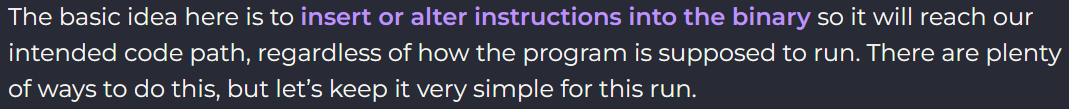
Patching x86 binaries:

Step 1: Open file in cutter & make sure thr “Load in write mode” box is selected like so Graphical user interface, application

Description automatically generated

<https://notes.huskyhacks.dev/notes/on-patching-binaries> see this link for a lot of helpful x86 assembly stuff!

Patching is editing a piece of malware so that a condition will always (or never) run depending on what we want. For example, if a certain function is called only if the SHA256 sum hardcoded into the program matches the SHA256 sum of some file/ webpage. We have no way of verifying weather these two hashes are the same so we can change the binary to make them always be the same and therefore execute the function in question.

In other words:

We can change an instruction in cutter by right clicking on it going to “edit” then “reverse jump”, this will simple change a jne (jump if not equal) instruction to je (jump if equal) and vice versa therefore reversing the jump to the opposite of what the program was doing.

After we change the binary, make sure to save it in cutter (default saves to a .rzdb aka Rizin Database file)

Once we remove the extension and run it we should get the alternate result we were looking for.

There are other techniques for patching binaries as well … do your research.