

gist:

Break a restricted shell to get the flag with a very limited set of input chars

steps:

1. run Docker container
2. open in browser
3. find limitations of the shell
4. try and research commands and flags
5. get the flag

So after running the Docker container and opening it in a browser via localhost:80 you see a very basic input prompt. A first attempt to submit input yielded that all input is restricted to the letters constituting "fiberglass" and spaces.

With this set, "ls" was the first command on my mind.

There is the flag file already! Would be so easy to read it out via cat, grep, printf, strings ... all of which have letters that are just not allowed.

So I found: <https://ss64.com/bash/> - a wholesome list of shell commands and browsed through them which ones fit with the letters. Dismissed some ideas and soon found, I should concentrate on the "file" command. Since I tried ls with some flags, I knew that this works. So I asked DeepSeek for flags:

which flags can be used with the bash command "file" and what do they do?

The `file` command in Linux/Unix determines file types using magic tests (file signatures, extensions, etc.). Below is a breakdown of its **most useful flags** and their purposes:

Basic Syntax

```
bash
file [OPTIONS] <file>
```

Key Flags & Their Functions

Flag	Description
<code>-b</code>	Brief mode - Omits filenames in output (shows only file type).
<code>-e <test></code>	Exclude a test (e.g., <code>-e soft</code> to skip symlink checks).
<code>-f <file></code>	Read filenames from a file (treats each line as a file to test).

This was the thing I needed, so I entered:

```
file -f flag
```

and it yielded an error message, as the line it printed was of course not a file but rather **the flag**:

CTF{race_condition_symlink}