ELEC 377
Operating Systems

Tutorial / Lab

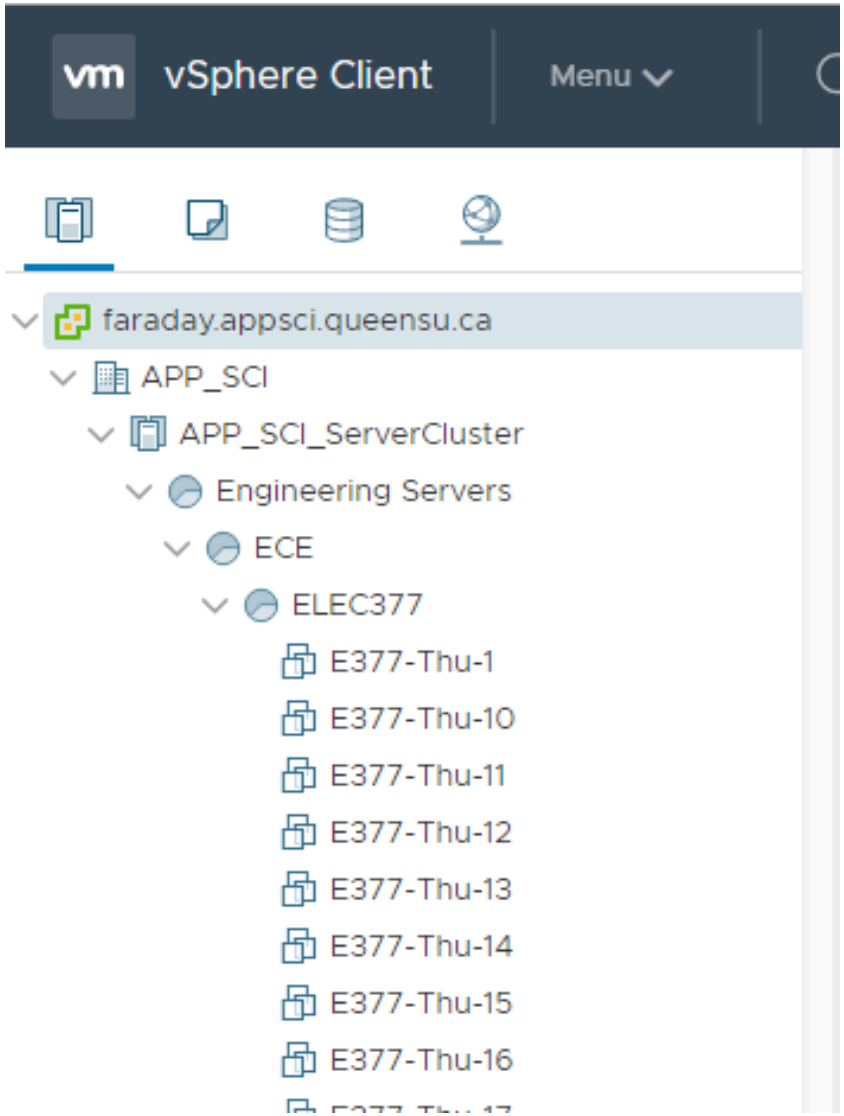
Labs - General Procedure

- First Lab Period of a Given Lab Assignment
  1 Hand in PreLab documentation (if required)
  2. Connect to your linux virtual machine
  3. Log in using the account *student (password: student)*
  4 Check out initial code from the git server
  5 Edit and compile your code
  6 Edit test data files and add to git control
  7 submit everything to the git server
  8 Test your program
  - (repeat from step 6 as necessary)
  9 Add output files to git control, submit
  10 log out of linux (exit, logoff)
  11 Log in as root and shut down linux machine

