This software is a must for AutoCAD software developers who wish to protect their software from unauthorized copying.

VisualLISP Copy Protection is a security software that binds your program to run ONLY on the computer on which you have entered an authorization code (key code) .

VisualLISP® Copy Protection

USER GUIDE

How it works

VisualLISP Copy Protection (or more briefly VCP) is a system software protection based on authorization codes related to the customer's computer that hosts your application.

To use the VCP system, the programmer must insert **just one line of code** in its application.

When the customer load your application in AutoCAD the VCP request an authorization code: the 'Key Code'.

To generate the ' Key Code ' the programmer has available a code generator that comes with VCP .

Once you have created the ' Key Code ' send it to the customer. With this code customer will enable the program.

The generation of the ' Key Code ' is made on the basis of two variable parameters that will make the code valid only for the computer for which it was requested and only for that application.
In this way , copying your program on another computer, to enable it, the customer will need a new ' Key Code ' .

VisualLISP Copy Protection consists of 3 elements :

- A **FAS** module to be inserted into your application AutoLISP / VisualLISP

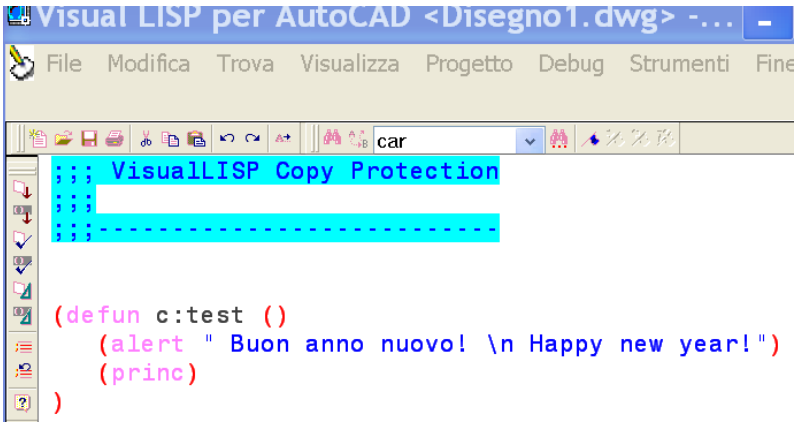
- A **DCL** module to be inserted into your application AutoLISP / VisualLISP

- An executable file , **GENCODE.EXE** , to generate the ' Key Code '

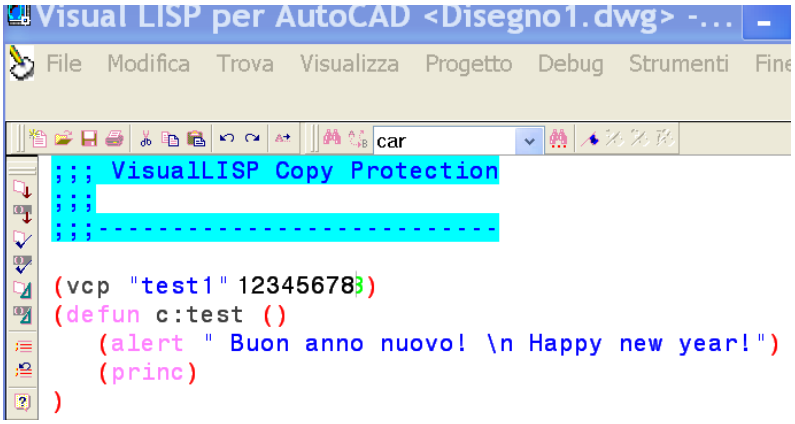
A practical example

Now a simple example to understand how VCP system work.

You want to protect the following program to unauthorized copy.



Insert one line of code to call the VCP function

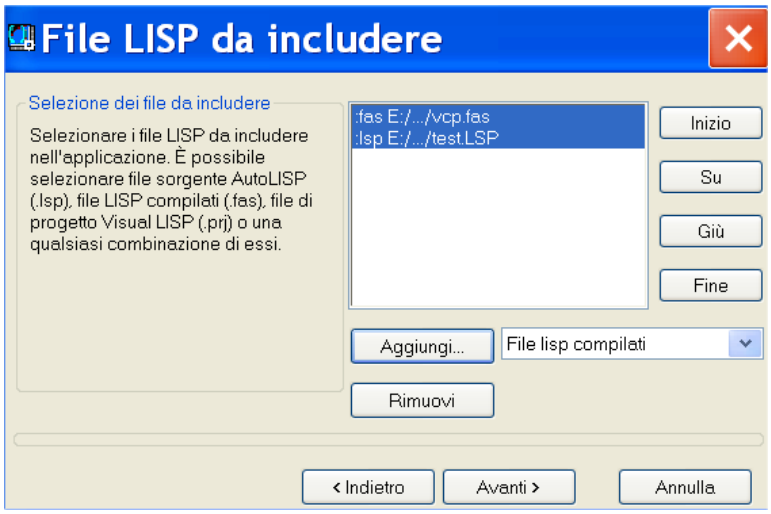


VCP function - contained in VCP.FAS - requires 2 arguments :

