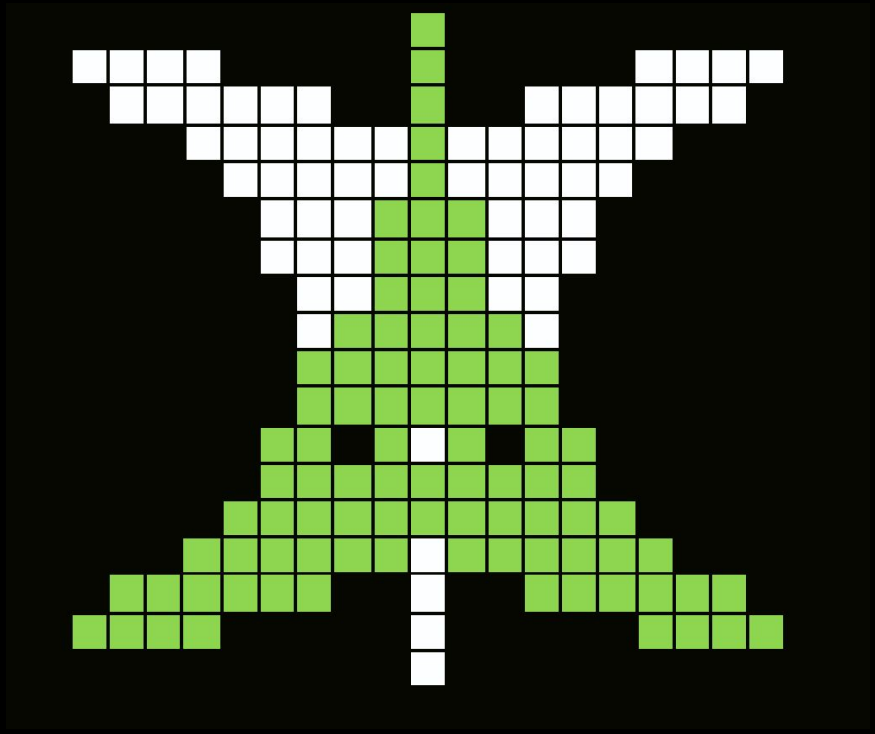
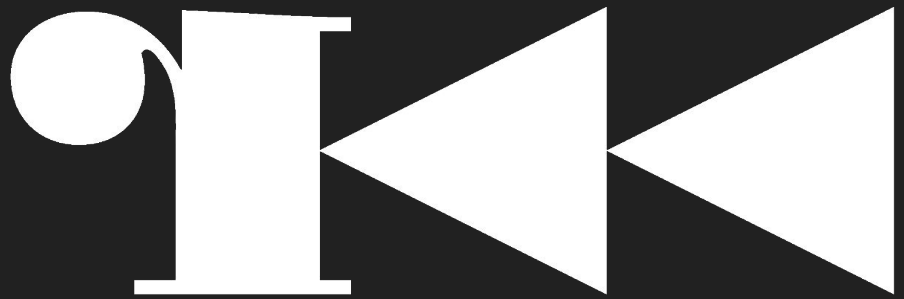


Digging into radare2 for fun and profit

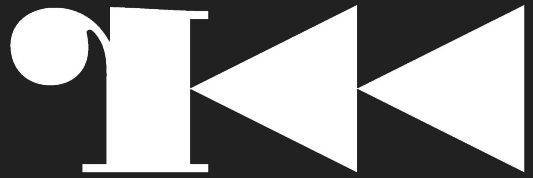
AvTokyo2017 // by pancake

INTRO & Translation: unixfreaxjp

Radare2とは?



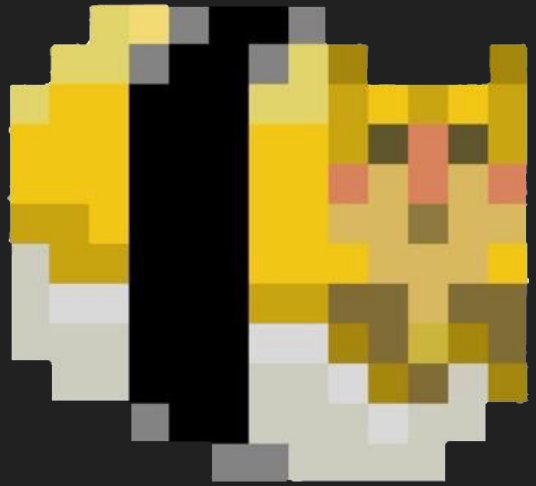
Radareとは?



- 12年間開発されているオープンソースプロジェクト（無料）
- リバースエンジニアリングのフレームワークとツールセット
- 最初段階では私が開発したソフトウェア
- コミュニティとコントリビューターのコーダーが結構増えて
- 個人でメンテナンスしたプロジェクトからプロジェクトを分けて、それぞれの開発プロジェクト責任者がたちあがりました

- 現状開発のペースは6週間1回でマイナーバーションアップをリリース
- メージャーバーションはr2conの後にリリース
- r2con 自体はスペイン、バルセロナから始まり、2017年の自体で
  大体230人の参加皆さんが着ました
- r2conのコンファレンスにあるプレセンテーションビデオを必ず youtubeにアップロードします

Who Am I?



- フリーソフトの開発者
  - Barcelona, Cataloniaの生まれ
  - いくつかフリーソフトウェアを開発しています
  - defcon CTF 3年間連続の参加しました
  - ペイントが大好き
  - 父親として頑張ります
- Links
  - 1）githubそして　2）bitbucket でradare コードをアップロード (user: trufae)
  - ツイッター: t https://twitter.com/trufae
- 仕事は NowSecure社 (Mobile Security Analyst, フォレンジックとR+D)
  - codecs 開発（assembly）mips, arm と x86向け
  - Embeddedデバイスのファームウェア開発 (リアルタイム・デバッギング調査機能を含む) フォレンジックについてほとんど Windows案件が多い
  - プログラミングとCTF関係の講師

