Where is Reverse Engineering used? [closed]

Asked 15 years, 10 months ago Modified 11 years, 9 months ago

Viewed 8k times

19

As it currently stands, this question is not a good fit for our Q&A format. We expect answers to be supported by facts, references, or expertise, but this question will likely solicit debate, arguments, polling, or extended discussion. If you feel that this question can be improved and possibly reopened, visit the help center for guidance.

Closed 12 years ago.

I ask myself **where** reverse engineering is used. I'm interested at learning it. But I don't know if I can/should put it on my CV.

I don't want my new chief to think I am an evil Hacker or something. :)

- So is it worth it?
- Should I learn it or put my effort somewhere else?
- Is there a good Book or tutorial out there? :)

reverse-engineering

Share

Improve this question

Follow

edited Feb 15, 2009 at 10:12

Jeff Atwood

63.9k ● 48 ● 151 ● 153

asked Feb 15, 2009 at 10:02

n00ki3

14.8k ● 18 ● 57 ● 65

8 Answers

Sorted by: Highest score (default)

18

Reverse engineering is commonly used for **deciphering file formats** for improving interoperability. For example, many popular commercial Windows applications don't run on Linux, which necessitates reverse engineering of files produced by those applications, so that they can be used in Linux. A good example of this would be the various formats supported by Gimp, OpenOffice, Inkscape, etc.

Another common use of reverse engineering is **deciphering protocols**. Good examples include Samba, DAAP support in many non-iTunes applications, cross platform IM clients like Pidgin, etc. For protocol reverse engineering, common tools of the trade include Wireshark and libpcap.

No doubt reverse engineering is often associated with software cracking, which is primarily understanding

program disassembly. I can't say that I've ever *needed* to disassemble a program other than out of pure curiosity or to make it do something it wasn't. One plus side to reverse engineering programs is that to make any sense of it, you will need to learn assembly programming. There are however legal ways to hone your disassembly skills, specifically using Crackmes. An important point to be made is that when you're developing security measures in your applications, or if you're in that business, you **need** to know how reverse engineers operate to try to stay one step ahead.

IMHO, reverse engineering is a very powerful and useful skill to have. Not to mention, it's usually fun and addictive. Like hmemcpy mentioned, I'm not sure I would use the term "reverse engineering" on my CV, only the skills/knowledge associated with it.

edited Feb 15, 2009 at 11:05

answered Feb 15, 2009 at 10:50

codelogic
73.5k ● 9 ● 61 ● 55

8

Reverse engineering is usually something you do because you have to, not because you want to. For example, there are legal issues with simply reverse engineering a product! But there are necessary cases - where (for example) the supplier has gone and no longer exists or is not contactable. A good example would be the WMD editor that you typed your question into. The SO team/community [had to reverse engineer this from obfuscated source] to apply some bug fixes.

Share Improve this answer

Follow

edited Jan 18, 2021 at 12:38

Community Bot

1 • 1

answered Feb 15, 2009 at 10:10

Marc Gravell

1.1m • 272 • 2.6k • 3k

8

One of the fields, in my opinion, where reverse engineering skills might be useful is anti-virus industry, for instance. However, I wouldn't place "reverse engineering" on my CV, but rather I'd write down experience in the Assembly language, using miscellaneous disassemblers/debuggers (such as IDA, SoftIce or OllyDbg) and other relevant skills.

Share Improve this answer

Follow

edited Feb 15, 2009 at 12:49

answered Feb 15, 2009 at 10:07

Igal Tabachnik
31.5k ● 15 ● 95 ● 157

Yes, reverse-engeneering can be used for search vulnerabilities at the code (low-level or architecture errors and 'Undocumented features'). This project is about this problem (but with Russian lang.): demono.ru – Egg Head Oct 19, 2013 at 22:57

4

I have worked on reverse engineering projects, and they certainly had nothing to do with hacking. We had the source code for all such projects (legitimately), but for one of the projects nobody actually knew what the code did behind the scenes, and how it interacted with other systems. That information had long been lost. In another project, we had the source code and some documentation, but the documentation wasn't up to date, so we had to reverse-engineer the source to update the documentation.

I don't mind having such projects on my CV. In fact, I believe I've learned a lot during the process.

Share Improve this answer

Follow

edited Feb 15, 2009 at 13:29

answered Feb 15, 2009 at 12:26

Hosam Aly
42.4k ● 37 ● 146 ● 182

2

Reverse engineering is needed whenever the documentation is lost or it never existed. Having the source helps, but you still have to reverse engineer the original logic, flow control and bugs out of it.

Share Improve this answer

Follow

answered Feb 15, 2009 at 10:11

Petteri H

12.2k ● 12 ● 67 ● 96

1

Working with strange hardware often forces you to reverse engineer. For instance, I was once working with an old signal acquisition card that behaved strangely; putting in beautiful sine wave produced awfully crippled data. It turned out that every other byte was two's complement and every other one's complement - or at least, when interpreted that way, the data became quite beautiful. Of course, this wasn't documented anywhere, and the card worked perfectly when used with its own proprietary software.

Share Improve this answer

Follow

answered Feb 15, 2009 at 10:53

Joonas Pulakka

36.6k ● 29 ● 108 ● 171

1

It is very common (in my experience) to encounter older code which has defects, has become outdated due to changing requirements, or both. It's often the case that there's inadequate documentation, and the original developer(s) are no longer available. Reverse

engineering that code to understand how it works (and sometimes to make a repair-or-replace decision) is an important skill.

If you have the source, it's often reasonable to do a small, carefully-planned, strictly-scoped amount of cleanup. (I'm hinting out loud that this can't be allowed to become a sinkhole for valuable developer time!)

It's also very helpful to be able to exercise the code in a testbed, either to verify that it does what was expected or to identify, document, isolate, and repair defects.

Doing so safely requires careful work. I highly recommend Michael Feathers' book *Working with Legacy Code* for its practical guidance in getting such code under test.

Share Improve this answer

Follow

answered Feb 15, 2009 at 12:48

joel.neely

30.9k ● 9 ● 57 ● 64

0

RCE is great skill for security guys (research, exploitation, IDS, IPS, AV etc.) but also it proves that you've got a deep and low level understanding of the subject.

Finding your way way around easier when working with 3rd party libraries as well.

If you are not working in security industry, if you are not good at ASM don't bother to learn it, generally it's hard to

learn.

Books

Hacking the art of exploitation talks about the subject from security point of view.

Also you might want to read books about Ollydbg and IDA Pro

Share Improve this answer

Follow

edited Mar 25, 2013 at 14:27

asheeshr

4,114 ● 6 ● 32 ● 50

answered Feb 15, 2009 at 13:16

dr. evil

27.2k ● 37 ● 134 ● 202