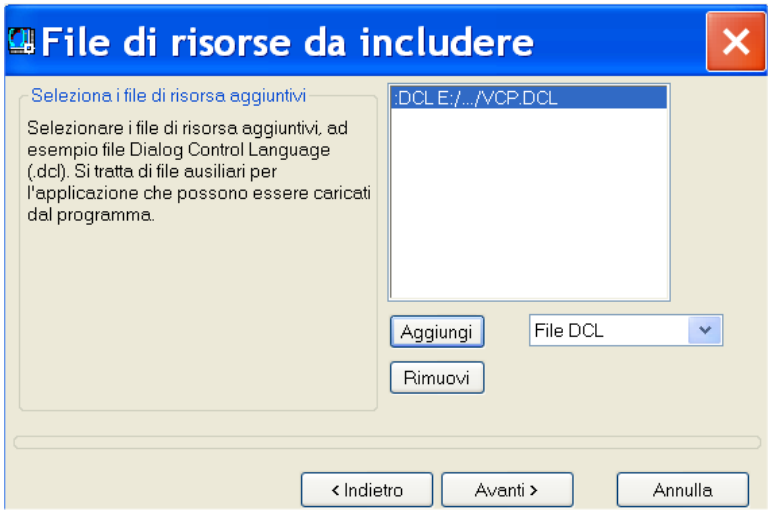
- first is a string that indicates the name of your application (for example, " test", " app1 " etc,)

- second is any integer greater than 1000000

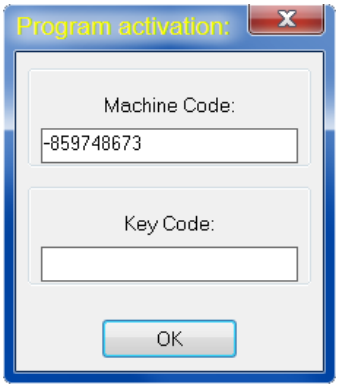
At this point we create the VLX file program which will include the VCP.FAS file …



and the VCP.DCL file (which will display the dialog box for requesting the key code)



Compile and create the file test.vlx ready to be sent to the customer . When the customer load it (with the AutoCAD command Appload), it will see the following window

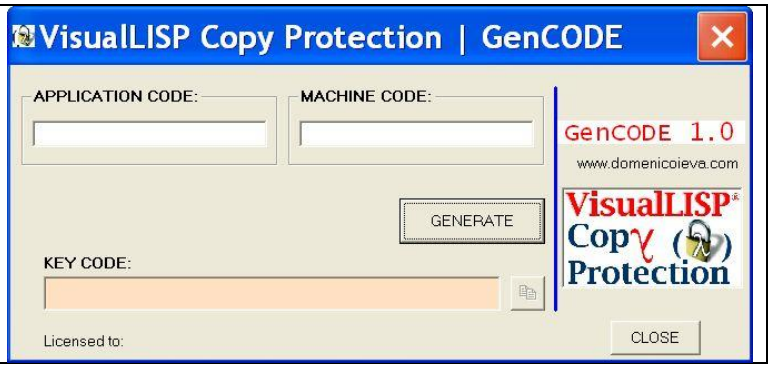


In this window in the first edit box appears the ' Machine Code ' .

This number identifies the computer on which you installed the program. This number will be used to generate the ' Key Code ' and the customer must notify the programmer .

At the bottom of the window there is an edit box in which to enter the ' Key Code ' that the customer will receive from the programmer.

Now you have the two elements to generate the ‘Key Code’. Start the executable GenCODE.exe and inrert :

1 - the ' Application Code ' which is the number used as the second argument of the VCP (in our example, 12345678) ;	
2 - the ' Machine Code ' sent by the client (in our example -859549476)	

included these two values , by selecting the button ' Generate ' , the ' Key Code ' will be generated. You will be sent it to the customer .

The customer, by entering the ' Key Code ' , enable the program .

