10.5 Imodule examples

These examples assume the instructor is working from a Labtainers distribution, e.g., one of the VM appliance.

10.5.1 Modify a lab manual for the telnet-lab

In this example, the instructor wants his or her students to work with a customized version of the telnet-lab manual.

- Change directory to `$LABTAINER_DIR/labs`
- Initialize the git archive:

  ```
  git init
  ```

  (Do this only once, no need to repeat for each IModule.)
- Add the original Labtainer file as the baseline:

  ```
  git add telnet-lab/docs/telnet-lab.docx
  ```

- Edit the telnet-lab/docs/telnet-lab.docx file
- Commit your change:

  ```
  git commit telnet-lab/docs/telnet-lab.docx
  ```

This change has no effect on any Docker container, so we need only generate the updated tar:

```
create-imodules.sh
```

Then publish the imodule.tar to the website.

10.5.2 Create a new lab

In this example, the instructor wants to create a new lab for use by his or her students. This example assumes the instructor has created a DockerHub registry that is publicly accessible.

- Change directory to labtainer/labs
- Initialize git archive: git init (Do this only once, no need to repeat for each IModule.)
- Create the lab per the Lab Designer User Guide, for this example, we assume the lab is my-new-lab.
- Include the name of your Docker Hub registry the lab config/start.config file `REGISTRY` attribute.
- Complete development and testing of the lab, e.g., build a SimLab test.
- While in the my-new-lab directory, run `cleanlab4svn.py` to remove temporary files that should not be under source control.
- While in the lab directory (parent of my-new-lab), add the lab to source control: