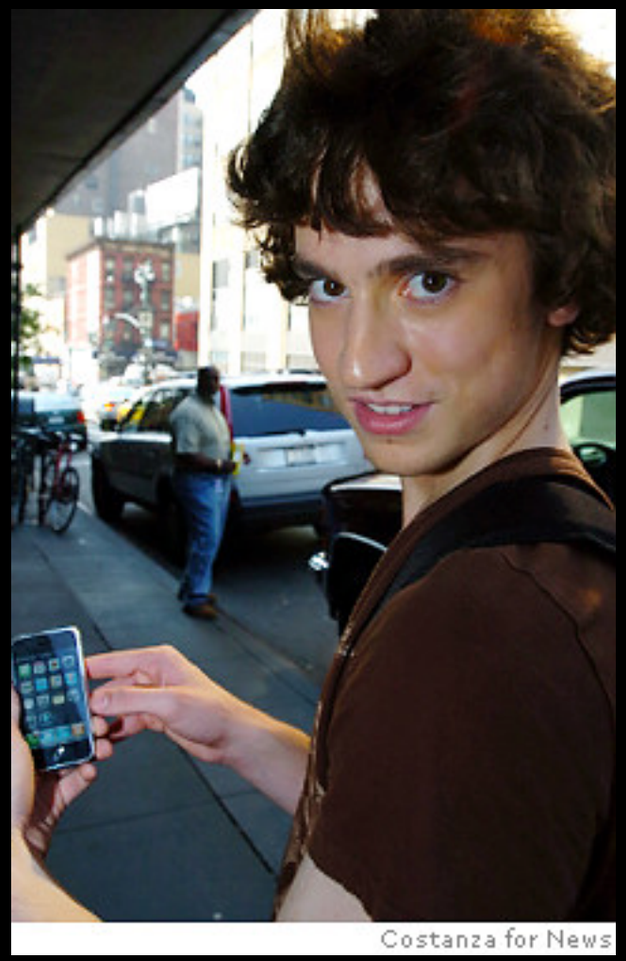
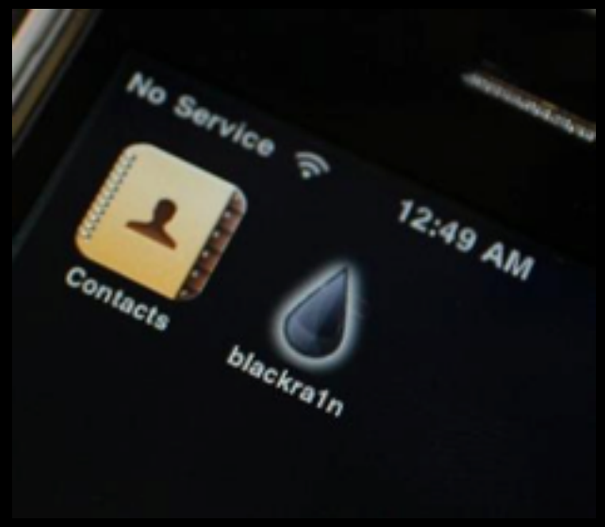
Why are our tools terrible?

(and a bit about other things)

George Hotz (geohot)

“geohot”


Costanza for News


No Service
12:49 AM
Contacts
blackra1n


GEORGE HOTZ

pwnium

- `ab.__defineGetter__("byteLength", function() { return 0xFFFFFFFC; });`
- spawn crosh with fake messages
- send commands to crosh by spoofing the window id
- try_touch_experiment %s command injection
- race condition in mount to get root
- magic symlinks to persist

towelroot

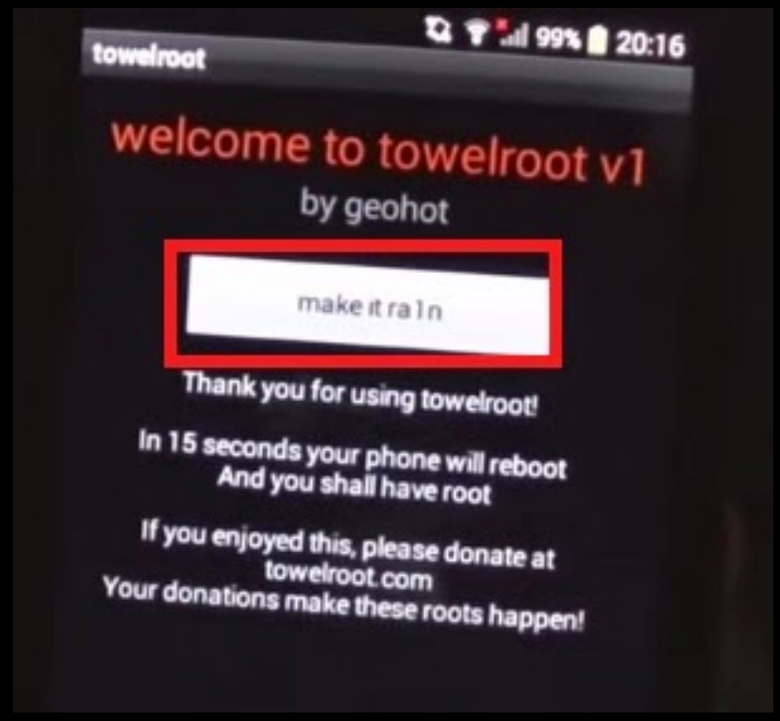
- CVE-2014-3153 found by comex, 6/7/14
- futex() syscall, break out of Linux sandbox
  - (yes, this would’ve worked for pwnium)
- towelroot, universal android root
- Used by 50 million people