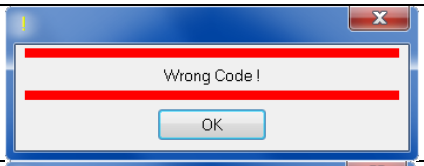
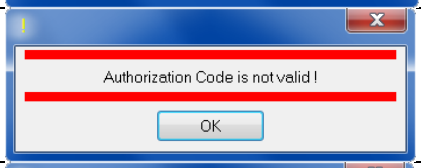
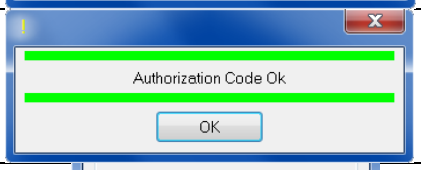
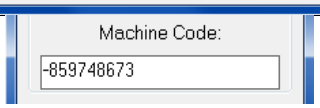
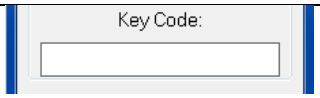
Customizable

You can customize the VisualLISP Copy Protection interface to conform it at main application layout and language.

You can modify the .DCL file, adding other graphics or rearranging existing ones.

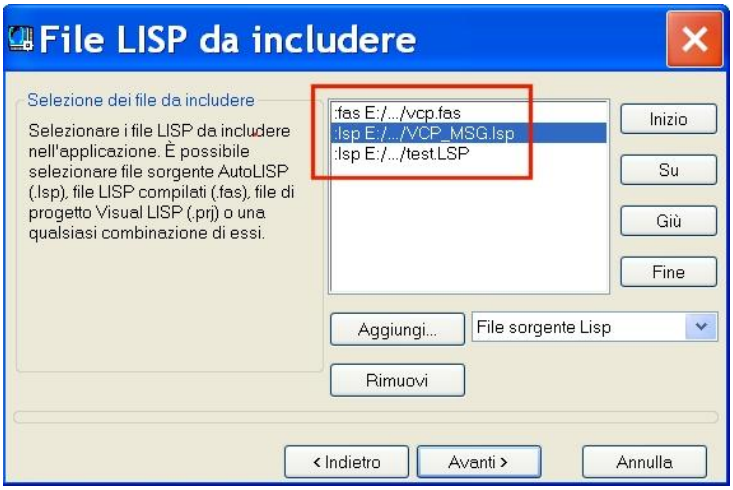
In the file VCP.DCL is important not to change the items that are commented: / * <---- DO NOT EDIT * /

Even prompt message and the dialog box are customizable. There are 6 internal messages :

"Wrong Code !"	Invoked when you enter an incorrect key code , or null	
"Authorization Code is not valid !"	Invoked when it detects a machine code and a incorrect key code	
"Authorization Code Ok"	Invoked when the code is entered correctly	
"Machine Code:"	1th edit_box title of the dialog box	
"Key Code:"	2nd edit_box title of the dialog box	
"Program activation"	dialog box title	

If you want to change them you have to edit the **VCP_MSG.LSP** file and add this file in the files to be included in the .VLX application.

It's' important add VCP_MSG.LSP file **after** VCP.FAS , as shown below:



Formatting HD

Formatting the HDD may need to provide, to the customer, again the ' Key Code ' because VCP function , on loading , would detect a different 'Machine Code ' .

Vcpreset

When customer insert the right ' Key Code ' VCP records in the computer this event.

To remove this record use the vcpreset function.

This function is contained in the file vcp.fas then , to call it , that file must be loaded .

The function need a single argument , of string type, which is the name of the application (ie the first argument used with the function vcp) .

```
( vcpreset "app_name" )
```

After calling the function , a subsequent loading of your application, you will be prompted again the ' Key Code ' .

With this function, the developer can be tests the functionality of VCP on own workstation.

VCP error handling

VCP is structured to push an error that break the Autolisp loading procedure when the user insert a wrong 'key code'.
If you need to manage VCP error you can use **`vl-catch-all-apply`** vlisp function. See an example

```
;; ================================================
;;              Intercept VCP error
;; ================================================

;; call the VCP function with 2 parameters in the list:
;;'("error_test" 123456) and store result in ERRO variable .
(setq erro(vl-catch-all-apply 'VCP '("error_test" 123456)))

(if (vl-catch-all-error-p erro); if ERRO is an error object...
 (progn
  (alert"Exit with error") ; show the alert box and set
  (setvar"errno"1000)      ; the system variable errno to custom value
 )
 (progn ;then all is ok and define my function...
      (alert"Exit with ok")
      (defun c:test ()
            (alert "TEST")
      )
 )
)
```