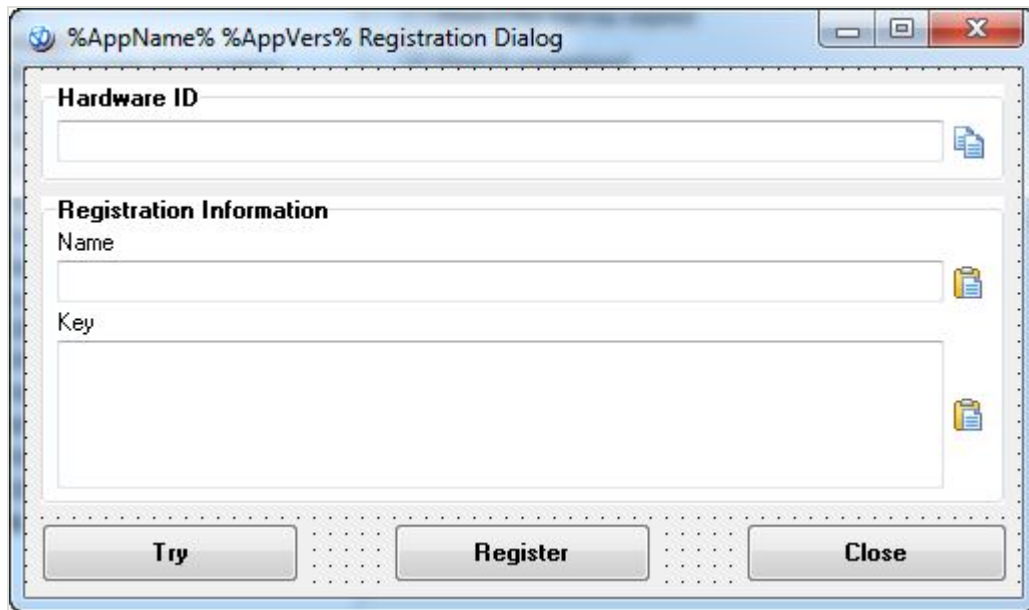


Use a TEdit object to put a standard Windows edit control on a form. Edit controls are used to retrieve text that users type. Edit controls can also display text to the user. Edit may contain Hardware ID string or custom string, allows to enter registration name and key.

Properties

- | [Align](#)
- | [Anchors](#)
- | [BevelEdges](#)
- | [BevelInner](#)
- | [BevelKind](#)
- | [BevelOuter](#)
- | [BevelWidth](#)
- | [BiDiMode](#)
- | [BorderStyle](#)
- | [CharCase](#)
- | [Color](#)
- | [Constraints](#)
- | [Content](#)
- | [Ctl3D](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Font](#)
- | [Height](#)
- | [Hint](#)
- | [Left](#)
- | [MaxLength](#)
- | [Name](#)
- | [PasswordChar](#)
- | [ReadOnly](#)
- | [ShowCopyButton](#)
- | [ShowHint](#)
- | [ShowPasteButton](#)
- | [TabOrder](#)
- | [TabStop](#)
- | [Top](#)
- | [Visible](#)
- | [Width](#)

TForm

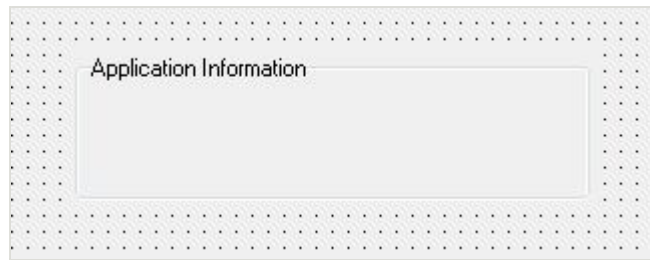


TForm represents the main registration window.

Properties

- | AlphaBlend
- | AlphaBlendValue
- | BiDiMode
- | BorderIcons
- | BorderStyle
- | BorderWidth
- | Caption
- | ClickAction
- | ClientHeight
- | ClientWidth
- | CloseAction
- | Color
- | Constraints
- | CtI3D
- | Cursor
- | Enabled
- | Font
- | FormStyle
- | Height
- | Hint
- | HorzScrollBar
- | Left
- | Name
- | Position
- | ShowHint
- | Top
- | TransparentColor
- | TransparentColorValue
- | VertScrollBar
- | Width
- | WindowState

TGroupBox

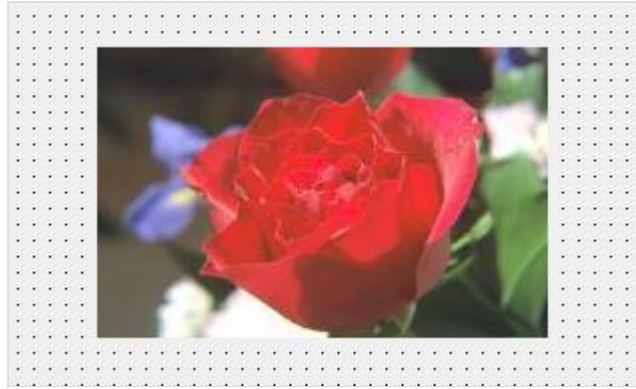


The TGroupBox component represents a standard Windows group box, used to group related controls on a form.

Properties

- | [Align](#)
- | [Anchors](#)
- | [BiDiMode](#)
- | [Caption](#)
- | [Color](#)
- | [Constraints](#)
- | [Ctl3D](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Font](#)
- | [Height](#)
- | [Hint](#)
- | [Left](#)
- | [Name](#)
- | [ShowHint](#)
- | [TabOrder](#)
- | [TabStop](#)
- | [Top](#)
- | [Visible](#)
- | [Width](#)

TImage




Use TImage to display a graphical image on a form.

Properties

- | [Align](#)
- | [Anchors](#)
- | [AutoSize](#)
- | [Center](#)
- | [Constraints](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Height](#)
- | [Hint](#)
- | [Left](#)
- | [Name](#)
- | [Picture](#)
- | [Proportional](#)
- | [ShowHint](#)
- | [Stretch](#)
- | [Top](#)
- | [Transparent](#)
- | [Visible](#)
- | [Width](#)

TLabel




Enter Registration Information

Use TLabel to add text that the user can't edit to a form.

Properties

- | [Align](#)
- | [Alignment](#)
- | [Anchors](#)
- | [AutoSize](#)
- | [BiDiMode](#)
- | [Caption](#)
- | [Color](#)
- | [Constraints](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Font](#)
- | [Height](#)
- | [Hint](#)
- | [Layout](#)
- | [Left](#)
- | [Name](#)
- | [ShowHint](#)
- | [Top](#)
- | [Transparent](#)
- | [Visible](#)
- | [Width](#)
- | [WordWrap](#)

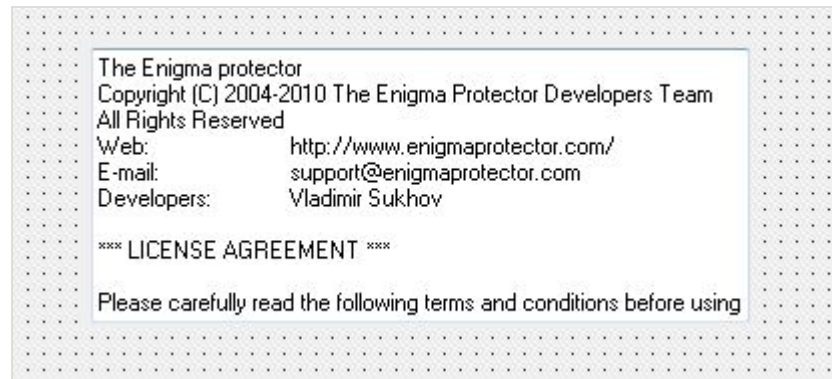


[Click Here to Go to the Online Order Page](#)

Use TLabel to a hyper link to the form that will be able to open an url in the browser, open default email client, open some external file.

Properties

- | [Align](#)
- | [Alignment](#)
- | [Anchors](#)
- | [AutoSize](#)
- | [BiDiMode](#)
- | [Caption](#)
- | [Color](#)
- | [Constraints](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Font](#)
- | [FontHovered](#)
- | [Height](#)
- | [Hint](#)
- | [Layout](#)
- | [Left](#)
- | [Name](#)
- | [ShowHint](#)
- | [Top](#)
- | [Transparent](#)
- | [Url](#)
- | [Visible](#)
- | [Width](#)
- | [WordWrap](#)



Use TMemo to put a standard Windows multiline edit control on a form. Multiline edit boxes allow the user to enter more than one line of text. They are appropriate for representing lengthy information. Memo may contain Hardware ID string or custom string, allows to enter registration name and key.

Properties

- | Align
- | Alignment
- | Anchors
- | BevelEdges
- | BevelInner
- | BevelKind
- | BevelOuter
- | BiDiMode
- | BorderStyle
- | Color
- | Constraints
- | Content
- | CtI3D
- | Cursor
- | Enabled
- | Font
- | Height
- | Hint
- | Left
- | Lines
- | MaxLength
- | Name
- | ReadOnly
- | ScrollBars
- | ShowCopyButton
- | ShowHint
- | ShowPasteButton
- | TabOrder
- | TabStop
- | Top
- | Visible
- | WantReturns
- | Width
- | WordWrap

TPanel

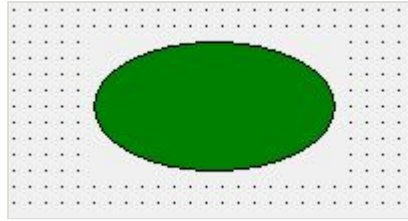


Use TPanel to put an empty panel on a form.

Properties

- | [Align](#)
- | [Alignment](#)
- | [Anchors](#)
- | [BevelEdges](#)
- | [BevelInner](#)
- | [BevelKind](#)
- | [BevelOuter](#)
- | [BevelWidth](#)
- | [BiDiMode](#)
- | [BorderStyle](#)
- | [BorderWidth](#)
- | [Caption](#)
- | [Color](#)
- | [Constraints](#)
- | [Ctl3D](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Font](#)
- | [Height](#)
- | [Hint](#)
- | [Left](#)
- | [Name](#)
- | [ShowHint](#)
- | [TabOrder](#)
- | [TabStop](#)
- | [Top](#)
- | [Visible](#)
- | [Width](#)

TShape



Add a TShape object to a form to draw a simple geometric shape on the form.

Properties

- | [Align](#)
- | [Anchors](#)
- | [Brush](#)
- | [Constraints](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Height](#)
- | [Hint](#)
- | [Left](#)
- | [Name](#)
- | [Pen](#)
- | [Shape](#)
- | [ShowHint](#)
- | [Top](#)
- | [Visible](#)
- | [Width](#)

TSpeedButton



TSpeedButton is almost same [TButton](#) control but allows to add an image onto the button.

Properties

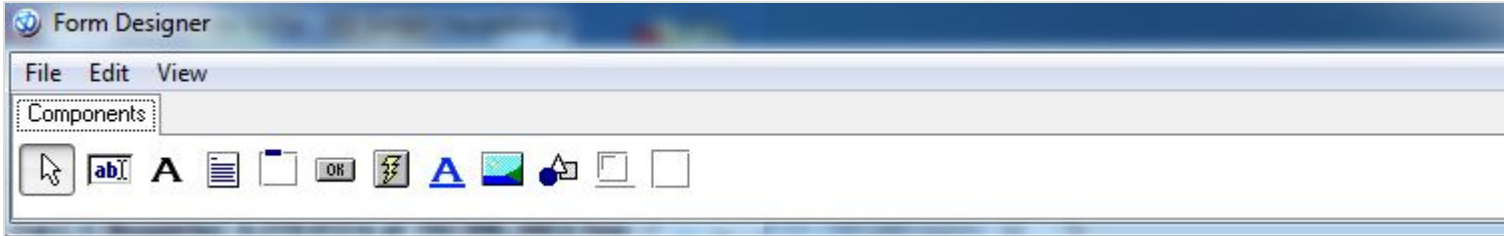
- | [Anchors](#)
- | [BiDiMode](#)
- | [Caption](#)
- | [ClickAction](#)
- | [Constraints](#)
- | [Cursor](#)
- | [Enabled](#)
- | [Flat](#)
- | [Font](#)
- | [Glyph](#)
- | [Height](#)
- | [Hint](#)
- | [Layout](#)
- | [Left](#)
- | [Margin](#)
- | [Name](#)
- | [NumGlyphs](#)
- | [ShowHint](#)
- | [Spacing](#)
- | [Top](#)
- | [Visible](#)
- | [Width](#)

Registration Dialog Designer











The Registration Dialog Designer allows to desing a custom registration dialog, apply own styles and window controls con are able to put own controls on the form and define registration events. Registration dialog designer contains 3 forms:

Components Palette

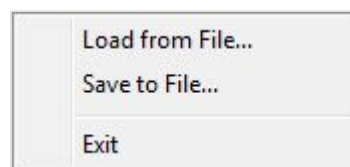
Contains a set of available components that you can put on the registration form. Main menu of the form allows to sho and Designed Form. Description of components' properties see below. To place component on the registration form clic component on the Component Palette and then click on the Registration Form to put component at necessary position.



There are following controls available:

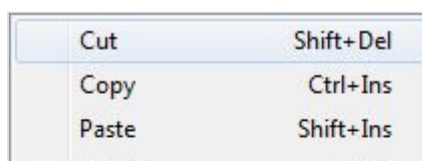
- | [TForm](#) - the registration form itself;
- |  [TEdit](#) control allows to enter registration name and key, show a hardware id or contain any user text;
- | **A** [TLabel](#) control allows to show any text;
- |  [TMemo](#) control is very similar to TEdit control but may have multiple lines of the text;
- |  [TGroupBox](#) control allows to group few controls on the form and also may contain a header text;
- |  [TButton](#) is a push button control that allows manage users actions like registering, closing application or cont
- |  [TSpeedButton](#) control is very similar to TButton, but also may contain an image;
- |  [TLink](#) control allows to open a browser window, default email client, open some text file in the e open/executed any other file;
- |  [TImage](#) control allows to draw a simple geometric shape on the form;
- |  [TShape](#) control allows to place an image to the form;
- |  [TBevel](#) control allows to create beveled boxes, frames, or lines;
- |  [TPanel](#) control allows to put an empty panel on a form. Panels have properties for providing a beveled control, as well as methods to help manage the placement of child controls embedded in the panel;

File menu



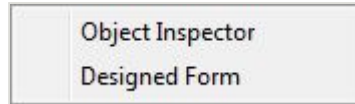
- | Load from File - allows to load registration form from external file;
- | Save to File - allows to save current registration form to external file;
- | Exit - exit designer;

Edit menu



- Copy - copy selected control(s) to clipboard;
- Paste - paste copied/cutted control(s) from clipboard to form;
- Delete - delete selected control(s) from the form;
- Select All - select all control(s) on the form;
- Bring to Front - bring selected control(s) to the front;
- Send to Back - send selected control(s) to the back;
- Align to Grid - align selected control(s) to the grid;

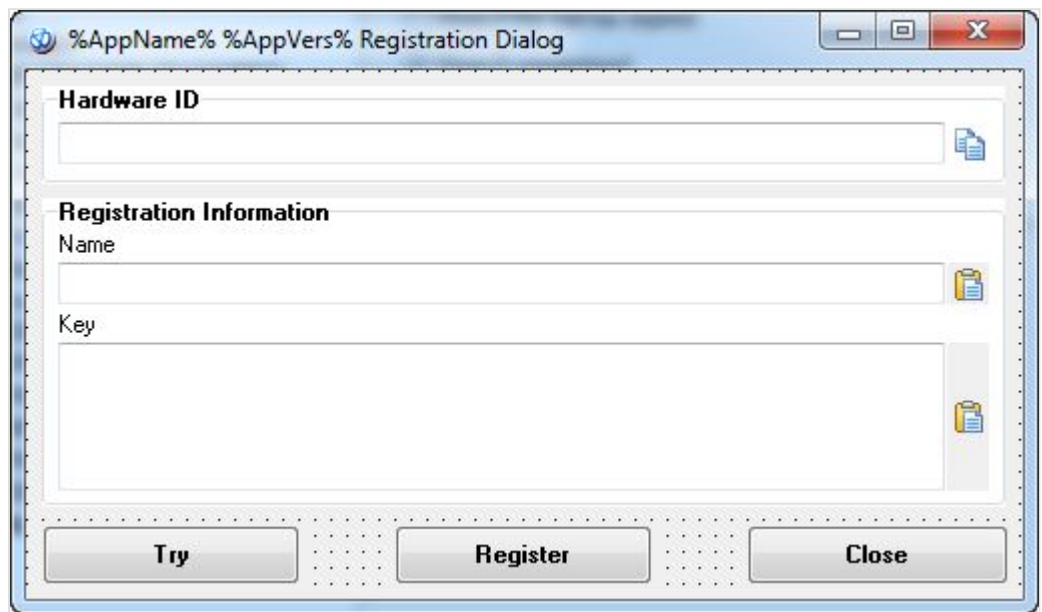
View menu



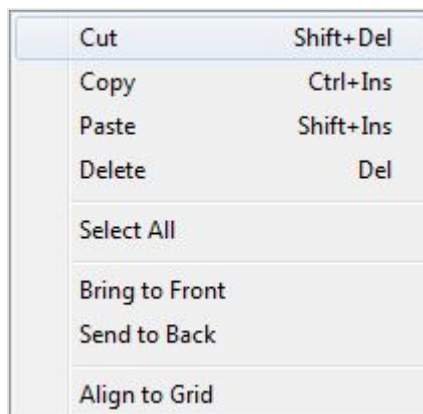
- Object Inspector - show Object Inspector;
- Designed Form - show designing registration form;

Registration Form

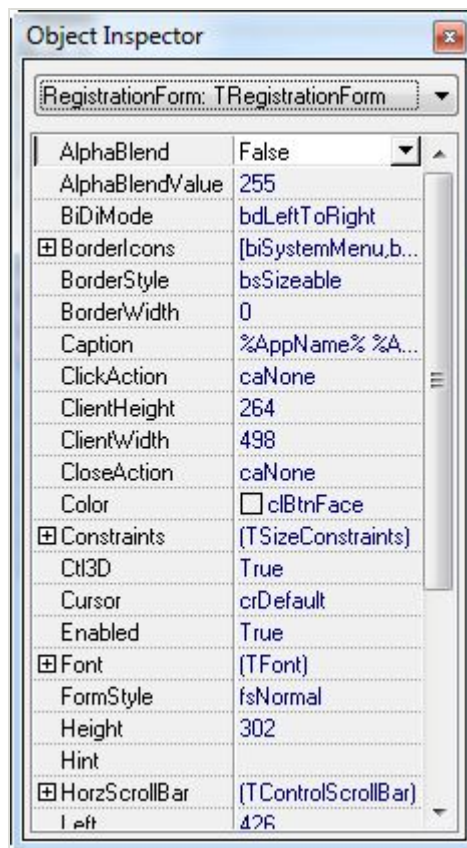
This is a form that will be shown to the users to allow enter registration information. Click onto the necessary component of this form and its properties will be shown in the Object Inspector. To delete component from the form click right mouse button on the component and choose Delete item in the appeared popup menu, or click Del button (Note: registration form could not be deleted).




Right mouse click shows registration form popup menu.




- Cut - cut selected control(s) to clipboard;
- Copy - copy selected control(s) to clipboard;
- Paste - paste copied/cutted control(s) from clipboard to form;



Click here to view TForm properties;

 Click here to view TEdit properties;

 Click here to view TLabel properties;

 Click here to view TMemo properties;

 Click here to view TGroupBox properties;

 Click here to view TButton properties;

 Click here to view TSpeedButton properties;

 Click here to view TLink properties;

 Click here to view TImage properties;

 Click here to view TShape properties;

 Click here to view TBevel properties;

 Click here to view TPanel properties;

显示密钥过期提示

开始显示密钥过期提示的天数

30

编辑消息

启用反调试器保护

进程结束时显示提示消息

编辑消息

--

SoftIce, OllyDbg, TWD

--

"

"

启用校验和保护

运行时文件完整性检查

进程结束时显示提示消息

编辑消息

使用密码启动

动作

总是询问密码

只在第一次询问密码

按天数询问 30

附加选项

星号显示密码

允许更换密码

密码

密码	注释
123	密码备忘：123

添加 编辑 删除

编辑消息

“*”

密码编辑器

密码：
123

注释：
密码备忘：123

确定 取消

启用文件名称保护

原文件名称：

进程结束时显示提示消息

启用磁盘类型检测

磁盘类型	运行权限
软盘	允许
硬盘	允许
网络驱动器	允许
光驱	允许
闪存	允许

进程结束时显示提示消息 编辑消息

--

:

| ():

| .

| .

| .

-

" "

启用进程实例数量限制

允许的进程实例数量：

结束进程

如果进程实例数量超标，显示提示信息

EP_CheckupCopies Enigma API

Enigma

SDK



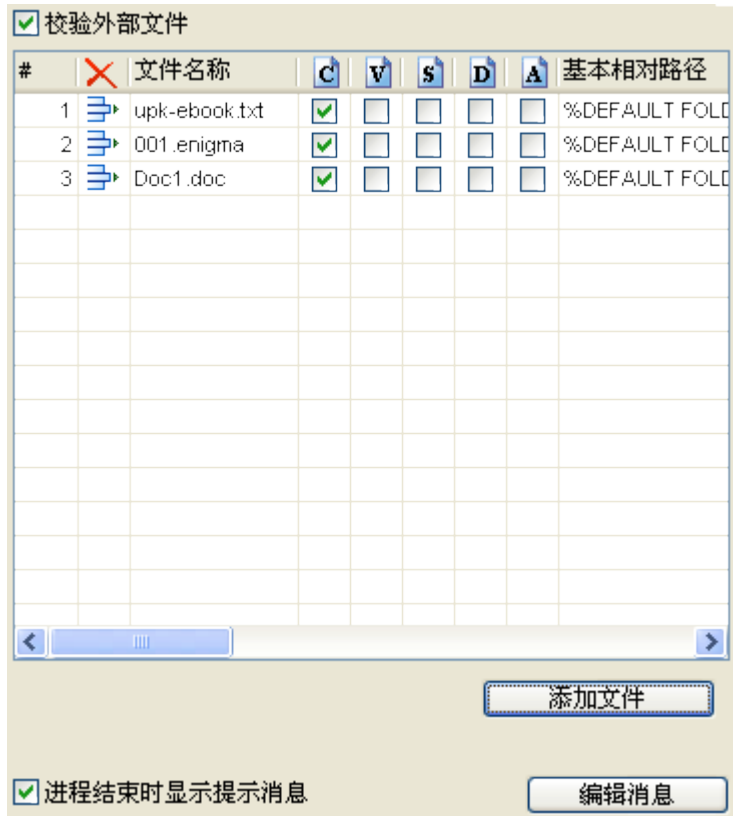
- / " "

- / /

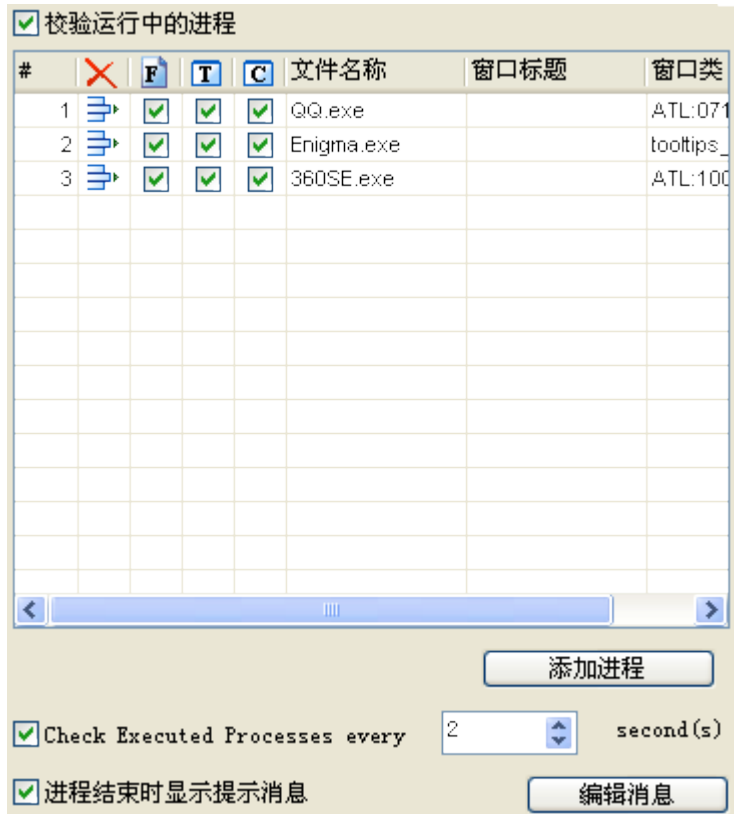
- / /

-

- " "



" " - :
 | |
 | File checksum -
 | File Version - Win32
 | File Size -
 | File Date -
 | File Attributes -
 " " -
 " " - " " %
 Examples " " %
 DEFAULT FOLDER% " " Examples.
 -
 - Enigma



Enigma

OllyDbg

SnagIt

Enigma

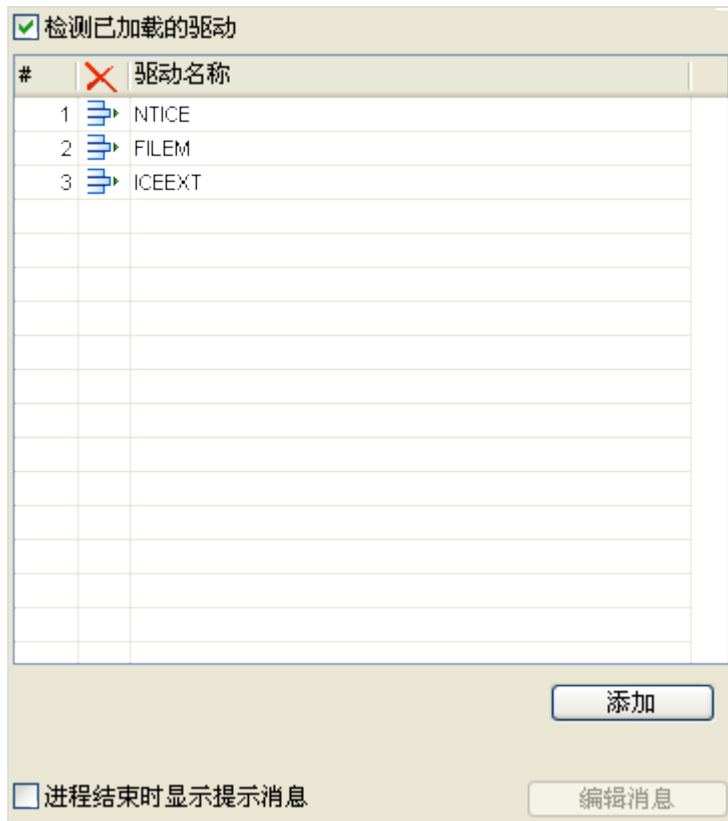
- | ☞ Delete process from the list
- | F File Name -
- | T Window Text -
- | C Window Class -

