



# **Manual ArchiCrypt Stealth -Plugin Editor-**

Dok.-Nr.: ACSAF-HB-0003

Issue date: 03.03.2005

Issue-Nr.: 4.0

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without explicit authorization is prohibited. Offenders will be held liable for the payment of damages. All rights reserved.  
ArchiCrypt is a registered trademark of  
Dipl.-Ing. Patric Remus

© 2005 Software Development Dipl.-Ing. Patric Remus

# Inhalt

<b>Teil I Plugin Editor</b>	<b>2</b>
1 Copyright .....	2
2 General .....	2
3 Plugin Editor .....	3
<b>Index</b>	<b>5</b>

# 1 Plugin Editor

## 1.1 Copyright

Copyright

**ArchiCrypt® Stealth Plugin Editor**  
Copyright © 2005 Dipl.-Ing. Patric Remus  
Am Brunneck 6, D-85521 Ottobrunn  
All rights reserved

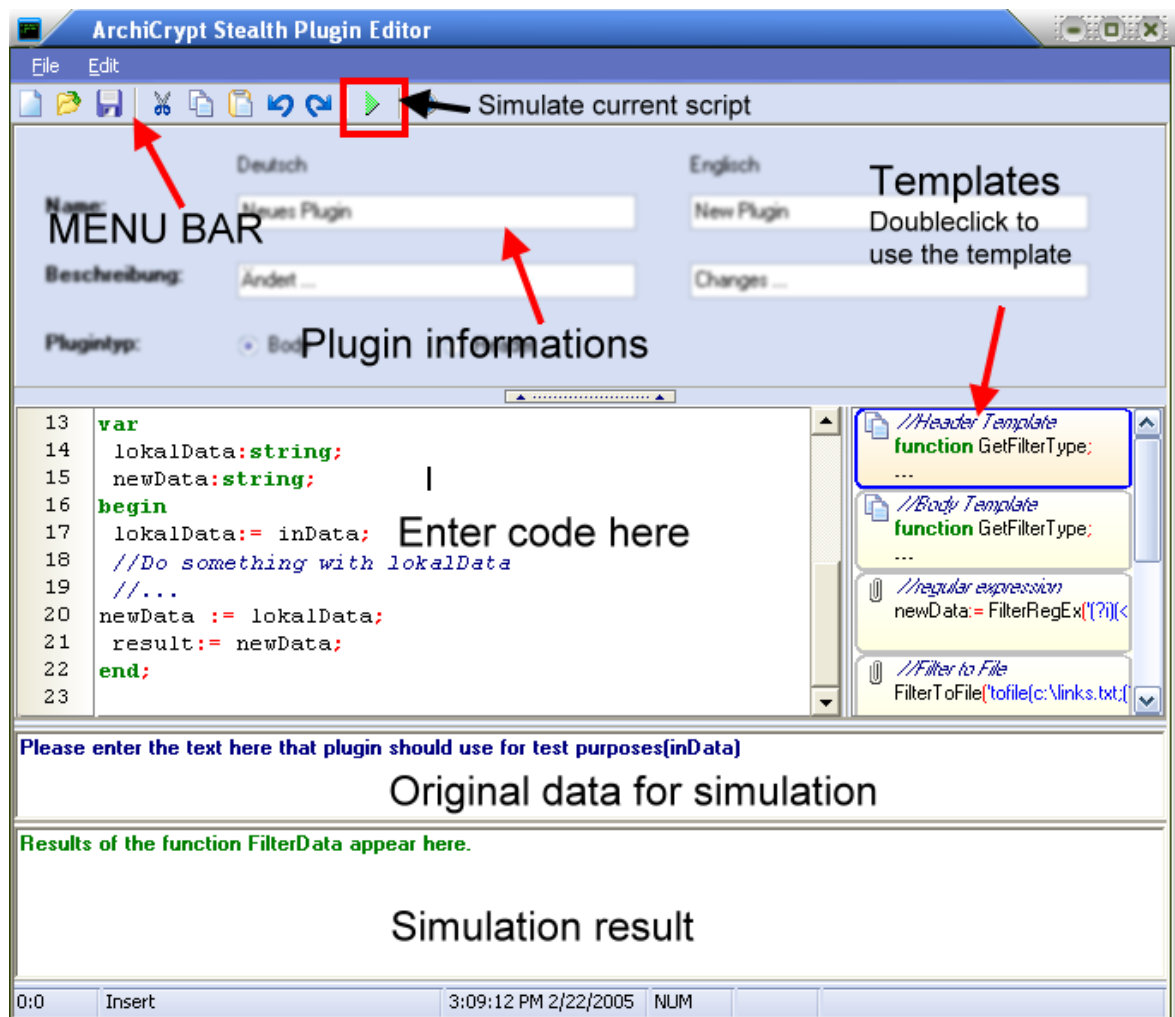
## 1.2 General

"Help" for the ArchiCrypt Stealth Plug-in Editor is not an introduction to the programming language, Delphi/Pascal, nor to the syntax of regular expressions.

You can find very comprehensive information about both issues at numerous places in the Internet. Tips for using individual functions are available, for example, via a search engine if you enter "Delphi" along with the name of the function. For example, "AnsiSameText Delphi"

There are many good sources for regular expressions in the Internet. You can use Perl-Syntax in the plug-ins.

## 1.3 Plugin Editor



What are Plug-ins?

Plug-ins are script files in Pascal syntax

How is a Plug-in structured?

Every plug-in must contain the function:

```
function GetFilterData(inString:string):string;
begin
end;
```

This function will receive the page content currently being received as variable `inString`. With the available functions, you can manipulate `inString` as you like and return the `result`. ArchiCrypt Stealth sends the altered content on to the current browser.

How can you convert existing plug-ins to the new format?

With the help of both functions

```
