

# **RADARE2**

## First r2babies steps

---

Maxime Morin (@Maijin212)

June 22, 2015

Nuit du Hack 2015

- 22 y/o french expat @ Luxembourg
- Food?, Food., Food! <3
- I hate Bullshit
- Malware.lu CERT team leader (2days/week) and incident response @ European Commission CSIRC (3days/week)
- User of radare2 (impossibru!)
- I'm creating tests + documentation

- r1 2006, r2 2009
- Multi-(OSes|Archs|Bindings|FileFormats|...)
- 10 tools based on the framework
- Around 111 contributors
- GSOC + RSOC
- CLI/VisualMode/GUI/WebGUI
- around 350K LOC

INSTALLATION !

- Always use git version!
- Use the provided VM on SSH ([radare:radare](#) / [root:root](#))
- git clone <http://github.com/radare/radare2> && cd radare2 && [./sys/install.sh](#)
- Use the Windows installer <http://bin.rada.re/radare2.exe>

## UTILITIES

---

- rax2
- rabin2
- rasm2
- radiff2
- rafind2
- rahash2
- radare2
- rarun2
- ragg2/ragg2-cc

- `rax2`
- `rabin2`
- `rasm2`
- `radiff2`
- `rafind2`
- `rahash2`
- `radare2`
- `rarun2`
- `ragg2/ragg2-cc`



## rax2 — Base converter

```
$ rax2 10
```

0xa

```
$ rax2 33 0x41 0101b
```

0x21 65 0x5

```
$ rax2 -s 4142434445
```

ABCDE

```
$ rax2 0x5*101b+5
```

30

- rax2
- rabin2
- rasm2
- radiff2
- rafind2
- rahash2
- radare2
- rarun2
- ragg2/ragg2-cc

## rabin2 — Binary program info extractor

---

```
$ rabin2 -e
```

Entrypoints

```
$ rabin2 -i
```

Shows imports

```
$ rabin2 -zz
```

Shows strings

```
$ rabin2 -g
```

Show all possible information

- rax2
- rabin2
- rasm2
- radiff2
- rafind2
- rahash2
- radare2
- rarun2
- ragg2/ragg2-cc

rasm2 — assembler and disassembler tool

---

```
$ rasm2 -a x86 -b 32 'mov eax, 33'
```

Assemble

```
$ rasm2 -d 9090
```

Disassemble

```
$ rasm2 -L
```

List supported asm plugins

```
$ rasm2 -a x86 -b 32 'mov eax, 33' -C
```

Output in C format

- rax2
- rabin2
- rasm2
- **radiff2**
- rafind2
- rahash2
- radare2
- rarun2
- ragg2/ragg2-cc

### radiff2 — unified binary diffing utility

---

```
$ radiff2 original patched
```

Code diffing

```
$ radiff2 -C original patched
```

Code diffing using graphdiff algorithm

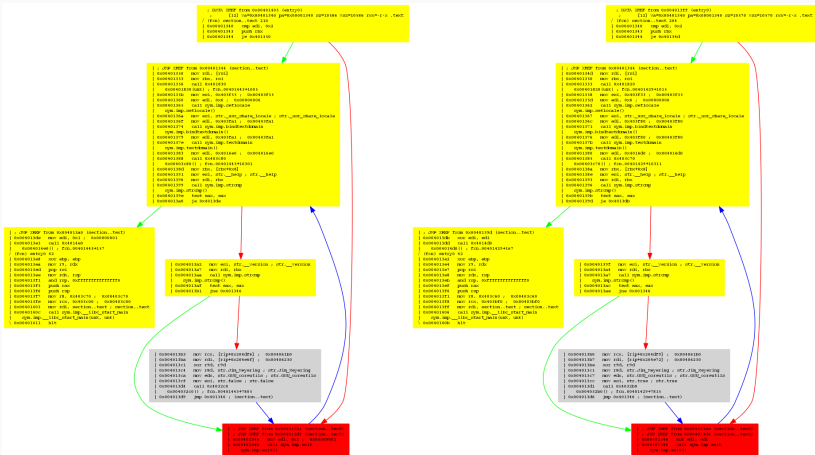
```
$ radiff2 -g main -a x86 -b32 original patched
```

Graph diff output of given symbol, or between two functions, at given offsets: one for each binary.

