cs 143A
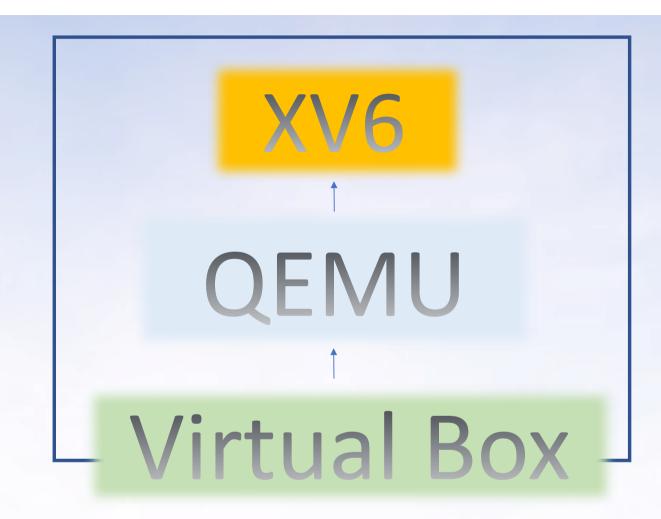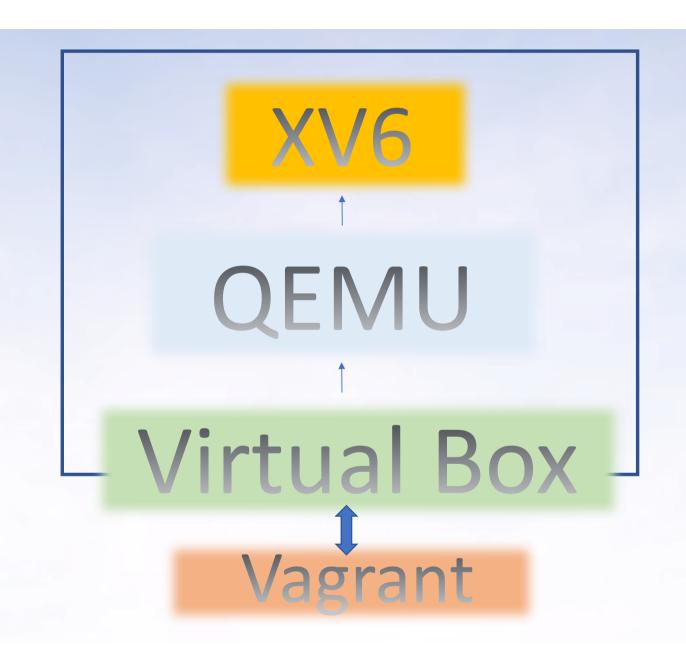
Principles of Operating Systems

https://goo.gl/UzpgBP

http://www.ics.uci.edu/~aburtsev/143A/hw/xv6-setup.html

ssh into Vagrant

We suggest you create a new folder for your ics143a homeworks, like
- andromeda$mkdir ics143a

Change into that directory:
- andromeda$cd ics143a

Fetch a version of the vagrant environment that explains to vagrant what kind of virtual machine you're planning to run:
- andromeda$ wget http://www.ics.uci.edu/~aburtsev/143A/hw/xv6-vagrant-master.tgz
- andromeda$ tar -xzvf xv6-vagrant-master.tgz

Change into the new folder
- andromeda$ cd xv6-vagrant-master

Change the name of the vagrant VM to something unique (otherwise we all end up with the same VM and vagrant is confused). In the `Vagrantfile` file change the following line

```
vb.name = "xv6_box_anton" # <--- You should change this to make VM names unique
```

Start vagrant VM (this will take several minutes as it is building QEMU inside)

```
odin$ vagrant up
```

If vagrant fails with the following message:

```
==> default: Clearing any previously set forwarded ports...
Vagrant cannot forward the specified ports on this VM, since they
would collide with some other application that is already listening
on these ports. The forwarded port to 20000 is already in use
on the host machine.

To fix this, modify your current project's Vagrantfile to use another
port. Example, where '1234' would be replaced by a unique host port:

  config.vm.network :forwarded_port, guest: 26001, host: 1234
```

Go ahead with the suggested fix. Change the following line in the Vagrantfile setting the host port to something random below 64000:

```
config.vm.network "forwarded_port", guest: 26001, host: 30000
```

If vagrant VM is up, you're ready to log in inside and start working on your xv6 Linux environment. Log in inside the vagrant VM. From the same folder where Vagrantfile is (i.e., from ics143a/xv6-vagrant-master) type

```
odin$ vagrant ssh
```

Boot xv6

From inside your Vagrant VM fetch the xv6 source:

```
vagrant@odin$ cd /vagrant
vagrant@odin$ mkdir ics143a
vagrant@odin$ cd ics143a
vagrant@odin$ git clone git://github.com/mit-pdos/xv6-public.git
Cloning into xv6...

...
```

Build xv6:

```
vagrant@odin$ cd xv6-public
vagrant@odin$ make

...

gcc -O -nostdinc -I. -c bootmain.c
gcc -nostdinc -I. -c bootasm.S
ld -m    elf_i386 -N -e start -Ttext 0x7C00 -o bootblock.o bootasm.o bootmain.o
objdump -S bootblock.o > bootblock.asm
objcopy -S -O binary -j .text bootblock.o bootblock

...
vagrant@odin$
```

make qemu-nox

- make qemu:
  Build everything and start qemu with the VGA console in a new window and the serial console in your terminal. To exit, either close the VGA window or press Ctrl-c or Ctrl-a x in your terminal.

- make qemu-nox
  Like make qemu, but run with only the serial console. To exit, press Ctrl-a x. This is particularly useful over SSH connections to Athena dialups because the VGA window consumes a lot of bandwidth.

Questions ?