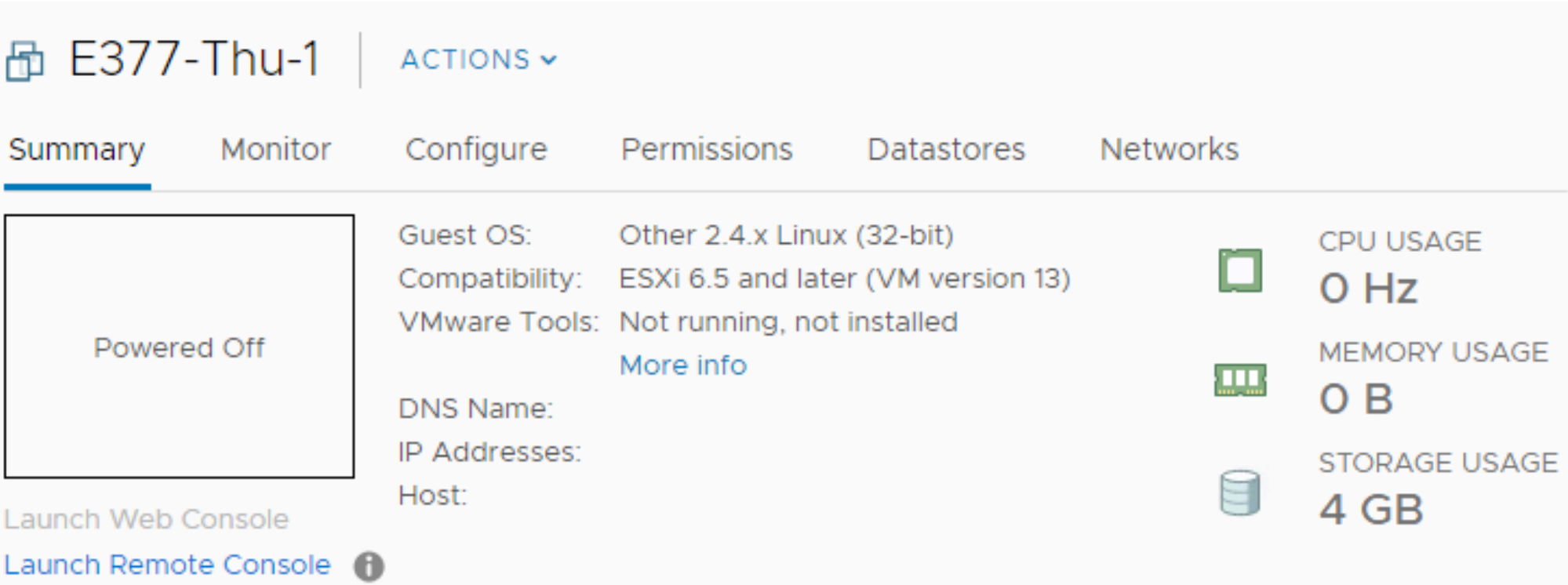
Labs - Connecting to Linux VM

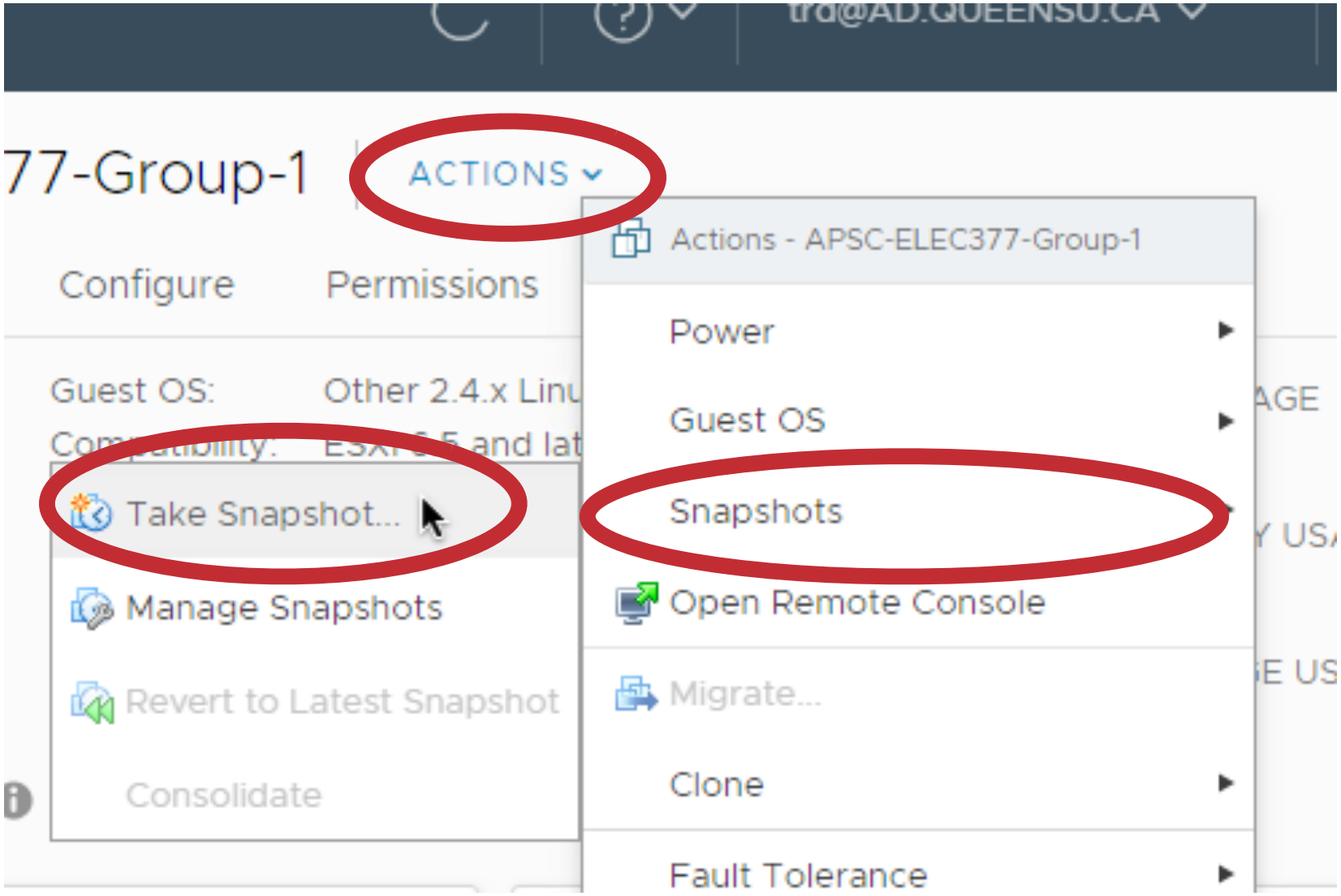
- Because of some issues in the current version of the server software, you can't log in using the firefox client