“tomcr00se”

twitch

My 2014

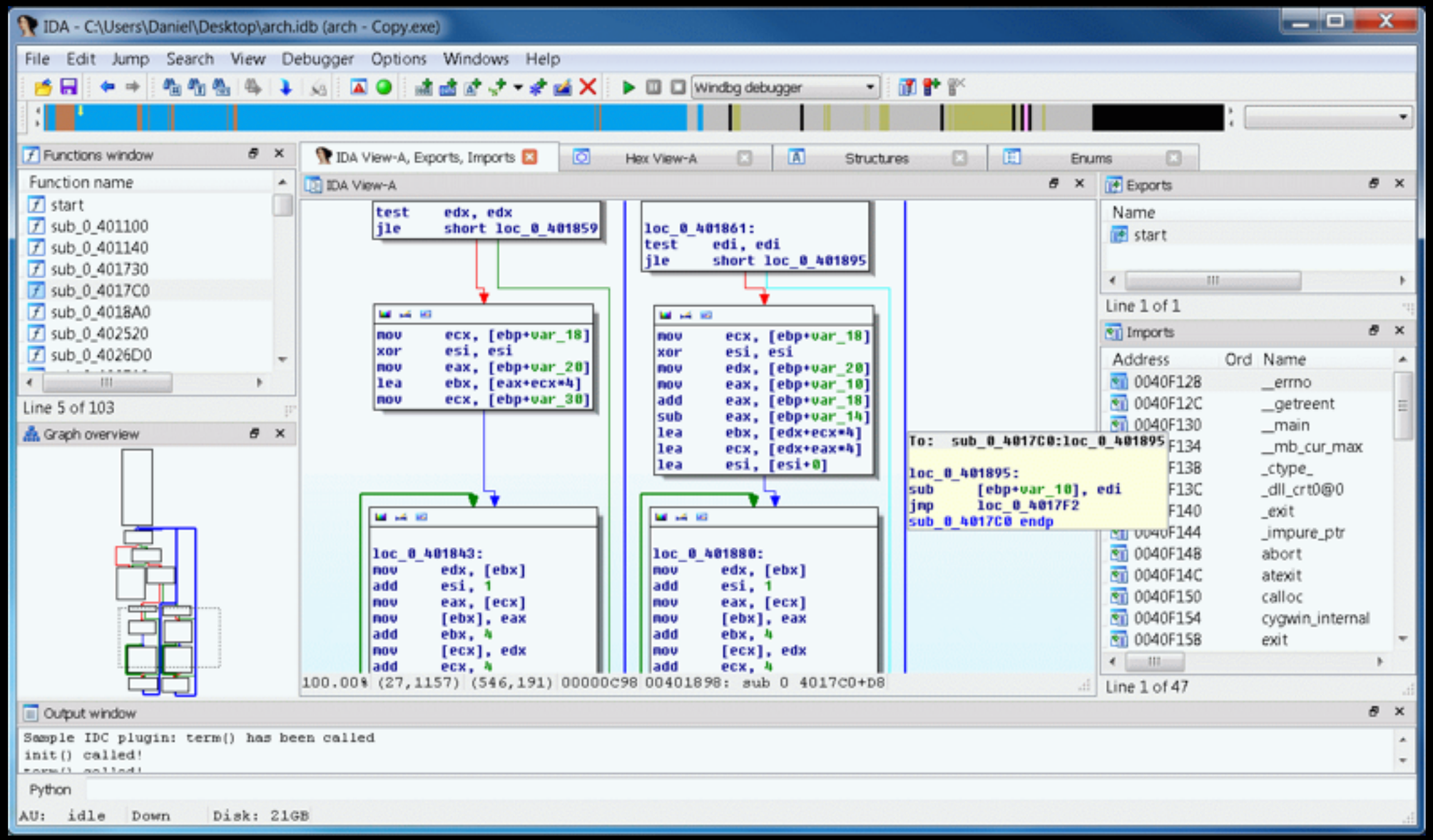
boston
key
party
2014



GDB is Terrible

```
vagrant@eagle:~/demoqira$ gdb demo_1
GNU gdb (Ubuntu 7.7-0ubuntu3.1) 7.7
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from demo_1...(no debugging symbols found)...done.
(gdb) r
Starting program: /home/vagrant/demoqira/demo_1
number: ^C
Program received signal SIGINT, Interrupt.
0xf7fdb430 in __kernel_vsyscall ()
(gdb) bt
#0  0xf7fdb430 in __kernel_vsyscall ()
#1  0xf7eec483 in read () from /lib/i386-linux-gnu/libc.so.6