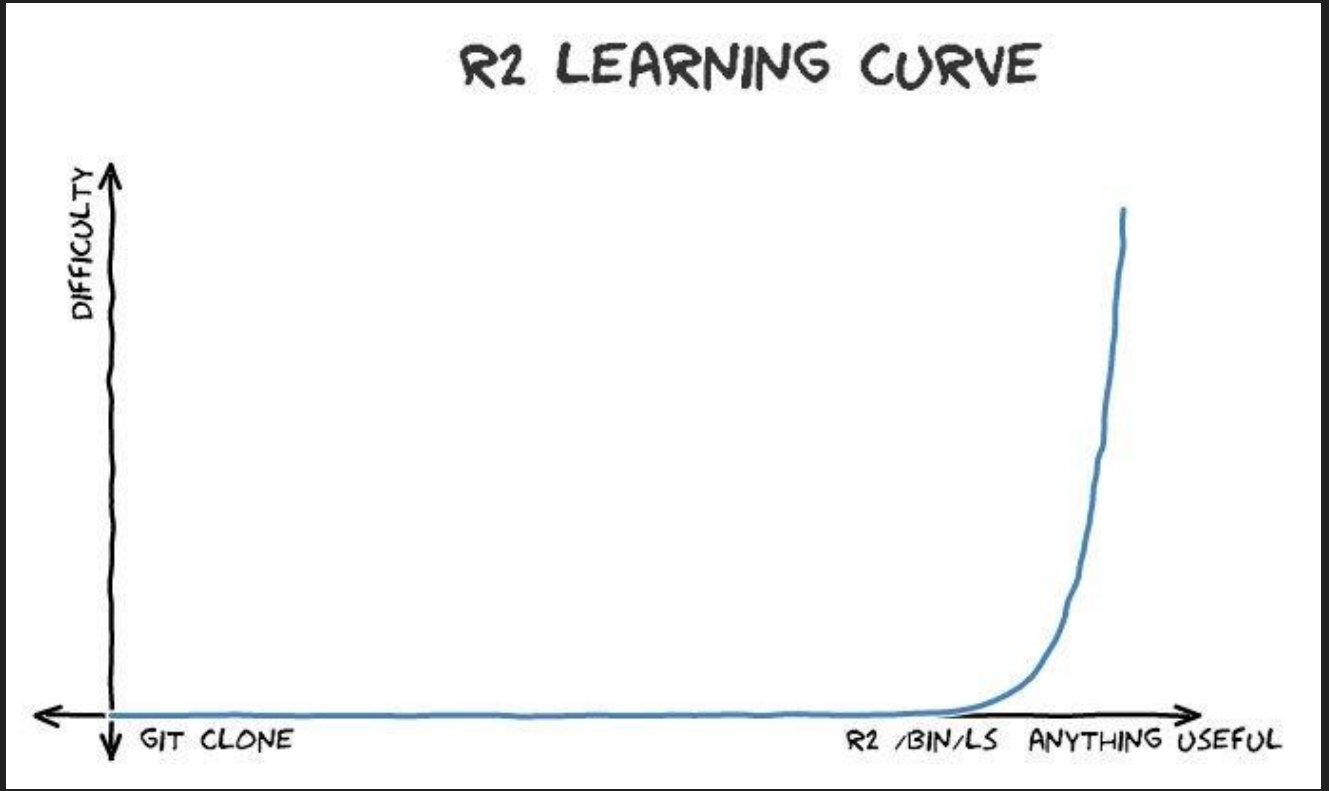
radare2の便利機能について

- プログラムのデバッギング調査ツール
- エンコードされたcharsetの検知と検索 (日本語,中国語, 韓国語, cyrilic, など)
- ストリングスのリファレンス調査、特徴な技術方法が持っている
- メモリダンプのカービングツール（"magic"を調べるフォレンジック調査）
- 色んなファイルシステムのマウントと認識機能、partition tableの読み込み
- その他のデバッガーとの連携機能(gdb, r2, frida, windbg,など）
- エムレーター機能を使い、解読の調査ツールとして非常に便利
- 外部デコンパイラまたはグラフ作成ツールを使用する
- バイナリのdiff 機能
- ゲームやシムレーションツーツ、例えば2048 や r2wars!

ほぼどんな環境にも使えます！

- 様々なOSの対応:
  - Windows, Linux, Mac, QNX, Solaris, NetBSD, FreeBSD, BeOS, Android, iOS, ….
- 色んなcpuアーキテクチャーにも対応しています、例えば:
  - x86, arm, mips, sparc, ppc, z80, 6502, 8051, avr, wasm, snes, java, dalvik, hppa, ...
- ネイティブデバッガの対応
- web-assembly と asm.jsのコンパイルする事もできます。
- ローカルとリモート(socket)の使い方
- Foll

(demo rasm2 -L rabin2 -L r2 -L)


R2 LEARNING CURVE
DIFFICULTY
GIT CLONE
R2 /BIN/LS
ANYTHING USEFUL

学習曲線

- 最初段階はステップバイステップでの学ぶ事ですが、長期的には楽しい
- １０個の実行コマンドが分かれば使えるようになる
- 直交性がコマンドを可能ですし、拡張し改善することができます
- 初めての方は2週間ぐらいの学ぶ事が可能

- 実践から学ぶ
- 関心と献身が必要
- 他の似たようなツールと比べたらやり方がいくつか違う
- オープンソースなので、rwx をどうぞ！

It’s Documented®

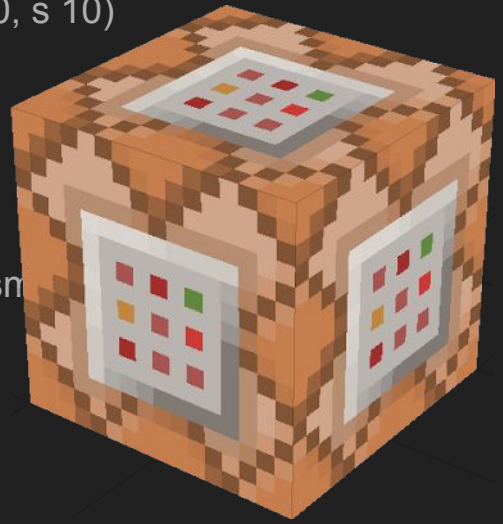
- ソースコードはCでドキュメンテーション課されている
- オンラインでマニュアルブックがあり（古いバーション）
- ブログの記事(フォロー: @radareorg)
- 使い方としてはYouTube とVimeoビデオが多い
  - r2con 2016, 2017のビデオ
- コマンドラインのヘルプマニュアル（"?"コマンド）
- UNIX系のmanページ
- IRC と Telegram チャンネルのQ & Aコニュニティ(800人)

