

VM

VM setup(Not needed for ubuntu)

1. Setup <https://ranger.uta.edu/~alex/courses/3318/slides.html>
2. VM virtualBox on linux <https://linuxize.com/post/how-to-install-virtualbox-on-ubuntu-20-04/>
3. VM setup https://uta.instructure.com/courses/100996/pages/omega-vm-valgrind?module_item_id=4226388
4. VM setup <https://www.youtube.com/watch?v=0dK8MEktWSk>

VM NOT NEEDED ON UBUNTU - DISCONTINUED

Omega

1. Setup Pulse – For auth, to ssh from bash

1. Download from here

https://uta.service-now.com/selfservice?id=ss_kb_article&sys_id=KB0010780

<https://pulse-vpn.uta.edu/dana/home/index.cgi>

2. Old vpn doc(Incorrect and not required)
https://ranger.uta.edu/~alex/courses/3318/resources/omega_server_directions_linux.pdf
3. Start Pulse
4. Open bash ->

- Type on local(bash)

“ssh nag3506@omega.uta.edu”

(nag3506 is your netId)

- This command is from old doc.

- <ISSUE 2>

5. <RESOLUTION TO ISSUE 2>

- Type on local

“ssh -oKexAlgorithms=+diffie-hellman-group1-sha1 nag3506@129.107.56.23”

6. For ease of access, Add the ssh keys -

- Type on local

“ssh-copy-id -oKexAlgorithms=+diffie-hellman-group1-sha1 nag3506@129.107.56.23”

- U can verify the ssh keys added on the remote(omega) server .ssh directory

7. Verify by scp(secured copy) a file onto omega location, compile it there with gcc, and verify by executing the .out file.
8. Done!

Valgrind

<https://stackoverflow.com/questions/24935217/how-to-install-valgrind-properly>

https://ranger.uta.edu/~alex/courses/3318/resources/Valgrind_Minimal_Tutorial.pdf

Valgrind-3.18.1

valgrind --leak-check=full ./a.out

Done

Setup VsCode

- <https://medium.com/floppy-disk-f/setting-up-visual-studio-code-vs-code-with-c-on-ubuntu-linux-506aa02cacb5>
- <https://ranger.uta.edu/~alex/courses/3318/code.html>
- Code Runner extension install
- C/C++ install
- Done
- Debugger - compile separately(gcc -g progName.c) then hit the debugger with the breakpoints
- <https://www.youtube.com/watch?v=X2tM21nmzfk>
- Done
- GREAT WRITE UP, required for multiple file debugging: https://code.visualstudio.com/docs/cpp/config-linux#_modifying-tasksjson
 - Create tasks.json in Terminal -> Configure Default Build Task -
 - Create launch.json -> Run Add Config
 - Ctrl + Shift + B -> Compile
 - Run - Debug -> Debug
 - For multiple files debugging, REFER the article
 - Modify the launch.json, and tasks.json
 - Do the *.c files thing, the directory thing
 - This shd work now
 - Ctrl + Shift + B and then Debug the code
 - eg – my tasks.json file args part

```
    "args": [
    "-fdiagnostics-color=always",
    "-g",
    "${workspaceFolder}/thisFolder/thisIsAnotherFolder/Homework3LiesHere/*.c",
    "-o",
    "${fileDirname}/${fileBasenameNoExtension}"
    ],
```

- Done

ISSUE 1. Kernel driver not installed (rc=-1908)

> Native ubuntu does not need VM, just valgrid

Kernel driver not installed (rc=-1908)

The VirtualBox Linux kernel driver is either not loaded or not set up correctly. Please reinstall virtualbox-dkms package and load the kernel module by executing

```
modprobe vboxdrv'
```

as root.

If your system has EFI Secure Boot enabled you may also need to sign the kernel modules (vboxdrv, vboxnetflt, vboxnetadp, vboxpci) before you can load them. Please see your Linux system's documentation for more information.

where: suplibOsInit what: 3 VERR_VM_DRIVER_NOT_INSTALLED (-1908) - The support driver is not installed. On linux, open returned ENOENT.

> **No needed for native Ubuntu, DIDN'T SOLVE THIS**

ISSUE 2. ssh nag3506@omega.uta.edu

Unable to negotiate with 129.107.56.23 port 22: no matching key exchange method found. Their offer: diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1,diffie-hellman-group1-sha1

Source: <https://unix.stackexchange.com/questions/340844/how-to-enable-diffie-hellman-group1-sha1-key-exchange-on-debian-8-0>

Resolution2:

> ssh -oKexAlgorithms=+diffie-hellman-group1-sha1 nag3506@129.107.56.23