



# Cutter

A graphical user interface for  
radare2

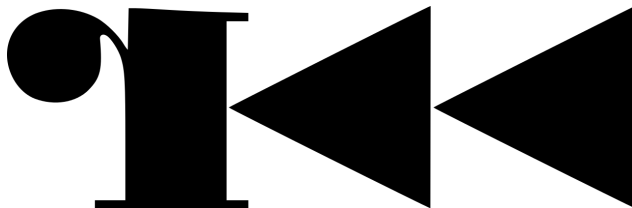
@xarkes\_

# whoami

Antide Petit

Twitter: @xarkes\_

GitHub: @xarkes



# # history

- \* project named iaio
- \* developed for long by Hugo Theso alone
- \* was less and less maintained
- \* cross platform GUI (Windows, Linux, OS X)



*"Cutter is not aimed at existing radare2 users. It instead focuses on those who are not yet radare2 users because of the learning curve, because they don't like CLI applications or because of the difficulty/instability of radare2."*

# # how to use cutter



- \* compile from source (cmake, qmake, meson)
- \* download binaries on the GitHub releases page

<https://github.com/radareorg/cutter/releases>



Search or jump to...

Pull requests Issues Marketplace Explore



radareorg / cutter

Unwatch 157

Unstar 3,145

Fork 244

Code

Issues 69

Pull requests 3

Projects 5

Insights

Settings

A Qt and C++ GUI for radare2 reverse engineering framework

Edit

radare2

cutter

gui

reverse-engineering

security

Manage topics

1,060 commits

9 branches

10 releases

71 contributors

GPL-3.0

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download



PabloCastellano and xarkes Update README.md (#678)

Latest commit da0db41 5 hours ago

.github

Version 1.7.1

12 days ago

docker

Docker: Fix typo in README.md, travis: fix image name (#564)

2 months ago

docs

Travis: use Qt 5.9.6 (#654)

10 days ago

&lt;&gt; Code

🔔 Issues 69

🔗 Pull requests 3

📁 Projects 5

📊 Insights

⚙️ Settings

Releases

Tags

Draft a new release

Latest release

📦 v1.7.1

🔑 d510897

# Cutter 1.7.1

Edit

 thestr4ng3r released this 11 days ago · [16 commits](#) to master since this release

## ▼ Assets 6

📦 <a href="#">Cutter-v1.7.1-win32.zip</a>	99 MB
📦 <a href="#">Cutter-v1.7.1-win64.zip</a>	111 MB
📦 <a href="#">Cutter-v1.7.1-x86_64.ApplImage</a>	125 MB
📦 <a href="#">Cutter-v1.7.1.dmg</a>	109 MB
📄 <a href="#">Source code</a> (zip)	
📄 <a href="#">Source code</a> (tar.gz)	

# Cutter 1.7.1

Patch release primarily for fixing the Strings Widget.