https://twitter.com/radareorg

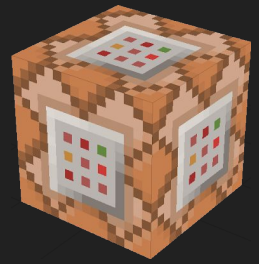
https://t.me/radare

基本の機能とコマンド

- 検索
  - Relative / Absoluteのアドレスと部分的なアドレス (s+0x10, s..10, s 10)
  - 履歴/History (!, s!)
  - ブロックサイズ / blocksize (b, @!)
- ダンプ/プリント
  - Hexdump (px, pxr, pxa, prc, pxA, …)
  - アセンブリの種類に夜、ディスアセンブリ (pd, pD) @a:mips, e asm
  - デコード構造 (pf)
  - Checksums, entropy, statistics (p=, ph)
- バイナリの書き込み
  - アセンブリの書き込み (wa)
  - ストリヌスや文字列 (w)
  - Hexpairs (wx)
  - 全体的なファイルの中身 (wf)



コマンド修飾子



プレフィックス

- [1-9]
  - Repeat command n times
- . (backtick)
  - Interpret the output of the command as r2
- “
  - Ignore special characters
- `
  - Insert the output of a command
- !
  - Shell escape
- $
  - Alias command
- \
  - Alias for =!

サフィックス

- @
  - temporal seek
- |
  - system pipe
- ~
  - internal grep
- >
  - file redirect
- #
  - Comment
- ?
  - Show help
- j
  - Output in JSON

Demo

ファイルシステムのマウントと検索機能

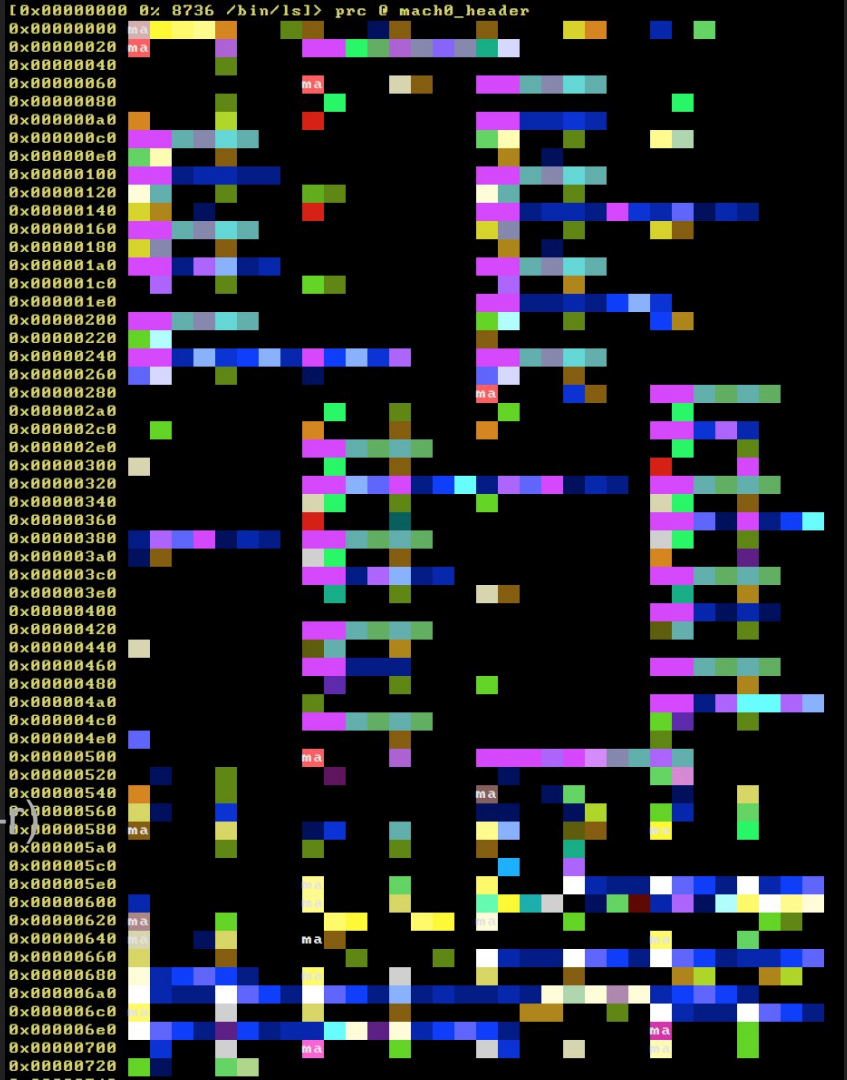
- 最初段階でradare2はフォレンジックツーツ飲みでした。
- ファイルの中身のoffset 検索,その逆もOK
- バイナリのパターンの検索、そして結果をダンプする機能
- HFS, FAT, NTFS, EXT2に対応
- まだ対応していません：Squash, jffs2