#2  0xf7e81503 in _IO_file_underflow () from /lib/i386-linux-gnu/libc.so.6
#3  0xf7e823e9 in _IO_default_uflow () from /lib/i386-linux-gnu/libc.so.6
#4  0xf7e8221b in __uflow () from /lib/i386-linux-gnu/libc.so.6
#5  0xf7e61d3d in _IO_vfscanf () from /lib/i386-linux-gnu/libc.so.6
#6  0xf7e673d6 in __isoc99_scanf () from /lib/i386-linux-gnu/libc.so.6
#7  0x0804853b in main ()
(gdb)
```

Think of the first time you used IDA…

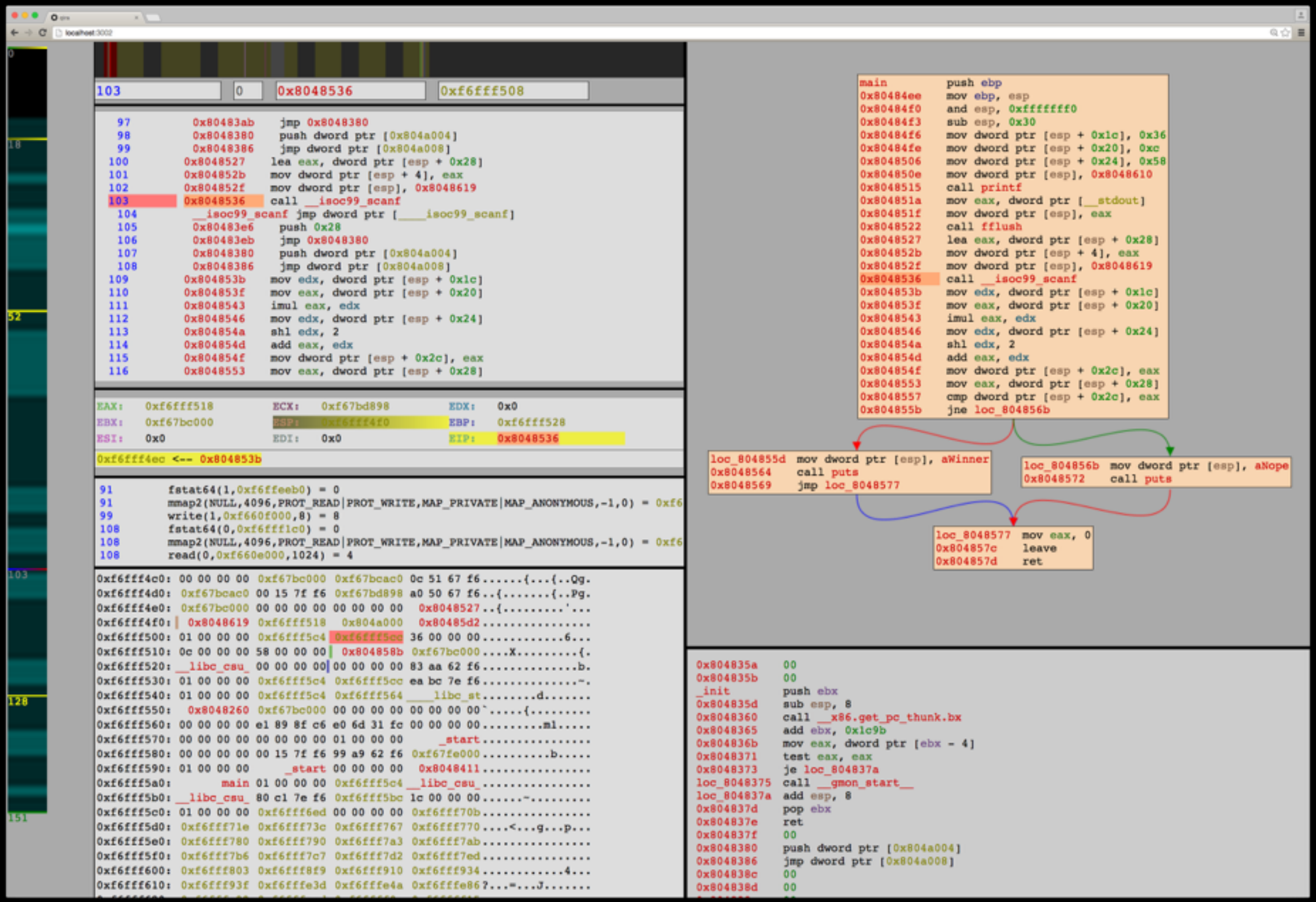


…now think of what is possible.

