Kiskometer_english

Angela Schmidt

COLLABORATORS			
	TITLE : Kiskometer_english		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Angela Schmidt	February 12, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Kiskometer_english

1.1 Index

Author
How to reach the author

Contents
Back to the contents

Copyright
Only non-commercial distribution allowed

Disclaimer
No liability

Getting started
Starting by a simple mouse-click

History
Changes since Version 1.0

Offset files
How to create own offset files

Options
Options for Kiskometer

Syntax
How to call the program

Thanx
People who earn a "thank you"

1.2 Getting started

To help you in working with Kiskometer, I've pre-installed some commands here. Just click on them using your mouse and the actions will be performed.

```
Start Kiskometer  Start Kiskometer using an Auto-Popup-Window
Quit Kiskometer
Redirect Kiskometer's output to the serial device
```

1.3 Kiskometer - Contents

```
              Kiskometer  is  a  program  that  checks  which programs use the  ←
              function
SetFunction()  and  that makes visible which functions have been patched.
You  may  pass a format string and so configure the format of the output.
With  the  help  of  an  extensive  filter you can suppress the output of
specific  calls of SetFunction(). Output occurs to stdout, into a file or
on the serial device - whatever you prefer.
```

```
              Copyright
                      Only non-commercial distribution allowed

              Disclaimer
                      No liability

              Author
                      How to reach the author

              Syntax
                      How to call the program

              Getting started
                  Starting by a simple mouse-click

              Offset files
                      How to create own offset files

              History
                      Changes since Version 1.0

              Thanx
                       People who earn a "thank you"
```

1.4 Syntax

```
              This program can be only started using the CLI. The simplest way  ←
              to start
it  is  by  just typing in its program name without any options. However,
one  should  really  start  it  with "Run", otherwise the CLI is unusable
until the next ^C.

    1> Run Kiskometer

There  appears  a  copyright  notice.  After  that, it seems that nothing
happens, but as soon as any function will be patched, there will appear a
```

patch notice in the CLI-window in which you started Kiskometer.

Do you want to try it? Then start Kai 'kiske' Iske's "MagicCX" or his
"ForceIcon". After that, there should happen something in that CLI
window... Do you now know why this program is called "Kiskometer"...? :-)

If you want more control, you can pass some
                Options
                   .

You can break the program using ^C - except if other programs that have
been started after Kiskometer, also have patched SetFunction() und have
not yet been terminated.

Do you want a quick test? Then look at
                this
                 page!

1.5 Options for Kiskometer

Syntax: Kiskometer [<options>] [<outputfile>] [LIBOFFSDIR <dir>]
or: Kiskometer QUIT | EXIT
or: Kiskometer HELP | ? | -?

You always can pass a filename for Kiskometer. This is the file in which
Kiskometer's output (except copyright notice, error messages etc.) will
be redirected. If you don't pass any filename, output will be done to
stdout, at least if option RAWIO is not given (see below).

If there's already an active Kiskometer, starting a new Kiskometer will
only cause the old Kiskometer to update some options. After updating the
options, the new Kiskometer will quit at once. The only thing you cannot
update is "LIBOFFSDIR".

You can quit Kiskometer by sending ^C or by starting a new Kiskometer
with either option QUIT or option EXIT.

If you pass "HELP", you will get a large help-page for Kiskometer.

The following options can be passed to Kiskometer. If any number is
expected, you can pass this number decimal, hexadecimal or octal. The
beginning of the number decides how this number is to be interpreted:

  1-9   => decimal
  0x or 0X   => hexadecimal
  0   => octal

LIBOFFSDIR <dir>
  Here you can pass the name of the directory which contains the library
  offset files. Default is "PROGDIR:LibOffsets". This option can be only
  passed during the real start of the program, so it cannot be updated
  later.

[IN]ACTIVE

»ACTIVE« activates Kiskometer after it has been des-activated using
option »INACTIVE«. If it is des-activated, it stays in memory (you
still can send "STATUS"-requests to it, the library offset files stay
in memory etc.), but the library vector "SetFunction" will be
completly freed. So after using option "INACTIVE" you'll have a
"clean system" again. If the patch cannot be des-installed – e. g.
because any other program has patched SetFunction() in the meantime –
an error message will occur.