# UTILITIES: RADIFF2 — GRAPH EXAMPLE

/bin/true

/bin/false





- rax2
- rabin2
- rasm2
- radiff2
- **rafind2**
- rahash2
- radare2
- rarun2
- ragg2/ragg2-cc

`rafind2` — Advanced commandline hexadecimal editor

---

```
$ rafind2 -X -s passwd dump.bin
```

Search for the string `passwd`

- rax2
- rabin2
- rasm2
- radiff2
- rafind2
- **rahash2**
- radare2
- rarun2
- ragg2/ragg2-cc

rahash2 — block based hashing utility

---

```
$ rahash2 -a all binary.exe
```

Display hashes of the whole file with all algos

```
$ rahash2 -B -b 512 -a md5
```

Compute md5 per block of 512

```
$ rahash2 -B -b 512 -a entropy
```

Compute md5 per block of 512

```
$ echo -n "admin" | rahash2 -a md5 -s "
```

Compute md5 of the string admin

- rax2
- rabin2
- rasm2
- radiff2
- rafind2
- rahash2
- radare2
- rarun2
- ragg2/ragg2-cc

## RADARE2 — COMMAND LINE

---

Keep in mind that:

1. Every character has a meaning i.e (w = write, p = print)
2. Every command is a succession of character i.e pdf = p  $\leftrightarrow$  print d  
 $\leftrightarrow$  disassemble f  $\leftrightarrow$  function
3. Every command is documented with **cmd?**, i.e pdf?,, ???, ???, ?\$?,  
?@?

1. Open a file with radare2 `radare2 file.exe`
2. Get Usage on the command `#! Usage: #algo <size> @ addr`
3. List of all existing algorithms `##`
4. SHA1 `!sha1`
5. Hashing from the begin `!sha1 @ 0`
6. with a hash block size corresponding to the size of the file `!sha1 $s @ 0x0`

This command is same as `rahash2 -a sha1 file.exe`



1. Get Usage on the command `i`?
2. Same as `rabin2`
3. `izj` for displaying in json
4. internal commands: `~`, `ls`, `{}`, `..`

Quick Demo

1. r2 -A or r2 then aaa : Analysis
2. s : Seek
3. pdf : Print disassemble function
4. af? : Analyse function
5. ax? : Analyse XREF
6. /? : Search
7. ps? : Print strings
8. C? : Comments
9. w? : Write

## RADARE2 — VISUAL MODE

---

1. V? : Visual help
2. p/P : rotate print modes
3. move using arrows/hjkl
4. o : seek to
5. e : r2configurator
6. v : Function list
7. \_ : HUD
8. V : ASCII Graph

## RADARE2 — WEBUI

---

```
r2 -A -c=H filename
```



" -- *When you sold that exploit, what they really bought, was your silence.* "

## Current Project

CurrentProject:

CurrentFile: /bin/ls

OtherProjects: \_\_\_\_\_

Layout:

## Files

# RADARE2 — DEBUGGER

---



1. radare2 -d
2. Quickly switch to Visual debugger mode: Vpp
3. OllyDBG/IDApro shortcuts friendly

- rax2
- rabin2
- rasm2
- radiff2
- rafind2
- rahash2
- radare2
- **rarun2**
- ragg2/ragg2-cc

## Rarun2 — run programs in exotic environments

---

1. Environment setup tools for radare2
2. most useful with debugger
3. aslr, stdout, arguments, r2preload ...

- rax2
- rabin2
- rasm2
- radiff2
- rafind2
- rahash2
- radare2
- rarun2
- [ragg2/ragg2-cc](#)

Ragg2/Ragg2-cc — frontend for compiling shellcodes

---

Demo time !

- **Website:** <http://rada.re/>
- **Blog:** <http://radare.today>
- **Book:** <http://maijin.gitbooks.io/radare2book/content/>

NOW YOUR TURN !