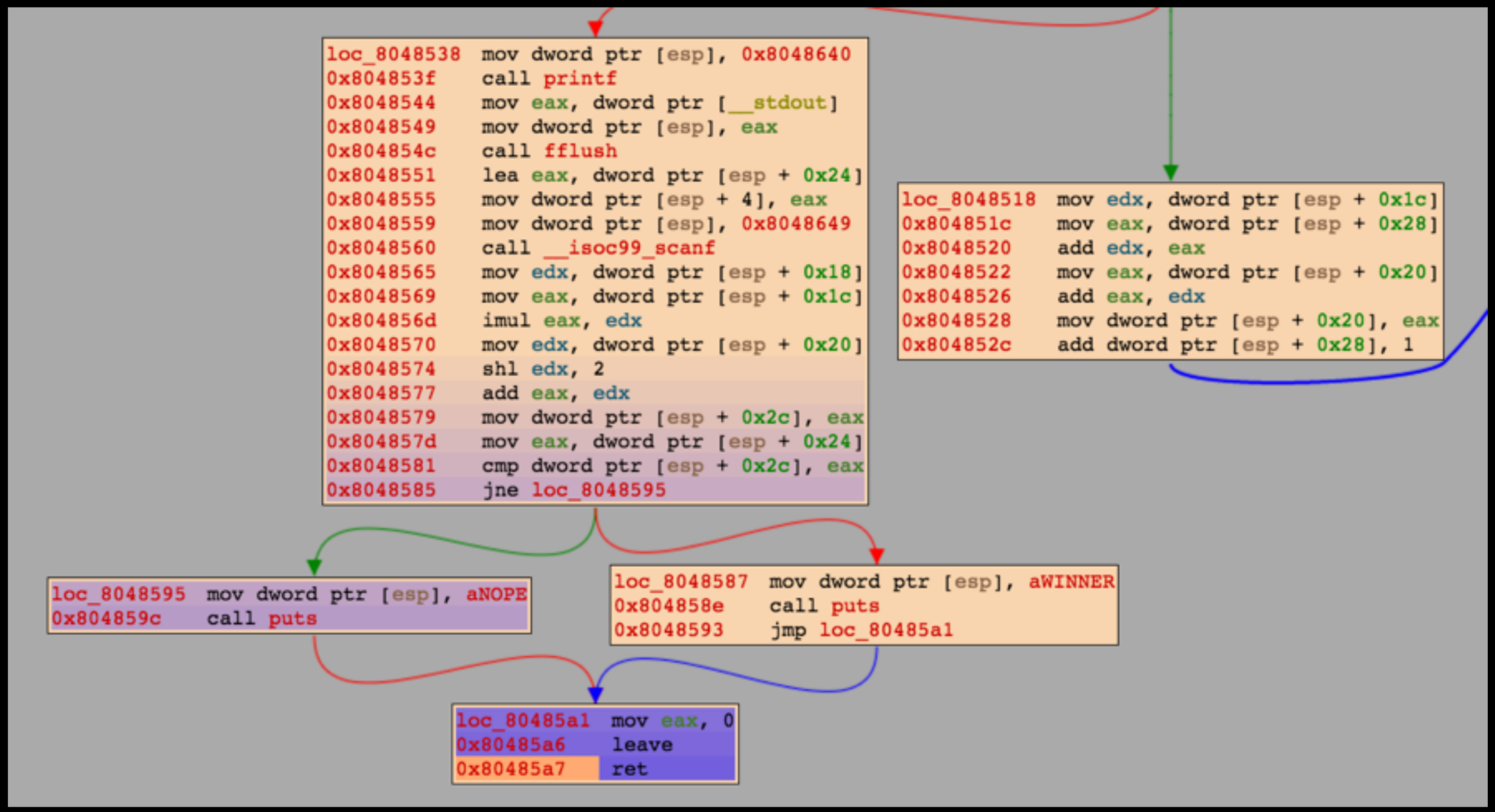
Version Control

```
EAX: 0x0          ECX: 0x3          EDX: 0x8049f38
EBX: 0x8049ff4    ESP: 0xf6ffeeac   EBP: 0xf6ffef48
ESI: 0xf67fe918   EDI: 0x0          EIP: 0x80483e4
0xf6ffeeac -- 0x8048531
0xf6ffee80:  0xf6ffeed8 00 00 00 01 03 00 00 00  0xf67fe53c ............<...
0xf6ffee90:  0xf6ffee50 03 00 00 00 03 00 00 00  02 00 00 00 P...............
0xf6ffeea0:   0x804825a  0x8049ff4  0xf6ffef48 | 0x8048531 Z.......H...1...
0xf6ffeeb0:  00 00 00 00 03 00 00 00  0xf67fdff4 65 c4 7e f6 ............e.~.
0xf6ffeec0:  0xf67fe4e4 0a 00 00 00  0xf66135e8  0xf67da928 .........5a.(.}.
0xf6ffeed0:  0xf67fe918  0xf67da928  0xf67fe53c 24 c3 7e f6 ....(.}.<...$.~.
0xf6ffeee0:  0xf67fe4e4  0xf67fea74  0xf67b4a20 75 f4 67 f6 ....t... J{.u.g.
0xf6ffeef0:  0xf67b4a20  0xf67fdff4  0xf67fe918  0xf67fe020  J{.........  ...
0xf6ffef00:  0xf67f73d2 ec 04 00 00 0c 00 00 00  20 08 00 00 .s..........  ...
0xf6ffef10:  0xf67fe4e4 00 00 00 00  0xf67fe024 00 00 00 00 .........$.......
0xf6ffef20:  0xf67b4a20 0a 00 00 00  0xf6ffeed0 03 00 00 00  J{..............
0xf6ffef30:   0x804822c  0xf6ffef84 8b c2 7e f6  0xf67b3ff4 ,..........~..?{.
0xf6ffef40:  0xf67b51e0 03 00 00 00 0xf67b43e4 81 ff 63 f6 .Q{......C{...c.
0xf6ffef50:  00 00 00 00 03 00 00 00  0xf6ffefa8 2f 9f 65 f6 ............/.e.
0xf6ffef60:  0xf67b4a20   0x804854c  0xf6ffef84 00 9f 65 01  J{.L.........e.
0xf6ffef70:  06 00 00 00  0xf67fe918  0xf67b3ff4  0xf67b3ff4 .........?{..?{.
```