进程	窗口	文件名称	窗口标题	窗口类	当前文
00000708	0049053E	Explorer.EXE		tooltips_class32	Explorer.E
00000708	00150980	Explorer.EXE		tooltips_class32	Explorer.E
00000708	0006008E	Explorer.EXE		BaseBar	Explorer.E
00000708	00180460	Explorer.EXE		Auto-Suggest Dr...	Explorer.E
00000708	00120364	Explorer.EXE		Auto-Suggest Dr...	Explorer.E
00000708	00490278	Explorer.EXE		Auto-Suggest Dr...	Explorer.E
00000708	000F0338	Explorer.EXE		Auto-Suggest Dr...	Explorer.E
00000708	000A0302	Explorer.EXE		Auto-Suggest Dr...	Explorer.E
00000708	0001006E	Explorer.EXE		tooltips_class32	Explorer.E
00000708	00030054	Explorer.EXE		Shell_TrayWnd	Explorer.E
00000708	00150688	Explorer.EXE	SysFader	SysFader	Explorer.E
00000708	0008079C	Explorer.EXE		tooltips_class32	Explorer.E
00000708	00090544	Explorer.EXE	SysFader	SysFader	Explorer.E
00000708	000409E0	Explorer.EXE		tooltips_class32	Explorer.E
00000708	0003040A	Explorer.EXE	SysFader	SysFader	Explorer.E
00000708	000F0982	Explorer.EXE		tooltips_class32	Explorer.E
00000708	0007052E	Explorer.EXE	SysFader	SysFader	Explorer.E
00000708	00010090	Explorer.EXE		tooltips_class32	Explorer.E
00000708	0007067E	Explorer.EXE	SysFader	SysFader	Explorer.E
00000708	0001007E	Explorer.EXE		tooltips_class32	Explorer.E

10

CPU

- Enigma



- " " " "

- Enigma





Windows 95, 98 and ME

" "

| Delete service from the list

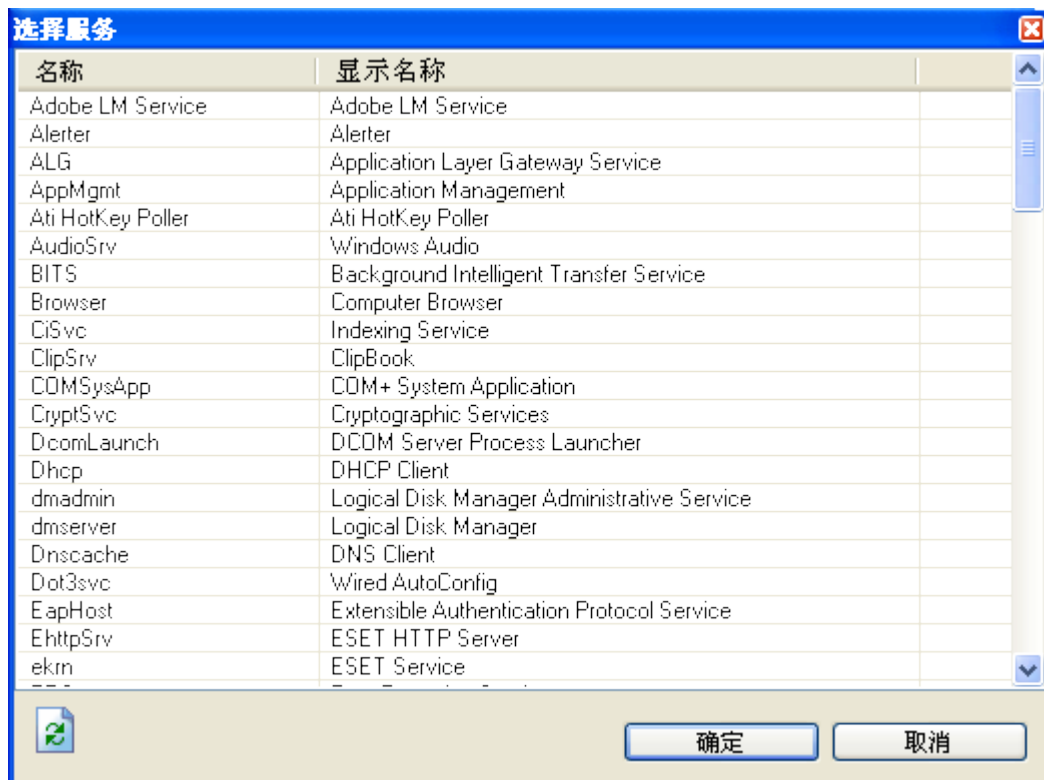
| Check Name -

| Check Display Name -

" "

" " -

-



- Enigma

Windows



Windows

-

Windows

Enigma

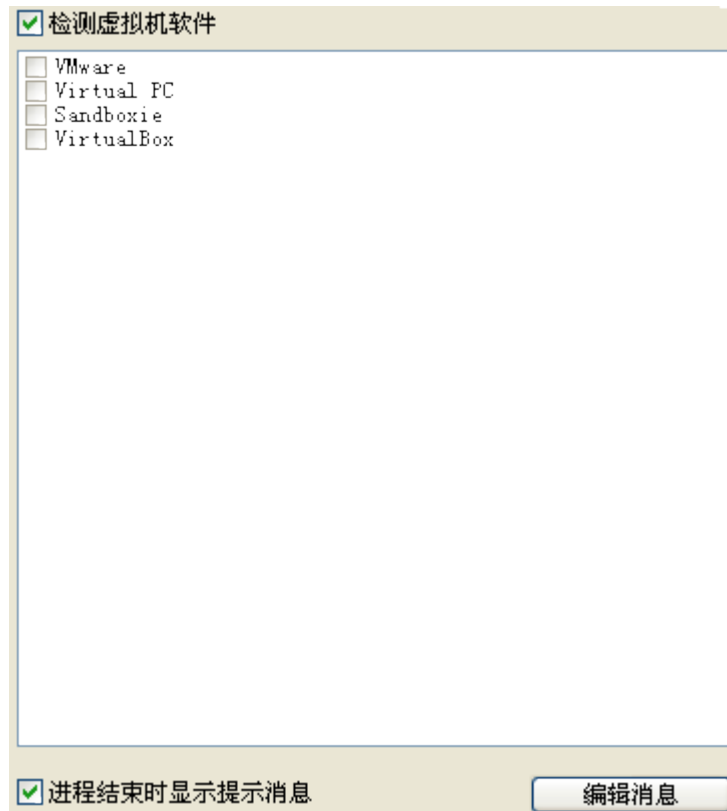
Windows

Windows

-

Enigma

Windows



Terminate Execution - if this option is enabled, Enigma Protector will terminate execution of the protected file if virtualization tools had been found. It is recommended to uncheck this option if you plan to use Enigma API function [EP_CheckupVirtualizationTools](#).

- Enigma

管理员权限

- 检测管理员权限
- 进程结束时显示提示消息

编辑消息

Windows NT

95/98/ME

Windows

- |
- |
- |
- |
- | WinAPI
- | WinAPI
- | Inline Patching
- |

区段名称：
data

入口伪装为：
Borland Delphi

PEiD

Enigma Protector

-

8

-

"None"

Visual Basic

"Microsoft Visual C++"

启用保留原文件大小功能

Enigma Protector

启用额外的资源保护

-

ResHack, Restorator

" " /"

启用高级输入表保护

-

Enigma Protector

启用 WinAPI 函数重定向功能

WinAPI

WinAPI

WinAPI

WinAPI

模拟标准 WinAPI 函数功能

WinAPI

Enigma

Enigma

WinAPI

Enigma API

启用内联补丁保护

选项

保护线程数量： 3

检测延时： 1000

-
CPU

1

3-5

15

-
CPU

Resources Protection


Select resources protection type

Protect all resources

Protect all resources, except icon

Protect all resources, except icon and version

Resources Protection allows to select types of resources in the executable file that will be protected. All resources in the file are being protected by default, but there you can enable/disable additional protection of ICON and VERSION resources.

- Protect all resources - all resources in the file will be protected;
- Protect all resources, except icon - do not protect ICON resources. Almost all executable files contain an ICON resources, default icon is displayed with the file is shown in the file explorer. Note, if the "Protect all resources" option is enabled, the default Windows icon  will be shown for this file in explorer.
- Protect all resources, except icon and version - disables protection of ICON and VERSION resources. Note, if any of two above options are enabled that means VERSION resources are protected, then the file version/information will not be shown in the properties dialog. Current option is recommended for most files.

Use first two options only if you exactly need protection of ICON and VERSION resources.

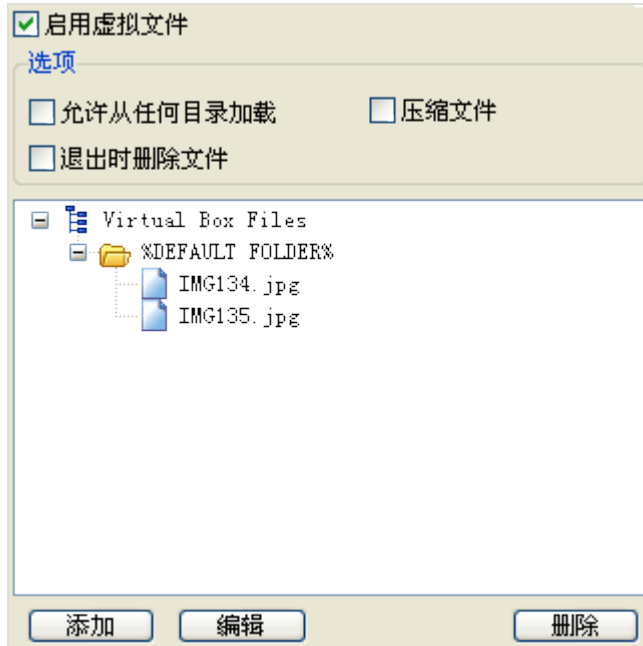
Note, following resources are not being protected also: STRINGS, MANIFEST, 'TYPELIB', 'REGISTRY'.

DLL/OCX

DLL/OCX

EXE

I



mp3 ocx

dll/OCX

dll/ocx

txt dll avi bpl

Warning: it is not recommended to use Virtual Box feature for protecting of Dynamic Link Libraries (DLL) files. Only DLLs that are calling once at main executable file start and never freed while application is working can be protected with Virtual Box. DLL files that are being called/freed multiple times per one session should not use Virtual Box, because it may cause application crash.

Enigma

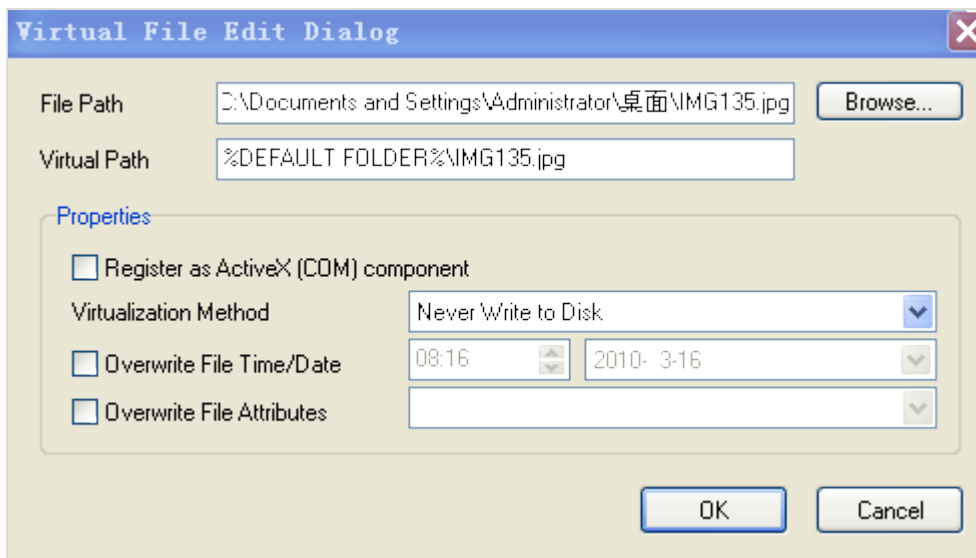
"Always Write to Disk

"Write if not Present

"

" Del"

/ / " "



-
-

" Browse "

- | %DEFAULT FOLDER% -
- | %SYSTEM FOLDER% - Windows System32 (WinNt) System (Win9X)
- | %WINDOWS FOLDER% - Windows
- | %My Documents FOLDER% - " "
- | %Program Files FOLDER% - " "
- | %Program Files\Common FOLDER% - " " " Common" " Program Files\Common folder"
- | %AllUsers\Documents FOLDER% - " All Users\Documents folder"
- | %History FOLDER% - " "
- | %Cookies FOLDER% - " Cookies"
- | %InternetCache FOLDER% - " Internet"

- | %DEFAULT FOLDER%\mydll.dll " mydll.dll"
- | %DEFAULT FOLDER%\data\intro.avi " data"
- | %SYSTEM FOLDER%\key.dat " key.dat" " c:\Windows\System32\"
- | C:\myfile.key C myfile.key

ActiveX COM - Enigma ActiveX COM

- | Never Write to Disk-
- | Always Write to Disk -
- | Write if not Present -

Overwrite File Date/Time - /

Overwrite File Attributes - / / /

Virtual Machine

Virtual Machine is a modern virtualization technology that allows to protect parts of the assembler source code of your module. Main idea of virtualization is converting original assembler code (which is well known for reverse engineers) to the PCODE - special programming language known only for Enigma Protector. When protected application requires to run virtualized code, the PCODE will be run on the internal Virtual Processor. Virtual Machine is very useful feature that allows to make reverse engineering/ analysing of the protected module very complex, our recommendation - using virtualization technology as much as possible. Also, note that not all code parts/functions should be virtualized. If the part of code or function executes too much times, it is not recommended to virtualize it, otherwise it slows down the work of your application and increase CPU loading. The better places for virtualization are parts of verifying registration code, trial control routines, cryptographic algorithms, in short, all the weak places that should not be reversed/cracked/analysed.

There are few ways the virtualization technology could be implemented:

- | by means [VM Markers](#). If you are software developer, have the sources code of your application and is able to compile it, then the VM Markers is the better and recommended way to apply virtualization;
- | by using MAP file (MAP file - contains a list of functions with their addresses, it has same name as an input file chosen on [Input Panel](#) but have a .map extension. MAP file allows accurate analysing of the input file), selecting necessary functions. This is also a choice for software developers who are able to compile modules and generate MAP files. To learn more how to create MAP files for different development IDEs, see [Making Of the Map File](#). Note, MAP file should be placed in the same folder with an input file and it should have same time span as an input file name. See [Functions Selecting](#);
- | third way, if you are not software developer and can't use two above methods then you may simply click Select Functions button and Enigma Protector will try to analyse the input file and find all functions that are possible to virtualize. Remember, this way is not so stable as a two previous, so you have double check workability of the protected modules. See [Functions Selecting](#).

Limitations:

- | Virtual Machine does not work with any kind of .NET files, either exe or dll files;
- | if the part of code/functions selected for virtualization contains any [Markers](#) inside, the workability of the marker will be lost, the code inside marker will be just executed;
- | for Delphi developers, the MAP file compiled with Delphi may contain a set of functions with the same names but different addresses. Enigma Protectors stores selected functions by the name in the project file. After loading of project file, the only first appeared function with necessary name will be selected;

Follow the links below for more information.

- | [File Entry Point](#)
- | [Functions Selecting](#)

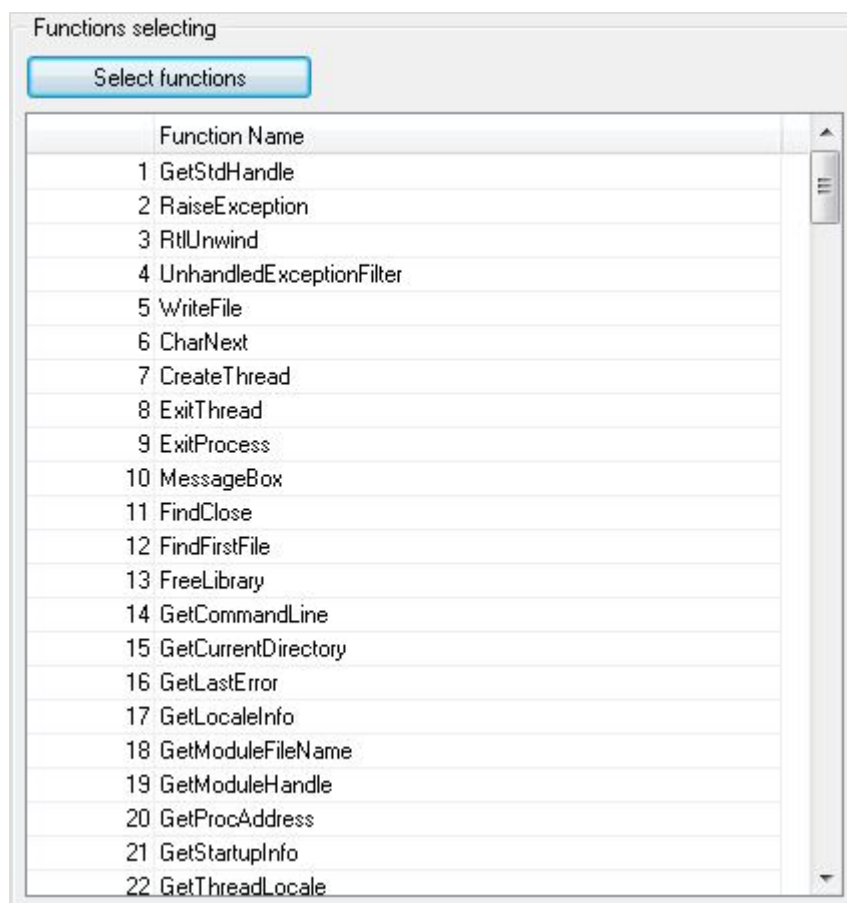
File Entry Point

Enable virtualization of file entry point

File Entry Point is a special function, starting with that the execution of any executable file begins. The file entry point function runs only one time per execution before any other code is executed.

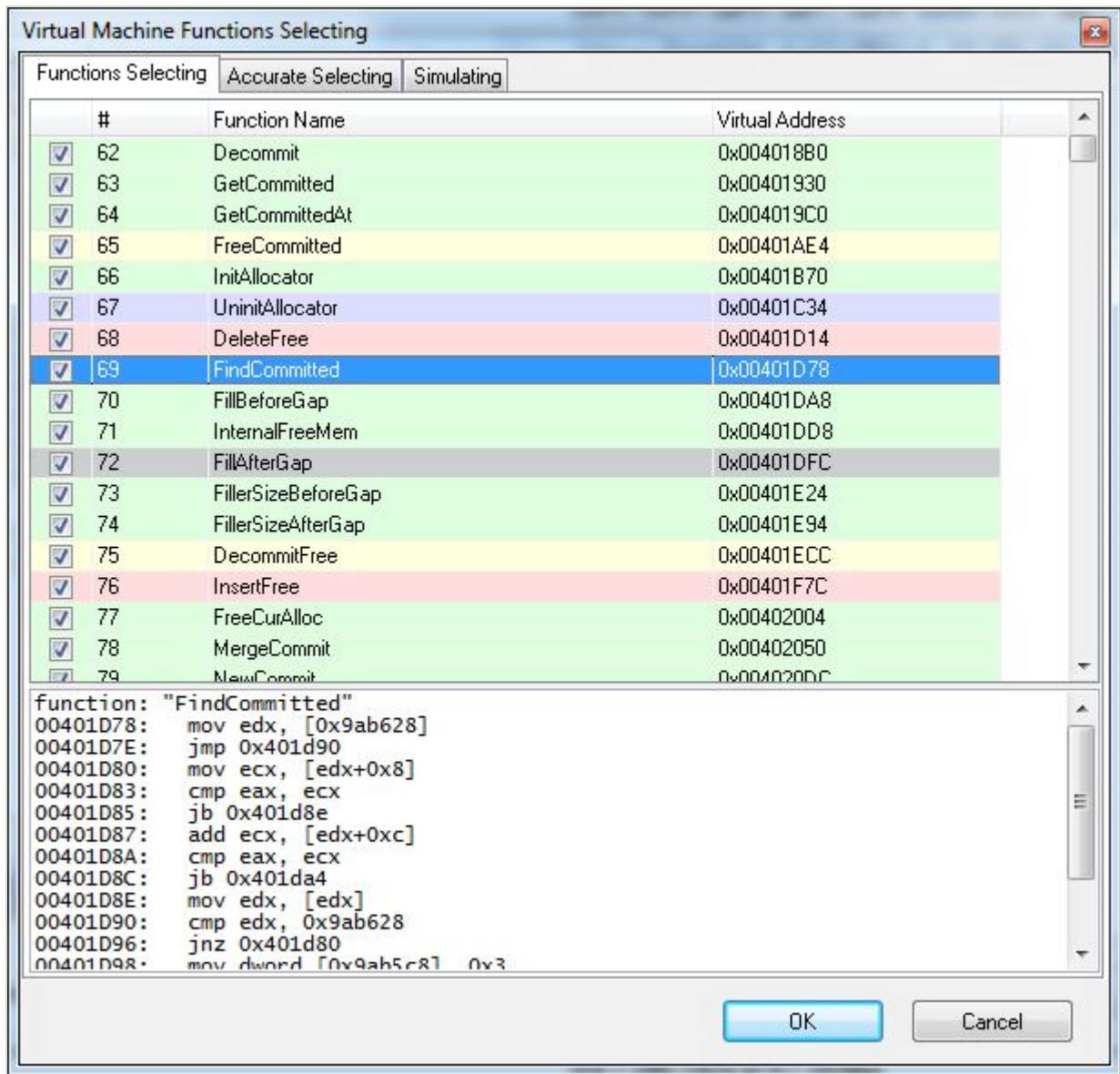
Enable virtualization of file entry point - enables virtualization of file entry point. It is always recommended to use this feature.

Functions Selecting

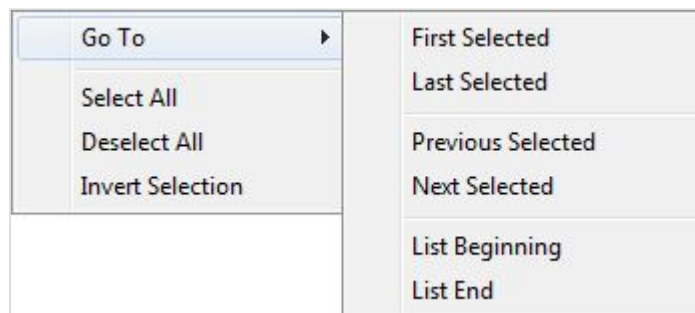


There is shown a list of functions names selected for virtualization. To change the list, click Select Functions button. The below windows with the possibility to select functions should appear. Note, if you have a MAP file for the Input file then the list of functions will be shown, otherwise, Enigma will try to analyse file itself and the functions names will be represented with their addresses.

Selecting Functions

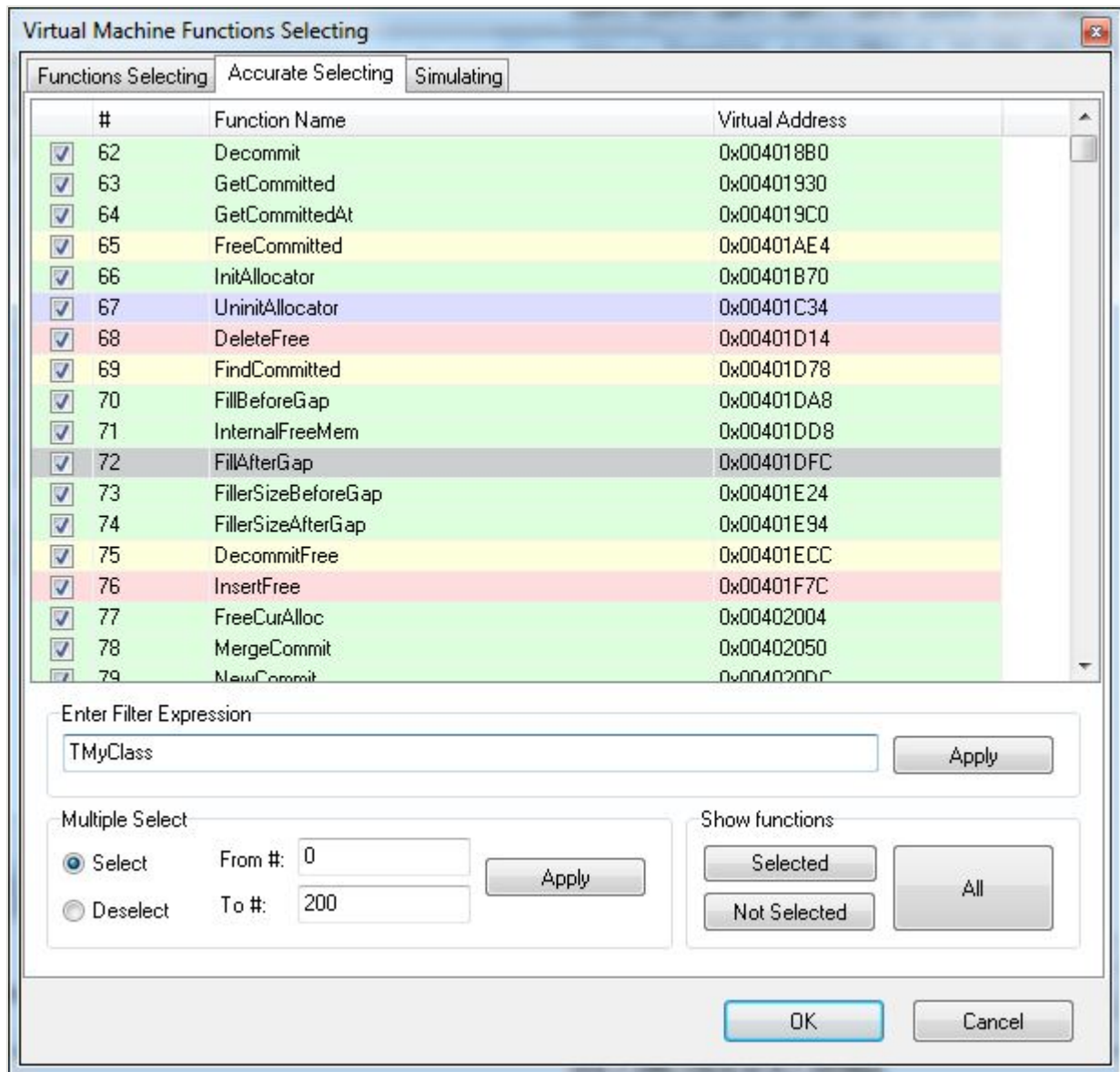


Select necessary functions for virtualization by checking them out. The information field at the bottom shows disassemble listing of selected function. For function coloring see below, Simulating topic. Right click on the list to show a popup menu.



- | Go To - First Selected - allows to scroll a list to the first checked function;
- | Go To - Last Selected - allows to scroll a list to the last checked function;
- | Go To - Previous Selected - selects the previous checked function;
- | Go To - Next Selected - selects the next checked function;
- | Go To - List Beginning - scrolls the list to the beginning;
- | Go To - List End - scrolls the list to the end;
- | Select All - checks all functions in the list;
- | Deselect All - unchecks all functions in the list;
- | Invert Selection - checks unchecked and unchecks checked functions;

Accurate Selecting



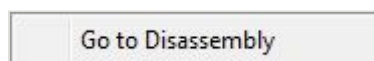
Accurate selecting allows few tools for easier navigation through the list of functions and their selecting.