## Functions

Name

entry0

entry1.init

fcn.00181cd0

fcn.00183230

fcn.00183390

fcn.00183410

fcn.001838d0

fcn.00183970

fcn.001858a0

fcn.001858a8

fcn.001858b0

fcn.00186a22

fcn.00188b7c

fcn.00188bb2

fcn.00188c1a

fcn.00188c41

fcn.0019f933

fcn.0019fd0c

fcn.0019fd5f

fcn.001a1a8a

fcn.001b8c78

fcn.001bb374

fcn.001bb7ac

fcn.001be024

fcn.001be07e

fcn.001be2ea

fcn.001becda

fcn.001bed0c

fcn.001beee0

fcn.001bfb0a

**fcn.001bfcf4**

fcn.001c0be6

fcn.001c0e9e

fcn.001c1520

fcn.001c16d0

fcn.001c17b2

fcn.001c2262

fcn.001c25ae

fcn.001c3e4e

fcn.001c445a

fcn.001c4f2e

fcn.001c6062

fcn.001c7104

fcn.001c72b0

fcn.001c7780

fcn.001ca3f5

fcn.001ca86a

fcn.001dd3b6

fcn.001dd752

fcn.001df08a

fcn.001e27d6

## Graph (fcn.001bfcf4)

```

mov rax, rax
call fcn.001bfb0a
mov qword [local_60h], rax

mov rdx, qword [local_68h]
mov rcx, qword [local_70h]
mov rax, qword [local_58h]
mov rsi, rcx
mov rdi, rax
call sym.std.__detail::_Hash_code_base_unsignedlonglong_std::pair_unsignedlonglongconst_DisassemblerGraphView::Function__std::__detail::_Select1st...
mov rdx, qword [local_70h]
mov rcx, qword [local_60h]
mov rax, qword [local_58h]
mov rsi, rcx
mov rdi, rax
call fcn.001c2262
mov rax, qword [local_58h]
mov rax, qword [rax + 0x18] ; [0x18:8]=0x1858c0 entry0
lea rdx, [rax + 1]
mov rax, qword [local_58h]
mov qword [rax + 0x18], rdx
mov rdx, qword [local_70h]
lea rax, [local_48h]
mov rsi, rdx
mov rdi, rax
call sym.std.__detail::_Node_iterator_std::pair_unsignedlonglongconst_DisassemblerGraphView::Function__false_false::_Node_iterator_std::__detail:...
mov rax, qword [local_48h]
mov rbx, qword [local_18h]
xor rbx, qword fs:[0x20]
je 0x1bfe6d

```

jmp 0x1bfe68

call sym.imp.\_\_stack\_chk\_fail ; void \_\_stack\_chk\_fail(void)

```

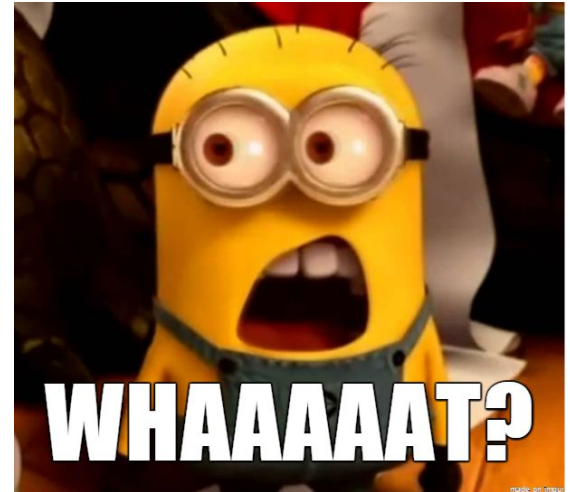
add rsp, 0x78
pop rbx
pop rbp
ret

```



# # a year before

- \* many “useless” features (from a reverse engineer PoV)
- \* the interface printed pseudo graphs in HTML  
(qt webengine)



