

20875 Software Engineering

Tutorial 2

1. Download the archive file `secret.zip` and uncompress it. It contains three files: `alpha`, `beta` and `gamma`. They are executables (but for safety reasons, it is recommended to not execute them).
2. Determine the ISA and OS that `alpha` and `beta` were built for.
3. `alpha` is a well-known app. Determine which one.
4. `beta` contains a (badly) hidden password. Find this password.
5. `beta` also contains a compressed image. Determine the image format and the library it uses to uncompress it. (Note: `gamma` is the same app as `beta`, but compiled for a different platform.)
6. [hard] Extract this image from the executable files.
7. Write a C program that takes two arguments: a file name, and the size of this file in bytes. The program maps the contents of the file into memory using `mmap()`. Then, it accesses the contents and prints them in hexadecimal (similar to what `hexdump` does).
8. [long] Download source code of the library ‘GMP’ from <https://gmplib.org/> (latest version). Compile the library and run a few of the tests.
9. Write a C program that allocates a huge amount of memory, then writes arbitrary data to it. Study the behavior of the program for different amounts of memory (in particular, less than currently free memory, more than currently free memory, more than total memory installed).
10. Consider the following task. We have a directory called `source/` that contains images. For each image, we want to create, in the directory `thumbnail/`, a version of this image whose width and height are both at most 256 pixels. You can use the `convert` or `magick` utility to that end. Write a `Makefile` that automates this task, and allows us to recompute the thumbnails only when source images are modified.

Hint: a few links to the `magick` documentation:

- General `magick` command documentation
- `-scale` parameter
- geometry specification