Enter Filter Expression - enter the text to sort selected functions. Only functions contained entered text will be shown after Apply button click. This tool is very useful when only members of particular class should be selected, just enter the class name, click Apply button and only members of this class will be shown.

Multiple Select - allows select and deselect a range of functions.

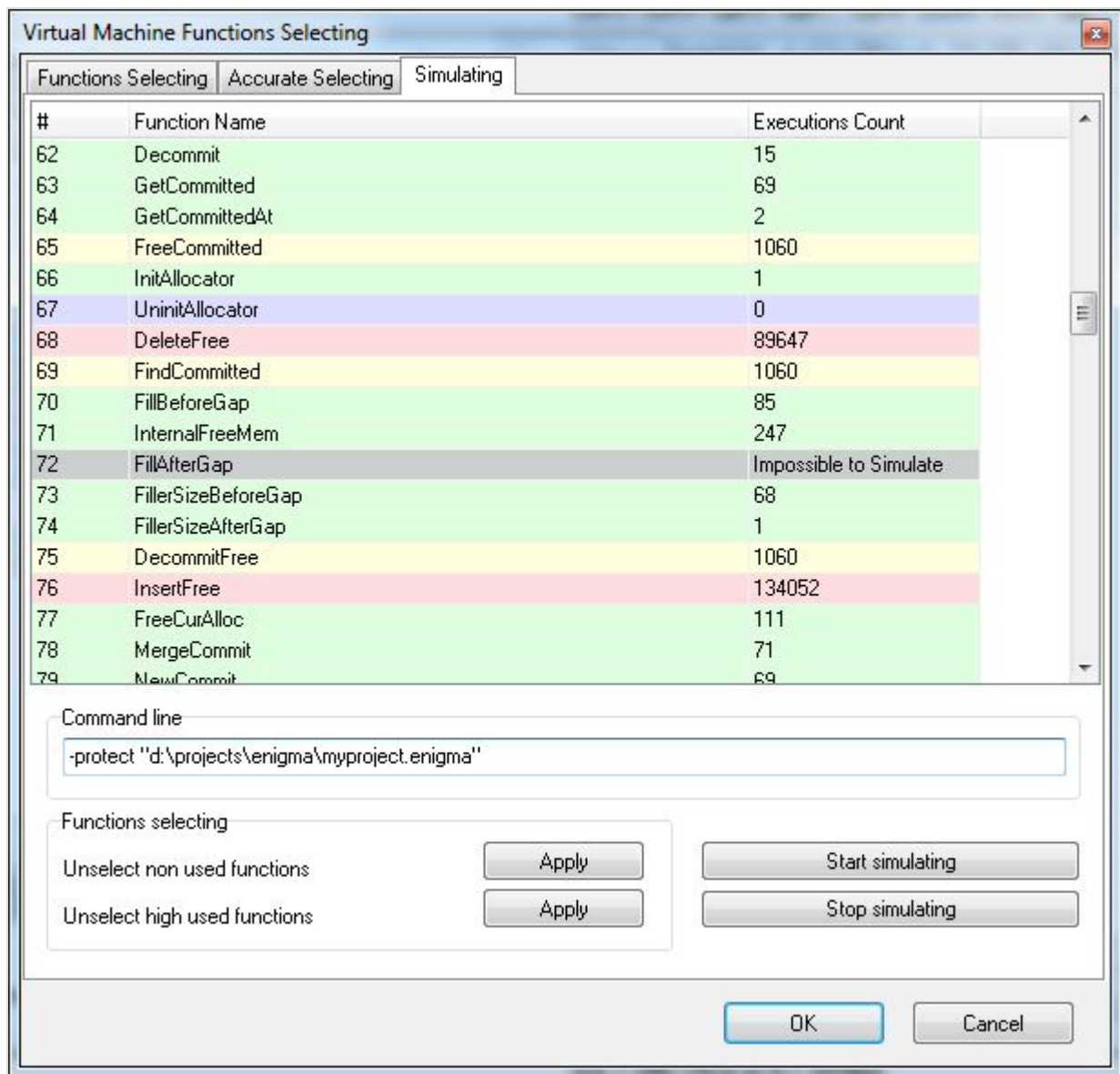
Show Functions - allows to show only Selected or Not Selected functions, or both.

Right click on the list to show a menu.



- Go to Disassembly - allows quick navigation to Functions Selecting tab, for viewing disassemble listing of the selected function

Simulating



Simulating allows to estimate how many times selected functions are executing in your application. While simulating Enigma Protector executes input file entered in [Input Panel](#), inserts some counters into executed processes that allows to count number of times particular function had been executed. Enigma Protector reads counters each second while application is working.

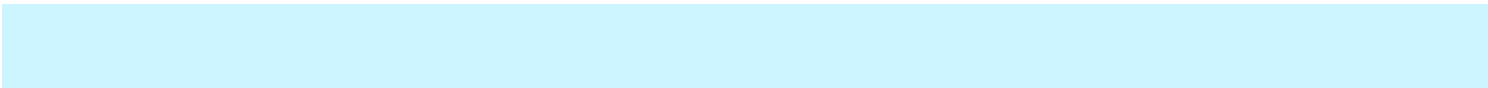
Click Start Simulating button to start simulating process. To stop simulating close executed application or click Stop Simulating button. If your application requires Command Line for some purposes, you enter this.

Note, this feature does not work with dll (COM/ActiveX etc) files, only exe files are supported.

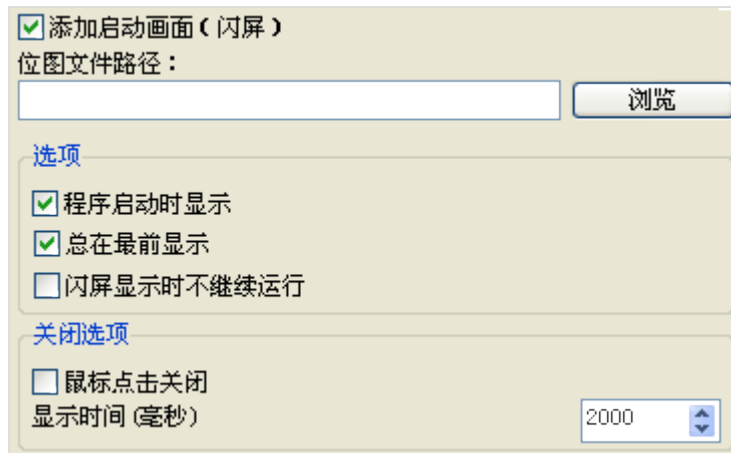
After simulating you may uncheck highly used or unused functions. But, unselecting these functions is not a rule, just an advice, if you think the functions should be virtualized, there is no difference how many times it had been executed while simulating. Also note that virtualization of highly used functions may slows down work of your application and increase CPU loading, and virtualization of non used functions may also be dangerous because there is no way to check out workability selected functions after protection.

Functions coloring:

- ▮ Gray color - Enigma Protector is not able to simulate execution of this function and know how many times it had been executed, or, there still was not attempt to simulate;
- ▮ Blue color - while last simulating the function was not executed;
- ▮ Green color - the number of times the functions was executed while last simulating is above zero and below the highly used value. Such functions are ideal for virtualization;
- ▮ Yellow color - the number of executions of this functions is near to the highly used. You will have to check out the working speed of your application after protection;
- ▮ Red color - function is highly used, it is not recommended to select this function, because it will slow down working speed of protected application.



|
|
|
|
|
|
|
|

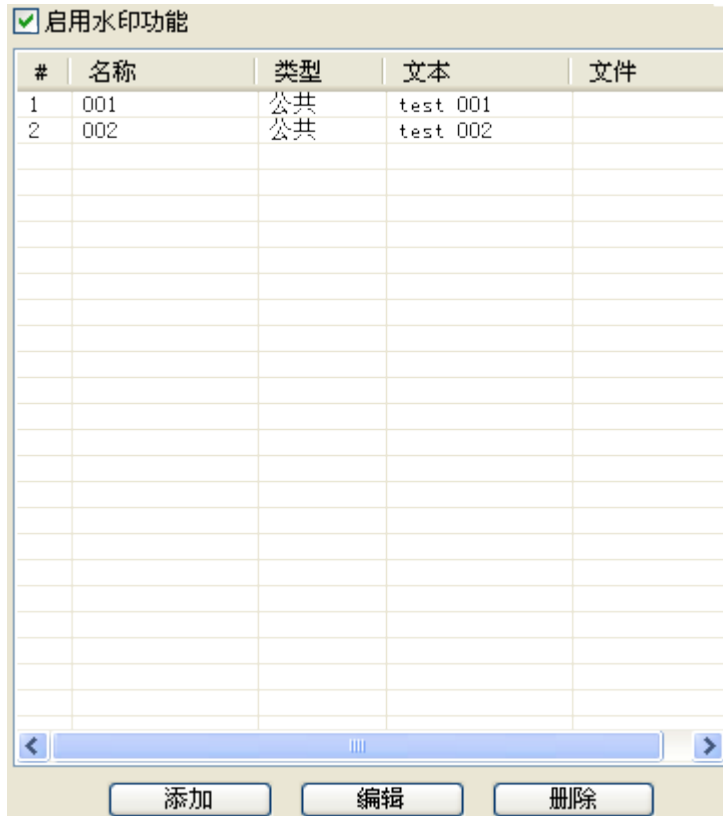


- Enigma Protector

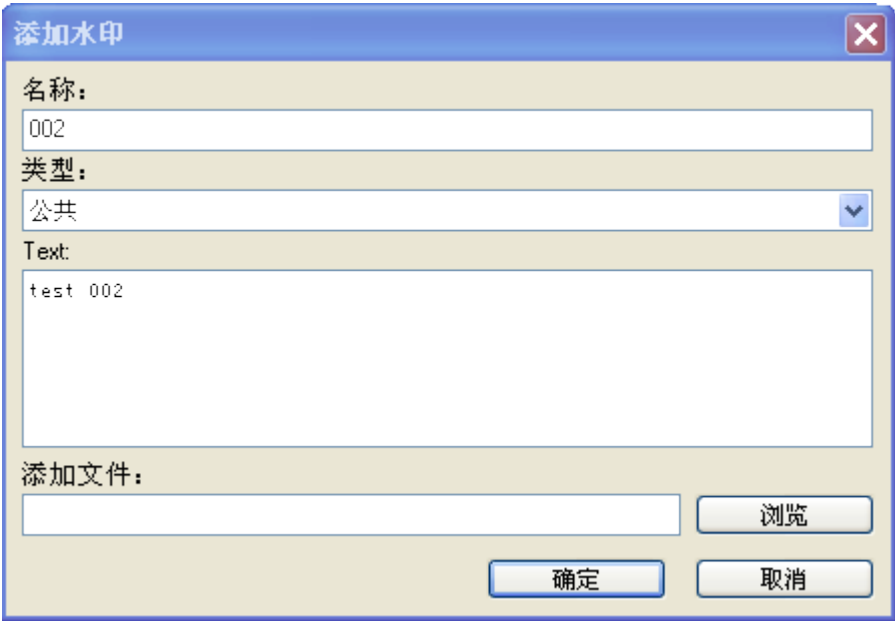
BMP, PNG, JPG (JPE, JPEG)

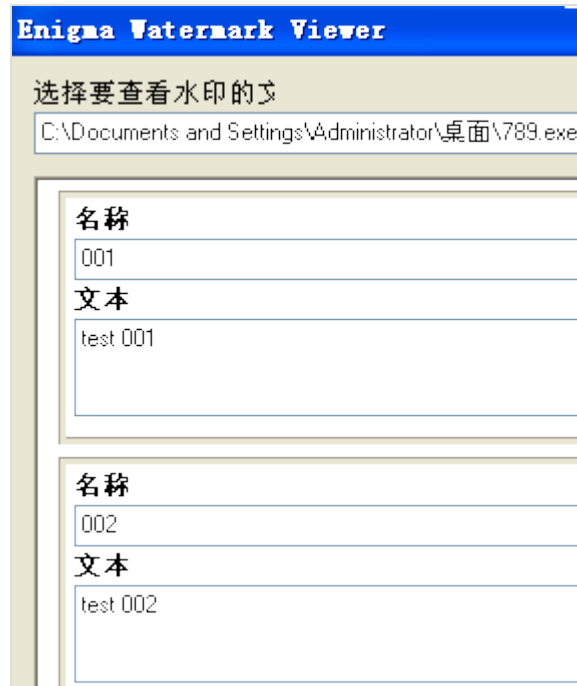
Enigma API [EP_SplashScreenShow](#)

[EP_SplashScreenHide](#)



Enigma Protector





[EP_MiscGetWatermark API](#)

双击插件查看描述

使..	名称	关于
<input type="checkbox"/>	checkremotedebugge...	Enigma anti-debugger plugin -
<input type="checkbox"/>	closehandle.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	debugobjects.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	emulatorsdetect.dll	Enigma anti-emulators plugin -
<input type="checkbox"/>	getstartupinfo.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	hardwarebreakpoint...	Enigma anti-debugger plugin -
<input type="checkbox"/>	heapcheck.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	hidecurrentthread.dll	Enigma hide process from debu
<input type="checkbox"/>	hideprocess.dll	Enigma hide process plugin -
<input type="checkbox"/>	int3.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	int3check.dll	Enigma anti-breakpoints plugi
<input type="checkbox"/>	isdebuggerpresent.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	isdebuggerpresentx...	Enigma anti-debugger plugin -
<input type="checkbox"/>	ntflags.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	ntforceflags.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	ntglobalflag.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	ntqueryinformation...	Enigma anti-debugger plugin -
<input type="checkbox"/>	outputdebugstring.dll	Enigma anti-debugger plugin -
<input type="checkbox"/>	sandboxiedetect.dll	Enigma Sandboxie Detect plugi
<input type="checkbox"/>	writeprocessmemory...	Enigma anti-WriteProcessMemor

Enigma Protector
Plugin
Enigma Protector

Enigma Protector
Enigma Protector\Plugins\

SDK

DLL
.NET

DLL

MS C++, Borland C++, MASM, Delphi

DLL

Enigma_Plugin_About - wide char (UNICODE)
Enigma_Plugin_Description - wide char (UNICODE)
Enigma_Plugin_OnInit -

" "

Enigma_Plugin_OnFinal -

Enigma_Plugin_OnSaveKey - function is calling when protected file is trying to save registration information. It can be used to skip standard saving routine to allow more flexibility to save registration information. Function has 4 parameters, ARegName and ARegKey contain pointers to registration name and key buffers and ARegNameLen and ARegKeyLen contain number of bytes of name and key buffers. Note, for handling of Wide Strings and UNICODE registrtration scheme you should divide ARegNameLen and ARegKeyLen values on 2 to get actual length of the string. This function should return FALSE if it is not saved registration information (or it is not needed to save it) and TRUE if function succeeded. Note, standard saving routine will not be called if this function returned TRUE.

Enigma_Plugin_OnLoadKey - function is calling when protected file is trying to load registration information. It can be used to skip standard loading routine to allow more flexibility to load registration information. Function has 4 output parameters. ARegName and ARegKey after successful loading should contain pointers to the buffers of registration name and key and ARegNameLen and ARegKeyLen should contain number of bytes of name and key buffers. Note, you have to allocate buffers for ARegName and ARegKey manually (somewhere in global space). For handling of Wide Strings and UNICODE registrtration scheme you should multiply lengths of the strings on 2. This function should return FALSE if it is not loaded registration information (or it is not needed to load it) and TRUE if function succeeded. Note, standard loading routine will not be called if this function returned TRUE.

增加自定义版本信息

版本号

主版本号: 1 次版本号: 0 发布: 0 构建: 0

程序属性

调试版本 特别版 DLL
 预发布版本 私人版

语言

地区 ID: 0x0409
English (United States)

密钥	值
CompanyName	The Enigma Protector Developers Team
FileDescription	Software Protection Tool
FileVersion	1.0.0.0
InternalName	ENIGMA.EXE
LegalCopyright	Copyrights (C) 2002-2009 Vladimir Su...
LegalTrademarks	Trademarks (R) 2002-2009 Vladimir S...
OriginalFilename	enigma.exe
ProductName	The Enigma Protector
ProductVersion	1.0.0.0
Comments	http://enigmaprotector.com/

常规 **版本** 兼容性

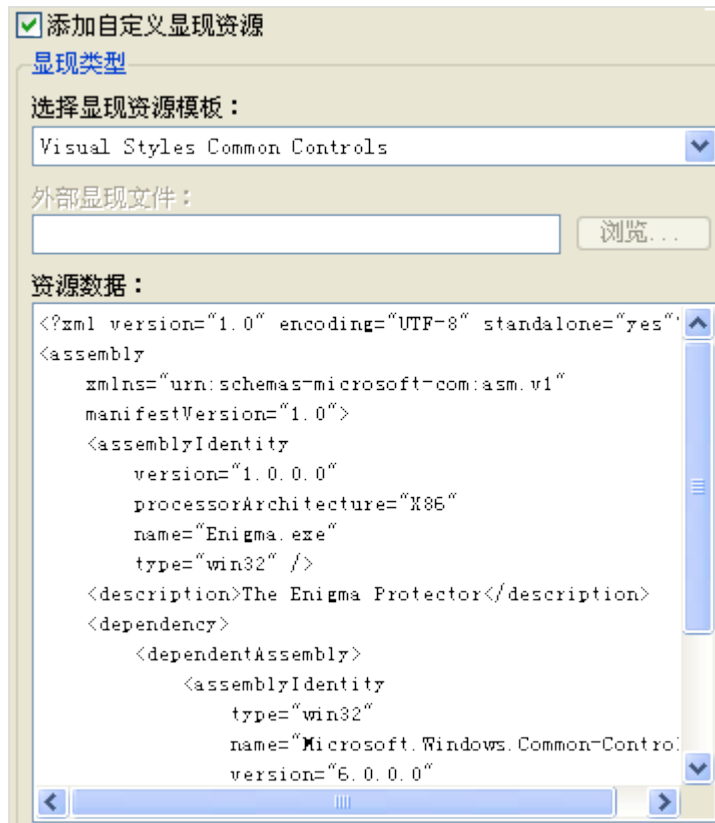
文件版本: 1.0.0.0

描述: Software Protection Tool

版权: Copyrights (C) 2002-2009 Vladimir Sukhov

其他版本信息

项目名称:	值:
备注	http://enigmaprotector.com/
产品版本	
产品名称	
公司	
合法商标	
内部名称	
文件版本	
语言	
源文件名	



- | Visual Styles Common Controls manifest allows to enable graphical themes of Windows XP/Vista for the protected file,
- | MS Visual Studio 2005/2008 manifest allows to check installation of Visual Studio 2005/2008 runtime libraries,
- | Each manifest can be manually edited and may contain information about product developer, it's description,
- | Manifest may help to define that the protected file should be run under administrative privileges and so on.
- | You may select own .manifest file that should be embedded into protected file.

The limitation there is only brain of developer!

已找到的标记

#	名称	虚拟地址

分析

属性

排除不分析的区段：

#	名称	原始地址	原始大小	虚拟地址	虚拟
<input type="checkbox"/>	.text	0x00001000	0x000B6000	0x00001000	0x0
<input type="checkbox"/>	.data	0x000B7000	0x00001000	0x000B7000	0x0
<input type="checkbox"/>	.rsrc	0x000B8000	0x00001000	0x000B9000	0x0

附加命令行参数

选项

附加参数

追加参数

输入命令行参数

/

-

-

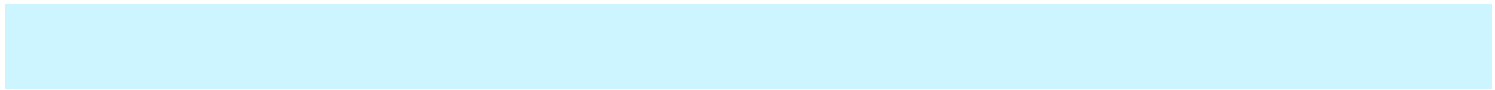

```
}
```

Visual Basic

```
Public Declare Function GetEnvironmentVariable Lib "kernel32.dll" Alias "GetEnvironmentVariableA" (ByVal Name
```

C# (.NET)

```
public class Enigma_EnvironmentVariables  
{  
    [DllImport("kernel32")] public static extern int GetEnvironmentVariable(string lpName, string lpBuffer, int  
}
```



|
|
|

API Enigma API Enigma API Examples Enigma
Enigma API API Enigma Protector

|
|
|
|
|
|
|
|
|
|

新版本重置试用期

过期时行为

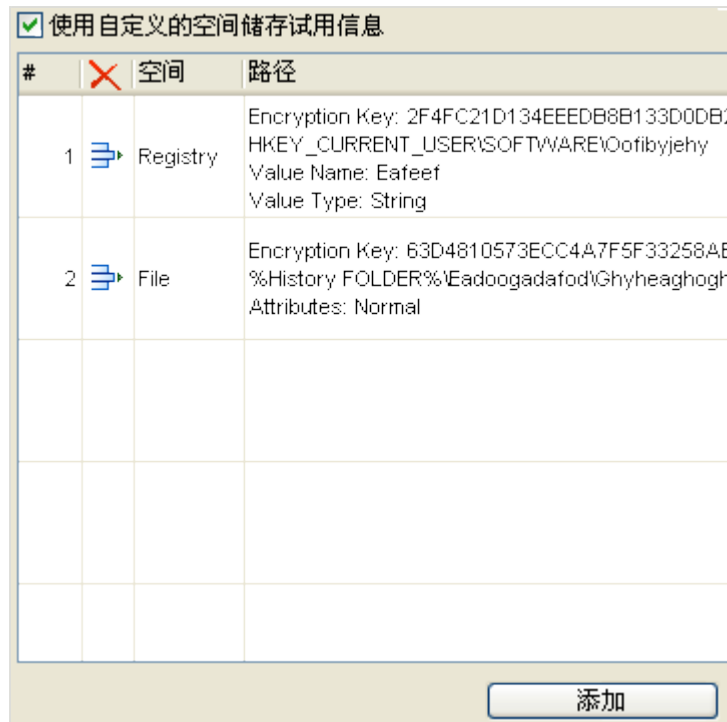
试用期过期时打开文件

文件名称：

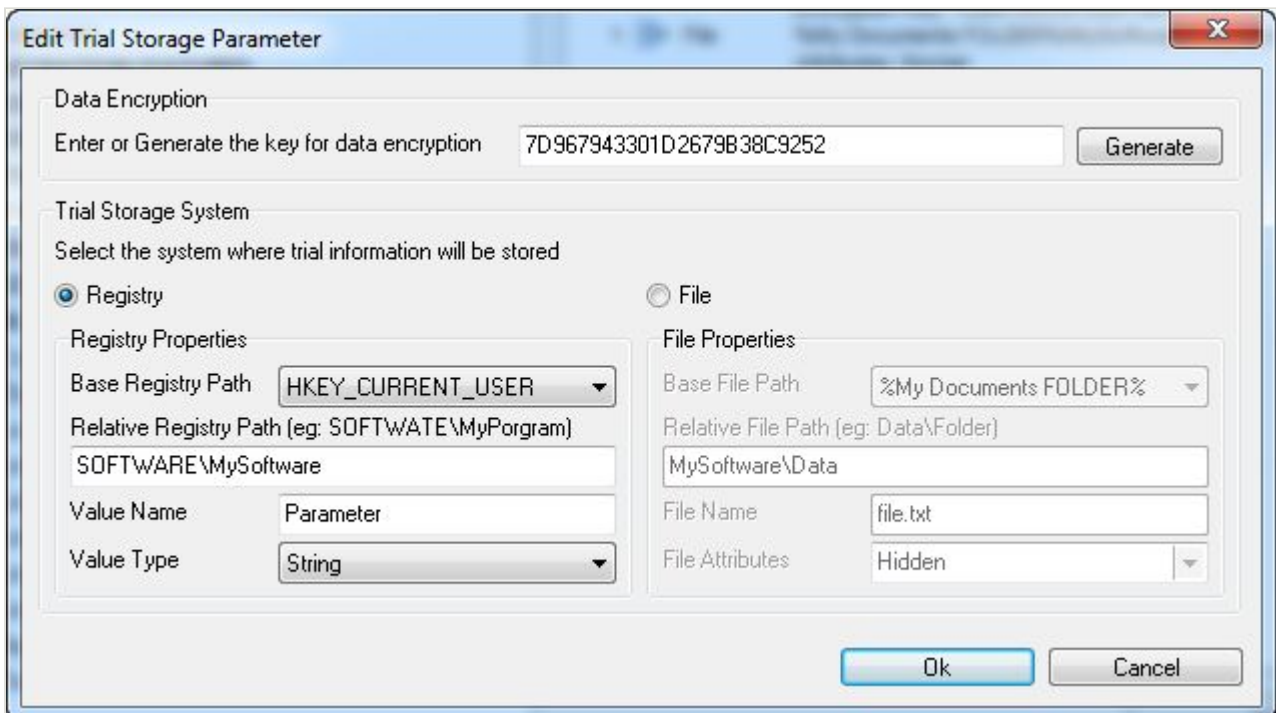
" " " "

Email

File Name field content	Description
http://www.mysite.com/	
mailto:admin@mysite.com	
License.txt	license.txt .txt
Relative\Register.exe	Register.exe
..\Register.exe	Register.exe
C:\Windows\Register.exe	c:\Windows Register.exe