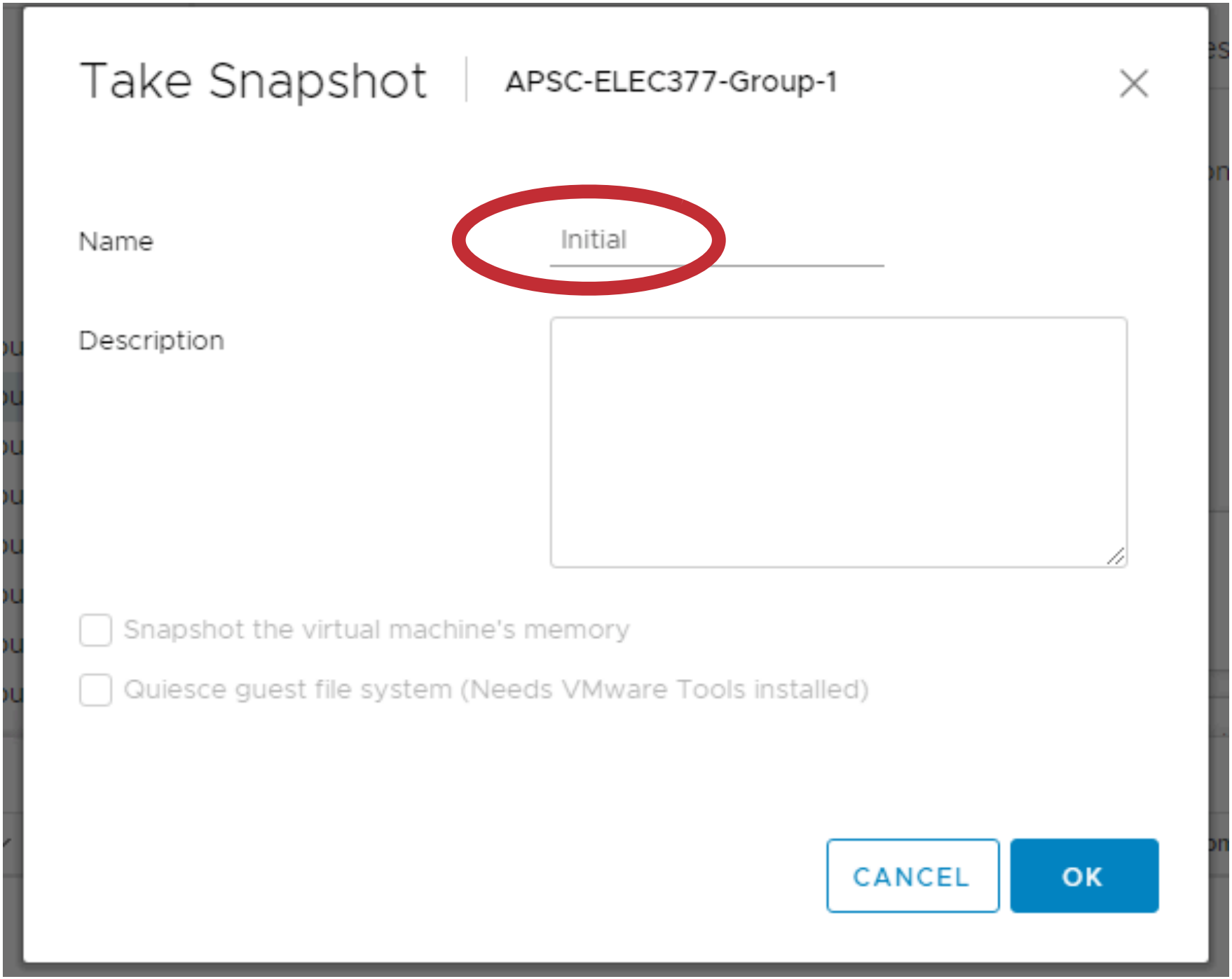
- Check that the network in the lower level says both appsci and VPN.
- In the chrome browser, go to https://faraday.appsci.queensu.ca/
- You will get a warning about an invalid certificate,
  - Click on advanced (lower left)
  - proceed to faraday.appsci.queensu.ca
- Select the html5 client.
- Log in with user name AD*yournetid*