QUIET
  Suppresses output of copyright notice etc.

VERYQUIET
  Does not only suppress the copyright notice, but even all error
  messages. This means that there will be only displayed patch messages.

[NO]RAWIO
RAWIOQUICK
STDIO
  The option RAWIO causes the patch messages to be sent to the serial
  device using KPrintF(). NORAWIO causes output to be written to stdout
  again. Error messages won't be redirected – only patch messages!

  STDIO does exactly the same as NORAWIO.

  Passing option RAWIOQUICK rather than option RAWIO will cause the
  output to be done during the SetFunction() call. But be careful – since
  OS 2.1, this needs a lot of stack and so might cause the patching
  program to crash!

[NO]SHAREOUTFILE
  If you want the output to be written into a file, usually this file
  will be opened exclusively and stays being opened exclusively until it
  is being closed. If SHAREOUTFILE is set and if at least OS 2.04 is
  available, the file will be only in exclusive mode while anything is
  being written to the file. All the other time, the file will be in
  shared mode. This means you can read the file most time. However, it
  might happen that some patch message is being lost because the
  filehandle cannot be turned to the exclusive mode when necessary.

  NOTE: some filesystems are buggy (e.g. the Ram-Disk of OS 3.1), so
  there might be problems with this option. For example, the OS 3.1
  Ram-Disk allows read access even after NOSHAREOUTFILE if SHAREOUTFILE
  has been set before or if the file already exists when opening it.

[NO]SLEEP
  This option puts Kiskometer into a sleeping state or brings it back
  from the sleeping state. While Kiskometer is sleeping, no output
  happens. But as soon as Kiskometer wakes up, all these output lines
  will be catched up.

FORMAT <string>
  With this option you can format the output. The following tokens are
  available:

    %o – Library offset as negative decimal number
    %O – Library offset as function name

```
      %l – Library base address
      %L – Library name
      %t – Task address
      %T – Taskname or name of the CLI process
      %c – Number of all SetFunction calls which have been shown
      %C – Number of all SetFunction calls. Includes those calles which
           have been suppressed by option FILTER
      %a – Address of new function
      %a – Address of old function
      \n – Line feed
      \t – Tabulator character
```

      Additionally, there is a very special token:

      %(<library-pattern>)(<library-offset>) – This token scans the library
         list for any library that matches with <library-pattern>. It stops
         at the first library found and returns the function name of the
         offset <library-offset>. This token is mainly useful for the
         function "STATUS" as described below.

      You can add format characters directly after the %-sign to format
      the   output.   Please   use   the   C-format-characters   to do so.
      Additionally, you can add ",<z>" to the format characers. This strips
      the output to <z> characters and adds three dots if the output was
      longer than <z> characers.

      Default is the following format line:

         "(%c) Offset %4o=%-20,20O of %-20,20L by %T\n"

      If you have a wide output window, I suggest the following line:

         "(%c) Offset %4o=%-20,20O of %-20,20L (%A->%a) by %T\n"

      Through this line you additionally find out about the old and the new
      library function address.

STATUS <formatstring>
  This option directly outputs <formatstring> to stdout. There are the
  same tokens available as described unter "Format", except those which
  only make sense when a program just calls SetFunction – %o, %O, %l, %L,
  %t, %T and %a. If you create a nice alias, you get a nice, fast
  function which prints the function name of a special library offset.

FILTER <filter>
  In many cases one does not want every patch to be printed. Therefore
  you can use the following filters:

      (AND arg1 arg2 ...) – And-connection of all arguments.

      (OR arg1 arg2 ...)   – Or-connection of all arguments.

      (NOT arg)   – Negation for the argument.