Enigma

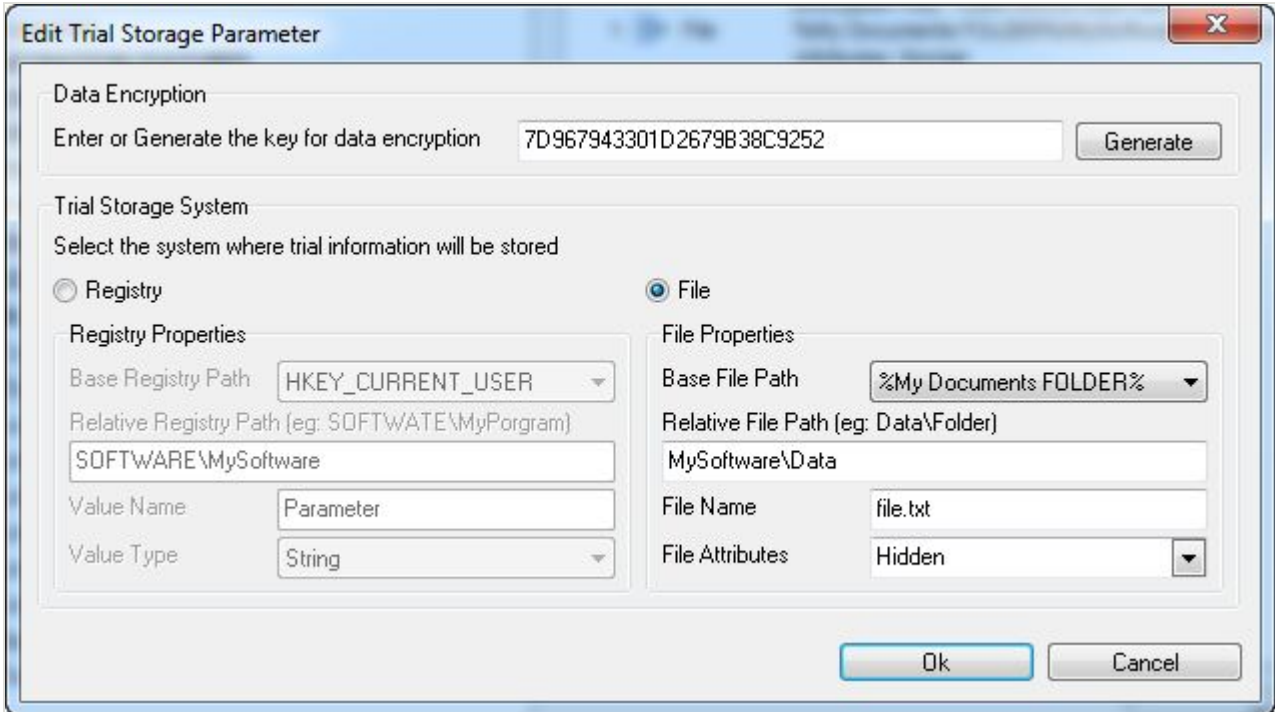


HKEY_CURRENT_USER

HKEY_LOCAL_MACHINE
HKEY_CLASSES_ROOT, HKEY_LOCAL_MACHINE, HKEY_USERS, HKEY_CURRENT_CONFIG

Value Name - name

Value Type -



%My Documents FOLDER%, %Program Files FOLDER%, %Program Files\Common FOLDER%, %AllUsers\Documents FOLDER%, %AllUsers\ApplicationData FOLDER%, %My Pictures FOLDER%, %History FOLDER%, %Cookies FOLDER%, %InternetCache FOLDER% Internet Explorer for Windows 95, Windows 98, Windows NT 4.0 %DEFAULT FOLDER%
%My Documents FOLDER%, %My Pictures FOLDER%, %History FOLDER%, %Cookies FOLDER%, %InternetCache FOLDER% Windows

File Name -

File Attributes - / / /

启用试用次数限制

最多试用次数：

过期时行为

在 x 秒后结束进程：

进程结束时显示提示信息

Windows

Enigma Protector

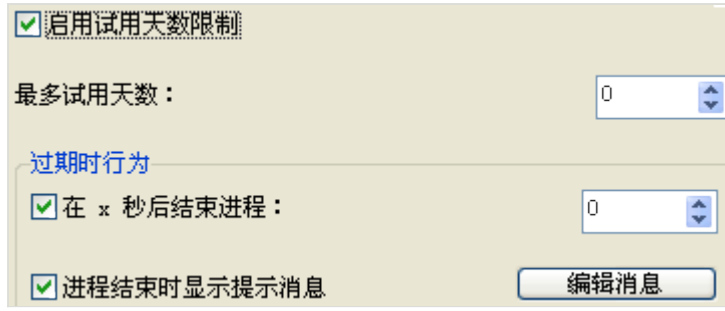
Enigma API

EP_TrialExecutions

Enigma Protector

x

x



-

Windows

Enigma Protector

Enigma API

EP_TrialDays

Enigma Protector

->

x

-

x

-

"

"

启用过期日期限制

过期日期： 2010-12-31

过期时行为

在 x 秒后结束进程： 0

进程结束时显示提示消息 编辑消息

- Enigma Protector

Enigma API
Enigma Protector

EP_TrialExecutionDate

->

x

-

x

-

"

"

启用从日期到日期限制

开始试用日期： 2010- 7- 1

试用结束日期： 2010-12- 1

过期时行为

在 x 秒后结束进程： 0

进程结束时显示提示消息 [编辑消息](#)

- Enigma

Enigma API
Enigma Protector

EP_TrialDateTillDate

x

-

x

-

"

"

启用运行时间限制

允许运行的总时间（分）：

过期时行为

过期时结束进程

进程结束时显示提示消息

Enigma

Enigma API EP_TrialExecutionTime
Enigma Protector

锁定试用到用户区域

选择允许试用的国家

<input checked="" type="checkbox"/>	Albania
<input checked="" type="checkbox"/>	Algeria
<input checked="" type="checkbox"/>	Argentina
<input checked="" type="checkbox"/>	Armenia
<input checked="" type="checkbox"/>	Australia
<input checked="" type="checkbox"/>	Austria
<input checked="" type="checkbox"/>	Azerbaijan
<input checked="" type="checkbox"/>	Bahrain
<input checked="" type="checkbox"/>	Belarus
<input checked="" type="checkbox"/>	Belgium
<input checked="" type="checkbox"/>	Belize
<input checked="" type="checkbox"/>	Bolivia
<input checked="" type="checkbox"/>	Brazil
<input checked="" type="checkbox"/>	Brunei Darussalam
<input checked="" type="checkbox"/>	Bulgaria
<input checked="" type="checkbox"/>	Cambodia
<input checked="" type="checkbox"/>	Canada
<input checked="" type="checkbox"/>	Central African Republic
<input checked="" type="checkbox"/>	Chad
<input checked="" type="checkbox"/>	Chile
<input checked="" type="checkbox"/>	People's Republic of China
<input checked="" type="checkbox"/>	Colombia
<input checked="" type="checkbox"/>	Costa Rica
<input checked="" type="checkbox"/>	Croatia
<input checked="" type="checkbox"/>	Czech Republic
<input checked="" type="checkbox"/>	Denmark
<input checked="" type="checkbox"/>	Dominican Republic
<input checked="" type="checkbox"/>	Ecuador
<input checked="" type="checkbox"/>	Egypt
<input checked="" type="checkbox"/>	El Salvador
<input checked="" type="checkbox"/>	Estonia

选择所有

不选所有

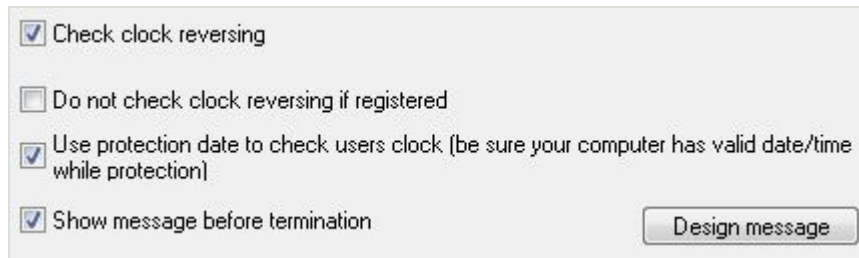
反向选择

启用提示消息

提示消息显示间隔（秒）：

[编辑消息](#)

Enigma Protector



The image shows a settings dialog box for Enigma Protector. It contains four checked options: 'Check clock reversing', 'Use protection date to check users clock (be sure your computer has valid date/time while protection)', and 'Show message before termination'. The option 'Do not check clock reversing if registered' is unchecked. A 'Design message' button is located at the bottom right of the dialog.

<input checked="" type="checkbox"/>	Check clock reversing
<input type="checkbox"/>	Do not check clock reversing if registered
<input checked="" type="checkbox"/>	Use protection date to check users clock (be sure your computer has valid date/time while protection)
<input checked="" type="checkbox"/>	Show message before termination

Design message

-

Enigma Protector

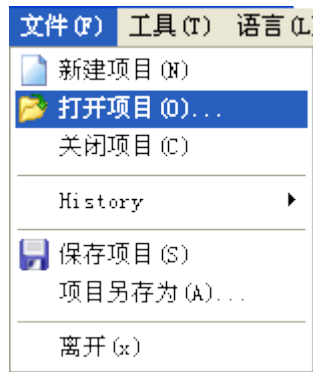
Do not check clock reversing if registered - use this to disable checking of the user time when application is registered. Please note, it is not recommended to disable clock reversing if the registration keys with (eg) expiration date are using.

Use protection date to check users clock - if the option enabled, Enigma Protector will also check if the current date on the user PC is not less than the date of protection. If the user PC date is less than date of protection then Enigma thinks the clock are back reversed. Please note, be sure your system clock (and Time Zone information) are correct.

- " "

" "

" ..."



" "



" "



" ..."



Enigma Protector

Enigma Protector " "



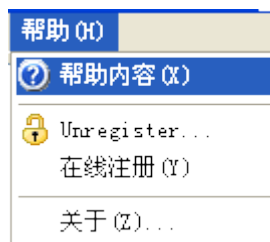
" "

Enigma Protector

\lang



Enigma Protector



/

/ Enigma Protector



输入注册名称：

输入注册密钥：

取消注册 确定 取消

Enigma Protector

Enigma Protector

Enigma Protector



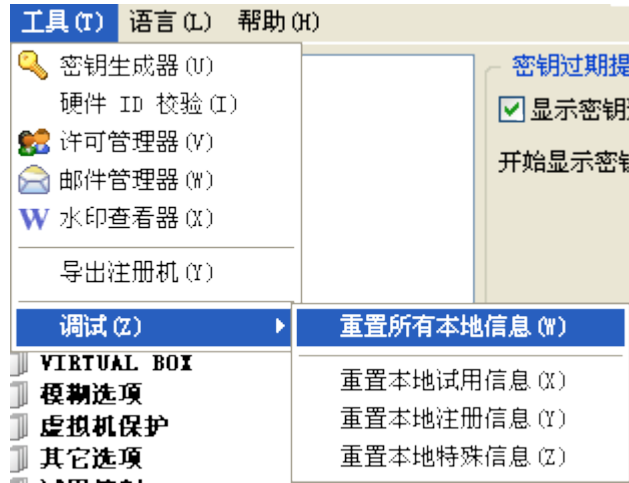
Enigma Protector



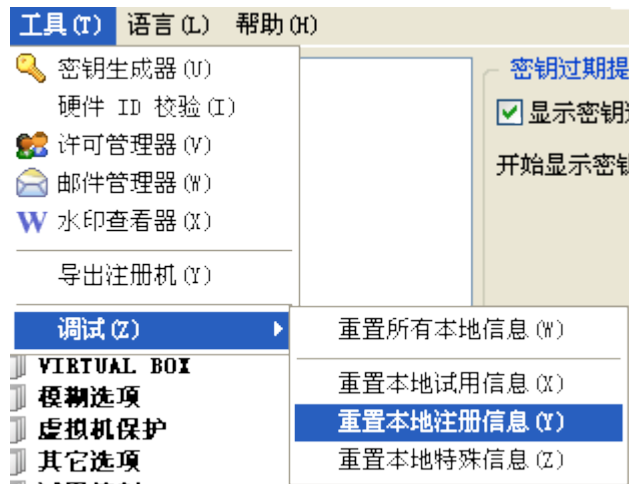
Enigma Protector



Enigma Protector



Enigma Protector



- [-] 项目设置
 - [-] 项目细节
 - [-] 输入参数
 - [-] 高级输入
 - [-] 输出参数
 - [-] 压缩选项
- [-] 注册特性
 - [-] 常规选项
 - [-] 密钥特性
 - [-] 注册信息存储
 - [-] 硬件锁定
 - 使用硬件 ID 加密
 - 注册对话框
 - 密钥过期提示
- [-] 检测选项
 - 反调试器保护
 - 校验和保护
 - 启动密码保护
 - 文件名称
 - 磁盘驱动器保护
 - 进程实例数量限制
 - 用户区域限制
 - 外部文件
 - 运行中的进程
 - 已加载的驱动
 - 已安装的服务
 - Windows 版本
 - 虚拟机软件
 - 权限设置
- [-] 保护特性
 - [-] 欺骗文件分析工具
 - 保留原文件大小
 - 额外的资源保护
 - 高级输入表保护
 - WinAPI 重定向
 - 模拟标准 WinAPI
 - 内联补丁
 - [-] 字符串加密
- [-] VIRTUAL BOX
 - 文件
- [-] 模糊选项
 - 入口点模糊



%App_Name%		
%App_Vers%		
%CU_EXTFILES%		
%CU_EXECPR%		
%CU_INSTSERV%		
%CU_WINVER%	Windows	Windows
%CU_VIRTTOOLS%		
%HardwareID%	ID	
%RegName%		Keys generator
%RegKey%		
%KeyExpYear%		
%KeyExpMonth%		
%KeyExpDay%		
%TrialExecsTotal%		
%TrialExecsLeft%		
%TrialDaysTotal%		
%TrialDaysLeft%		

%TrialExpYear%		
%TrialExpMonth%		
%TrialExpDay%		
%TrialStartYear%		
%TrialStartMonth%		
%TrialStartDay%		
%TrialEndYear%		
%TrialEndMonth%		
%TrialEndDay%		
%TrialExecMinsTotal%		
%TrialExecMinsLeft%		

-

X -

X OK - " OK/ "

-

-

MAP

MAP

Enigma Protector

Delphi

Borland C++

MS Visual Studio

MAP

[Delphi up to version 7](#)

[Delphi 2007 and newer \(2009, 2010\)](#)

[Borland C++ Builder up to version 7](#)

[Microsoft Visual C++ up to 6.0](#)

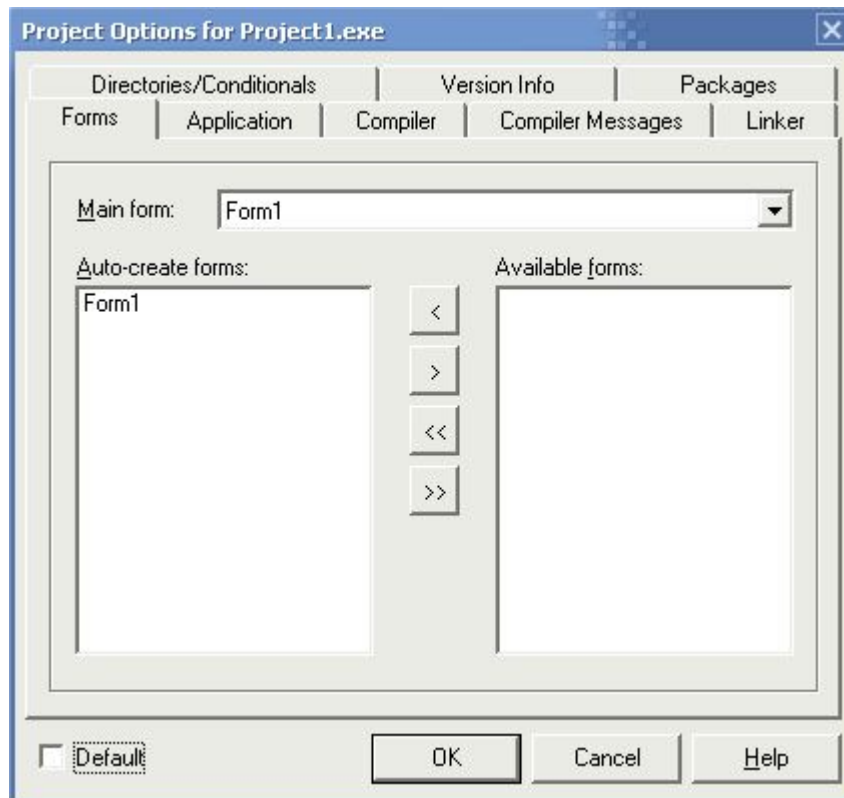
[Microsoft Visual Studio 2003 and newer \(2005, 2008\)](#)

Delphi

MAP

| Delphi

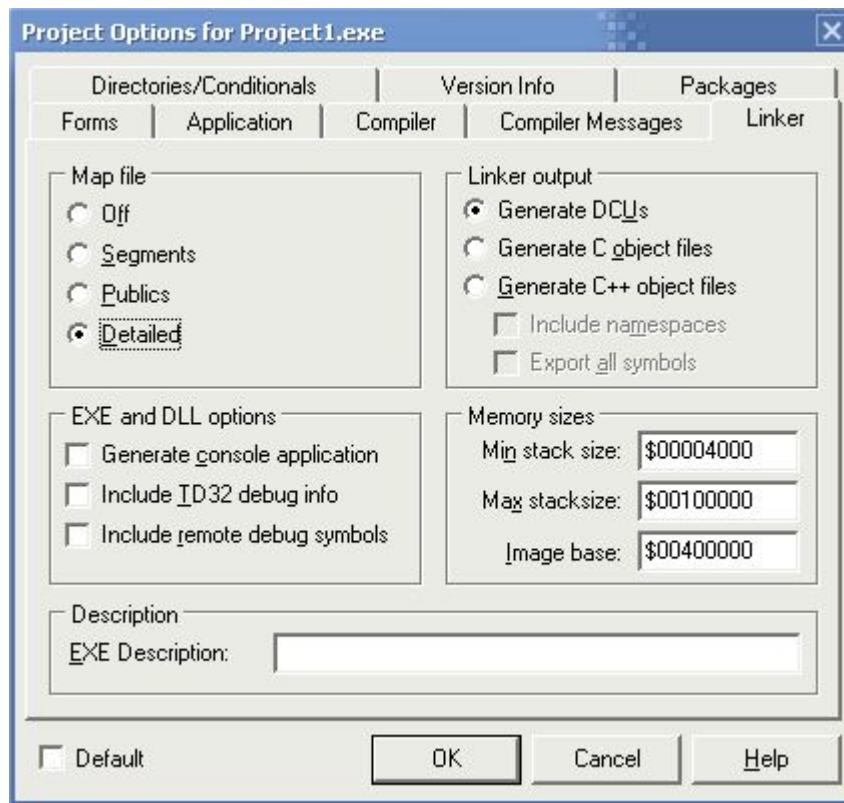
| "Project" -> "Options" Shift+Ctrl+F11



| "Linker"

"Map File"

"Details"



.map

.map

Delphi 2007

Delphi

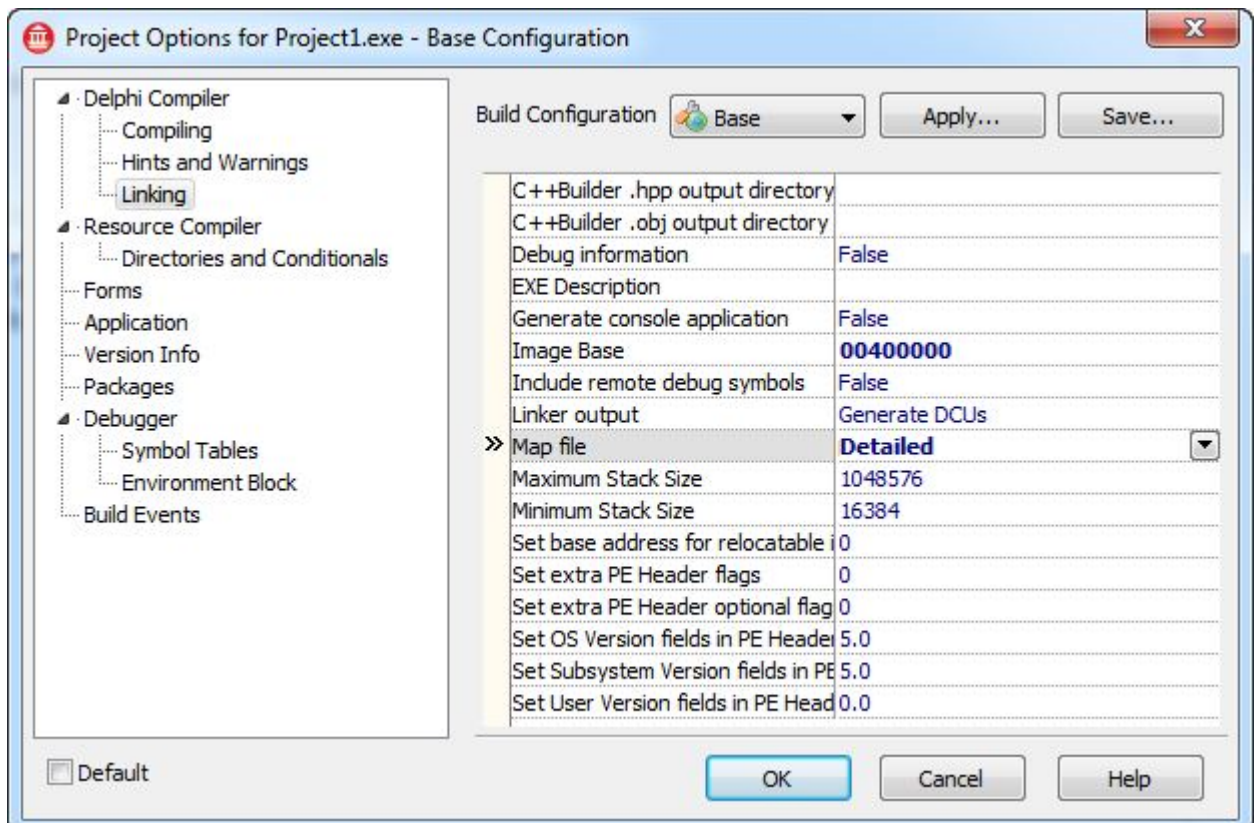
"Project" -> "Options"

Shift+Ctrl+F11

"Linker"

"Map file"

Detailed"

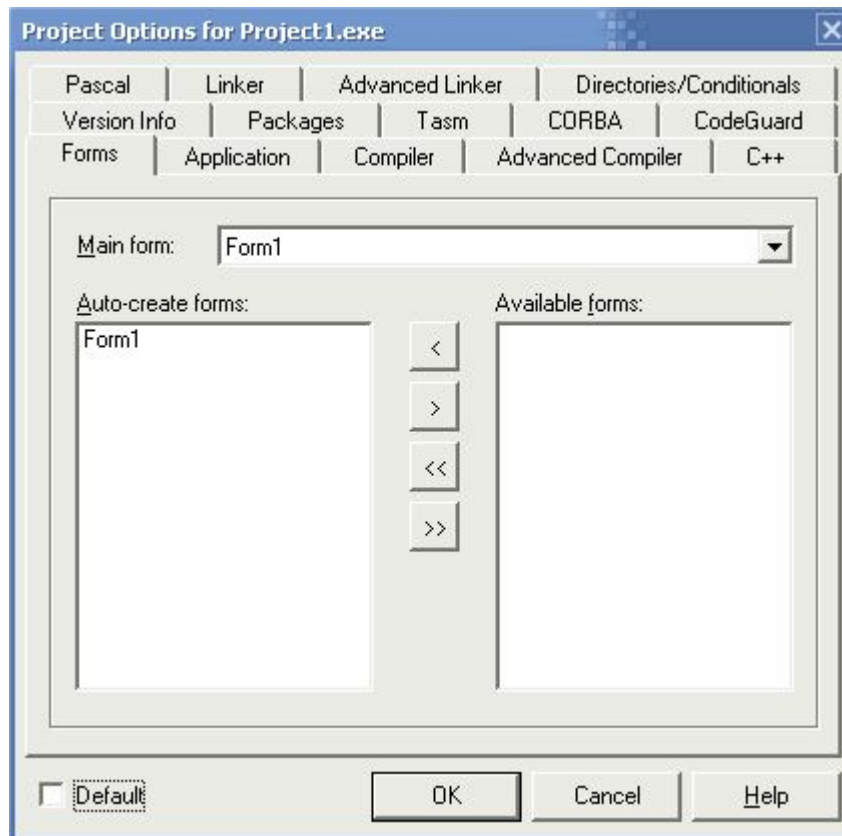


.map

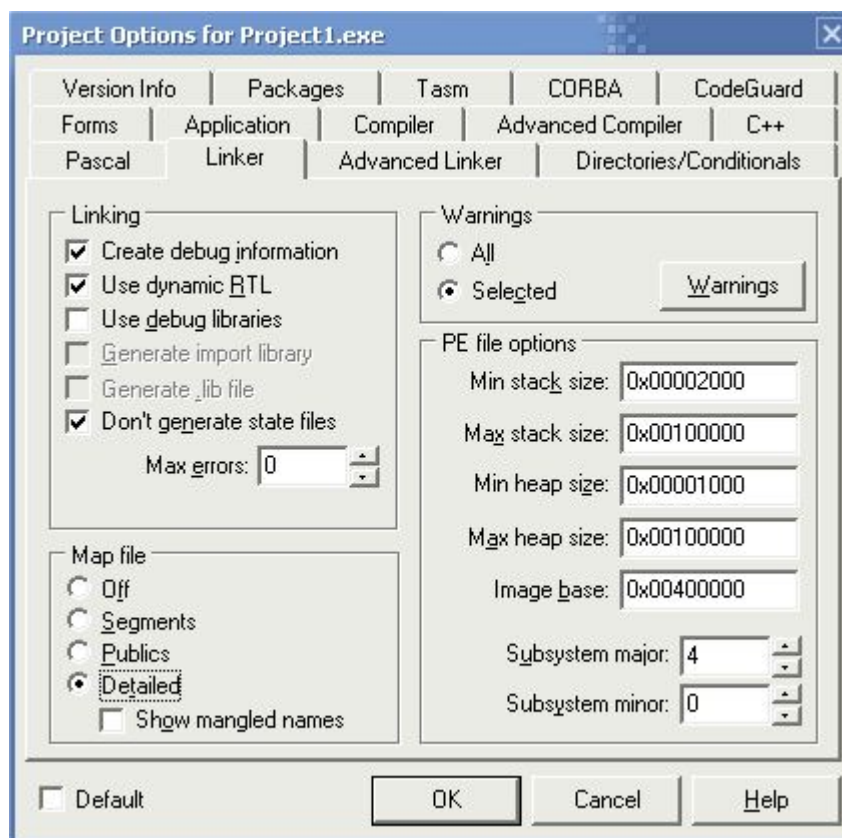
.map

Borland C++ Builder

MAP



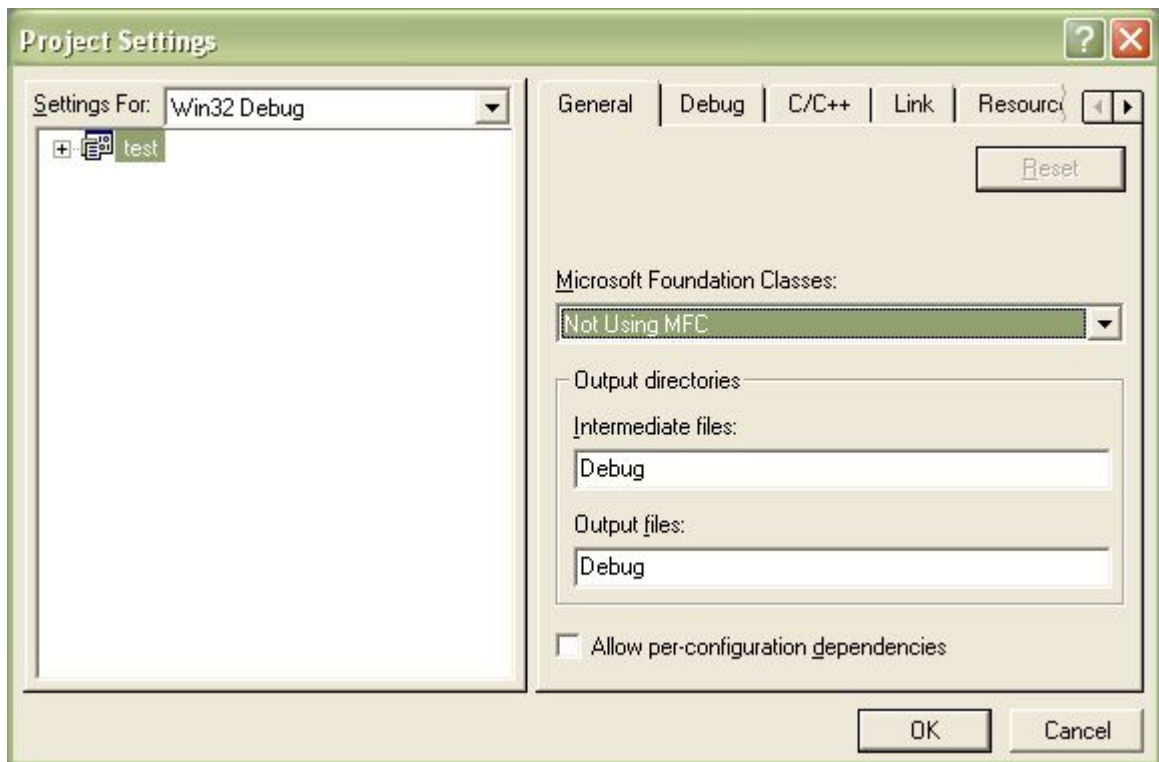
1 "Linker" "Map File" "Details"