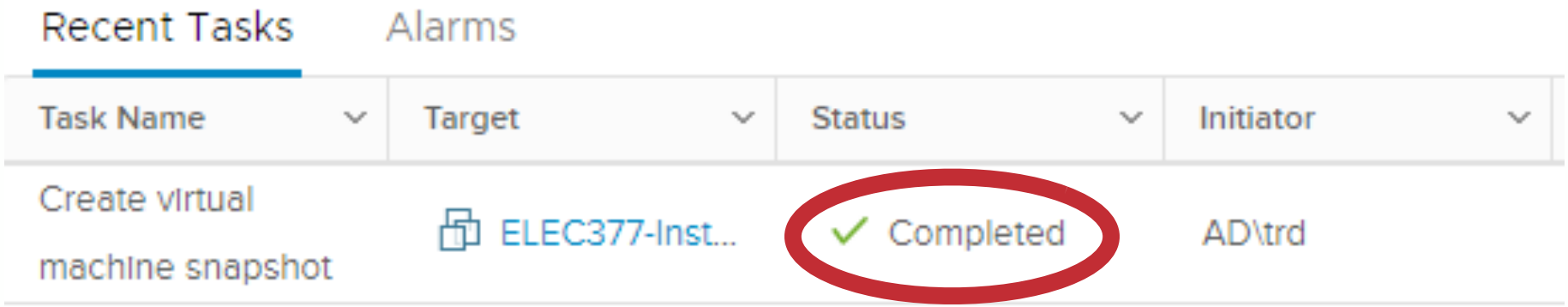


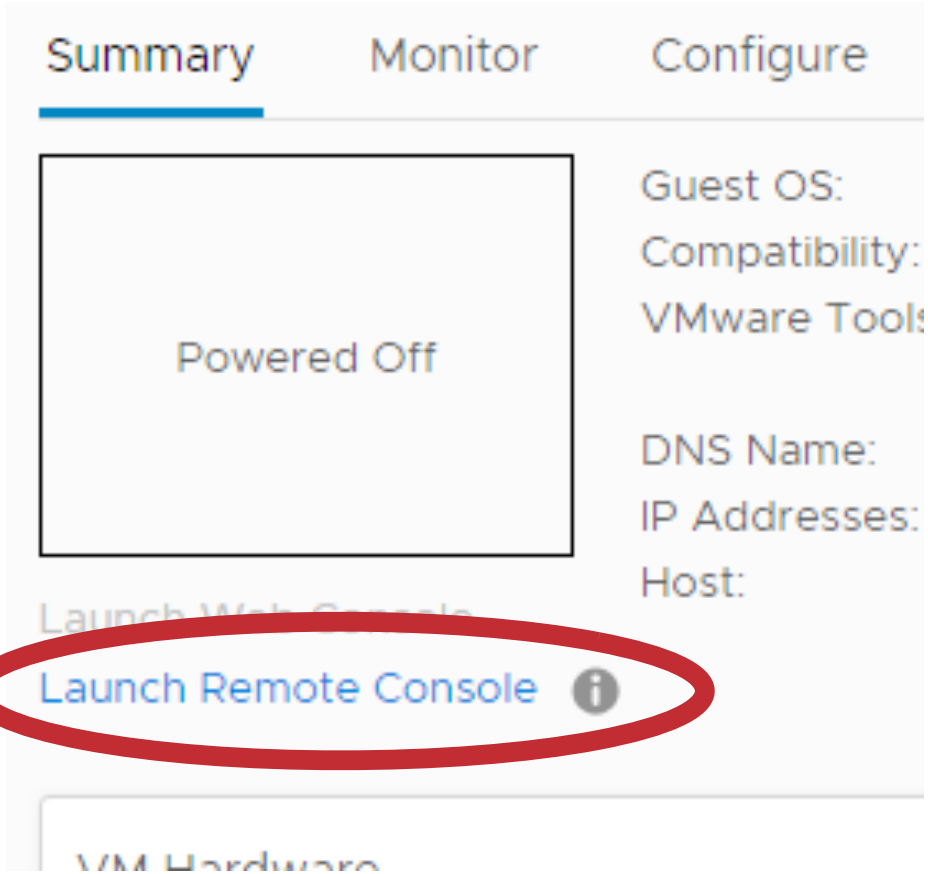
- You will only see one. DO NOT start it yet!!. If you see a green arrow on your icon, let me know right now.

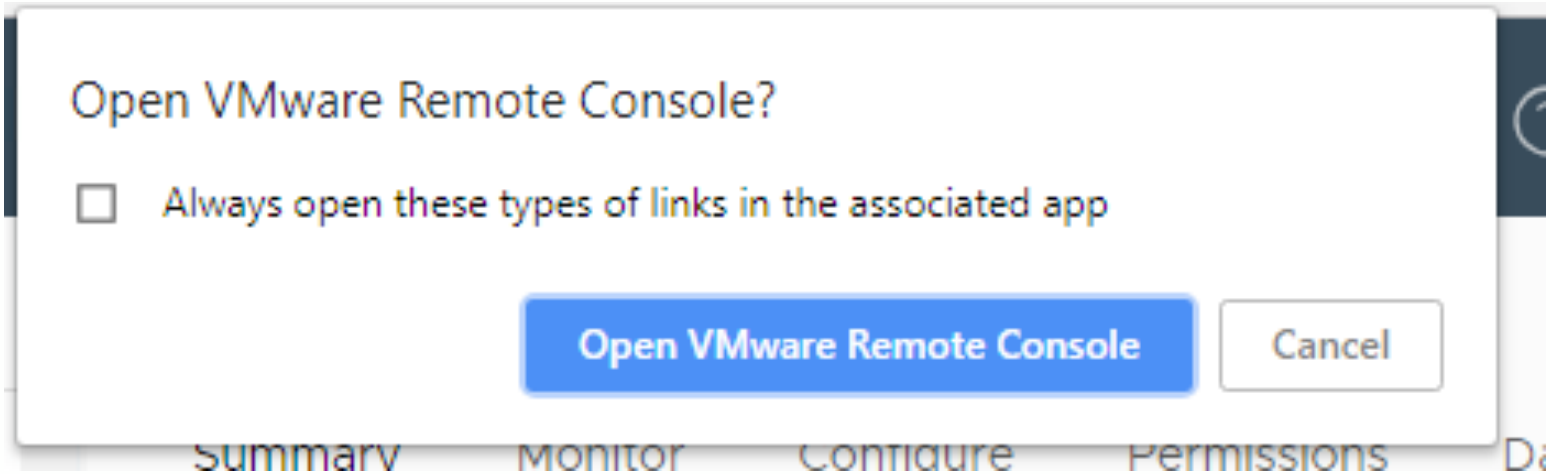

E377-Thu-1
ACTIONS
Summary
Monitor
Configure
Permissions
Datastores
Networks
Powered Off
Launch Web Console
Launch Remote Console
Guest OS:
Other 2.4.x Linux (32-bit)
Compatibility:
ESXi 6.5 and later (VM version 13)
VMware Tools:
Not running, not installed
More info
DNS Name:
IP Addresses:
Host:
CPU USAGE
0 Hz
MEMORY USAGE
0 B
STORAGE USAGE
4 GB