# # a year after

- \* reworked almost 100% of the codebase
- \* added features



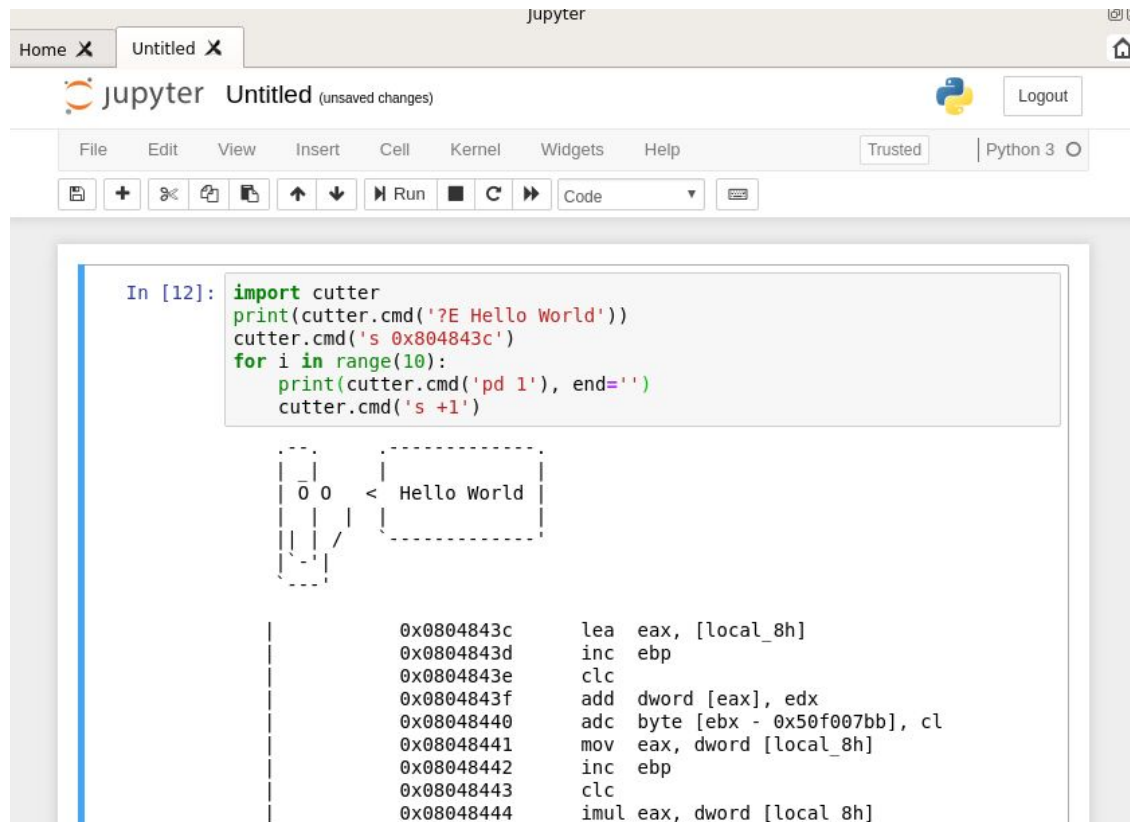
# # vtables

		<u>V</u> Table
Name	Address	
✓ VTable 1	0x0030d47d	
sym.Ui_SetToDataDialog::setupUi_QDialog__::_lambda___4_::_FUN	0x002f2a5c	
✓ VTable 2	0x0031a6a6	
sym.MemoryMapModel::tr_charconst_charconst_int	0x002e2e2e	

# # classes

beginReload()	method	0x002b0342
endReload()	method	0x002b035e
tr(charconst*,charconst*,int)	method	0x002b0d81
qt_static_metacall(QObject*,QMetaObject::Call,int,void**)	method	0x002fd57e
metaObject()const	method	0x002fd594
qt_metacast(charconst*)	method	0x002fd5dc
qt_metacall(QMetaObject::Call,int,void**)	method	0x002fd62e
~ClassesModel()	method	0x002fd928
▶ ClassesModel::RowTypeQVariant::value<ClassesModel	class	0x00000000
▶ ClassesModel::RowTypeqvariant_cast<ClassesModel	class	0x00000000
▶ ClassesSortFilterProxyModel	class	0x00000000
▼ ClassesWidget	class	0x00000000
ClassesWidget(MainWindow*,QAction*)	method	0x002b07ce
~ClassesWidget()	method	0x002b0ab6
getSource()	method	0x002b0b4c
flagsChanged()	method	0x002b0b8c
refreshClasses()	method	0x002b0bbe
on_classesTreeView_doubleClicked(QModelIndexconst&)	method	0x002b0cc0
qt_static_metacall(QObject*,QMetaObject::Call,int,void**)	method	0x002fd746
metaObject()const	method	0x002fd7b6
qt_metacast(charconst*)	method	0x002fd7fe
qt_metacall(QMetaObject::Call,int,void**)	method	0x002fd850

# # jupyter scripting



The screenshot shows a Jupyter Notebook window titled "jupyter" with a tab for "Untitled". The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a status bar showing "Trusted" and "Python 3".

The active cell is a code cell labeled "In [12]:". It contains the following Python code:

```
import cutter
print(cutter.cmd('?E Hello World'))
cutter.cmd('s 0x004843c')
for i in range(10):
    print(cutter.cmd('pd 1'), end='')
    cutter.cmd('s +1')
```

Below the code, there is a diagram illustrating the state of the program. It shows a memory location containing the value "00" and a label "Hello World" with a less-than sign "<" between them, indicating a comparison or a specific memory state.

At the bottom, the assembly code generated for the Python code is displayed:

```
0x004843c    lea    eax, [local_8h]
0x004843d    inc    ebp
0x004843e    clc
0x004843f    add    dword [eax], edx
0x0048440    adc    byte [ebx - 0x50f007bb], cl
0x0048441    mov    eax, dword [local_8h]
0x0048442    inc    ebp
0x0048443    clc
0x0048444    imul   eax, dword [local_8h]
```

