Reverse Engineering Software

Miles Lajoie

CS 305

Outline

This presentation will cover:

- What reverse engineering software entails.
- The legality of reverse engineering software.
- Past legal cases

What is Reverse Engineering?

- Reverse Engineering is understanding how a proprietary piece of software works.
- Often stated in EULAs as being forbidden, however it is rarely ever "defined."
- RE might just be understanding the program on a technical level.
- But RE might also be taking it apart for evil reasons!
- Regardless of legality, reverse engineering is usually frowned upon.

Black Boxing

- The easiest and most common form of RE is observation of behavior. It is akin to design in the development of new software.
- Strings, some comments, and variables are usually visible in plaintext inside compiled files or binaries.
- A step up is viewing memory, to see how the program works specific portions of itself at runtime.
- This can lead to cracking a specific feature.

Example

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C
0000A6A0	4D	00	E8	D9	A9	FF	FF	53	6A	03	B9	B8	9C
0000A6C0	00	A3	8C	BF	4D	00	E8	B5	A9	FF	FF	68	A0
0000A6E0	CC	93	05	00	A1	24	7C	4C	00	B9	F0	9C	4D
0000A700	5C	BF	4D	00	53	68	00	00	80	3F	8B	11	FF
0000A720	3C	BB	FF	FF	53	B9	58	60	4D	00	E8	41	A5
0000A740	BC	03	00	00	A1	E0	BF	4D	00	83	C4	08	3B
0000A760	04	3B	C3	74	ED	68	4C	81	4C	00	E8	B1	08
0000A780	81	EC	20	04	00	00	68	58	83	4C	00	6A	01
0000A7A0	81	C4	20	04	00	00	C3	FF	15	64	F0	4B	00
0000A7C0	EF	00	00	6A	00	FF	15	28	F2	4B	00	6A	30
0000A7E0	15	D0	BF	4D	00	52	FF	15	48	F0	4B	00	33
0000A800	4D	00	F3	AB	8D	44	24	04	50	FF	15	90	F0
0000A820	81	F9	00	00	00	08	73	63	8D	54	24	20	68
0000A840	44	24	20	68	80	00	00	00	50	8D	8C	24	28
0000A860	94	24	20	02	00	00	6A	30	68	58	83	4C	00
0000A880	4B	00	33	C0	81	C4	20	04	00	00	C3	B8	01
0000A8A0	51	53	8B	5C	24	0C	F6	C3	01	74	13	E8	50
0000A8C0	BB	4D	00	85	C0	75	0D	6A	00	FF	15	A4	F0
0000A8E0	74	05	E8	39	02	00	00	FF	15	98	F5	4B	00
0000A900	FF	FF	F6	C3	04	74	10	A1	A8	BF	4D	00	B9

Offset	0	1	2	3	4	5	6	7	8	9	A	B
0000A6A0	4D	00	E8	D9	A9	FF	FF	53	6A	03	B9	B8
0000A6C0	00	A3	8C	BF	4D	00	E8	B5	A9	FF	FF	68
0000A6E0	CC	93	05	00	A1	24	7C	4C	00	B9	F0	9C
0000A700	5C	BF	4D	00	53	68	00	00	80	3F	8B	11
0000A720	3C	BB	FF	FF	53	B9	58	60	4D	00	E8	41
0000A740	BC	03	00	00	A1	E0	BF	4D	00	83	C4	08
0000A760	04	3B	C3	74	ED	68	4C	81	4C	00	E8	B1
0000A780	81	EC	20	04	00	00	68	58	83	4C	00	6A
0000A7A0	81	C4	20	04	00	00	C3	FF	15	64	F0	4B
0000A7C0	EF	00	00	6A	00	FF	15	28	F2	4B	00	6A
0000A7E0	15	D0	BF	4D	00	52	FF	15	48	F0	4B	00
0000A800	4D	00	F3	AB	8D	44	24	04	50	FF	15	90
0000A820	81	F9	00	00	00	08	EB	63	8D	54	24	20
0000A840	44	24	20	68	80	00	00	00	50	8D	8C	24
0000A860	94	24	20	02	00	00	6A	30	68	58	83	4C
0000A880	4B	00	33	C0	81	C4	20	04	00	00	C3	B8
0000A8A0	51	53	8B	5C	24	0C	F6	C3	01	74	13	E8
0000A8C0	BB	4D	00	85	C0	75	0D	6A	00	FF	15	A4
0000A8E0	74	05	E8	39	02	00	00	FF	15	98	F5	4B
0000A900	FF	FF	F6	C3	04	74	10	A1	A8	BF	4D	00

White Boxing

- Disassembly reproduces assembly code out of a compiled binary or program.
- Decompilation generates an equivalent source code in a target language, minus comments and names.
- Decompiling can lead to theft of algorithms and other integral pieces of software, but it might also be used for data recovery.
- In many developer communities, decompiling a piece of software is almost always frowned upon, even if its legal.

Example

Original Source:

```
int fib(int x)
{
    if (x > 1)
        return(fib(x - 1) + fib(x - 2));
    else
        return x;
}
```

Decompiled Code:

```
int fib(int param5)
{
    int local8;
    int local9;

    if (param1 <= 1) {
        local9 = param5;
    } else {
        local8 = fib(param5 - 1);
        local9 = fib(param5 - 2);
        local9 += local8;
    }
    return local9;
}
```

(But they are equivalent)

When is it Legal?

- You own the software, even if the software says it is “protected.”
- Attempting to achieve interoperability or compatibility.
- Company or original programmer disappeared in a poof of smoke.
- Investigation of patent or copyright infringement.

When is it Illegal?

- Some goals of interoperability.
- When possible theft of code can be ruled.
- If it leads directly to a criminal act.
- If it is done for commercial purposes.

A Gray Area

- The legality of reverse engineering tends to be handled on a case-by-case basis.
- Reverse engineering a piece of software for private purposes, such as learning, is not exactly enforceable.
- Reverse engineering a piece of software to crack something such as a copyright check can be hard to catch.
- Disassembly and decompilation is frequently done for nefarious, but strangely legal purposes.

Some Legal Examples

- Atari Games v. Nintendo (1992)
- Sega v. Accolade (1992)
- Computer Associates v. Altai (1992)
- Sony v. Connectix (2000)
- Blizzard v. bnetd (2005)

Conclusion

- Reverse engineering software is a grayscale for legality, but usually frowned upon in terms of ethics.
- The laws of RE are subject to interpretation and legal incidents of it are evaluated on a case-by-case basis.
- EULAs are just a scare tactic, they don't legally prevent you from reversing their software.
- But sometimes, the law does get superseded in court.

References

- 1). Intro to Reverse Engineering: http://www.acm.uiuc.edu/sigmil/RevEng/index.html
- 2). Frequently Asked Questions (and Answers) about Reverse Engineering: http://www.chillingeffects.org/reverse/faq.cgi#QID188
- 3). The Laws and Economics of Reverse Engineering: http://www.yalelawjournal.org/pdf/111-7/SamuelsonFINAL.pdf
- 4). Software, Reverse Engineering and the Law: https://lwn.net/Articles/134642/
- 5). Copyright of Computer Programs: http://digital-law-online.info/lpdi1.0/treatise25.html
- 6). Ethics in Computing: http://ethics.csc.ncsu.edu/intellectual/reverse/study.php
- 7). Intel Architecture Software Developer's Manual (Volume 2): http://download.intel.com/design/PentiumII/manuals/24319102.PDF