- コマンド ‘m’ を使うとパーティションテーブルとファイルシステムの認識・読み込みができます。
  - GRUB開式のプラグイン. (GPL warning)
  - 他のコマンドもあり、例えば：io and r2 filesystems (wip)

Demo

バイナリヘッタのパーシング

- 色んな種類バイナリヘッタの対応
- rabin2 -I
- 破損されたバイナリの認識・調査機能
- ディスクやメモリからの読み込み機能
- IO レーヤーのAPIやりとりの調査
- Virtual Address えむレーションスペース
- r2 -nn
- rabin2 -H
- memory ヘッタのパーシング (oba, .!rabin2 -H)
- リソースなどのダンプ機能



Demo

調査とディスアセンブリについて

初心者向けのジェネリック調査機能

- -A, aa, aaa, aaaa, aaaaa, aaaaaaaah!

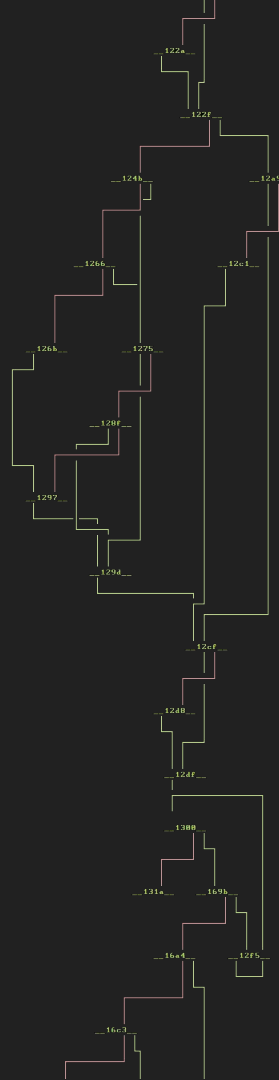
実はもっと面白いコマンドが沢山あります:

- e??anal.

それとも、もっと詳細なデータを狙う為に別途コマンド:

- aac, aar, aae, aav, aab, ...

ESILエミュレーターのコマンドもあります



調査とディスアセンブリについて

Listing functions

- afl

Listing basic blocks

- afb

Graphing them

- agf

Rename function

- afn

Analyzing a single opcode

- ao

Analyze single function

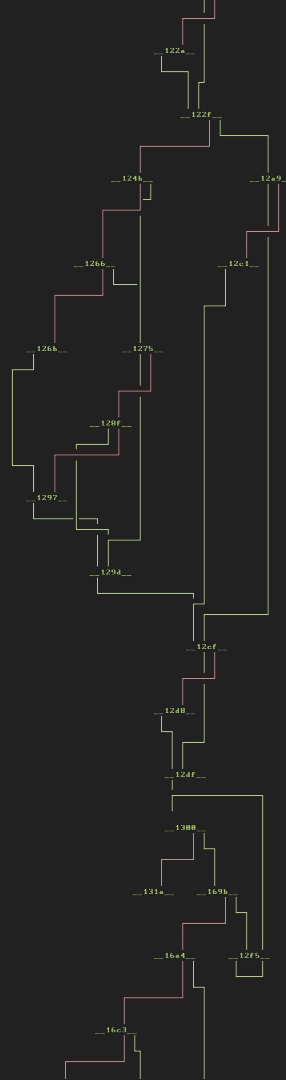
- af (e anal.hasnext)