QIRA


103
0
0x8048536
0xf6fff508
97 0x80483ab jmp 0x8048380
98 0x8048380 push dword ptr [0x804a004]
99 0x8048386 jmp dword ptr [0x804a008]
100 0x8048527 lea eax, dword ptr [esp + 0x28]
101 0x804852b mov dword ptr [esp + 4], eax
102 0x804852f mov dword ptr [esp], 0x8048619
103 0x8048536 call __isoc99_scanf
104 __isoc99_scanf jmp dword ptr [____isoc99_scanf]
105 0x80483e6 push 0x28
106 0x80483eb jmp 0x8048380
107 0x8048380 push dword ptr [0x804a004]
108 0x8048386 jmp dword ptr [0x804a008]
109 0x804853b mov edx, dword ptr [esp + 0x1c]
110 0x804853f mov eax, dword ptr [esp + 0x20]
111 0x8048543 imul eax, edx
112 0x8048546 mov edx, dword ptr [esp + 0x24]
113 0x804854a shl edx, 2
114 0x804854d add eax, edx
115 0x804854f mov dword ptr [esp + 0x2c], eax
116 0x8048553 mov eax, dword ptr [esp + 0x28]
EAX: 0xf6fff518 ECX: 0xf67bd898 EDX: 0x0
EBX: 0xf67bc000 ESP: 0xf6fff4f0 EBP: 0xf6fff528
ESI: 0x0 EDI: 0x0 EIP: 0x8048536
0xf6fff4ec <-- 0x804853b
91 fstat64(1,0xf6ffeeb0) = 0
91 mmap2(NULL,4096,PROT_READ|PROT_WRITE,MAP_PRIVATE|MAP_ANONYMOUS,-1,0) = 0xf6
99 write(1,0xf660f000,8) = 8
108 fstat64(0,0xf6fff1c0) = 0
108 mmap2(NULL,4096,PROT_READ|PROT_WRITE,MAP_PRIVATE|MAP_ANONYMOUS,-1,0) = 0xf6
108 read(0,0xf660e000,1024) = 4
0xf6fff4c0: 00 00 00 00 0xf67bc000 0xf67bcac0 0c 51 67 f6 ......{...{..Qg.
0xf6fff4d0: 0xf67bcac0 00 15 7f f6 0xf67bd898 a0 50 67 f6 ..{.......{..Pg.
0xf6fff4e0: 0xf67bc000 00 00 00 00 00 00 00 00 0x8048527 ..{.........'...
0xf6fff4f0: 0x8048619 0xf6fff518 0x804a000 0x80485d2 ....
0xf6fff500: 01 00 00 00 0xf6fff5c4 0xf6fff5cc 36 00 00 00 ....6...
0xf6fff510: 0c 00 00 00 58 00 00 00 0x804858b 0xf67bc000 ....X.........{.
0xf6fff520: __libc_csu_ 00 00 00 00 00 00 00 00 83 aa 62 f6 ....b.
0xf6fff530: 01 00 00 00 0xf6fff5c4 0xf6fff5cc ea bc 7e f6 ....~.
0xf6fff540: 01 00 00 00 0xf6fff5c4 0xf6fff564 ____libc_st ....d.......
0xf6fff550: 0x8048260 0xf67bc000 00 00 00 00 00 00 00 00 `.....{.........
0xf6fff560: 00 00 00 00 e1 89 8f c6 e0 6d 31 fc 00 00 00 00 ....m1.....
0xf6fff570: 00 00 00 00 00 00 00 00 01 00 00 00 _start ....
0xf6fff580: 00 00 00 00 00 15 7f f6 99 a9 62 f6 0xf67fe000 ....b.....
0xf6fff590: 01 00 00 00 _start 00 00 00 00 0x8048411 ....
0xf6fff5a0: main 01 00 00 00 0xf6fff5c4 __libc_csu_ ....
0xf6fff5b0: __libc_csu_ 80 c1 7e f6 0xf6fff5bc 1c 00 00 00 ......~.........
0xf6fff5c0: 01 00 00 00 0xf6fff6ed 00 00 00 00 0xf6fff70b ....
0xf6fff5d0: 0xf6fff71e 0xf6fff73c 0xf6fff767 0xf6fff770 ....<...g...p...
0xf6fff5e0: 0xf6fff780 0xf6fff790 0xf6fff7a3 0xf6fff7ab ....
0xf6fff5f0: 0xf6fff7b6 0xf6fff7c7 0xf6fff7d2 0xf6fff7ed ....
0xf6fff600: 0xf6fff803 0xf6fff8f9 0xf6fff910 0xf6fff934 ....4...
0xf6fff610: 0xf6fff93f 0xf6fffe3d 0xf6fffe4a 0xf6fffe86 ?...=...J.......
main push ebp
0x80484ee mov ebp, esp
0x80484f0 and esp, 0xfffffff0
0x80484f3 sub esp, 0x30
0x80484f6 mov dword ptr [esp + 0x1c], 0x36
0x80484fe mov dword ptr [esp + 0x20], 0xc
0x8048506 mov dword ptr [esp + 0x24], 0x58
0x804850e mov dword ptr [esp], 0x8048610
0x8048515 call printf
0x804851a mov eax, dword ptr [__stdout]
0x804851f mov dword ptr [esp], eax
0x8048522 call fflush
0x8048527 lea eax, dword ptr [esp + 0x28]
0x804852b mov dword ptr [esp + 4], eax
0x804852f mov dword ptr [esp], 0x8048619
0x8048536 call __isoc99_scanf
0x804853b mov edx, dword ptr [esp + 0x1c]
0x804853f mov eax, dword ptr [esp + 0x20]
0x8048543 imul eax, edx
0x8048546 mov edx, dword ptr [esp + 0x24]
0x804854a shl edx, 2
0x804854d add eax, edx
0x804854f mov dword ptr [esp + 0x2c], eax
0x8048553 mov eax, dword ptr [esp + 0x28]
0x8048557 cmp dword ptr [esp + 0x2c], eax
0x804855b jne loc_804856b
loc_804855d mov dword ptr [esp], aWinner
0x8048564 call puts
0x8048569 jmp loc_8048577
loc_804856b mov dword ptr [esp], aNope
0x8048572 call puts
loc_8048577 mov eax, 0
0x804857c leave
0x804857d ret
0x804835a 00
0x804835b 00
_init push ebx
0x804835d sub esp, 8
0x8048360 call __x86.get_pc_thunk.bx
0x8048365 add ebx, 0x1c9b
0x804836b mov eax, dword ptr [ebx - 4]
0x8048371 test eax, eax
0x8048373 je loc_804837a
loc_8048375 call __gmon_start__
loc_804837a add esp, 8
0x804837d pop ebx
0x804837e ret
0x804837f 00
0x8048380 push dword ptr [0x804a004]
0x8048386 jmp dword ptr [0x804a008]
0x804838c 00
0x804838d 00

