Golang System Software Engineer - Technical Assessment

Git repository link:

Exercise 1 - Bootable Linux image via QEMU

In this exercise, I created a system for downloading and automatically building an `AMD64` kernel image, that prints "hello world" after booting successfully. It is then ran using `QEMU`.

I did two decisions on this project:

1. I decided to implement the whole thing as a "naked" kernel image, that then ran a binary that printed "hello world" as `PID1`. I did this to ensure that no login shell or anything that could be unimportant (like user-land tools) was present on the image.
2. I implemented the build tool as a `Makefile`. Instead of writing a normal shell script, by implementing it as a `Makefile`, we get all the advantages of using a build system (like automatically managing rule dependencies and only having to rebuild what is necessary).

Exercise 2 - Shred tool in Go

In this exercise, I implemented a `Shred(path)` function that will overwrite the given file (e.g. "randomfile") 3 times with random data and delete the file afterwards. Note that the file may contain any type of data.

Some of the most important test cases are:

1. Check if it succeeds on a normal file (data is different, file has the same size, file was removed, etc.)
2. Check if it fails when it tries to shred a file with wrong permission/that does not exist.
3. Check if for two different files if the random data generated is different (both for different function calls and for each iteration of a write of random bytes).
4. Check if the random writes were properly flushed before removing the file.

`Shred` is an important tool when dealing with the removal of confidential files (files we DON'T want to be able to recover after being deleted). The reason for this is that some filesystems and disk firmware don't actually delete a file when asked for it to be removed; for performance reasons they may simply just delete it from the file system metadata. Unfortunately, sometimes this allows some tools to recover previously deleted files.

`Shred` makes sure that:

- A file was deleted from the disk in the normal way
- And that all someone can recover is random data, since that was what was written in the file.