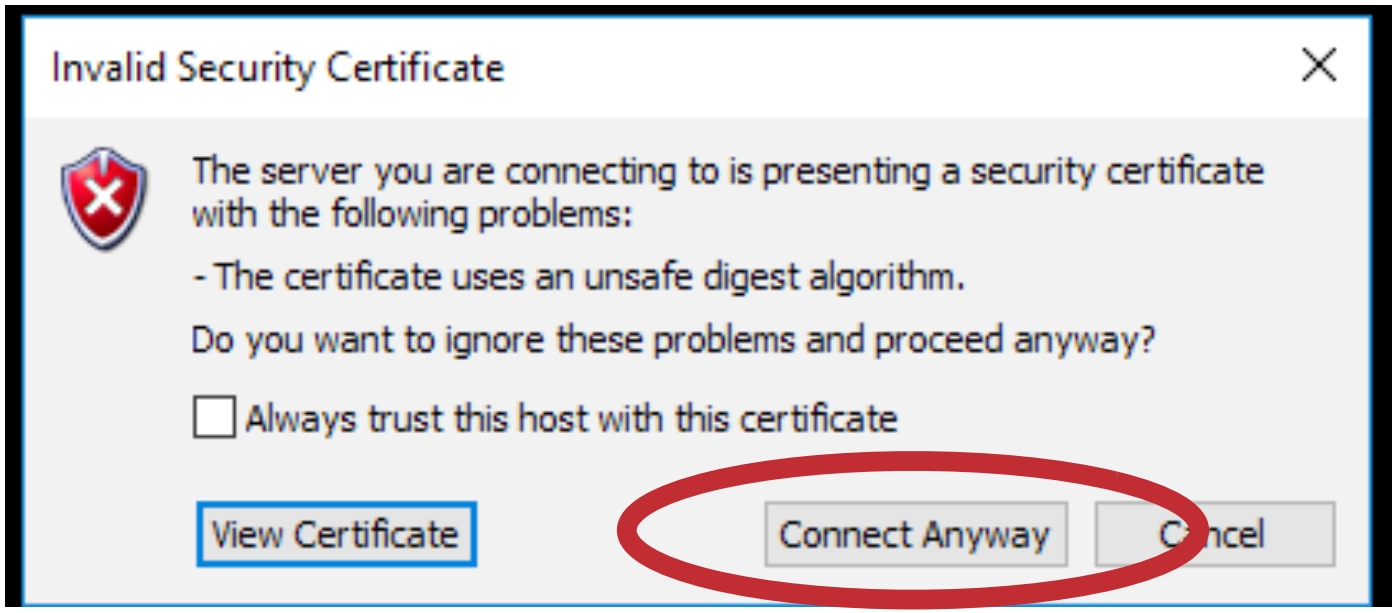

77-Group-1
ACTIONS
Configure
Permissions
Guest OS:
Other 2.4.x Linu
Take Snapshot...
Manage Snapshots
Revert to Latest Snapshot
Consolidate
Actions - APSC-ELEC377-Group-1
Power
Guest OS
Snapshots
Open Remote Console
Migrate...
Clone
Fault Tolerance

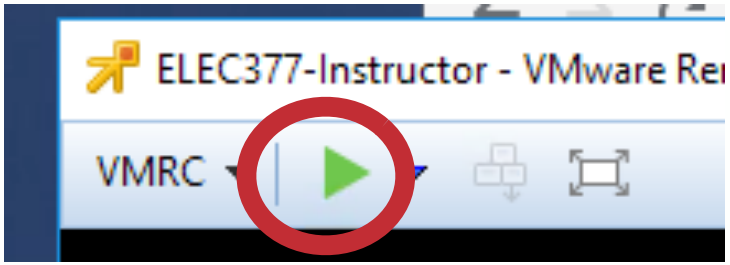

Take Snapshot
APSC-ELEC377-Group-1
Name
Initial
Description
Snapshot the virtual machine's memory
Quiesce guest file system (Needs VMware Tools installed)
CANCEL
OK


Recent Tasks
Alarms
Task Name
Target
Status
Initiator
Create virtual machine snapshot
ELEC377-Inst...
Completed
AD\trd


Summary
Monitor
Configure
Powered Off
Guest OS:
Compatibility:
VMware Tools
DNS Name:
IP Addresses:
Host:
Launch Remote Console

