CS 491/591 Spring 2016 Lab 1

Due 11:59pm on Wednesday, 25 March 2016

Please send your submission for lab 1 as a gzipped tar ball attachment to an email to “crandall@cs.unm.edu”. The subject of the email should contain both the strings “lab1” and “reclass”. Do not submit your lab writeup to any other address, attach any files that are not the one single tar ball, or put any part of your lab submission in the text body of the email. Include your name in the body of the email in case it's not obvious from your email address.

The tar ball should contain a single directory, with all files within that single directory. It should include your writeup in PDF format (no other formats will be accepted, especially not Microsoft Word), plus two or more PE binaries as per below.

You should take the minesweeper PE *.exe binary (winmine.exe) from the System32 directory of the standard VMWare image that Bob and Carry provided and make some modifications to it. You should add a backdoor way for you to win easily, but the game logic doesn't change for people that don't know the backdoor. You should also use anti-disassembly or other obfuscation tricks to make it harder to discover and/or reverse engineer your modifications.

Lab 1 is worth 100 points, based on the following rubric:

- 70 points for your one-page writeup describing the modifications you made and your attempts to hide/obfuscate them. Since different students have a different skill level with x86 assembly, etc., the grade will be based on originality and demonstration of material covered in the class rather than on raw technical sophistication.
- 25 points for properly updating the CRC32 checksum in the PE header of your modified binary.
- 5 points if the CRC32 checksum of your modified binary matches the CRC32 checksum of the original binary
- 50 extra credit points if you can produce two modified binaries that have the same MD5 sum but different SHA1 sums.
- Automatic A+ in the class if you can produce a modified binary with the same MD5 sum as the original (don't spend too much time on this, it's probably not feasible).

You are expected to do your own work. From reverse engineering to modifications to writing the writeup, for all phases of this project you should do your own work. Any instance of not doing your own work will be considered cheating. For your writeup, if you copy even a single sentence from an existing source without clearly attributing it to the correct authors, that will be considered cheating. If you're not sure whether something will be considered cheating or not, ask me before you do it. You are encouraged to discuss the assignment with your classmates at a high level. Exchanging tools, source code that existed before the assignment was assigned, and thoughts about approaches to specific problems is okay. As a reminder of the course policy, if you cheat on any assignment in this class including this assignment (cheating includes, but is not limited to, representing somebody else's work as your own or fabricating files to make it look like you completed the assignment) you will receive an F in the class. Every student's modifications to the binary should be unique, both in terms of code and
in terms of functionality. If you want to share source code written for the assignment with a classmate, you should get my permission first and share it with the whole class. *E.g.*, if you write some Python code to pull apart a PE file and then modify it and put it back together, that source code is the kind of thing that I would encourage you to share with your classmates as long as it doesn't have assignment specific things in it (like code to make the CRC32 checksum match the original, for example).

The number of binaries that you include in your tar ball depends on what parts of the assignment and extra credit you did, but at a minimum you should include the original unmodified binary (which should be the same for all students because you're supposed to get it directly from the VMWare image---but I still want you to include it to verify you started with the right binary) and your modified version.

You can also include source code in your tar ball if you wish. You can use any tools or existing source code (*i.e.*, that existed before the assignment was assigned) from any source so long as you cite it in your writeup. Tools I'd recommend looking into include Python libraries for dealing with PE file formats, hex editors, assemblers, and all the tools we've talked about in class so far.