function FilterRegEx()
```

**function** [FilterToFile](#)()

you can very simply convert existing plug-ins to the new format.

Create a new plug-in and specify the name and a description. Then use the [BodyTemplate](#) and convert the [FilterData](#) function as shown below. At [SEARCHEXPRESSION](#) enter the search expression inside apostrophes; at [TEMPLATE](#) enter the plug-in template inside apostrophes.

```
//Body Template
function GetFilterType;
begin
  result:= 'Body';
end;

function FilterData(inData:string);
var
  lokalData:string;
  newData:string;
begin
  result:= FilterRegEx(SEARCHEXPRESSION,inData,TEMPLATE);
end;
```

Example:

```
function FilterData(inData:string);
var
  lokalData:string;
  newData:string;
begin
  result:= FilterRegEx('(?i)(<title>(.*)</title>)',inData,'<title>Stealth --
$2</title>');
end;
```

In order to convert a plug-in and save the data to a file, you can use the [FilterToFile](#) function.

Example:

```
function FilterData(inData:string);
var
  lokalData:string;
  newData:string;
begin
  lokalData := inData;
  //web site content is not changed!
  FilterToFile('tofilec:\links.txt;(?) (HTTP):(//|\\|\\\)([_a-z\d\-\ ]+(\.[_a-z\d\-\ ]+)((//[_a-z\d\-\ \\\.]+)*)tofile',lokalData,'-');
  result:= inData;
end;
```

Finally, save the plug-in to the plug-in directory. The directory is found in the ArchiCrypt Stealth installation directory.

Which functions are supported?

When entering code you can use the key combination <STRG>+space at any point and will receive a list of all available functions.

Attention: *The script engine cannot interpret pointers. Therefore, using external DLLs is not possible.*

# Index

## - F -

FilterRegEx 3  
FilterToFile 3

## - H -

How can you convert existing plug-ins to the new format? 3  
How is a Plug-in structured? 3

## - R -

regular expressions 2

## - S -

SEARCHEXPRESSION 3

## - W -

What are Plug-ins? 3  
Which functions are supported? 3