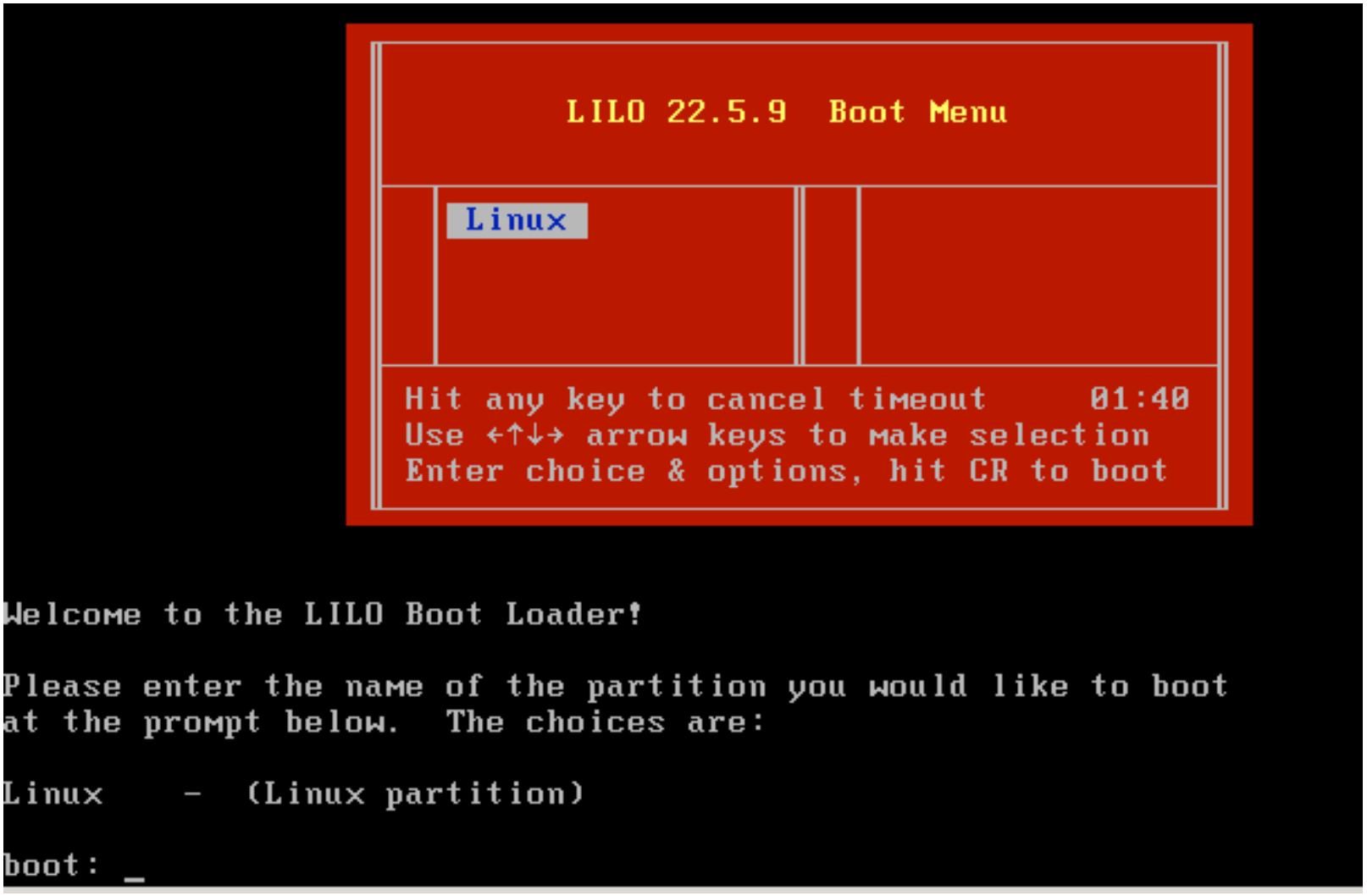

Open VMware Remote Console?
Always open these types of links in the associated app
Open VMware Remote Console
Cancel
Summary
Monitor
Configure
Permissions


Invalid Security Certificate
The server you are connecting to is presenting a security certificate with the following problems:
- The certificate uses an unsafe digest algorithm.
Do you want to ignore these problems and proceed anyway?
Always trust this host with this certificate
View Certificate
Connect Anyway
Cancel