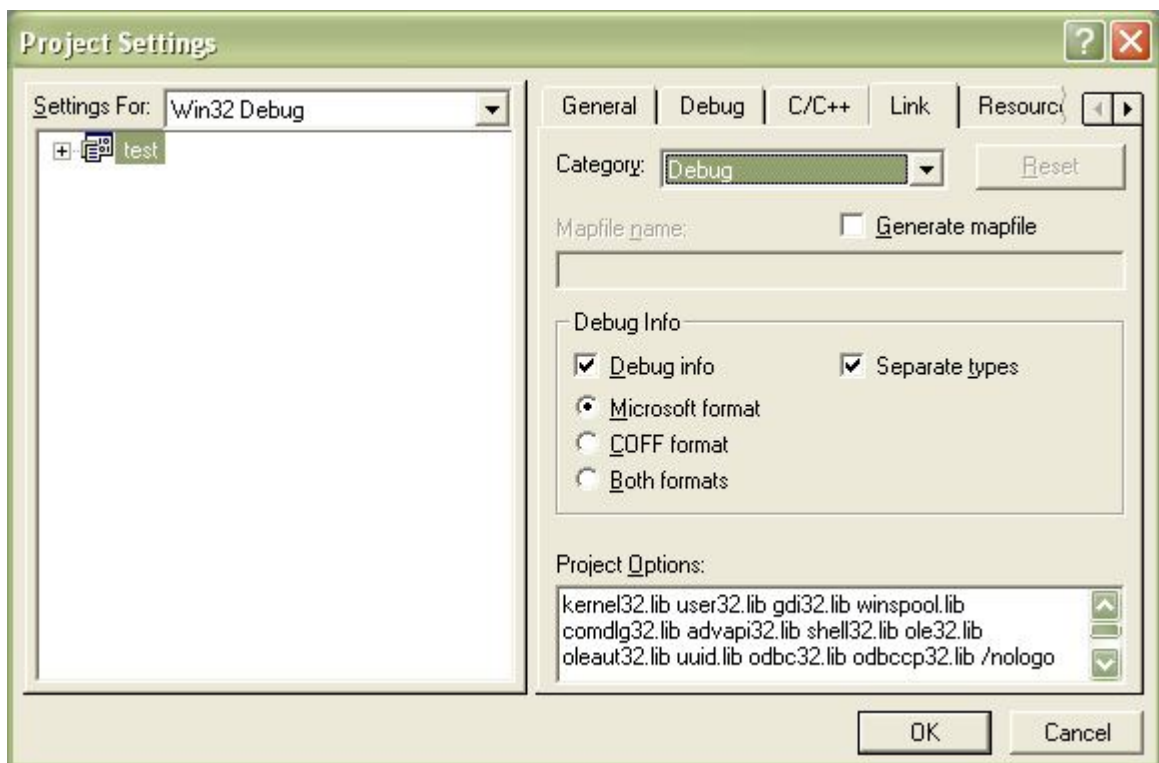
1 " .map .map "

MS Visual C++ MAP

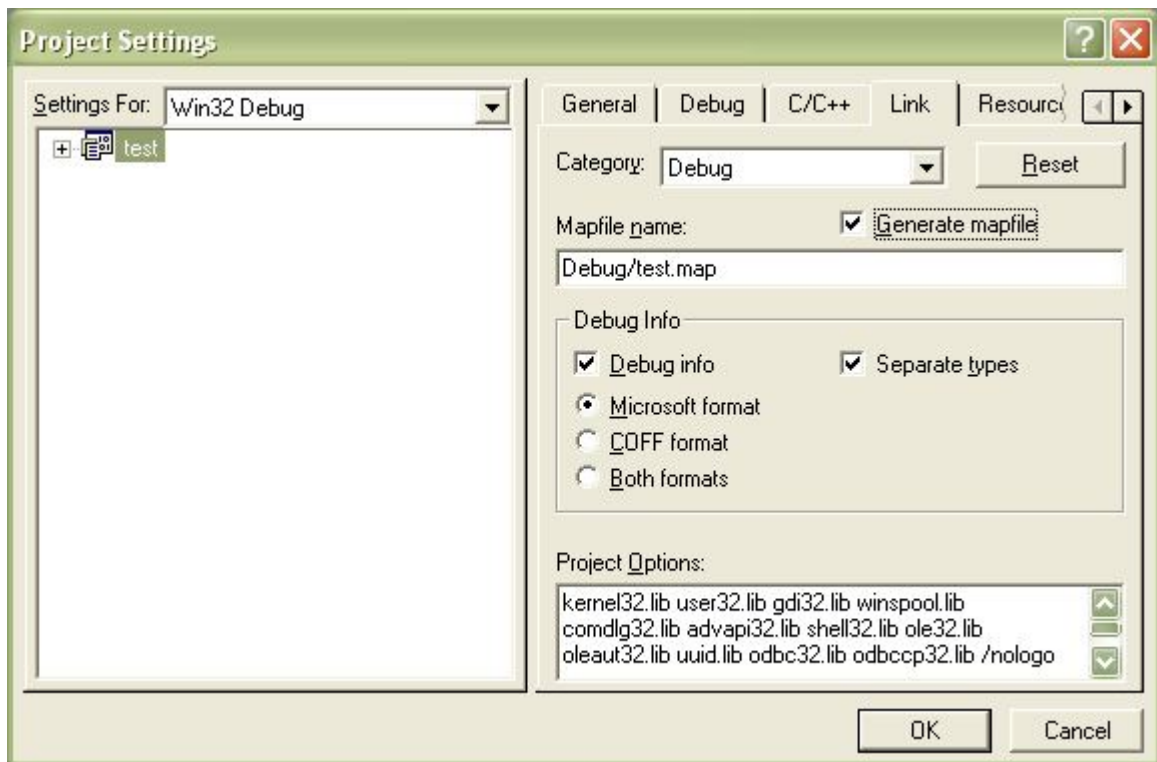
1 MS Visual ++
1 "Project" -> "Settings" Alt+F7



| "Link" "Category" - "Debug"

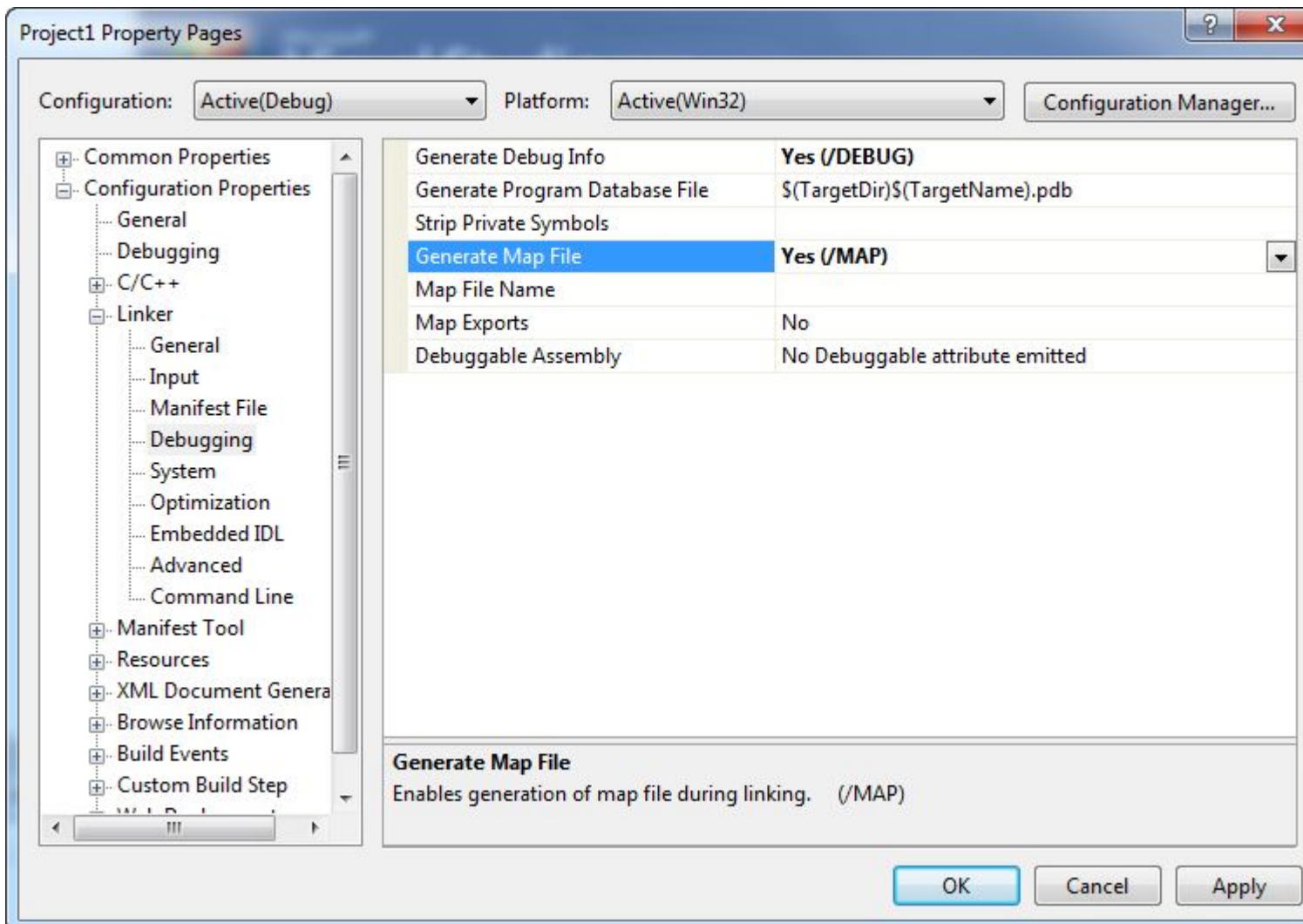


| "Generate mapfile"



MS Visual Studio 2003 :

- | MS Visual C++
- | Project - Properties
- | Configuration Properties - Linker - Debugging



MAP

MAP

Enigma Protector
Win32/Win64

Windows 32/64 PE
C++, Delphi, MASM, Visual Basic, Free Pascal, FoxPro, Power Basic, TASM

*.exe, *.dll, *.ocx, *.scr

Work with command line

Enigma Protector can be used to protect files through the command line. Command line version enigma32.exe or enigma64.exe is located in the installation folder.

- | To load project file into Enigma Protector: Enigma.exe Project.enigma
- | To start protection use console version:

Version	Example
Enigma Protector x86 (usual 32-bits version)	enigma32.exe [options] Project.enigma
Enigma Protector x64	enigma64.exe [options] Project.enigma

Project.enigma is the Enigma Protector project file.

Following options can be used:

- | -q enables quite protection mode, all messages are suppressed
- | -qe enables quite protection mode, all messages are suppressed except error messages

API

Enigma API

Enigma API

stdcall

- | [Registration API](#) -
- | [Trial API](#) -
- | [Crypt API](#) -
- | [Miscellaneous API](#) -

[Enigma API](#)

API

Enigma API

Enigma

API

API

- | [EP_RegCheckKey](#)
- | [EP_RegCheckKeyA](#)
- | [EP_RegCheckKeyW](#)
- | [EP_RegLoadKey](#)
- | [EP_RegLoadKeyA](#)
- | [EP_RegLoadKeyW](#)
- | [EP_RegSaveKey](#)
- | [EP_RegSaveKeyA](#)
- | [EP_RegSaveKeyW](#)
- | [EP_RegDeleteKey](#)
- | [EP_RegLoadAndCheckKey](#)
- | [EP_RegCheckAndSaveKey](#)
- | [EP_RegCheckAndSaveKeyA](#)
- | [EP_RegCheckAndSaveKeyW](#)
- | [EP_RegHardwareID](#)
- | [EP_RegHardwareIDA](#)
- | [EP_RegHardwareIDW](#)
- | [EP_RegKeyExpirationDate](#)
- | [EP_RegKeyExpirationDateEx](#)
- | [EP_RegKeyCreationDate](#)
- | [EP_RegKeyCreationDateEx](#)
- | [EP_RegKeyExecutions](#)
- | [EP_RegKeyExecutionsLeft](#)
- | [EP_RegKeyExecutionsTotal](#)
- | [EP_RegKeyDays](#)
- | [EP_RegKeyDaysLeft](#)
- | [EP_RegKeyDaysTotal](#)
- | [EP_RegKeyRuntime](#)
- | [EP_RegKeyRuntimeLeft](#)
- | [EP_RegKeyRuntimeTotal](#)
- | [EP_RegKeyGlobalTime](#)
- | [EP_RegKeyGlobalTimeLeft](#)
- | [EP_RegKeyGlobalTimeTotal](#)
- | [EP_RegKeyRegisterAfterDate](#)
- | [EP_RegKeyRegisterAfterDateEx](#)
- | [EP_RegKeyRegisterBeforeDate](#)
- | [EP_RegKeyRegisterBeforeDateEx](#)
- | [EP_RegShowDialog](#)

[Enigma API](#)

[API](#)

[Enigma API](#)

[Enigma](#)

EP_RegCheckKeyW

EP_RegCheckKey

```
| Name - - \0  
| Key - - \0
```

1 0

UNICODE

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegCheckKey( char* Name, char* Key );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples

EP_RegCheckKeyA

EP_RegCheckKeyA

[EP_RegCheckKey](#)

```
| Name - - \0  
| Key - - \0
```

1 0

UNICODE

[REGISTRATION FEATURES - Common panel.](#)

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegCheckKeyA( char* Name, char* Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples

EP_RegCheckKey

EP_RegCheckKeyA

UNICODE

EP_RegCheckKey

unicode

```
| Name - - \0 unicode  
| Key - - \0 unicode
```

1 0

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegCheckKeyW( wchar_t* Name, wchar_t* Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples

EP_RegCheckAndSaveKey

EP_RegCheckAndSaveKey

[EP_RegCheckKey](#)

[EP_RegSaveKey](#)

```
| Name - - \0  
| Key - - \0
```

1 0

UNICODE

```
|  
|  
|
```

[Show/Hide C++ function definition](#)

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegCheckAndSaveKey( char* Name, char* Key );
```

[Show/Hide Delphi function definition](#)

[Show/Hide Visual Basic function definition](#)

[Show/Hide C# \(.NET\) function definition](#)

[Show/Hide Delphi function example](#)

[Show/Hide C++ function example](#)

[Show/Hide C# \(.NET\) function example](#)

Examples

EP_RegCheckAndSaveKeyA

EP_RegCheckAndSaveKeyA [EP_RegCheckAndSaveKey](#)

```
| Name - - \0  
| Key - - \0
```

1 0

UNICODE

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegCheckAndSaveKeyA( char* Name, char* Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples

EP_RegCheckAndSaveKeyW

EP_RegCheckAndSaveKeyW
UNICODE

[EP_RegCheckAndSaveKey](#)

unicode (wide)

```
| Name - - \0  
| Key - - \0
```

1 0

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegCheckAndSaveKeyW( wchar_t* Name, wchar_t* Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\RegistrationUnicode

EP_RegDeleteKey

EP_RegDeleteKey

1 0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegDeleteKey();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples

EP_RegHardwareID

EP_RegHardware

-

\0

0

UNICODE

[REGISTRATION FEATURES - Common panel.](#)

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall PCHAR EP_RegHardwareID();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples

EP_RegHardwareIDA

EP_RegHardwareIDA

\0

0

UNICODE

[REGISTRATION FEATURES - Common panel.](#)

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall PCHAR EP_RegHardwareIDA();
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples

EP_RegHardwareIDW

EP_RegHardwareIDW

Registration Features - Hardware Lock

UNICODE

\0

0

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall PWCHAR EP_RegHardwareIDW();
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\RegistrationUnicode

EP_RegLoadKey

EP_RegLoadKey

```
| Name - - \0  
| Key - - \0
```

1 0

UNICODE

REGISTRATION FEATURES - Common panel.

/

[EP_RegCheckKey](#)

[EP_RegLoadAndCheckKey](#)

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegLoadKey( char** Name, char** Key );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples

EP_RegLoadKeyA

EP_RegLoadKey

EP_RegLoadKey

```
| Name - - \0  
| Key - - \0
```

1 0

UNICODE

REGISTRATION FEATURES - Common panel.

/

EP_RegCheckKey

EP_RegLoadAndCheckKey

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegLoadKeyA( char** Name, char** Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples

EP_RegLoadKeyW

EP_RegLoadKey

UNICODE

[EP_RegLoadKey](#)

REGISTRATION FEATURES - Common panel.

unicode (wide)

	Name	-	-	\0	unicode
	Key	-	-	\0	unicode

1 0

/

[EP_RegCheckKey](#)

[EP_RegLoadAndCheckKey](#)

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegLoadKeyW( wchar_t** Name, wchar_t** Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\RegistrationUnicode

EP_RegSaveKey

EP_RegSaveKey

```
| Name - - \0  
| Key - - \0
```

1 0

/

[EP_RegCheckKey](#)

[EP_RegCheckAndSaveKey](#)

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegSaveKey( char* Name, char* Key );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples

EP_RegSaveKeyA

EP_RegSaveKeyA

[EP_RegSaveKey](#)

```
| Name - - \0  
| Key - - \0
```

1 0

/

[EP_RegCheckKey](#)

[EP_RegCheckAndSaveKey](#)

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegSaveKeyA( char* Name, char* Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples

EP_RegSaveKeyW

EP_RegSaveKeyA

UNICODE

[EP_RegSaveKey](#)

unicode (wide)

-

	Name	-	-	\0	unicode
	Key	-	-	\0	unicode

1 0

/

[EP_RegCheckKeyW](#)

[EP_RegCheckAndSaveKeyW](#)

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegSaveKeyW( wchar_t* Name, wchar_t* Key );
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\RegistrationUnicode

EP_RegLoadAndCheckKey

EP_RegLoadAndCheckKey

[EP_RegLoadKey](#)

[EP_RegCheckKey](#)

1 0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegLoadAndCheckKey();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples

EP_RegKeyCreationDate

EP_RegKeyCreationDate

[EP_RegKeyCreationDateEx](#)

```
| Year -  
| Month -  
| Day -
```

```
1 0
```

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyCreationDate( int* Year, int* Month, int* Day );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

EP_RegKeyCreationDateEx

EP_RegKeyCreationDateEx

32

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyCreationDateEx();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyExpirationDate

EP_RegKeyExpirationDate

[EP_RegKeyExpirationDateEx](#)

| `Year` -
| `Month` -
| `Day` -

1 0

|
|
|

[Show/Hide C++ function definition](#)

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyExpirationDate( int* Year, int* Month, int* Day );
```

[Show/Hide Delphi function definition](#)

[Show/Hide Visual Basic function definition](#)

[Show/Hide C# \(.NET\) function definition](#)

[Show/Hide Delphi function example](#)

Examples\KeyExpiration

EP_RegKeyExpirationDateEx

EP_RegKeyExpirationDateEx

32

0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyExpirationDateEx();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples\KeyExpiration

EP_RegKeyExecutions

EP_RegKeyExecutions

EP_RegKeyExecutionsLeft EP_RegKeyExecutionsTotal

```
| Total -  
| Left -
```

0

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyExecutions( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyExecutionsLeft

EP_RegKeyExecutionsLeft

EP_RegKeyExecutions

0

|
|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyExecutionsLeft();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyExecutionsTotal

EP_RegKeyExecutionsTota

EP_RegKeyExecutions

0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyExecutionsTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyDays

EP_RegKeyDays function

EP_RegKeyDaysLeft EP_RegKeyDaysTotal

```
| Total -  
| Left -
```

```
1      0
```

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyDays( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyDaysLeft

EP_RegKeyDaysLeft

EP_RegKeyDays

0

|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyDaysLeft();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyDaysTotal

EP_RegKeyDaysTotal

EP_RegKeyDays

0

The function fails in the following cases:

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyDaysTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyRuntime

EP_RegKeyRuntime

Creating Keys

EP_RegKeyRuntimeLeft EP_RegKeyRuntimeTotal

```
| Total -  
| Left -
```

```
1 0
```

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyRuntime( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyRuntimeLeft

EP_RegKeyRuntimeLeft

f

[EP_RegKeyRuntime.](#)

[Creating Keys](#)

0

|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyRuntimeLeft();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyRuntimeTotal

EP_RegKeyRuntimeTotal

EP_RegKeyRuntime

Creating Keys

0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyRuntimeTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyGlobalTime

EP_RegKeyGlobalTime

EP_RegKeyGlobalTimeLeft EP_RegKeyGlobalTimeTotal

```
| Total -  
| Left -
```

1 0

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyGlobalTime( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyGlobalTimeLeft

EP_RegKeyGlobalTimeLeft

[EP_RegKeyGlobalTime](#)

EP_RegKeyGlobalTimeLeft

0

|
|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyGlobalTimeLeft();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyGlobalTimeTotal

EP_RegKeyGlobalTimeTotal

[EP_RegKeyGlobalTime](#)

0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyGlobalTimeTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyRegisterAfterDate

EP_RegKeyRegisterAfterDate

EP_RegKeyRegisterAfterDateE

```
| Year -  
| Month -  
| Day -
```

```
1 0
```

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyRegisterAfterDate( int* Year, int* Month, int* Day
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyRegisterAfterDateEx

EP_RegKeyRegisterAfterDateEx

1 32

0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyRegisterAfterDateEx();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyRegisterBeforeDate

EP_RegKeyRegisterBeforeDate

[EP_RegKeyRegisterBeforeDateEx](#)

```
| Year -  
| Month -  
| Day -
```

1 0

```
|  
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_RegKeyRegisterBeforeDate( int* Year, int* Month, int* Da
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegKeyRegisterBeforeDateEx

EP_RegKeyRegisterBeforeDateEx

32

0

|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_RegKeyRegisterBeforeDateEx();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_RegShowDialog

EP_RegShowDialog shows a registration dialog that is designed and enabled on the [REGISTRATION FEATURES - Registration Dialog](#) panel. Note, Registration Dialog feature should be enabled, otherwise dialog will not be shown. Also, any Exit application actions in the Registration Dialog will not exit the application, but just close dialog window.

Return Value

Function does not return any value.

Remark

The function fails in the following cases:

- | application is not protected;
- | [REGISTRATION FEATURES - Registration Dialog](#) feature is not enabled.

Definition

[Show/Hide C++ function definition](#)

```
extern "C" __declspec( dllimport ) __stdcall void EP_RegShowDialog();
```

[Show/Hide Delphi function definition](#)

[Show/Hide Visual Basic function definition](#)

[Show/Hide C# \(.NET\) function definition](#)

Examples

[Show/Hide Delphi function example](#)

See function examples in the installation folder, Examples\RegistrationShowDialog subfolder.

API

- | EP_TrialExecutions
- | EP_TrialExecutionsLeft
- | EP_TrialExecutionsTotal
- | EP_TrialDays
- | EP_TrialDaysLeft
- | EP_TrialDaysTotal
- | EP_TrialExpirationDate
- | EP_TrialExpirationDateEx
- | EP_TrialDateTillDate
- | EP_TrialDateTillDateStartEx
- | EP_TrialDateTillDateEndEx
- | EP_TrialExecutionTime
- | EP_TrialExecutionTimeLeft
- | EP_TrialExecutionTimeTotal

Enigma API

API

Enigma API

Enigma

EP_TrialExecutionsTotal

```
| Total -  
| Left -
```

1 0

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_TrialExecutions( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples

EP_TrialExecutionsLeft

EP_TrialExecutionsLeft

[EP_TrialExecutions](#)

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialExecutionsLeft();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples

EP_TrialExecutionsTotal

EP_TrialExecutionsTotal

EP_TrialExecutions

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialExecutionsTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples

EP_TrialDays

EP_TrialDays

EP_TrialDaysLeft EP_TrialDaysTotal

```
| Total -  
| Left -
```

1 0

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_TrialDays( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples\Trial subfolder

EP_TrialDaysLeft

EP_TrialDaysLeft

EP_TrialDays

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialDaysLeft();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples

EP_TrialDaysTotal

EP_TrialDaysTotal

EP_TrialDays

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialDaysTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples

EP_TrialExpirationDate

EP_TrialExpirationDate

EP_TrialExpirationDateEx

```
| Year -  
| Month -  
| Day -
```

```
1 0
```

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_TrialExpirationDate( int* Year, int* Month, int* Day );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Examples\Trial

EP_TrialExpirationDateEx

EP_TrialExpirationDateEx

EP_TrialExpirationDate

32

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialExpirationDateEx();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_TrialDateTillDate

EP_TrialDateTillDate

-

EP_TrialDate

```
| StartYear -  
| StartMonth -  
| StartDay -  
| EndYear -  
| EndMonth -  
| EndDay -
```

1 0

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_TrialDateTillDate( int* StartYear, int* StartMonth, int*
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples\Trial

EP_TrialDateTillDateStartEx

EP_TrialDateTillDateStartEx

-

EP_TrialDateTillDate

32

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialDateTillDateStartEx();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_TrialDateTillDateEndEx

EP_TrialDateTillDateEndEx

EP_TrialDateTillDate

32

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialDateTillDateEndEx();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_TrialExecutionTime

EP_TrialExecutionTime

EP_TrialExecutionTimeLeft EP_TrialExecutionTimeTotal

```
| Total -  
| Left -
```

1 0

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_TrialExecutionTime( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C++ function example

Examples\TrialExecutionTime

EP_TrialExecutionTimeLeft

EP_TrialExecutionTimeLeft

[EP_TrialExecutionTime](#)

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialExecutionTimeLeft();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_TrialExecutionTimeTotal

EP_TrialExecutionTimeTotal

EP_TrialExecutionTime

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_TrialExecutionTimeTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

API

Crypt API

/

- | [EP_CryptHashBuffer](#)
- | [EP_CryptHashFileA](#)
- | [EP_CryptHashFileW](#)
- | [EP_CryptHashStringA](#)
- | [EP_CryptHashStringW](#)
- | [EP_CryptDecryptBuffer](#)
- | [EP_CryptDecryptBufferEx](#)
- | [EP_CryptEncryptBuffer](#)
- | [EP_CryptEncryptBufferEx](#)

[Enigma API](#)

API

: [Enigma API](#)

EP_CryptDecryptBuffer

EP_CryptDecryptBuffer

| `Buffer` -
| `Size` -
| `Key` - ANSI

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall void EP_CryptDecryptBuffer( byte* Buffer, int Size, char* Key);
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\CryptBuffer

EP_CryptDecryptBufferEx

EP_CryptDecryptBufferEx