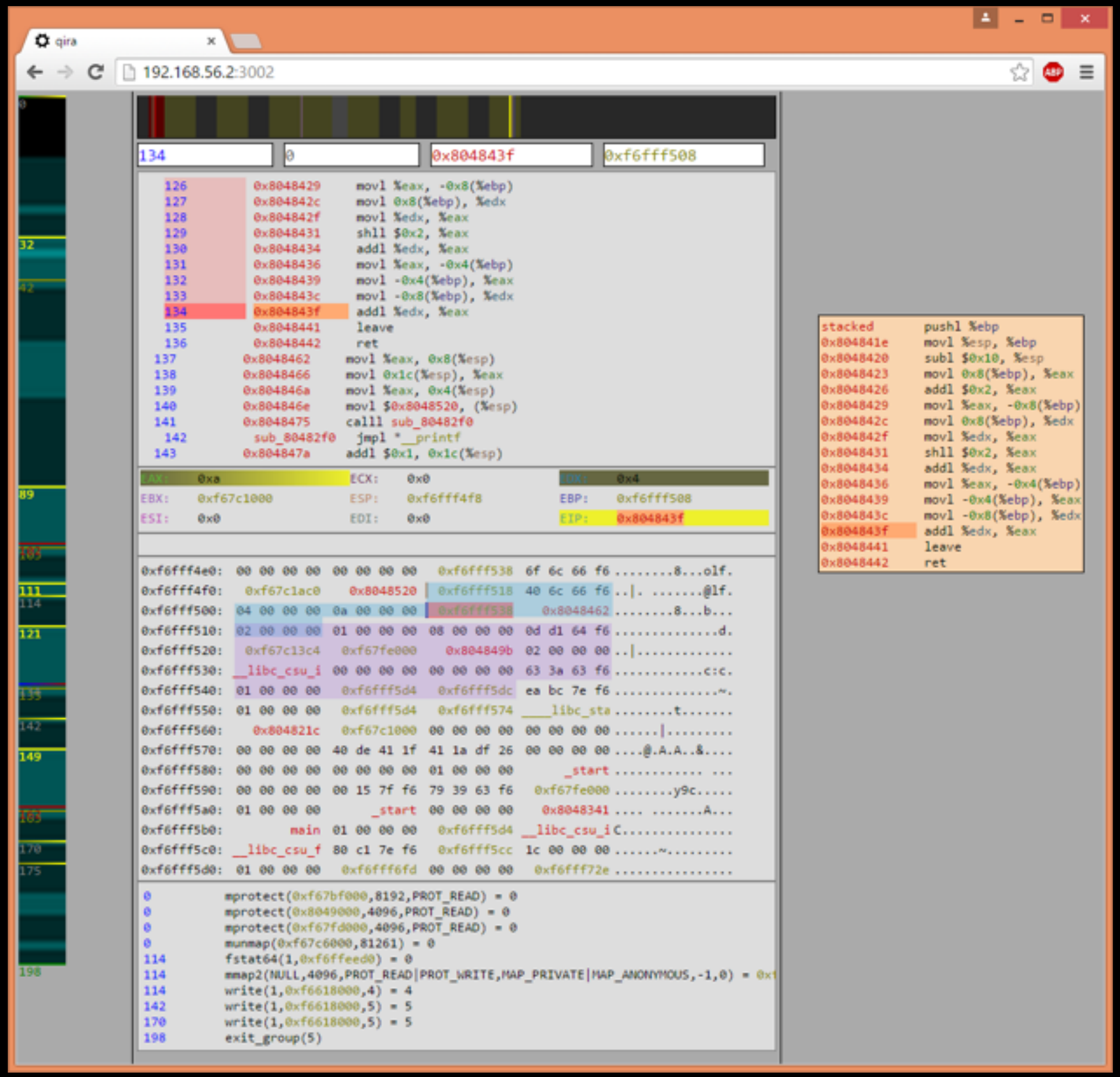
Where was eip?