ELEC377-Instructor - VMware Re
VMRC

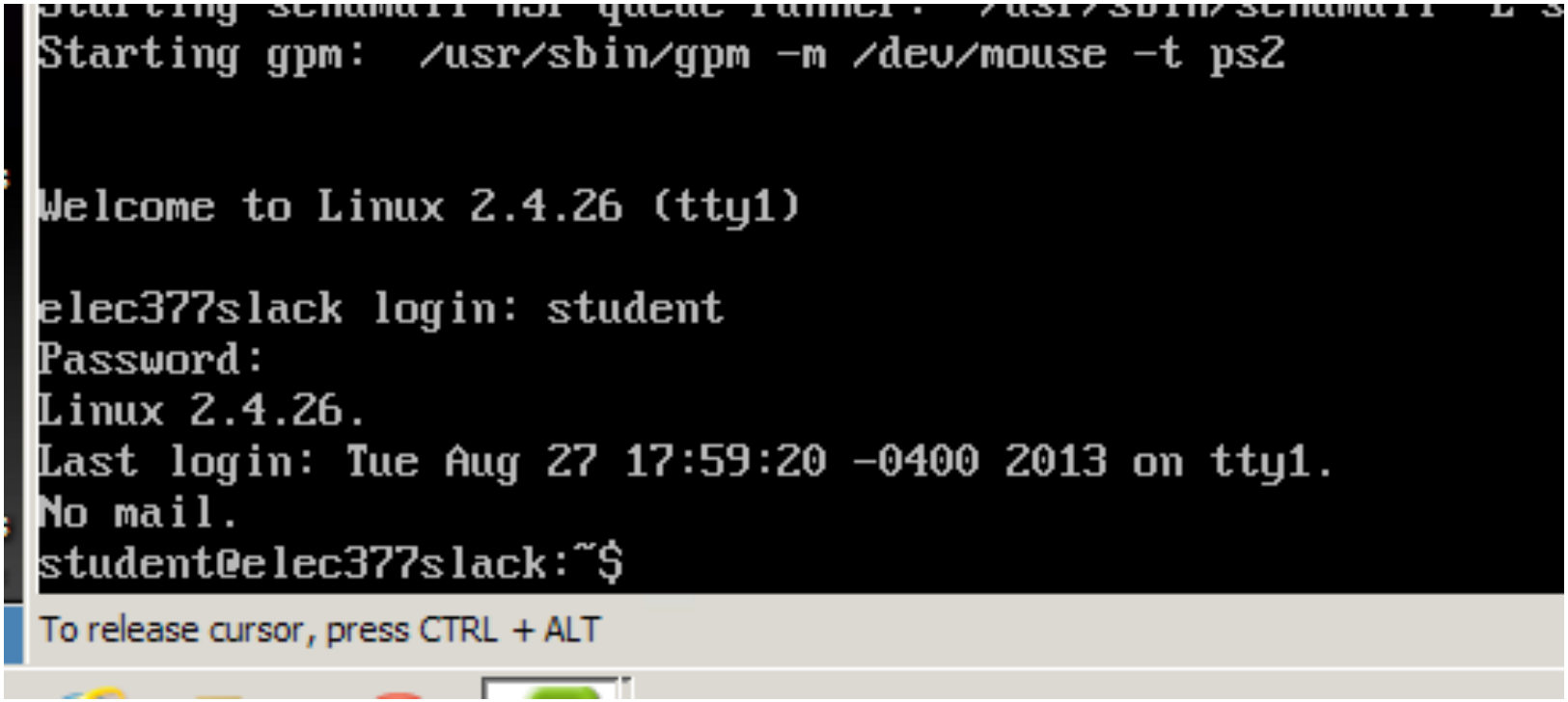
Labs - Starting the VM

- When you see the boot: prompt, press enter



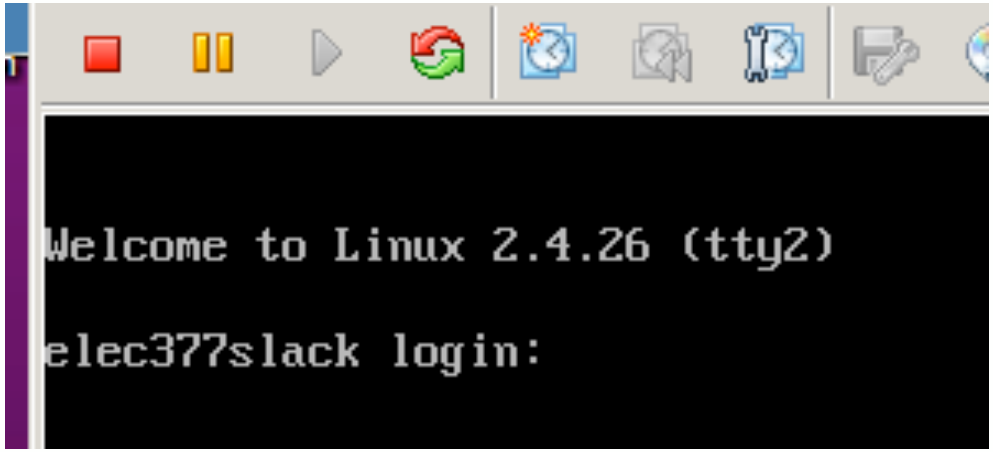
Labs - Logging into Linux

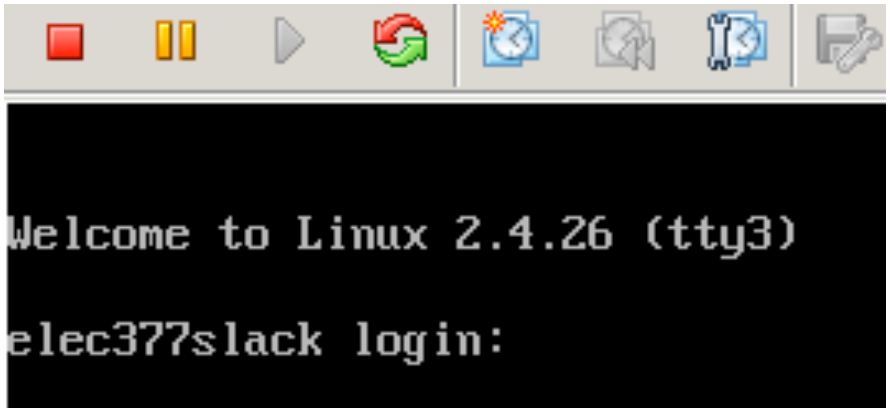
- username *student*, password *student*



Labs - Using Linux

- use alt-Function to switch between consoles





-- make sure you have clicked inside the window, as the function keys work on the outside window.

— there is a graphics environment, but not working on the VDI yet. Does work with a minor fix at home (slides at end of presentation)

Labs - Details

- Editors :
  vim, elvis (a vi clone), joe, emacs, jed (emacs clone)

- You can also edit in a windows editor and use git to transfer the code to linux.

Saving your Files

- Some of your labs will involve Linux Kernel Modules

– these operate in kernel mode, with no protection

– a bug in your code may crash your virtual machine and might corrupt your virtual hard drive

◊ Save your code on the git server before you run it

- git is a version control system. It keeps track of changes to each file.