```
| InBuffer -  
| OutBuffer -  
| Size -  
| Key -  
| KeySize -
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall void EP_CryptDecryptBufferEx( byte* InBuffer, byte* OutBuffer, i
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\CryptBuffer

EP_CryptEncryptBuffer

EP_CryptEncryptBuffer

| **Buffer** -
| **Size** -
| **Key** - ANSI

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall void EP_CryptEncryptBuffer( byte* Buffer, int Size, char* Key);
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\CryptBuffer

EP_CryptEncryptBufferEx

EP_CryptEncryptBufferEx

```
| InBuffer -  
| OutBuffer -  
| Size -  
| Key -  
| KeySize -
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall void EP_CryptEncryptBufferEx( byte* InBuffer, byte* OutBuffer, i
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\CryptBuffer

EP_CryptHashBuffer

EP_CryptHashBuffer

hash()

| **Hash** -

	=
XOR32	HASH_XOR32 = 0
MD2	HASH_MD2 = 1
MD5	HASH_MD5 = 2
RipeMD160	HASH_RipeMD160 = 3
SH1	HASH_SH1 = 4
SHA224	HASH_SHA224 = 5
SHA256	HASH_SHA256 = 6
SHA384	HASH_SHA384 = 7
SHA512	HASH_SHA512 = 8

| **Buffer** -

| **Size** -

| **Digest** -

0

	()
XOR32	4
MD2	16
MD5	16
RipeMD160	20
SH1	20
SHA224	28
SHA256	32
SHA384	48
SHA512	64

|
|
|
|
|

[\[-\] Show/Hide C++ function definition](#)

```
extern "C" __declspec( dllimport ) __stdcall int EP_CryptHashBuffer( int Hash, byte* Buffer, int Size, byte*
```

[\[+\] Show/Hide Delphi function definition](#)

[\[+\] Show/Hide C# \(.NET\) function definition](#)

Examples\Hashes

EP_CryptHashFileA

EP_CryptHashFileA

| Hash -

	=
XOR32	HASH_XOR32 = 0
MD2	HASH_MD2 = 1
MD5	HASH_MD5 = 2
RipeMD160	HASH_RipeMD160 = 3
SH1	HASH_SH1 = 4
SHA224	HASH_SHA224 = 5
SHA256	HASH_SHA256 = 6
SHA384	HASH_SHA384 = 7
SHA512	HASH_SHA512 = 8

| FileName-

| Digest -

0

	()
XOR32	4
MD2	16
MD5	16
RipeMD160	20
SH1	20
SHA224	28
SHA256	32
SHA384	48
SHA512	64

|
|
|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_CryptHashFileA( int Hash, char* FileName, byte* Digest);
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\Hashes

EP_CryptHashFileW

EP_CryptHashFileA

| Hash -

	=
XOR32	HASH_XOR32 = 0
MD2	HASH_MD2 = 1
MD5	HASH_MD5 = 2
RipeMD160	HASH_RipeMD160 = 3
SH1	HASH_SH1 = 4
SHA224	HASH_SHA224 = 5
SHA256	HASH_SHA256 = 6
SHA384	HASH_SHA384 = 7
SHA512	HASH_SHA512 = 8

| FileName- UNICODE

| Digest -

0

	()
XOR32	4
MD2	16
MD5	16
RipeMD160	20
SH1	20
SHA224	28
SHA256	32
SHA384	48
SHA512	64

|
|
|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_CryptHashFileW( int Hash, wchar_t* FileName, byte* Digest
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\Hashes

EP_CryptHashStringA

EP_CryptHashStringA

ANSI

| Hash -

	=
XOR32	HASH_XOR32 = 0
MD2	HASH_MD2 = 1
MD5	HASH_MD5 = 2
RipeMD160	HASH_RipeMD160 = 3
SH1	HASH_SH1 = 4
SHA224	HASH_SHA224 = 5
SHA256	HASH_SHA256 = 6
SHA384	HASH_SHA384 = 7
SHA512	HASH_SHA512 = 8

| Str - ANSI

| Digest -

0

	()
XOR32	4
MD2	16
MD5	16
RipeMD160	20
SH1	20
SHA224	28
SHA256	32
SHA384	48
SHA512	64

|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_CryptHashStringA( int Hash, char* Str, byte* Digest);
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\Hashes

EP_CryptHashStringW

EP_CryptHashFileW

UNICODE

| **Hash** -

	=
XOR32	HASH_XOR32 = 0
MD2	HASH_MD2 = 1
MD5	HASH_MD5 = 2
RipeMD160	HASH_RipeMD160 = 3
SH1	HASH_SH1 = 4
SHA224	HASH_SHA224 = 5
SHA256	HASH_SHA256 = 6
SHA384	HASH_SHA384 = 7
SHA512	HASH_SHA512 = 8

| **Str** - UNICODE

| **Digest** -

0

	()
XOR32	4
MD2	16
MD5	16
RipeMD160	20
SH1	20
SHA224	28
SHA256	32
SHA384	48
SHA512	64

|
|
|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_CryptHashStringW( int Hash, wchar_t* Str, byte* Digest);
```

Show/Hide Delphi function definition

Show/Hide C# (.NET) function definition

Examples\Hashes

API

Enigma

- | [EP_MiscCountryCode](#)
- | [EP_MiscGetWatermark](#)
- | [EP_ProtectedStringByID](#)
- | [EP_ProtectedStringByKey](#)
- | [EP_CheckupCopies](#)
- | [EP_CheckupCopiesCurrent](#)
- | [EP_CheckupCopiesTotal](#)
- | [EP_CheckupIsEnigmaOk](#)
- | [EP_CheckupIsProtected](#)
- | [EP_CheckupVirtualizationTools](#)

[Enigma API](#)

[API](#)

[Enigma API](#)

[Enigma](#)

EP_CheckupCopies

EP_CheckupCopies

EP_CheckupCopiesTotal

EP_CheckupCopiesCurrent

```
| Total -  
| Current -
```

```
1 0
```

```
:
```

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_TrialDays( int* Total, int* Left );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Examples\ExecutedCopies

EP_CheckupCopiesCurrent

EP_CheckupCopiesCurrent

[EP_CheckupCopies](#)

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_CheckupCopiesCurrent();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_CheckupCopiesTotal

EP_CheckupCopiesTotal

EP_CheckupCopies

0

|
|

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_CheckupCopiesTotal();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_CheckupIsEnigmaOk

EP_CheckupIsEnigmaOk

1 0

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_CheckupIsEnigmaOk();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Examples\CheckEnigma

EP_CheckupIsProtected

EP_CheckupIsProtected

Enigma

1 0

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_CheckupIsProtected();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Examples\CheckEnigma

EP_EnigmaVersion

EP_EnigmaVersion

Enigma

Enigma

0x014E (334)

1.78

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_EnigmaVersion();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Examples\CheckEnigma

EP_MiscCountryCode

EP_MiscCountryCode

Afghanistan	CN_AFGHANISTAN	114
Albania	CN_ALBANIA	1
Algeria	CN_ALGERIA	2
Argentina	CN_ARGENTINA	3
Armenia	CN_ARMENIA	4
Australia	CN_AUSTRALIA	5
Austria	CN_AUSTRIA	6
Azerbaijan	CN_AZERBAIJAN	7
Bahrain	CN_BAHRAIN	8
Bangladesh	CN_BANGLADESH	115
Belarus	CN_BELARUS	9
Belgium	CN_BELGIUM	10
Belize	CN_BELIZE	11
Bolivia	CN_BOLIVIA	116
Bosnia and Herzegovina	CN_BOSNIA	117
Brazil	CN_BRAZIL	13
Brunei Darussalam	CN_BRUNEI	14
Bulgaria	CN_BULGARIA	15
Cambodia	CN_CAMBODIA	16
Canada	CN_CANADA	17
Caribbean	CN_CARRIBEAN	118
Chile	CN_CHILE	20
China	CN_CHINA	21
Colombia	CN_COLOMBIA	22
Costa Rica	CN_COSTARICA	23
Croatia	CN_CROATIA	24
Czech Republic	CN_CZECH	25
Denmark	CN_DENMARK	26
Dominican Republic	CN_DOMINICAN	27
Ecuador	CN_ECUADOR	28
Egypt	CN_EGYPT	29
El Salvador	CN_ELSALVADOR	30
Estonia	CN_ESTONIA	31
Ethiopia	CN_ETHIOPIA	119
Faroe Islands	CN_FAROE	32
Finland	CN_FINLAND	33
France	CN_FRANCE	34
Georgia	CN_GEORGIA	35
Germany	CN_GERMANY	36
Greece	CN_GREECE	37
Greenland	CN_GREENLAND	120
Guatemala	CN_GUATEMALA	38
Honduras	CN_HONDURAS	39
Hong Kong	CN_HONGKONG	40
Hungary	CN_HUNGARU	41
Iceland	CN_ICELAND	42
India	CN_INDIA	43
Indonesia	CN_INDONESIA	44
Iran	CN_IRAN	45
Iraq	CN_IRAQ	46
Ireland	CN_IRELAND	47
Israel	CN_ISRAEL	48
Italy	CN_ITALY	49
Jamaica	CN_JAMAICA	50
Japan	CN_JAPAN	51
Jordan	CN_JORDAN	52
Kazakhstan	CN_KAZAKHSTAN	53

Kenya	CN_KENYA	54
Korea	CN_KOREA	56
Kuwait	CN_KUWAIT	57
Kyrgyzstan	CN_KYRGYZSTAN	58
Laos	CN_LAOS	121
Latvia	CN_LATVIA	59
Lebanon	CN_LEBANON	60
Libyan	CN_LIBYAN	122
Liechtenstein	CN_LIECHTENSTEIN	62
Lithuania	CN_LITHUANIA	63
Luxembourg	CN_LUXEMBOURG	64
Macao	CN_MACAO	65
Macedonia	CN_MACEDONIA	66
Malaysia	CN_MALAYSIA	67
Maldives	CN_MALDIVES	123
Malta	CN_MALTA	124
Mexico	CN_MEXOCI	68
Monaco	CN_MONACO	70
Mongolia	CN_MONGOLIA	71
Montenegro	CN_MONTENEGRO	125
Morocco	CN_MOROCCO	72
Nepal	CN_NEPAL	126
Netherlands	CN_NETHERLANDS	73
New Zealand	CN_NEWZEALAND	74
Nicaragua	CN_NICARAGUA	75
Nigeria	CN_NIGERIA	127
Norway	CN_NORWAY	76
Oman	CN_OMAN	77
Pakistan	CN_PAKISTAN	78
Panama	CN_PANAMA	79
Paraguay	CN_PARAGUAY	80
Peru	CN_PERY	81
Philippines	CN_PHILIPPINES	82
Poland	CN_POLAND	83
Portugal	CN_PORTUGAL	84
Puerto Rico	CN_PUERTORICO	85
Qatar	CN_QATAR	86
Romania	CN_ROMANIA	87
Russia	CN_RUSSIA	88
Rwanda	CN_RWANDA	128
Saudi Arabia	CN_SAUDIARABIA	89
Senegal	CN_SENEGAL	129
Serbia	CN_SERBIA	130
Serbia and Montenegro	CN_SERBIAMONTENEGRO	90
Singapore	CN_SINGAROPE	91
Slovakia	CN_SLOVAKIA	92
Slovenia	CN_SLOVENIA	93
South Africa	CN_SOUTHAFRICA	94
Spain	CN_SPAIN	95
Sri Lanka	CN_SRILANKA	131
Sweden	CN_SWEDEN	96
Switzerland	CN_SWITZERLAND	97
Syrian	CN_SYRIAN	132
Taiwan	CN_TAIWAN	98
Tajikistan	CN_TAJIKISTAN	99
Thailand	CN_THAILAND	100
Trinidad and Tobago	CN_TRINIDADTOBAGO	101
Tunisia	CN_TUNISIA	102
Turkey	CN_TURKEY	103
Turkmenistan	CN_TURKMENISTAN	133
Ukraine	CN_UKRAINE	104
United Arab Emirates	CN_UAE	105
United Kingdom	CN_UNITEDKINGDOM	106

United States	CN_USA	107
Uruguay	CN_URUGUAY	108
Uzbekistan	CN_UZBEKISTAN	109
Venezuela	CN_VENEZUELA	110
Viet Nam	CN_VIETNAM	111
Yemen	CN_YEMEN	112
Zimbabwe	CN_ZIMBABWE	113

0

:

|

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_MiscCountryCode();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

EP_MiscGetWatermark

EP_MiscGetWatermark

```
| ID -  
| WM -  
| WM.WMType -  
| WM.Name -  
| WM.NameLen -  
| WM.Text -  
| WM.TextLen -  
| WM.FileName -  
| WM.FileNameLen -  
| WM.AFile -  
| WM.AFileLen -
```

0

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllexport ) __stdcall int EP_MiscGetWatermark( int ID, PWMCContent WM );
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Examples\Watermarks

EP_ProtectedStringByID

EP_ProtectedStringByID

ID

```
| ID - ID  
| Buffer -  
| Len -
```

0

```
| ID  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_ProtectedStringByID( int Total, char* Left, int Len);
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C# (.NET) function example

Show/Hide Visual Basic function example

Examples\ProtectedStrings

EP_ProtectedStringByKey

EP_ProtectedStringByKey

```
| Key -  
| Buffer -  
| Len -
```

0

```
|  
|
```

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_ProtectedStringByKey( char* Key, char* Left, int Len);
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Show/Hide C# (.NET) function example

Show/Hide Visual Basic function example

Examples\ProtectedStrings

EP_SplashScreenShow

EP_SplashScreenShow

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall int EP_SplashScreenShow();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Examples\SplashScreen

EP_SplashScreenHide

EP_SplashScreenHide

Show/Hide C++ function definition

```
extern "C" __declspec( dllimport ) __stdcall void EP_SplashScreenHide();
```

Show/Hide Delphi function definition

Show/Hide Visual Basic function definition

Show/Hide C# (.NET) function definition

Show/Hide Delphi function example

Examples\SplashScreen

EP_CheckupVirtualizationTools

EP_CheckupVirtualizationTools allows to check if the file is started under any of the known Virtualization Tools like VMWare/VirtualBox/VirtualPC. Note, you have to enable [CHECK-UP - Virtualization Tools](#) feature, select necessary virtualization tools you would like to check and disable option Terminate Execution.

Return Value

Function returns 1 (true) if the file is started under any virtualization tool or 0 (false) if the function failed. See Remark section below to know the cases when the function fails.

Remark

The function fails in the following cases:

- | application is not protected;
- | [CHECK-UP - Virtualization Tools](#) feature is not enabled;
- | necessary Virtualization Tools for checking were not selected on [CHECK-UP - Virtualization Tools](#) panel.

Definition

[Show/Hide C++ function definition](#)

```
extern "C" __declspec( dllimport ) __stdcall BOOL EP_CheckupVirtualizationTools();
```

[Show/Hide Delphi function definition](#)

[Show/Hide Visual Basic function definition](#)

[Show/Hide C# \(.NET\) function definition](#)

Examples

[Show/Hide Delphi function example](#)

See function examples in the installation folder, Examples\CheckVirtualizationTools subfolder.

Enigma API

enigma_ide.dll	Enigma Protector	API	GetModuleHandleA, LoadLibraryA
GetProcAddress	Enigma Protector	Examples	Enigma API
Enigma API			

Delphi

```
var
  pEP_RegHardwareID : function : pchar;
  pEP_RegCheckKey : function(pcName, pcKey : pchar) : boolean;
  pcName : pchar;
  pcKey : pchar;

begin
  // Show message box with a Hardware ID
  pEP_RegHardwareID := GetProcAddress(GetModuleHandle('enigma_ide.dll'), 'EP_RegHardwareID');
  MessageBox(0, pEP_RegHardwareID, 'Application' , MB_OK);
  // Check registration information
  pcName := 'Registration Info';
  pcKey := 'Registration Key';
  pEP_RegCheckKey := GetProcAddress(GetModuleHandle('enigma_ide.dll'), 'EP_RegCheckKey');
  if pEP_RegCheckKey(pcName, pcKey) then
    MessageBox(0, 'Valid Registration Key', 'Application', MB_OK)
  else
    MessageBox(0, 'Invalid Registration Key', 'Application', MB_OK)
end;
```

- | API - Enigma API
- | -
- | Protector -
- | -
- | - Enigma Protector
- | - Enigma Protector

API

Visual Basic

Enigma API

```
If EP_RegLoadAndCheckKey = TRUE Then  
    | *****  
End If
```

If - End If

			EP_RegLoadAndCheckKey = TRUE	Visual Basic		Visual Basic
TRUE	\$FFFF	FALSE	0	Enigma API	0	1

```
If EP_RegLoadAndCheckKey Then  
    | *****  
End If
```

enigma_api.dll

enigma_api.dll

Enigma API

Enigma API

Visual Basic .NET

Enigma API Visual Basic .NET

Visual Basic

Example\

API

EnigmaSDK\VB\

API

Access

"

\

"

: UNICODE UNICODE

: A UNICODE UNICODE

: Enigma KEY KEY

:



Enigma Protector

Email

Protector

Enigma Protector

Enigma Protector

Enigma Protector

Reg_Crypt

Delphi

```
function IsEnigmaPresent : boolean;
begin
  Result := true;
  // this code returns TRUE at any ways,
  // but we should use it to fighting
  // with the Delphi code optimization
  if GetModuleHandle(nil) <> 0 then
  begin
    {$I reg_crypt_begin1.inc}
    Result := false;
    {$I reg_crypt_end1.inc}
  end;
end;
```



Enigma Protector

Enigma Protector

.enigmadb.

Enigma Protector

Enigma Protector

.enigmamail.

: gmail

: SSL

|

|

|

1. Enigma

2.

3.

4.

5.

6.

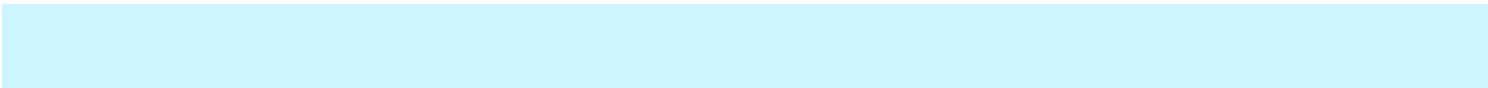
7.

- 1.
- 2.
- 3.

Enigma Protector



|
|
|
|



I Enigma Protector

I

Enigma Protector

RSA 512

1.

" "

2. Enigma API

Enigma Protector /

Enigma Protector
Enigma API

Enigma API " "

|
|
|
|
|
| CGI ()

Vladimir

Registration key :

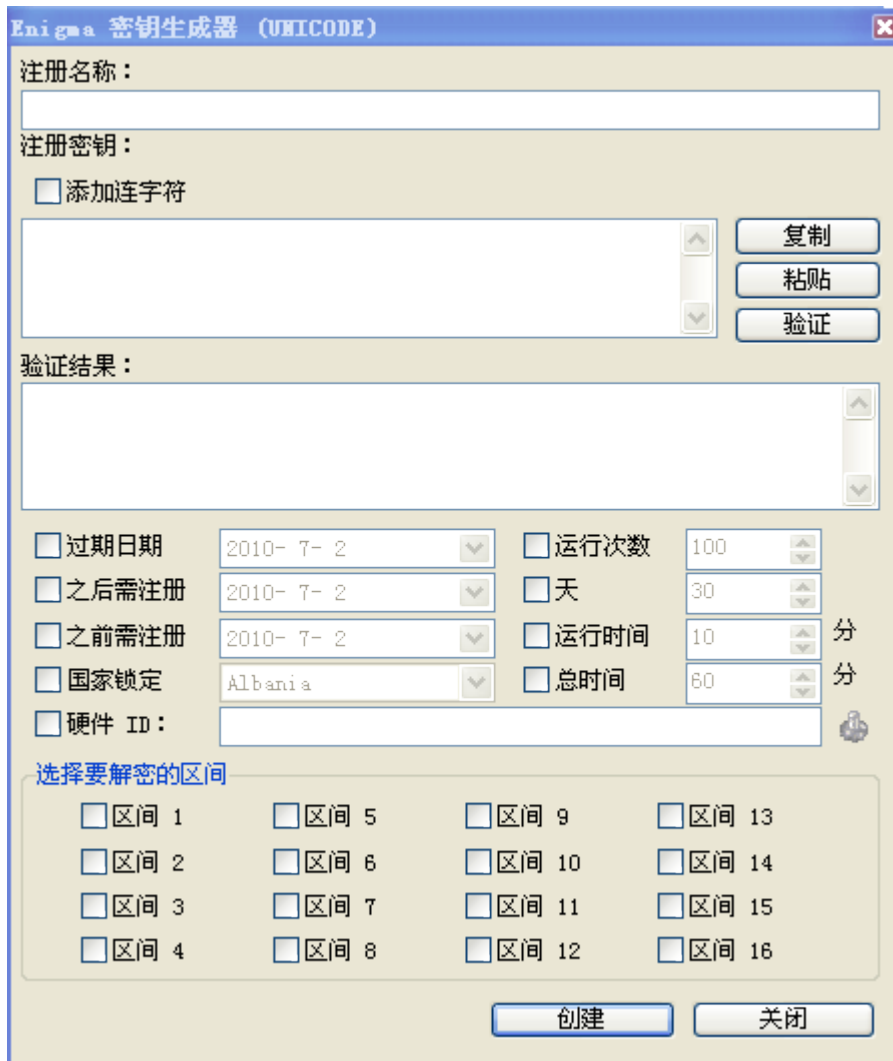
W3UH9K-L5EUT2-XUPUYJ-P8YYHR

Verifying results :

```
0x80DBC089
2007 4 17
2008 1 1
    #1
    #7
    #16
```

```
|          2007 4 17
|          2008 1 1
|          #1, #7, #16.
```


The Enigma Protector



EP_RegKeyExpirationDate
 ID
 EP_RegHardwareID

Enigma

API

ID

Enigma API

CGI

```
CGI                                     CGI                                     2                                     Windows Linux
EnigmaSDK\CGI Keygen                  cgi-bin
Key GET POST
CGI :
1.      cgi-bin
2.      ("keygen" for Linux      "keygen.exe" for Windows )
3.      :
      CGI :
| GenerateKey - ansi* ;
| GenerateKeyA - ;
| GenerateKeyW - unicode* ( ) ;
| GenerateKeyFromProject - ansi* , ;
| GenerateKeyFromProjectA - ;
| GenerateKeyFromProjectW - unicode* ( ) .
*      ansi unicode - ,
      UNICODE ANSI
**      -
CGI :
-      :
| GenerateKey
| GenerateKeyA
| GenerateKeyW
| GenerateKeyFromProject
| GenerateKeyFromProjectA
| GenerateKeyFromProjectW
-
-      ( %20 " ");
-      Enigma -- ;
-      Enigma -- ;
-      ID ;
-      *;
-      *;
-      *;
-      ;
-      ;
-      ;
-      ;
-      Enigma -- ,
*      2 +2 +2 2010 12 1 01122010
```

- [Show/Hide HTML \(GET method\) example](#)
- [Show/Hide HTML \(GET method\) getting parameters from project file example](#)
- [Show/Hide HTML \(POST method\) example](#)
- [Show/Hide HTML \(POST method\) getting parameters from project file example](#)
- [Show/Hide PHP \(POST method\) example](#)

Show/Hide PHP (GET method) example

POST

1. " - " "Do not allow to change hyphens value"



2. "Do not allow to change sections value"



3. "Do not allow to change expiration date value"

The Enigma Protector Export Keys Generator

Default value of key Expiration date

The screenshot shows a dialog box titled "The Enigma Protector Export Keys Generator" with a subtitle "Default value of key Expiration date". The main area contains the following options:

- Key with Expiration date
 - Fixed date
 - 2010-12-31
 - Custom Date
 - Today + 30 + Days
- Do not allow to change Expiration date

At the bottom right, there are three buttons: "< Back", "Next >", and "Cancel".

4. "Do not allow to change Register After date"

The Enigma Protector Export Keys Generator

Default value of Register After date

The screenshot shows a dialog box titled "The Enigma Protector Export Keys Generator" with a subtitle "Default value of Register After date". The main area contains the following options:

- Key with Register After date
 - Fixed date
 - 2008-12-12
 - Custom Date
 - Today + 30 + Days
- Do not allow to change Register After date value

At the bottom right, there are three buttons: "< Back", "Next >", and "Cancel".

5. "Do not allow to change Register Before date"



6. "Do not allow to change Executions "



7. "Do not allow to change Days "



The Enigma Protector Export Keys Generator

Default value of Days limit

Key with Days limit

Days count

Do not allow to change Days

< Back Next > Cancel

8. "Do not allow to change Run-time "



The Enigma Protector Export Keys Generator

Default value of Run-time limit

Key with Run-time limit

Run-time minutes

Do not allow to change Run-time

< Back Next > Cancel

9. "Do not allow to change Global Time "
-



The Enigma Protector Export Keys Generator

Default value of Global Time limit

Key with Global Time limit
Global Time minutes

Do not allow to change Global Time

< Back Next > Cancel

10. "Do not allow to change Country Lock "



The Enigma Protector Export Keys Generator

Default value of Country Lock

Key with Country Lock
Country Lock

Do not allow to change Country Lock

< Back Finish Cancel

11. " Finish"

Enigma Protector (Windows) libkeygen.so (Linux) (Windows Linux) keygen.dll
Examples\Keygen Examples\KeygenUnicode Enigma Protector ,

ansi :

- | KG_GenerateRegistrationKey
- | KG_GenerateRegistrationKeyFromProject
- | KG_GenerateRegistrationKeyA
- | KG_GenerateRegistrationKeyFromProjectA
- | KG_VerifyRegistrationInfo
- | KG_VerifyRegistrationInfoFromProject
- | KG_VerifyRegistrationInfoA
- | KG_VerifyRegistrationInfoFromProjectA

unicode :

- | KG_GenerateRegistrationKeyW
- | KG_GenerateRegistrationKeyFromProjectW
- | KG_VerifyRegistrationInfoW
- | KG_VerifyRegistrationInfoFromProjectW

TKeyGenParams

TKeyGenParams

KG_GenerateRegistrationKey

KG_GenerateRegistrationKeyFromProject

	<p>() (RSA 512/768/1024/2048/3072/4096), (- /);</p> <p>RM_512 = 0; RM_768 = 1; RM_1024 = 2; RM_2048 = 3; RM_3072 = 4; RM_4096 = 5;</p>
	<p>() (Base 2/8/16/32/64), (-):</p> <p>RB_2 = 0; RB_8 = 1; RB_16 = 2; RB_32 = 3; RB_64 = 4;</p>
	<p>() 1 0 ();</p>
	<p>() unicode , , 2048 ;</p>
	<p>() ;</p>
	<p>() ansi , ;</p>
	<p>() ;</p>
	<p>() 0 1 ();</p>
	<p>() ;</p>
	<p>() ;</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
ID	<p>() ansi , ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 (see);</p>
	<p>() ;</p>

	() 1 0 ();
	() (Enigma API EP_MiscCountryCode);
	() 1 0 ();
	() ;
	() ;
	() ;
	() 1 0 ();
	() ;
	() ;
	() ;
	() (-);
	() 16 ();
	() (-);
	() (-);

Show/Hide Delphi structure definition

Show/Hide C++ structure definition

Show/Hide C# (.NET) structure definition

Show/Hide Visual Basic structure definition

KG_GenerateRegistrationKey

ansi
PKeyGenParams TGenKeyParams (ansi -) -

EP_NO_ERROR=0	
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Visual Basic function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Borland C++ Builder function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

[⊕ Show/Hide Visual Basic function example](#)

, Examples\Keygen

KG_GenerateRegistrationKeyFromProject

KG_GenerateRegistrationKey ()
TGenKeyParam KG_GenerateRegistrationKeyFromProject ansi ,

EP_NO_ERROR=0	
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Visual Basic function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Borland C++ Builder function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

[⊕ Show/Hide Visual Basic function example](#)

, Examples\Keygen

TKeyGenParamsA

TKeyGenParamsA

KG_GenerateRegistrationKeyA

KG_GenerateRegistrationKeyFromProjectA

	<p>() (RSA 512/768/1024/2048/3072/4096), (-</p> <p>/):</p> <p>RM_512 = 0;</p> <p>RM_768 = 1;</p> <p>RM_1024 = 2;</p> <p>RM_2048 = 3;</p> <p>RM_3072 = 4;</p> <p>RM_4096 = 5;</p>
	<p>() (Base 2/8/16/32/64), (-</p> <p>):</p> <p>RB_2 = 0;</p> <p>RB_8 = 1;</p> <p>RB_16 = 2;</p> <p>RB_32 = 3;</p> <p>RB_64 = 4;</p>
	<p>() 1 0 ();</p>
	<p>() unicode , , 2048 ;</p>
	<p>() ;</p>
	<p>() ansi , ;</p>
	<p>() 0 1 ();</p>
	<p>() ;</p>
	<p>() ;</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
ID	<p>() ansi , ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>

	()	(Enigma API EP_MiscCountryCode));
	()	1	0 ();
	()		;
	()		;
	()		;
	()	1	0 ();
	()		;
	()		;
	()		;
	()		(-);
	()	16	();
	()		(-);
	()		(-);

[Show/Hide Delphi structure definition](#)

[Show/Hide C++ structure definition](#)

[Show/Hide C# \(.NET\) structure definition](#)

KG_GenerateRegistrationKeyA

KG_GenerateRegistrationKeyA KG_GenerateRegistrationKey ansi
- PKeyGenParamsA TGenKeyParamsA (
-)

Return Values

EP_NO_ERROR=0	
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY =2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY =4	
EP_ERROR_REGINFOTOOLARGE =5	()
EP_ERROR_PRIVATEKEYISNOTSET =6	
EP_ERROR_PUBLICKEYISNOTSET =7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY =16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY =17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE =19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

, Examples\Keygen

KG_GenerateRegistrationKeyFromProjectA