Alternative analysis loop

- a2f

Autoname function

- afna



調査についてのオップション

もっと調べるコードがある場合：

- anal.hasnext

調査した上でもっと文字列を探すコマンド：

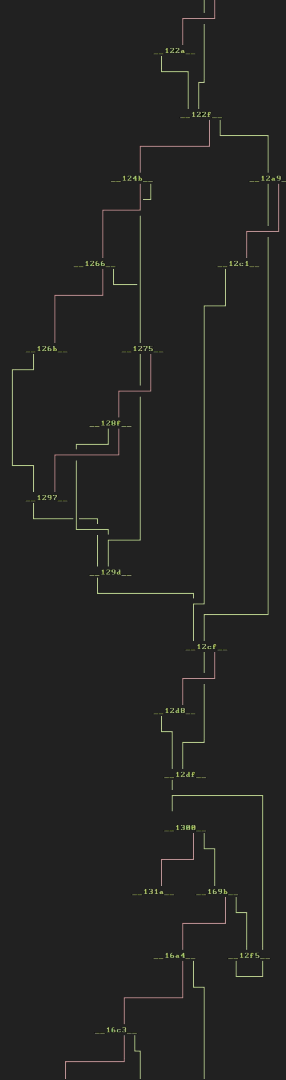
- anal.strings

実行コマンドなしの場所に下記のコマンドが便利

- anal.noncode

xrefやJump-Tables調査

- anal.jmptbl



Demo

マルチプルフォーアトのデータをプリント

‘b’ コマンドを使い、ブロックサイズ(blocksize)の設定

ズームニューワー機能, entropy調査, データブロックのカラーリング, 実行命令コードなど

- "p" コマンドとは？

Format string-like strings by parsing the memory at given address
特定なメモリアドレスにある文字列データのパーシングについて

- pf xxi foo bar cow @ addr

Demo

デバッギングとエミュレーター機能

スタティックバイナリの調査モードで エミュレーターエンジン(ESIL VM)を使い、radare2はデバッギング行動のシムレーションが出来る。実行命令コードはESILにパーシング、そしてシムレーションのレジスターのバリューの計算が出来る、ようはバイナリを実行せず、解読が出来る機能機能。なお、デバッギングをする時に -d のパラメーターを使えばダイナミック調査モードを動かす事が出来る

- r2 -d
- ジャンプ命令(addrのアドレス迄は) ⇨ dcu addr
- ステッピングは ⇨ ds や dso

デバッギングの上でエミュレータを動かす方法

ネイティブとリモートデバッギングサーバに接続が可能

- dbg:// winedbg:// gdb:// windbg:// qnx:// ..

その他 low level デバッギングについて:

- バックステップ (Thanks Ren Kimura!)
- メモリスナップショット
- Software/Hardware ブレークポイント
- Assisted debugging (emulation + debug)
- Tracing
- Filedescriptor manipulation

Rarun2 プロフィール

実行環境のコンフィグについて下記の方法で設定ができます:

- r2 -r or dbg.profileの情報でテキストファイルフォーマットの洗濯が可能
- カンマのof directives via dor or -R commandline flags

radare2環境の上でgid, uid, chroot, chdir, environment, arguments, filedescriptors の変更する事ができます。