# demo

# # internal structure

- \* QDockWidget

- \*\* can be moved around

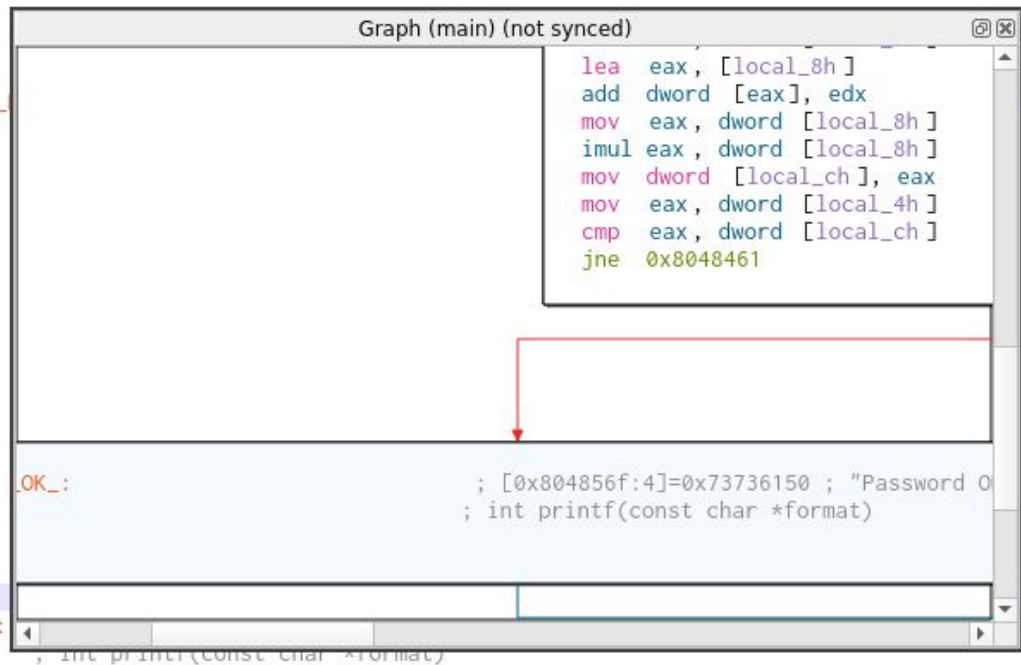
- \*\* can be docked

- \*\* can be windowed

```

/ (fcn) main 144
  main (int argc, char **argv, char **envp);
    ; var unsigned int local_ch @ ebp-0xc
    ; var signed int local_8h @ ebp-0x8
    ; var int local_4h @ ebp-0x4
    ; var int local_4h_2 @ esp+0x4
0x080483e4    push ebp
0x080483e5    mov  ebp, esp
0x080483e7    sub  esp, 0x18
0x080483ea    and  esp, 0xffffffff
0x080483ed    mov  eax, 0
0x080483f2    add  eax, 0xf
0x080483f5    add  eax, 0xf
0x080483f8    shr  eax, 4
0x080483fb    shl  eax, 4
0x080483fe    sub  esp, eax
0x08048400    mov  dword [esp], str.IOLI_Crackme_
0x08048407    call sym.imp.printf
0x0804840c    mov  dword [esp], str.Password:
0x08048413    call sym.imp.printf
0x08048418    lea  eax, [local_4h]
0x0804841b    mov  dword [local_4h_2], eax
0x0804841f    mov  dword [esp], 0x804856c
0x08048426    call sym.imp.scanf
0x0804842b    mov  dword [local_8h], 0x5a
0x08048432    mov  dword [local_ch], 0x1ec
0x08048439    mov  edx, dword [local_ch]
0x0804843c    lea  eax, [local_8h]
0x0804843f    add  dword [eax], edx
0x08048441    mov  eax, dword [local_8h]
0x08048444    imul eax, dword [local_8h]
0x08048448    mov  dword [local_ch], eax
0x0804844b    mov  eax, dword [local_4h]
0x0804844e    cmp  eax, dword [local_ch]
,=< 0x08048451    jne  0x8048461
    0x08048453    mov  dword [esp], str.Password_OK_:
    0x0804845a    call sym.imp.printf
,==< 0x0804845f    jmp  0x804846d
|'-> 0x08048461    mov  dword [esp], str.Invalid Password : [0x804857f:4]=0x61766e49 : "Invalid Password!\n" : const char *format

```





# # internal structure (dos)

- \* every widget uses radare2 JSON output
- \*\* commands are more stable than the API
- \*\* json is relatively easy to parse

{JSON}

# # contributing

- \* open issues

- \* make a PR

- \*\* .ui files

- \*\* .cpp

# # projects (work in progress)

- \* move main widgets into an external library

- \*\* help maintenance with other projects (e.g. x64dbg)

- \*\* give the possibility to other people to use those

# # projects (work in progress)

- \* C++ and Python plugins
- \* plugin manager (r2pm-go)
- \* debugger support

# # google summer of code

- \* @filipe\_casal (mandlebro)

- \* debugging platform

- \* some stuff require a lot of rework in r2  
codebase



Google  
Summer of Code

## # documentation?

- \* contribution guidelines are available in the repository
- \* no complete documentation (the software is still young)
- \* feel free to ask any question on **IRC** or **Telegram** or **GitHub**

<https://github.com/radareorg/cutter/blob/master/README.md>

## # efficiency

- \* more and more people use it in real world challenges

(still not enough)

- \* we can still add widgets and features that are available in radare2

## # conclusion

- \* a lot of changes and refinements in a year
- \* need to focus on useful features



# # special thanks

\* thestr4ng3r

\* maijin

\* pelijah

\* mandlebro

\* megabeets

<https://github.com/radareorg/cutter/>



@r2gui

@xarkes\_