```
KG_GenerateRegistrationKeyFromProjectA  
    (  
KG_GenerateRegistrationKeyFromProjectA  
TGenKeyParamA
```

```
KG_GenerateRegistrationKeyFromProject  
    )  
ansi
```

TGenKeyParamA

EP_NO_ERROR=0	
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

, Examples\Keygen

TKeyGenParamsW

TKeyGenParamsW

KG_GenerateRegistrationKeyW

KG_GenerateRegistrationKeyFromProjectW

	<p>() (RSA 512/768/1024/2048/3072/4096), (- /): RM_512 = 0; RM_768 = 1; RM_1024 = 2; RM_2048 = 3; RM_3072 = 4; RM_4096 = 5;</p>
	<p>() (Base 2/8/16/32/64), (-): RB_2 = 0; RB_8 = 1; RB_16 = 2; RB_32 = 3; RB_64 = 4;</p>
	<p>() 1 0 ();</p>
	<p>() unicode , , 4096 ;</p>
	<p>() ;</p>
	<p>() unicode ;</p>
	<p>() 0 1 ());</p>
	<p>(ter) ;</p>
	<p>() ;</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
ID	<p>() ID ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>
	<p>() 1 0 ();</p>
	<p>() ;</p>

	() 1 0 ();
	() (Enigma API EP_MiscCountryCode);
	() 1 0 ();
	() ;
	() ;
	() ;
	() 1 0 ();
	() ;
	() ;
	() ;
	() (-);
	() 16 ();
	() (-);
	() (-).

Show/Hide Delphi structure definition

Show/Hide C++ structure definition

Show/Hide C# (.NET) structure definition

KG_GenerateRegistrationKeyW

KG_GenerateRegistrationKeyW
 univode
).

KG_GenerateRegistrationKey
 -
 TGenKeyParamsW

unicode
 (

EP_NO_ERROR=0	
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

, Examples\KeygenUnicode

KG_GenerateRegistrationKeyFromProjectW

KG_GenerateRegistrationKeyFromProjectW KG_GenerateRegistrationKeyW unicode
 unicode - (
 TGenKeyParamW KG_GenerateRegistrationKeyFromProjectW uicode ,
 , TGenKeyParamW

Return Values

EP_NO_ERROR=0	
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

, Examples\KeygenUnicode

TKeyVerifyParams

TKeyVerifyParams

[KG_VerifyRegistrationInfo](#)

[KG_VerifyRegistrationInfoFromProject](#)

	<p>() (RSA 512/768/1024/2048/3072/4096), (-)</p> <p>/):</p> <p>RM_512 = 0; RM_768 = 1; RM_1024 = 2; RM_2048 = 3; RM_3072 = 4; RM_4096 = 5;</p>
	<p>() (Base 2/8/16/32/64), (-):</p> <p>RB_2 = 0; RB_8 = 1; RB_16 = 2; RB_32 = 3; RB_64 = 4;</p>
	() 2048
	()
	()
	()
	()
	()
	()
	() 0 , 1 ()
	() ;
	() ;
	() ;
	() 1 0 () ;
ID	() ;
	() , 1 0 () ;
	() ;
	() 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;

	() ;
	() , 1 0 ();
	() (Enigma API EP_MiscCountryCode);
	() ();
	() ;
	() ;
	() ;
	() (, 1 0);
	() ;
	() ;
	() ;
	() (-);
	() 16 ();
	() (-);

[⊕ Show/Hide Delphi structure definition](#)

[⊕ Show/Hide C++ structure definition](#)

[⊕ Show/Hide C# \(.NET\) structure definition](#)

[⊕ Show/Hide Visual Basic structure definition](#)

KG_VerifyRegistrationInfo

KG_VerifyRegistrationInfo
the Enigma Protector

TKeyVerifyParams
(-)

TKeyVerifyParams
()

EP_NO_ERROR=0	;
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Visual Basic function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Borland C++ Builder function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

[⊕ Show/Hide Visual Basic function example](#)

, Examples\Keygen

KG_VerifyRegistrationInfoFromProject

KG_VerifyRegistrationInfoFromProject

TKeyVerifyParams

ansi

TKeyVerifyParams

EP_NO_ERROR=0	;
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Visual Basic function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Borland C++ Builder function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

[⊕ Show/Hide Visual Basic function example](#)

, Examples\Keygen

TKeyVerifyParamsA

TKeyVerifyParamsA

[KG_VerifyRegistrationInfoA](#)

[KG_VerifyRegistrationInfoFromProjectA](#)

	<p>() (RSA 512/768/1024/2048/3072/4096), (-)</p> <p>/):</p> <p>RM_512 = 0; RM_768 = 1; RM_1024 = 2; RM_2048 = 3; RM_3072 = 4; RM_4096 = 5;</p>
	<p>() (Base 2/8/16/32/64), (-):</p> <p>RB_2 = 0; RB_8 = 1; RB_16 = 2; RB_32 = 3; RB_64 = 4;</p>
	() ;
	() ;
	() ;
	()
	()
	() 0 1 ()
	() ;
	() ;
	() ;
	() 1 0 () ;
ID	() ;
	() , 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;
	() (Enigma API EP_MiscCountryCode) ;
	() 1 0 ()

);	
	()	;	
	()	;	
	()	;	
	()	;	
	());	, 1 0
	()	;	
	()	;	
	()	;	
	()	(-);	
	() 16 Keys);	(Creating
	());	(-	

Show/Hide Delphi structure definition

Show/Hide C++ structure definition

Show/Hide C# (.NET) structure definition

KG_VerifyRegistrationInfoA

KG_VerifyRegistrationInfoA

KG_VerifyRegistrationInfo

TKeyVerifyParamsA

TKeyVerifyParamsA

the Enigma Protector

(-)

EP_NO_ERROR=0	;
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

, Examples\Keygen

KG_VerifyRegistrationInfoFromProjectA

KG_VerifyRegistrationInfoFromProjectA

ansi

TKeyVerifyParamsA

TKeyVerifyParamsA

EP_NO_ERROR=0	;
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

, Examples\Keygen

TKeyVerifyParamsW

TKeyVerifyParamsW

KG_VerifyRegistrationInfoW

KG_VerifyRegistrationInfoFromProjectW

	<p>() (RSA 512/768/1024/2048/3072/4096), (- /);</p> <p>RM_512 = 0; RM_768 = 1; RM_1024 = 2; RM_2048 = 3; RM_3072 = 4; RM_4096 = 5;</p>
	<p>() (Base 2/8/16/32/64), (-):</p> <p>RB_2 = 0; RB_8 = 1; RB_16 = 2; RB_32 = 3; RB_64 = 4;</p>
	()unicode , ;
	() ;
	()unicode , ;
	()
	()
	()
	() 0 1 () ;
	() ;
	() ;
	() ;
	() 1 0 () ;
ID	() unicode , ID ;
	() 1 0 () ;
	() ;
	() 1 0 () ;
	() ;
	() , 1 0 () ;
	() ;
	() 1 0 () ;
	() ;
	() 1 0 () ;

	());	(Enigma API	EP_MiscCountryCode
	()	1	0	();
	() ;			
	() ;			
	() ;			
	()	1	0	();
	() ;			
	() ;			
	() ;			
	()	(-);
	()	16	();
	());		(-

Show/Hide Delphi structure definition

Show/Hide C++ structure definition

Show/Hide C# (.NET) structure definition

KG_VerifyRegistrationInfoW

KG_VerifyRegistrationInfoW

TKeyVerifyParamsW

the Enigma Protector

TKeyVerifyParamsW

(

EP_NO_ERROR=0	;
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	()
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

, Examples\KeygeUnicode

KG_VerifyRegistrationInfoFromProjectW

KG_VerifyRegistrationInfoFromProjectW

unicode

TKeyVerifyParamsW

TKeyVerifyParamsW

EP_NO_ERROR=0	;
EP_ERROR_UNKNOWN=1	
EP_ERROR_KEYBUFFEREMPTY=2	
EP_ERROR_KEYBUFFERISLESS=3	
EP_ERROR_REGINFOEMPTY=4	
EP_ERROR_REGINFOTOOLARGE=5	
EP_ERROR_PRIVATEKEYISNOTSET=6	
EP_ERROR_PUBLICKEYISNOTSET=7	
EP_ERROR_PRIVATEKEYISINVALID=8	
EP_ERROR_PUBLICKEYISINVALID=9	
EP_ERROR_KEYMODEISINVALID=10	
EP_ERROR_KEYBASEISINVALID=11	
EP_ERROR_CURRENTDATEISINVALID=12	
EP_ERROR_EXPIRATIONDATEISINVALID=13	
EP_ERROR_KEYISINVALID=14	
EP_ERROR_HARDWAREID=15	ID
EP_ERROR_HARDWAREBUFFEREMPTY=16	ID
EP_ERROR_HARDWAREIDINVALIDFORKEY=17	ID
EP_ERROR_PROJECTFILENOTFOUND=18	
EP_ERROR_INVALIDPROJECTFILE=19	
EP_ERROR_EXECUTIONSNUMBERINVALID=20	
EP_ERROR_DAYSNUMBERINVALID=21	
EP_ERROR_COUNTRYCODEINVALID=22	
EP_ERROR_RUNTIMEINVALID=23	
EP_ERROR_GLOBALTIMEINVALID=24	
EP_ERROR_INSTALLBEFOREINVALID=25	
EP_ERROR_INSTALLAFTERINVALID=26	

[⊕ Show/Hide Delphi function definition](#)

[⊕ Show/Hide C++ function definition](#)

[⊕ Show/Hide C# \(.NET\) function definition](#)

[⊕ Show/Hide Delphi function example](#)

[⊕ Show/Hide Visual C++ function example](#)

[⊕ Show/Hide C# \(.NET\) function example](#)

Examples\KeygeUnicode



Please Note:

- | Markers cannot be used in DOT NET applications;
- | Visual Basic 6 applications should be compiled in native mode (not pcode).

Enigma Protector

- | VM -
- | Reg_Crypt -
- | UnReg_Crypt -
- | Decrypt_On_Execute -
- | Unprotected -
- | Check_Protection -
- | Run_Once -

Enigma Protector

```
{WRONG}
{$I include\reg_crypt_end1.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
{$I include\reg_crypt_begin1.inc}
```

```
{RIGHT}
{$I include\reg_crypt_begin1.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
{$I include\reg_crypt_end1.inc}
```

```
{WRONG}
{$I include\reg_crypt_begin1.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
{$I include\reg_crypt_begin2.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
{$I include\reg_crypt_end2.inc}
```

```
{RIGHT}
{$I include\reg_crypt_begin1.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
{$I include\reg_crypt_end1.inc}
{$I include\reg_crypt_begin2.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
{$I include\reg_crypt_end2.inc}
```

```
{WRONG}
{$I include\reg_crypt_begin1.inc}
  {$I include\reg_crypt_begin2.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
  {$I include\reg_crypt_end2.inc}
{$I include\reg_crypt_end1.inc}
```

```
{RIGHT}
{$I include\reg_crypt_begin1.inc}
  MessageBox(0, #10#13'This message appears only if application is registered' +
             #10#13'and section #1 unlocked by registration key' +
             #10#13, 'Application', 0);
```

```
{ $I include\reg_crypt_end1.inc }
```

```
{WRONG}  
{ $I include\reg_crypt_begin1.inc }  
if EP_RegLoadKey(pcUserInfo, pcKey) then  
begin  
    eName.Text := string(pcUserInfo);  
    { $I include\reg_crypt_end1.inc }  
    eKey.Text := string(pcKey);  
    ShowMessage('Thanks for registration');  
end;
```

```
{RIGHT}  
{ $I include\reg_crypt_begin1.inc }  
if EP_RegLoadKey(pcUserInfo, pcKey) then  
begin  
    eName.Text := string(pcUserInfo);  
    eKey.Text := string(pcKey);  
    ShowMessage('Thanks for registration');  
end;  
{ $I include\reg_crypt_end1.inc }
```

If-Then-Else, For-To-Do, While-Do, Repeat-Until, Try-Except, Try-Finally

```
{WRONG}  
if EP_RegLoadKey(pcUserInfo, pcKey) then  
begin  
    { $I include\reg_crypt_begin1.inc }  
    eName.Text := string(pcUserInfo);  
    eKey.Text := string(pcKey);  
    ShowMessage('Thanks for registration');  
    { $I include\reg_crypt_end1.inc }  
end;
```

```
{RIGHT}  
{ $I include\reg_crypt_begin1.inc }  
if EP_RegLoadKey(pcUserInfo, pcKey) then  
begin  
    eName.Text := string(pcUserInfo);  
    eKey.Text := string(pcKey);  
    ShowMessage('Thanks for registration');  
end;  
{ $I include\reg_crypt_end1.inc }
```

C C++ return

```
{WRONG}  
void CTestDlg::CheckRegistered(BOOL bReg)  
{  
    CWnd* wnd;  
    // Enable/disable unregister button  
    wnd = GetDlgItem(IDC_BUTTON_UNREGISTER);  
    wnd->EnableWindow(bReg);  
  
    // Enable/disable unregister static  
    wnd = GetDlgItem(IDC_BUTTON_REGISTER);  
    wnd->EnableWindow(!bReg);  
    // Enable/disable user info edit  
    wnd = GetDlgItem(IDC_EDIT_USERINFO);  
    wnd->EnableWindow(!bReg);  
    // Enable/disable key edit
```

```

wnd = GetDlgItem(IDC_EDIT_KEY);
wnd->EnableWindow(!bReg);

char* sName = NULL;
char* sKey = NULL;
{$I include\check_protection_begin.inc}
if (bReg)
{
    if (EP_RegLoadKey(&sName, &sKey))
    {
        SetDlgItemText(IDC_EDIT_USERINFO, sName);
        SetDlgItemText(IDC_EDIT_KEY, sKey);
        return;
    }
}
{$I include\check_protection_end.inc}
}

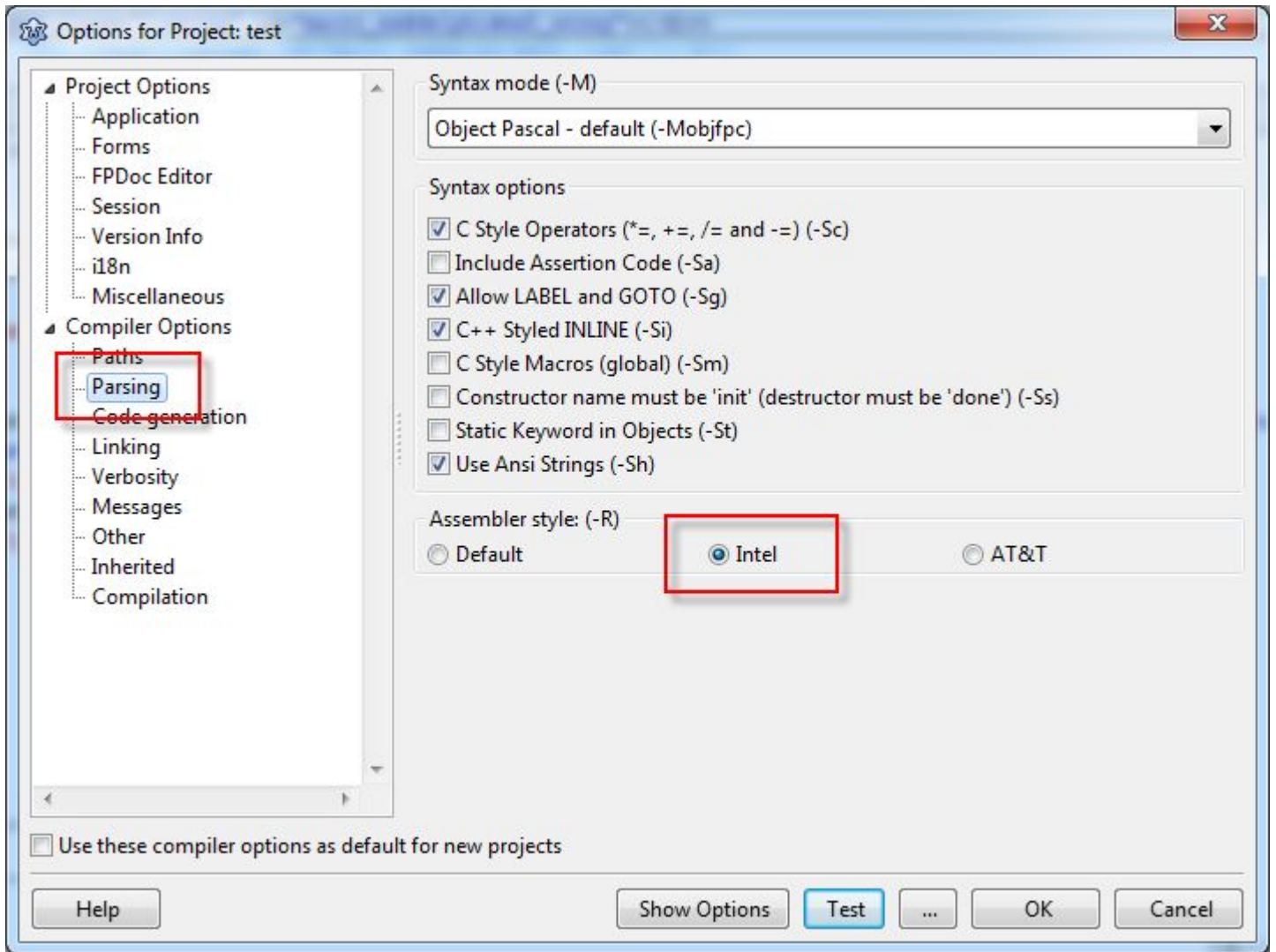
{RIGHT}
void CTestDlg::CheckRegistered(BOOL bReg)
{
    CWnd* wnd;
    // Enable/disable unregister button
    wnd = GetDlgItem(IDC_BUTTON_UNREGISTER);
    wnd->EnableWindow(bReg);

    // Enable/disable unregister static
    wnd = GetDlgItem(IDC_BUTTON_REGISTER);
    wnd->EnableWindow(!bReg);
    // Enable/disable user info edit
    wnd = GetDlgItem(IDC_EDIT_USERINFO);
    wnd->EnableWindow(!bReg);
    // Enable/disable key edit
    wnd = GetDlgItem(IDC_EDIT_KEY);
    wnd->EnableWindow(!bReg);

    char* sName = NULL;
    char* sKey = NULL;
    {$I include\check_protection_begin.inc}
    if (bReg)
    {
        if (EP_RegLoadKey(&sName, &sKey))
        {
            SetDlgItemText(IDC_EDIT_USERINFO, sName);
            SetDlgItemText(IDC_EDIT_KEY, sKey);
        }
    }
    {$I include\check_protection_end.inc}
    return;
}

```

- 1 Free Pascal users should enable Intel Assembler Style in Compiler Options.



Examples\MarkerVM

Content

[+ Show/Hide Delphi marker content](#)

[+ Show/Hide C++ marker content](#)

[+ Show/Hide Visual Basic marker content](#)

Examples

[+ Show/Hide Delphi marker example](#)

Examples\MarkerVM

Content

- [+ Show/Hide Delphi marker content](#)
- [+ Show/Hide C++ marker content](#)
- [+ Show/Hide Visual Basic marker content](#)

Examples

- [+ Show/Hide Delphi marker example](#)
- [+ Show/Hide C++ marker example](#)
- [+ Show/Hide Visual Basic marker example](#)

Examples\MarkersRegCrypt

Content

- [+ Show/Hide Delphi marker content](#)
- [+ Show/Hide C++ marker content](#)
- [+ Show/Hide Visual Basic marker content](#)

Examples

- [+ Show/Hide Delphi marker example](#)
- [+ Show/Hide C++ marker example](#)
- [+ Show/Hide Visual Basic marker example](#)

Examples\UnRegCryptMarkers

Decrypt_On_Execute

Decrypt_On_Execute

Content

[⊕ Show/Hide Delphi marker content](#)

[⊕ Show/Hide C++ marker content](#)

[⊕ Show/Hide Visual Basic marker content](#)

Examples

[⊕ Show/Hide Delphi marker example](#)

[⊕ Show/Hide C++ marker example](#)

[⊕ Show/Hide Visual Basic marker example](#)

Examples\MarkersDecryptOnExecute

Check_Protection

Check_Protection

Content

[+ Show/Hide Delphi marker content](#)

[+ Show/Hide C++ marker content](#)

[+ Show/Hide Visual Basic marker content](#)

Examples

[+ Show/Hide Delphi marker example](#)

[+ Show/Hide Visual Basic marker example](#)

Examples\CheckProtection

Run_Once

Run_Once

Content

[+ Show/Hide Delphi marker content](#)

[+ Show/Hide C++ marker content](#)

[+ Show/Hide Visual Basic marker content](#)

Examples

[+ Show/Hide Delphi marker example](#)

Examples\MarkerRunOnce

Unprotected

Unprotected

Content

[+ Show/Hide Delphi marker content](#)

[+ Show/Hide C++ marker content](#)

[+ Show/Hide Visual Basic marker content](#)

Examples

[+ Show/Hide Delphi marker example](#)

[+ Show/Hide Visual Basic marker example](#)

Examples\MarkersUnprotect



|
| 1
| 2

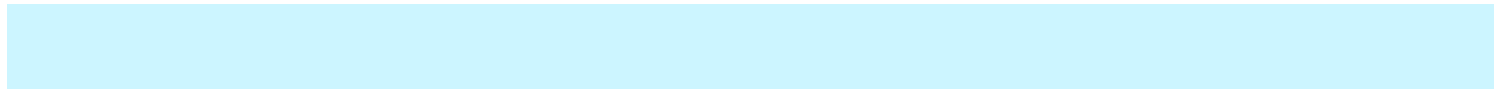
|
|
| " "

Delphi

```
"Hello World"
```

```
"Tutorials\Simple envelope protection\Delphi\"
```

```
program test;  
{$APPTYPE CONSOLE}  
uses  
    Windows, SysUtils;  
begin  
    MessageBox(0, 'This is The Enigma Protector test application.'#10#13 +  
        'If after protection you will see this message then the application works correctly!',  
        'Test Application', 0);  
end.
```

Enigma Protector Enigma Enigma

Protector

- 1.
- 2.
- 3.

a. Enigma Protector

b. " " "

c.

d. " " " "

4.

Windows

Enigma Protector

5.

:

Enigma

3.b

Windows

|

Windows

.reg

```
[HKEY_ROOT\HKEY_RELATIVE]
>Name="User Information"
| Key="XUCBSNJ9T642DCQP3FW8C6NJL75X7"
```

HKEY_ROOT - Windows

HKEY_LOCAL_MACHINE HKEY_CURRENT_USER - " "

HKEY_RELATIVE - Windows " "

User Information -

XUCBSNJ9T642DCQP3FW8C6NJL75X7 -

.reg

Windows

|

```
[Registration information]
Name=UserInfo
| Key=XUCBSNJ9T642DCQP3FW8C6NJL75X7
```

UserInfo -

XUCBSNJ9T642DCQP3FW8C6NJL75X7 -

" "

Delphi