- コマンドのパラメーターについてストリングス、ファイル、もしくはstdoutにも対応する事が可能

$ man rarun2

デバッガーでのデータをプリントする時に

スタックの中身を見せる事ができます

- dbt - backtrace
- pxr@r:SP

ローラル変数のバリューを見せる事が可能（varやレジスター）

- afvd

カラーバーの変更も可能

- p=

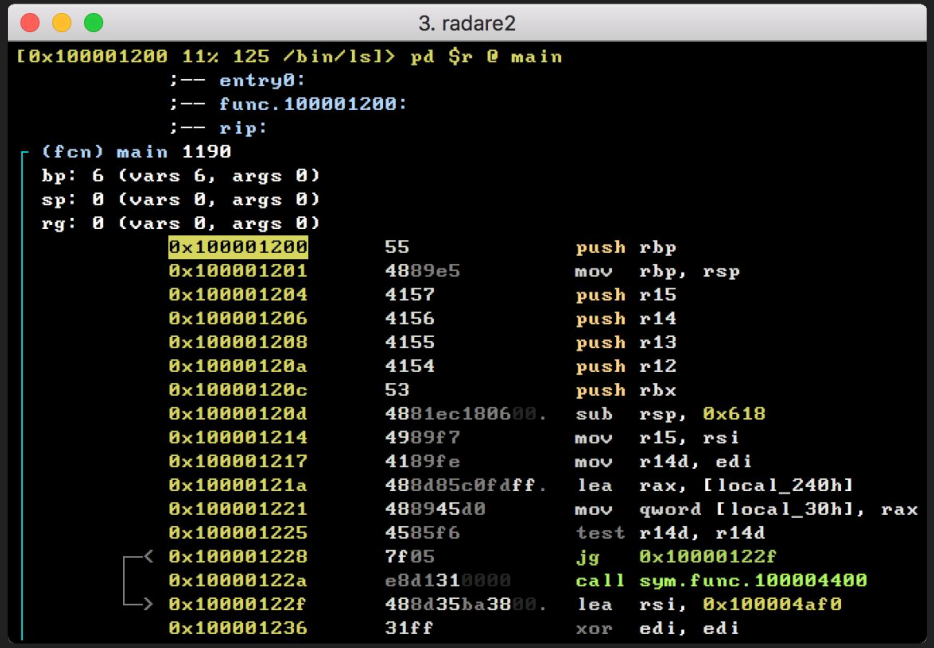
Demo

インターフェース

- メインはコマンドプルンプトCLI
- 最近はビジュアルモードが人気
- ウェブサーバのGUI (r2 -c=H)

ビジュアルモードの時に、コマンドインのターフェースはキースツロークとなります

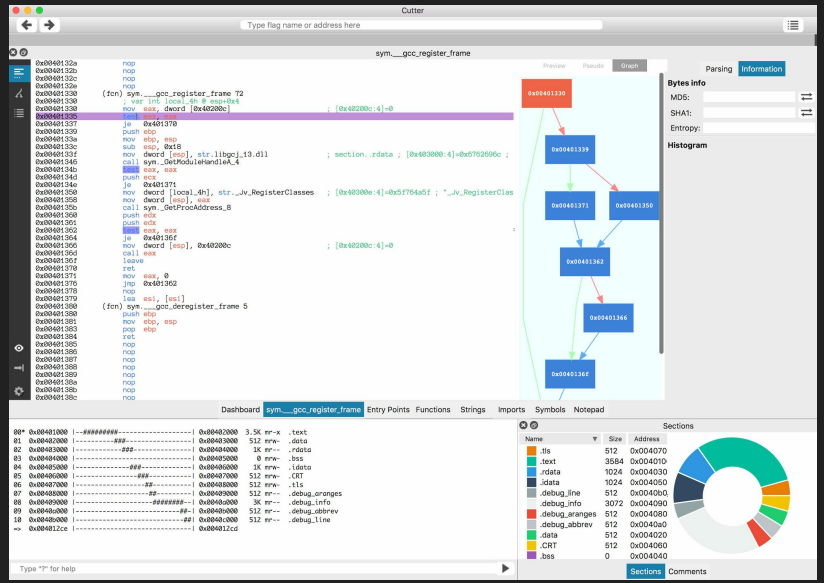
- ビューの変更 : pP"|=...
- テッピング : 's', bpトグル, など
- コマンド履歴/history
- ビジュアルアセンブラーr
- インタラクティブ・グラフ