      (LIBNAME <pattern>) – This filter returns TRUE if the name of the
                            library to patch matches with <pattern> and
                            FALSE otherwise.

```
(LIBBASE <address>)    - This filter returns TRUE  if the  library base
                         of the library to patch is <address> and FALSE
                         otherwise.

(TASKNAME <pattern>)   - This filter  returns  TRUE  if the name of the
                         task to patch matches with <pattern> and FALSE
                         otherwise.  If the task that patches a library
                         is a CLI process, the name of the program just
                         loaded will be compared instead.

(TASKADDR <address>)   - This filter  returns TRUE if the  task address
                         of the task which patches a library  equals to
                         <address> and FALSE otherwise.

(FUNCNAME <pattern>)   - This filter  returns TRUE  if the name  of the
                         function to patch matches  with <pattern>  and
                         FALSE otherwise.

(OFFSET <offset>)      - This  filter  returns  TRUE  if  the offset to
                         patch equals to <offset> and FALSE otherwise.

(SAMEADDR)             - This  filter  returns TRUE if the old function
                         address  is  the  same  as  the  new  function
                         address and FALSE otherwise.

(<string>)             - Useful   to   protect   spaces   in  <string>.
                         e.g. "(TASKNAME (Services Manager))"
```

  Example:

  1> Kiskometer FILTER "(AND (LIBNAME dos.library) (FUNCNAME Open))"

  You may create as complicate filters as you want.

1.6 Offset files

This software package comes with an ARexx script which is useful to
convert FD files into a format which can be parsed by Kiskometer. With
the help of these files Kiskometer is able to show the names of the
functions which are just going to be patched.

Syntax:

makelibnames.rexx <source> <dest> <libname>

<source>: Name of the FD file to be converted.
<dest>:   Name of the destination file which should be created.
<libname>: Name of the library, e.g. "exec.library". Please not that this
           is casesensitive!

1.7 Copyright

Kiskometer is Copyright © 1995,1997 by Angela Schmidt. This ←
                software
package may be used freely for non-commercial purposes. Duplication and
distribution of this software package is allowed too, as long as this
happens non-commercially and as long as the software remains unchanged.
Requesting a so-called "copy-fee" ist not allowed. This means you must
not request any penny when you give this software package to other
people. Exceptions require the written permission of the
                Author
                  .

Hereby permission is granted to distribute this software package on the
"Meeting Pearls" CD-ROM series and on "Amiga Developer CD V1.0 April
1996".

1.8 Disclaimer

                Neither the
                Author
                  nor any other person are liable for any damage caused
directly or indirectly by using this software.

1.9 Author

```
My address is:
  Angela Schmidt
  Finkenweg 26
  89233 Neu-Ulm
  Germany

Phone:
  +49 721 695307

EMail:
  Angela.Schmidt@stud.uni-karlsruhe.de
```

Please have in mind that I cannot answer all the letters or email sent to
me!

Thanks for your understanding!

1.10 History

```
Version 1.0 (10-Jun-95)
  - Initial release

Version 1.1 (3-Oct-95)
  - Option  RAWIO  no longer calls KPrintF() during SetFunction() because
    RawDoFmt()  needs  a  lot  of  stack  since OS 2.1 and so might cause
```

```
      crashes in some cases.
    - Option  RAWIOQUICK is new.  For debugging reasons you can still print
      the output to the serial.device during SetFunction().
    - Option NORAWIO now always prints to stdout.
    - Option STDIO is new and works like NORAWIO.
    - The format line now knows a new parameter:  %A shows the old function
      address.
    - One enforcer hit has been removed.

Version 1.2 (5-Apr-96)
    - Added options SHAREOUTFILE and NOSHAREOUTFILE

Version 1.3 (14-Okt-96)
    - If  SetFunction()  was called with a WORD value as offset, Kiskometer
      did  not  work  correctly. I hope this bug is fixed now. Thanx to Ben
      Hutchings  for  the bugreport. Unfortunately I did not find a program
      to test this. ;-)

Version 1.4 (10-Mar-97)
    - Neuer Filter "SAMEADDR". Zum Beispiel nützlich für NSDPatch.
    - Mögliches Problem, das vorkommen konnte, wenn Option "STATUS" benutzt
      wird, während ein SetFunction()-Hit angezeigt wird, gefixt.
```

1.11 Thanx

```
Thanks go to:

Bernhard Möllemann
For very good betatesting and very good bugreports!

Klaus Melchior
For bugreports.

Christoph Dietz
For bugreports and feature enhancement requests.

Ralph Babel
For the hint that RawDoFmt() needs a lot of stack since OS 2.1.

Kai Iske
For his name. :-)
```