```
"Hello World"
```

```
"Tutorials\Trial protection part 1\Delphi\"
```

```
program test;
{$APPTYPE CONSOLE}
uses
  Windows,
  SysUtils;
begin
  MessageBox(0, 'This is The Enigma Protector test application.'#10#13 +
    'If after protection you will see this message then the application works correctly!',
    'Test Application', 0);
end.
```

Enigma API

Enigma API

API

Enigma API

- | [EP_TrialExecutions](#)
- | [EP_TrialDays](#)
- | [EP_TrialExpirationDate](#)

Enigma API

Tutorials\Trial protection part 2\

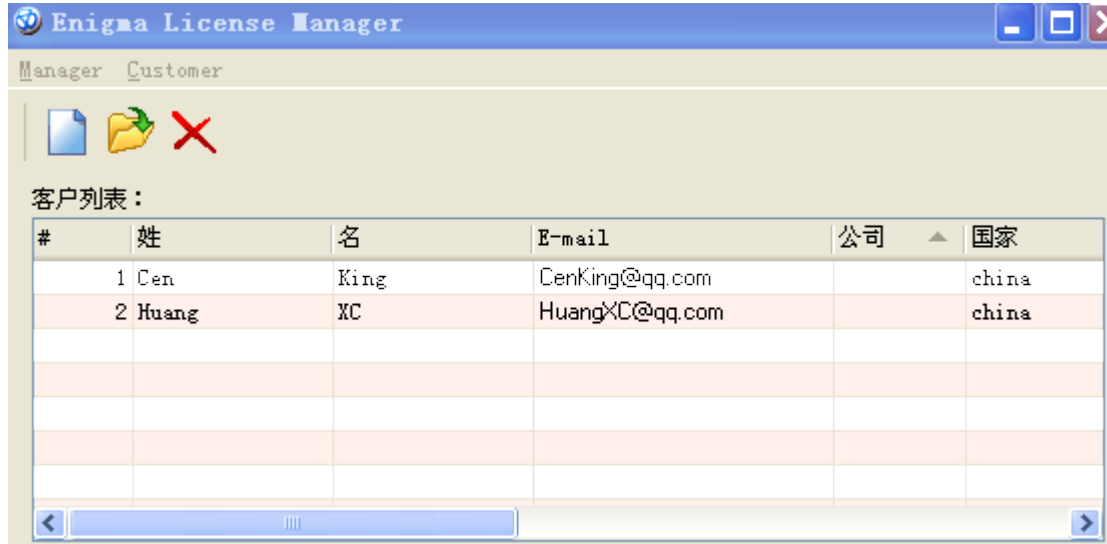
```
program test;
{$APPTYPE CONSOLE}
uses
  Windows,
  SysUtils,
  enigma_ide in 'include\enigma_ide.pas';
var
  Total, Left : word;
  sMessage : string;
begin
  // This is test application of trial Enigma API functions
  // If the function succeed then it returns true
  // If the function failed then
  // 1. the trial limit on executions number was not set before protection
  // 2. here is unexpected error
  if EP_TrialExecutions(Total, Left) then
  begin
    if Left = 0 then
    begin
      MessageBox(0, 'Your trial period has over!', 'Test application', 0);
    end else
    begin
      if (Total div Left) >= 2 then sMessage := 'You have left more then half executions of your trial! period';
      sMessage := sMessage + format('You have left %d from %d days of your trial period!', [Left, Total]);
      MessageBox(0, PChar(sMessage), 'Test application', 0);
    end;
  end;
  // You may also use EP_TrialDays and EP_TrialExpirationDate
  // functions to control trial period
end.
```

Enigma



Enigma Protector



" Customer"-



- | Adding/Editing / (- Customer - /)
- | (- Manager -)
- | (- Manager -)






编辑客户



公司:  



姓:  



名:  



国家: 



语言:  



街道地址:  



省:  

邮编:  


城市:  

电话:  

传真:  

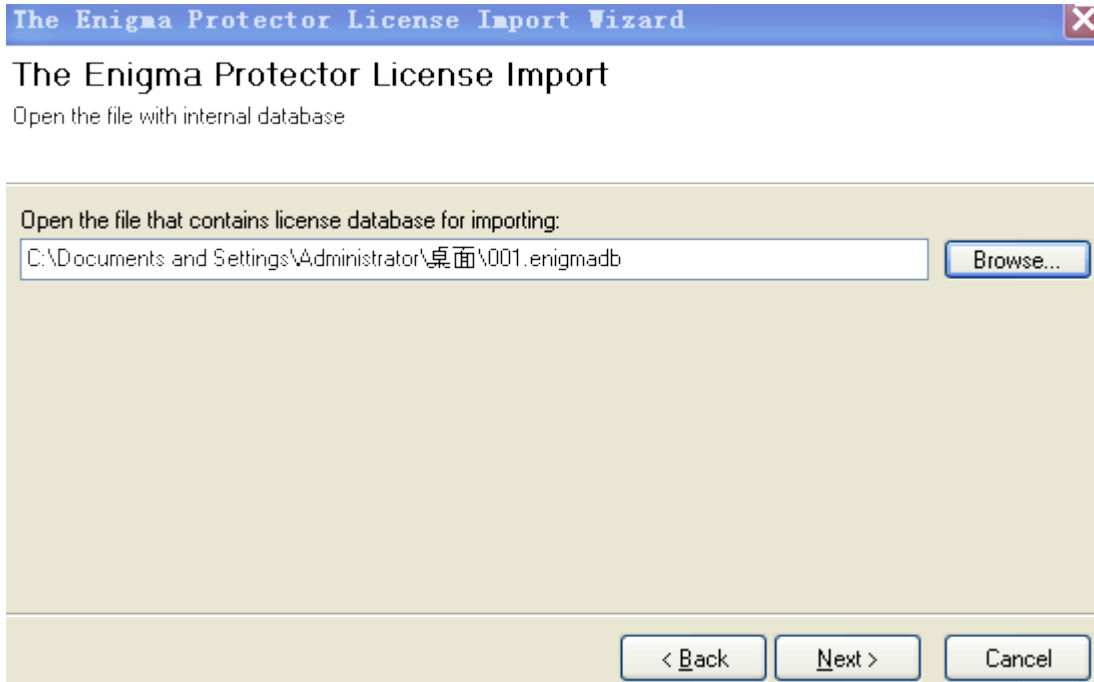
E-mail:  

客户授权

#	已泄漏	已续订	注册名称	注册密钥	硬件 ID
1	<input type="checkbox"/>		综合	Q8ZJKYWN...	

Enigma

1. *.enigmabd " Next"



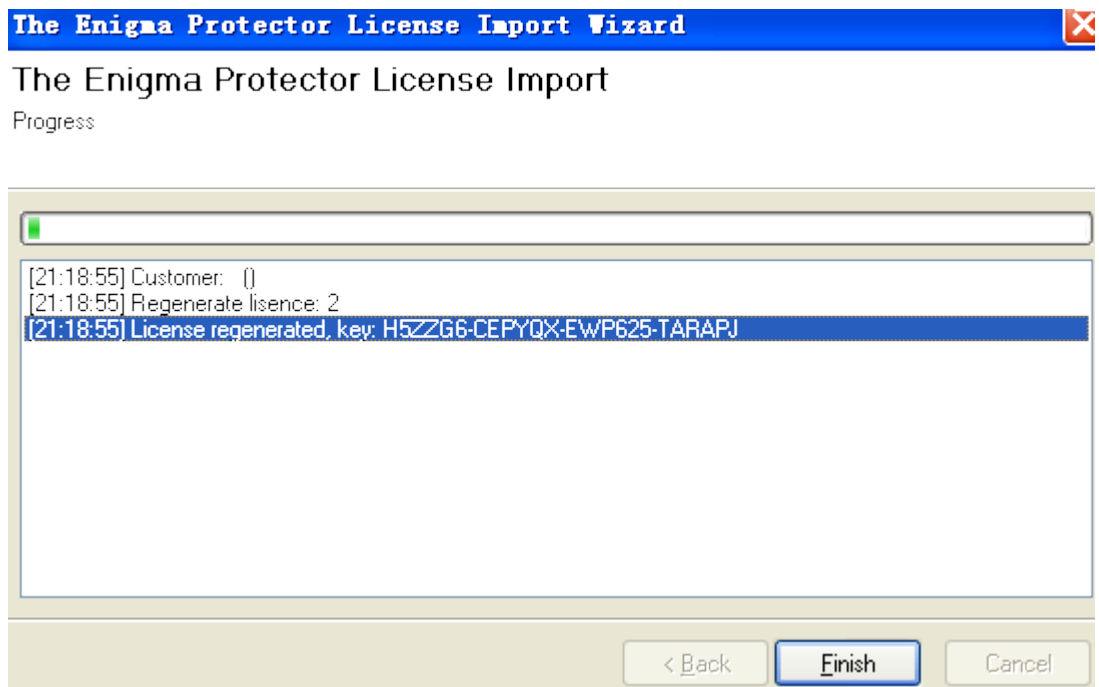
2.



3.

Enigma

Finish



编辑许可

注册名称：
综合

注册密钥：
 添加连字符
 Q8ZJKYWNQLWH6CGSS3YA8Z9S 复制
粘贴

过期日期 2010- 7- 2 运行次数 100
 之后需注册 2010- 7- 2 天 30
 之前需注册 2010- 7- 2 运行时间 10
 国家锁定 Albania 总时间 60

硬件 ID：

选择要解密的区间

<input type="checkbox"/> 区间 1	<input type="checkbox"/> 区间 5	<input type="checkbox"/> 区间 9	<input type="checkbox"/> 区间 13
<input type="checkbox"/> 区间 2	<input type="checkbox"/> 区间 6	<input type="checkbox"/> 区间 10	<input type="checkbox"/> 区间 14
<input type="checkbox"/> 区间 3	<input type="checkbox"/> 区间 7	<input type="checkbox"/> 区间 11	<input type="checkbox"/> 区间 15
<input type="checkbox"/> 区间 4	<input type="checkbox"/> 区间 8	<input type="checkbox"/> 区间 12	<input type="checkbox"/> 区间 16

附加属性

许可数量：

许可价格：

参考序数：

已泄漏的许可
续订 (邮件管理器使用)
 已续订的许可
 续订过期日期：2011- 7- 2

备注：

创建 Additional Parameters

" "

/ - /

ID - ID

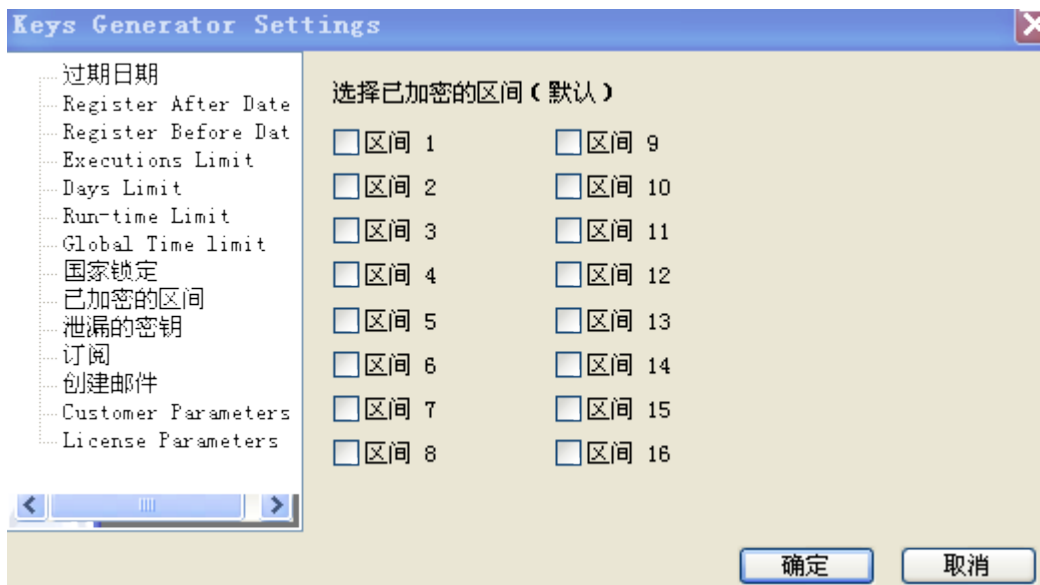
- ID

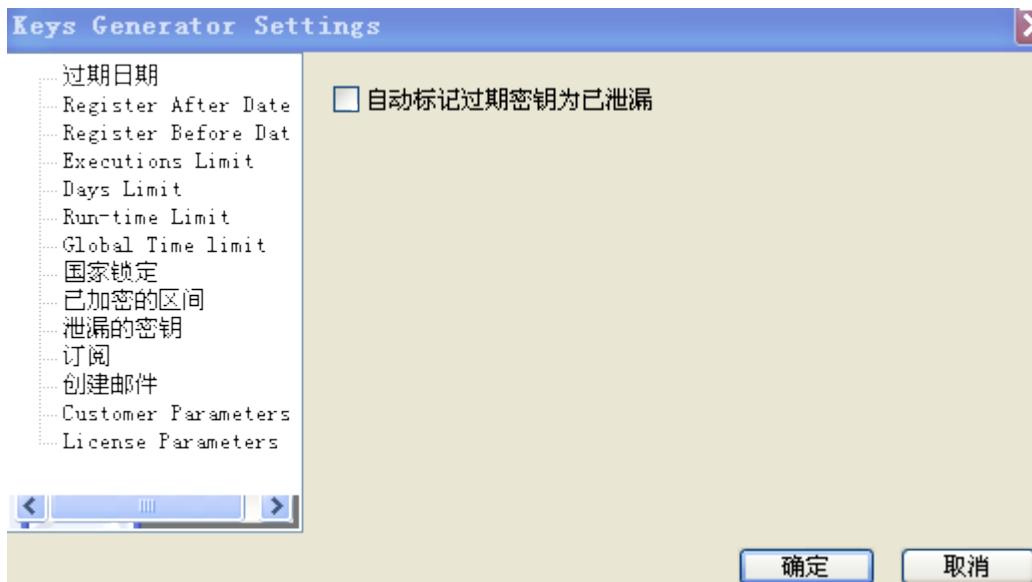
Enigma Mailer

Enigma

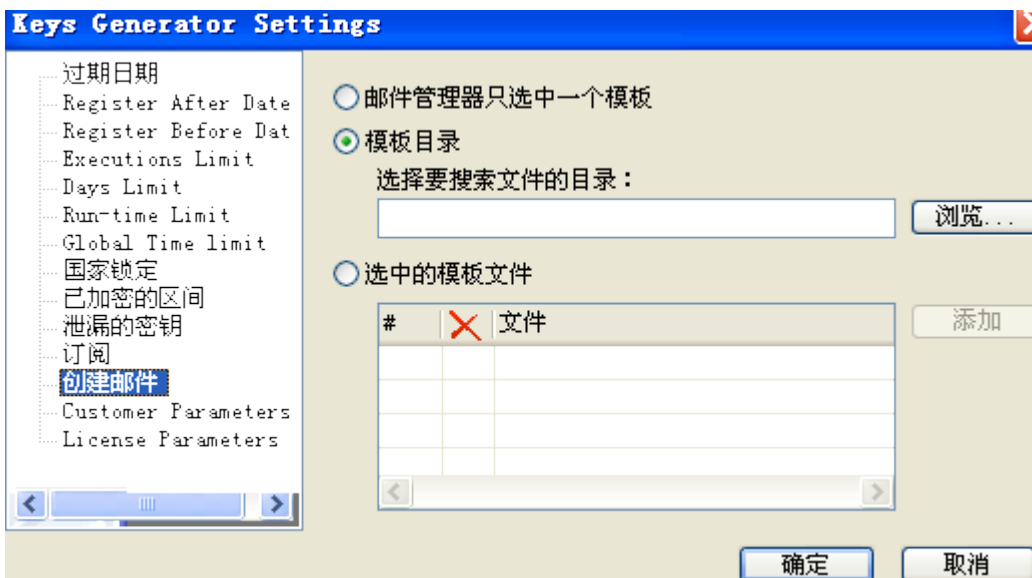
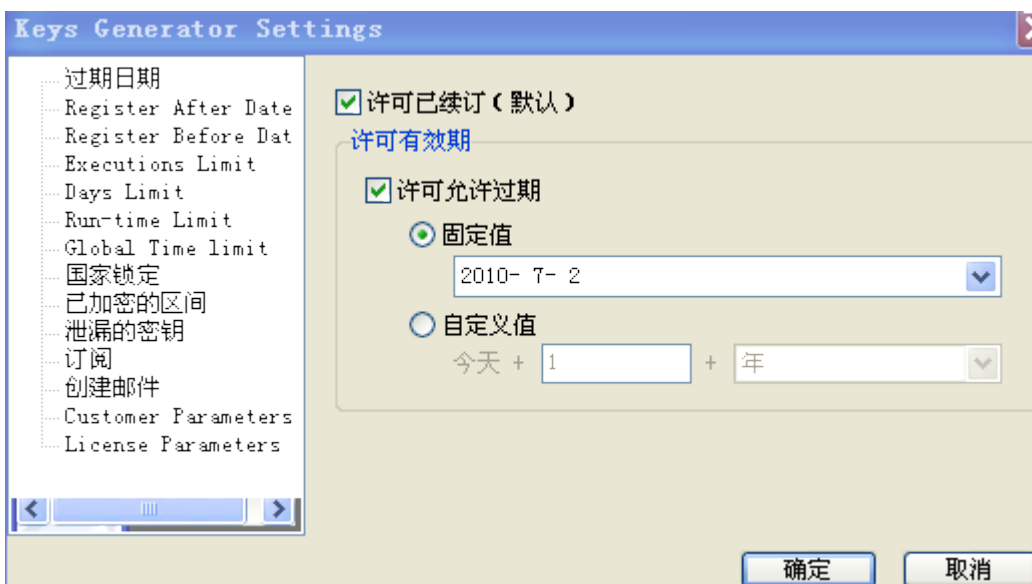
License Manager Properties

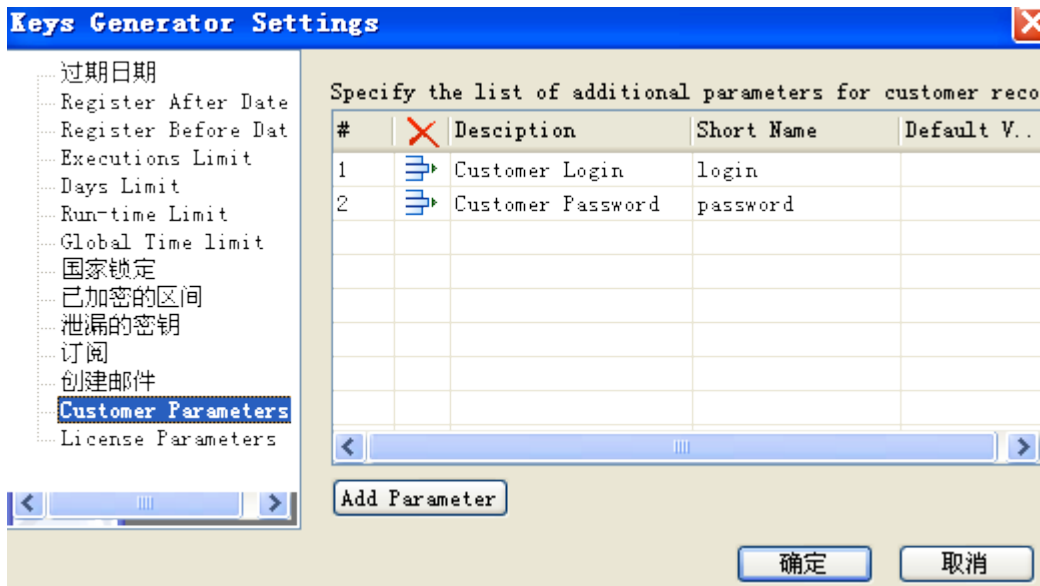
License Manager Properties





- Enigma



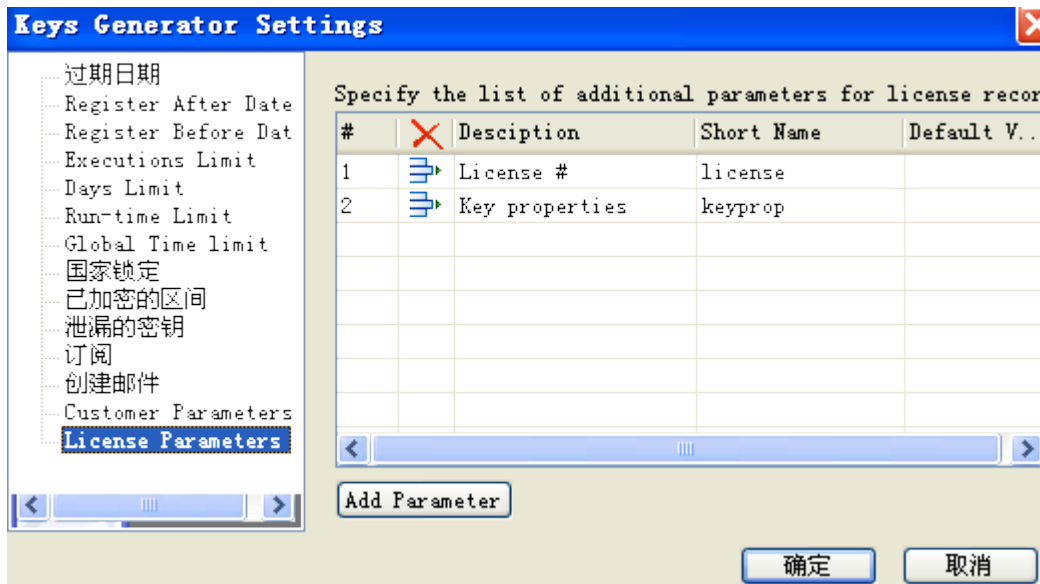


Description() -

Short Name() -

Default Value -

License Parameters -



Description() -

Short Name() -

Default Value -

Enigma

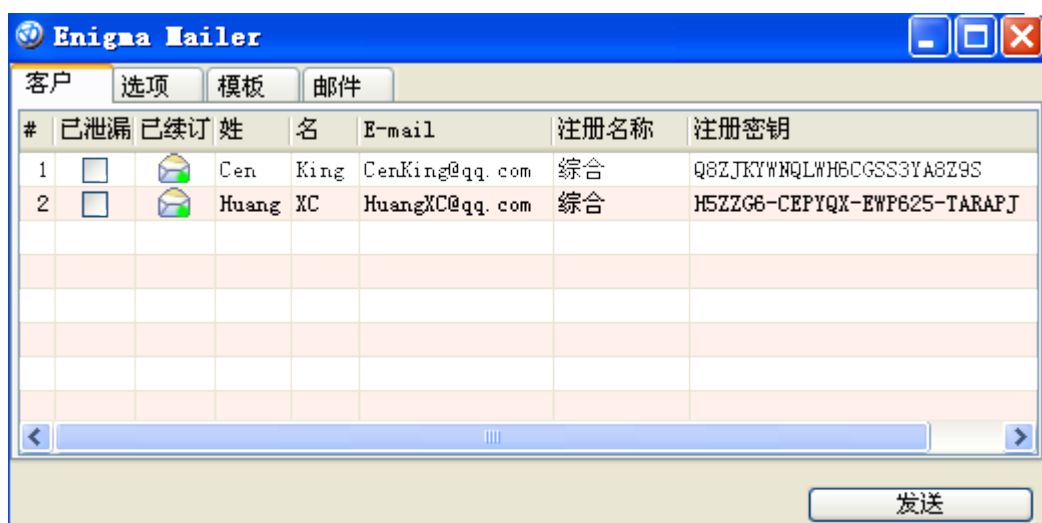
Enigma Protector

Enigma

Enigma

- Enigma

|
|
|
|



Enigma Mailer

客户 选项 模板 邮件

E-Mail Address
“发送”地址（此地址必须和下方的用户名和密码对应）
mail@server.com

帐户信息（用于连接服务器）

用户名： mail
密码： password

邮件服务器

SMTP 服务器： smtp.server.com
端口： 25
 Use SSL (for SSL connection it is recommended to use port

发送延迟
如果服务器错误，延迟发送 这个数值是在服务器停止响应后，再次尝试发送邮件所间隔的分钟。
3 分钟

Email

- Email
- SMTP Email
- SMTP Email
- SMTP Email
- SMTP - SMTP
- Email
-

Email

4

%CUSTOMERCOMPANY%	
%CUSTOMERFIRSTNAME%	
%CUSTOMERLASTNAME%	
%CUSTOMERSTREET%	
%CUSTOMERSTATE%	
%CUSTOMERZIP%	
%CUSTOMERCITY%	
%CUSTOMERCOUNTRY%	
%CUSTOMERLANGUAGE%	
%CUSTOMERPHONE%	
%CUSTOMERFAX%	
%CUSTOMEREMAIL%	Email
%REGNAME%	
%REGKEY%	
%REGKEYCREATEDATE%	
%REGKEYEXPIRATIONDATE%	
%REGKEYHARDWARE%	ID
%REGKEYLICNUM%	
%REGKEYPRICE%	
%REGKEYREFERENCE%	
%REGKEYCOMMENT%	

" Pattern" %CUSTOMER %

" Pattern" %REG %

The screenshot shows the 'Enigma Mailer' application window with the '模板' (Template) tab selected. The interface includes a menu bar with '客户', '选项', '模板', and '邮件'. Below the menu, there is a section for loading templates from files with '打开' (Open) and '另存为' (Save As) buttons. The main area is titled '模板' and contains several input fields: '从:' (From) with 'Software Company', '到:' (To) with '%CUSTOMERFIRSTNAME% %CUSTOMERLASTNAME%', '抄送地址:' (CC Address), '暗送地址:' (BCC Address), '主题:' (Subject) with 'Software Registration Order', and '正文:' (Body) with 'Hello %CUSTOMERFIRSTNAME%'. A '生成 Email' (Generate Email) button is located at the bottom of the template area, and a '发送' (Send) button is at the very bottom of the window.

Emails -

Email

Enigma Protector

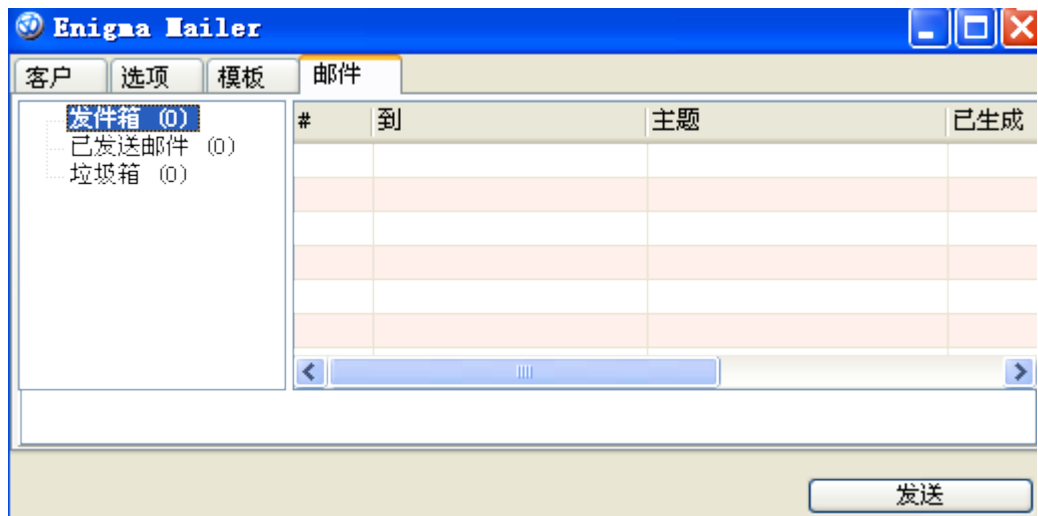
Email

"

"

"

"



Enigma Protector

- 1.
2. CRC
3. Enigma Protector /

Enigma Protector

压缩类型

- 快速压缩
- 最好压缩

压缩方式

- 压缩除图标外的所有资源
- 压缩除图标、版本外的所有资源
- 压缩所有资源

| -
| i -
i -
i -
i -
i -

Enigma Protector

STRINGS, 24, 'TYPELIB', 'REGISTRY',