グラフUI / GUIについて

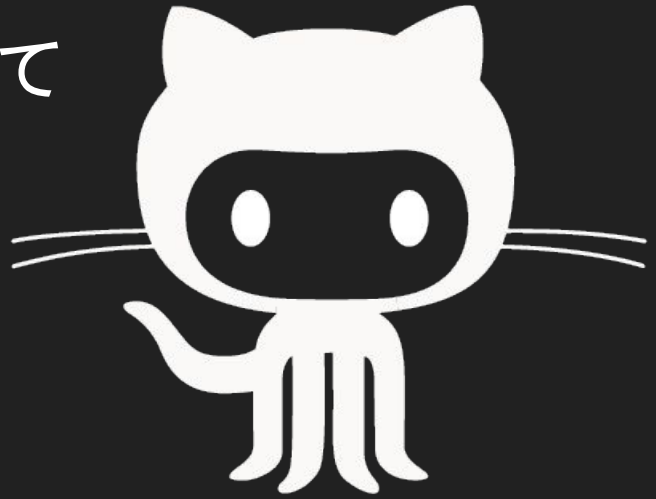
必要な時に r2pmでインストールが可能

- Gradare2 (simple gtk2/3 + vte ui)
- Ragui (unreleased)
- Bokken (unmaintained)
- Blessr2 (nodejs-blessed based UI)
- WebUIs (material, enyo, tiled, …)
- Radare2gui (.net for windows)
- Cutter (previously known as laito)
- …



Demo

カスタマイズ(read:"r2 hack")について



- ライブラリー
  - symlinksのライブラリー連携の読み込み
  - cd libr/* ; vim ; make; run
- プラグイン
  - ./configure-plugins
  - r2pm
- バインド
  - C バインドのAPI の使う事が可能(Python, Perl, Ruby, Scheme, Haskell)
  - Thanks to Valabind
- スクリプト
  - スクリプトは RLangでPython, C, やValaでコーディングが可能
  - バンドの自動ローディング設定も可能

r2pipe

r2の自動課されている方法

- シングル api 機能: 1個コマンドを実行し、結果を見る事
- その他のスクリプトを使いマルチプル実行コマンドが可能

マルチプルpipeコミュニケーションチャンネル

- Pipe
- Socket
- HTTP
- Native
- Spawn

Demo

Third Party アドオンについて

色んなアドオン機能の開発プロジェクトがあります、例えば:

- scripts, plugins, patchesの対応…
- r2pmまたはパッケージマネージャー経由のインストール
  - Install everything by default in your home (unless -g is used)
- Cでのデコンパイル機能, SMT Solvers, tools, …
- r2pm -sのコマンドで一覧を見えます。

(DEMO)

r2frida

- Fridaはradare2のフックエンジンであり, javascript インジェックションのコマンドを使い、ローカルやリモートのプロセスにフックした後色んなコマンドやりとりができます。
  - It comes with a REPL, a tracer, process list, etc..
- Radare2 を使えばFridaのフロントエンドとして使う事ができます。
  - Uses the power of the IO plugins
  - Access functionality via io->system
  - Using the \ or =! Command
- もしくはrarunにあるr2preload を使うと自分自身のプロセスインジェクションも可能です。コマンド： self://

WineDBG

- Wine is not a Windows Emulator
- Comes with winedbg, a very rustic commandline low level debugger
- The io.winedbg plugin allows to interface with it
- Similar to the bochs:// one
- Allows to debug window programs with r2 on Linux and Mac platforms.
- In early stage of development
  - Lot of potential here

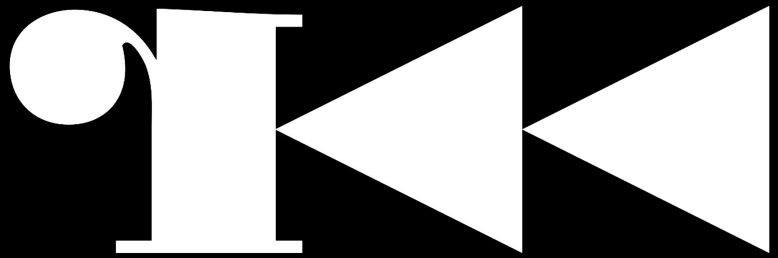
その他 3rd Partyデバッガーバックエンド

- GDB / LLDB
  - Debug kernels via the gdbserver embedded in qemu, vmware, vbox, ..
  - Apple’s debugserver, GNU’s gdbserver
  - AVR emulator and jtag
- WINDBG
  - Connect to a windbg server
- WINEDBG
  - Debug Windows programs on wine (Linux, Mac, ..)
- QNX
  - The debugserver used in automobile
- Bochs
  - X86 CPU debugger

Demo

しつもん が ありますか？

(Questions?)



Thanks For Watching!