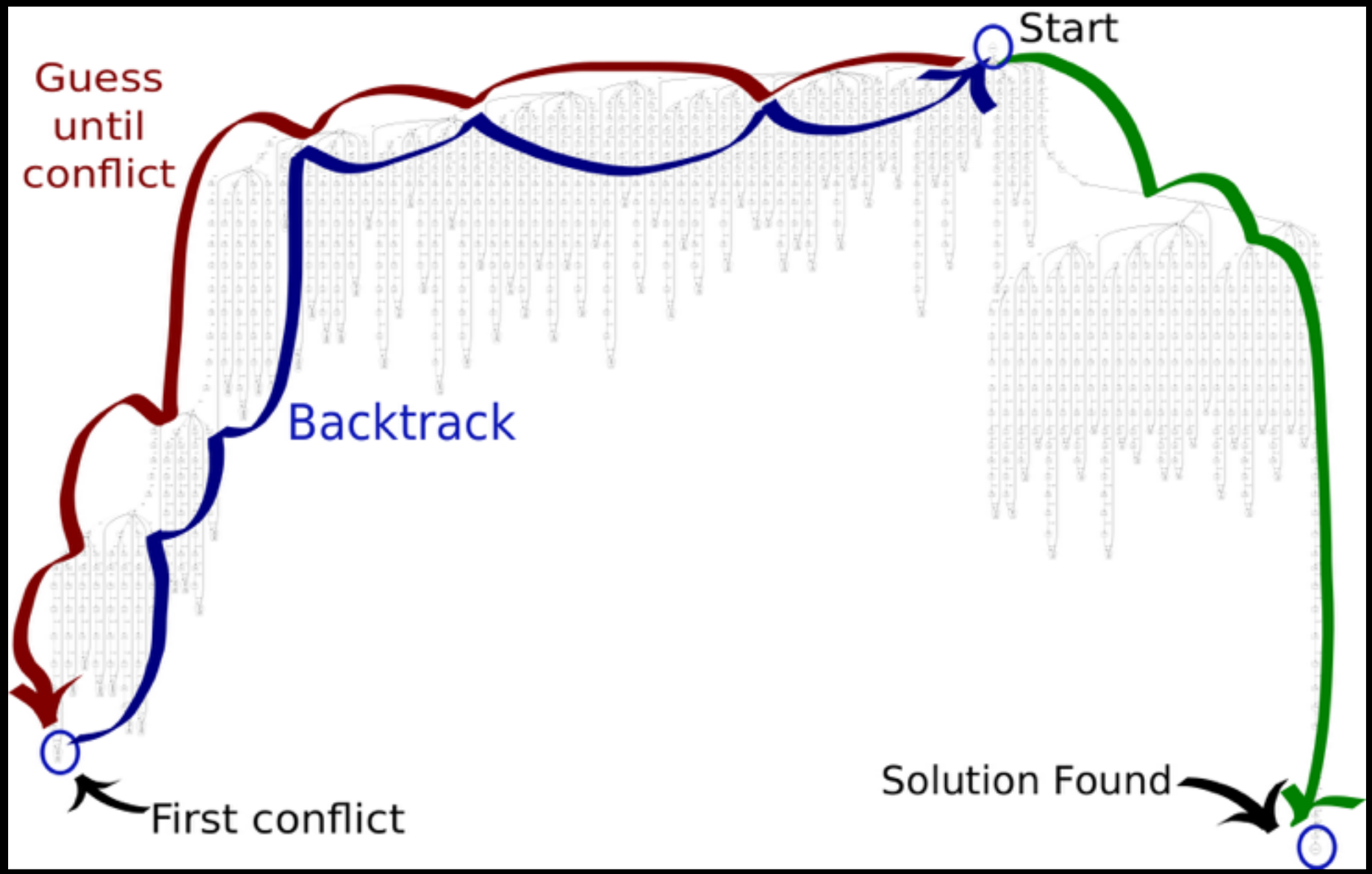
DEMO

qira.me

Type Information



SAT Solvers


Start
Guess
until
conflict
Backtrack
First conflict
Solution Found

Rewind Forking


```
0xf6fff430:  0xf6fff3ec 03 00 00 00 03 00 00 00 02 00 00 00
0xf6fff440:  0xf6fff4a0 81 b1 7e f6  0xf67daa10   0x804863d
0xf6fff450:  0xf6fff470 03 00 00 00  0xf67fe000 72 c3 7e f6
0xf6fff460:  0xf67fe504  0xf660f000  0xf6fff484  0xf67bc030
0xf6fff470:  0xf67fe938  0xf67da858  0xf67fe55c 2c c2 7e f6
0xf6fff480:  0xf67fe504  0xf67dd40c 08 00 00 00  0xf67dab48
0xf6fff490: 01 00 00 00  0xf67bc000 00 00 00 00  0xf67fe040
```

Future of the Project

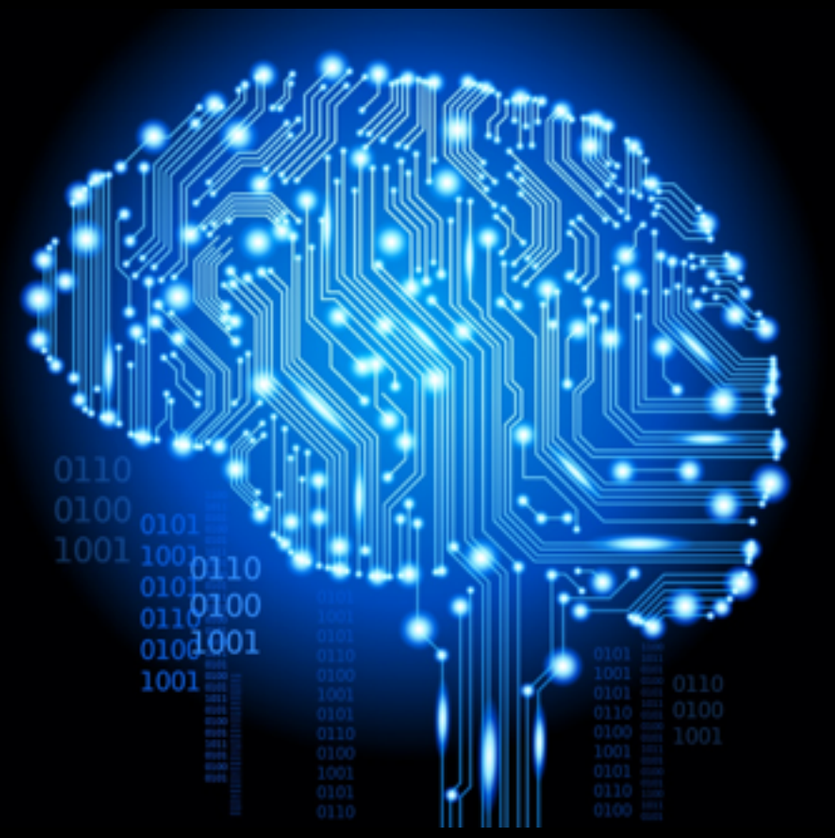
Carnegie Mellon University

BinaryAnalysisPlatform / **bap**

Companies spend millions of dollars to make puzzles for me to solve

But the puzzles are getting tedious and repetitive

My 2015



vicarious™

I’m retired from hacking

Questions?

https://github.com/BinaryAnalysisPlatform/qira

https://soundcloud.com/tomcr00se