Saving your Files

- An ssh key is needed if you don't want to type your password in every time you save changes to you files.

- in the student login window type:

```
ssh-keygen -t rsa -b 4096 -C "groupXXX@elec377.queensu.ca"
```

This is not a real email address, but we need an address for the git server.

It will ask for a file name, accept the default:

```
/home/student/.ssh/id_rsa
```

Saving your Files

- It will ask for a passphrase (empty for no passphrase)
  - choose a good password
  - do **NOT** use your netid password

- you now have a private/public key pair for access.
- We now have to get the keys out to register with the git server.
  - since this is an old version of linux, nobody wants to talk to it……

Saving your Files

- Some global settings to do:

```
git config --global user.email "groupXXX@elec377.queensu.ca"
git config --global user.name “groupXXX"
```

- If you run git on windows and intend to access your respository on Windows, use the following on Windows (not on linux)

```
git config --global core.autocrlf true
git config --global core.safecrlf true
```

Saving your Files

- check out the repository

git clone https://*netid*@code.engineering.queensu.ca/Elec377/E377-Wed-XXX

Git will ask for your netid password

- XXX is the group number of your VM
- Use E377-Thu-XXX if the Thursday lab.

Do *not* use

git clone https://code.engineering.queensu.ca/Elec377/E377-Wed-XXX

and let it ask for the user name and password. This will use a different authentication mechanism that doesn't work on the old version of linux.

Saving your Files

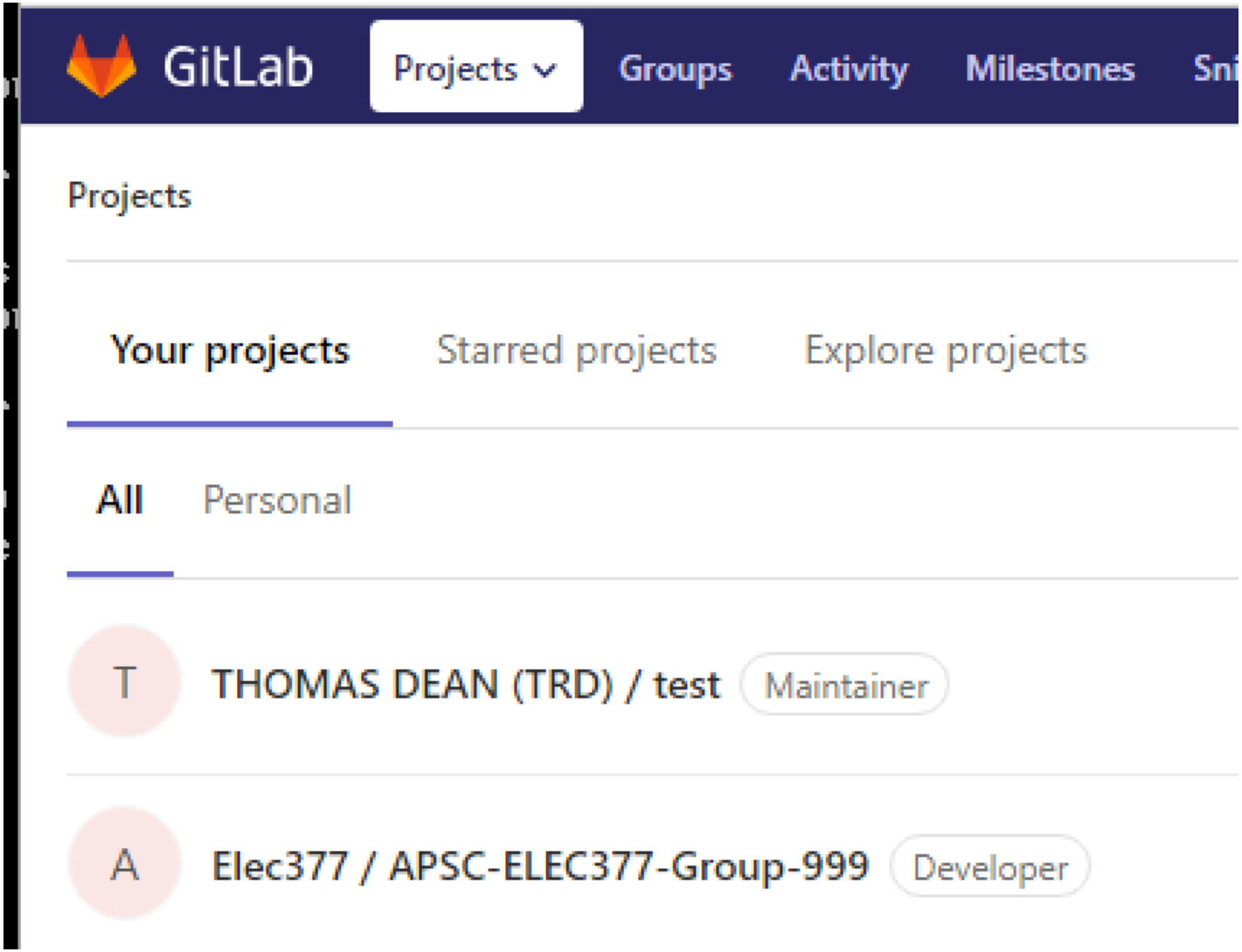
- You have a copy of your repository in the directory E377-Wed-XXX or E377-Thu-XXX

```
cd E377-Wed-XXX
mkdir ssh
cd
cd .ssh
cp id_rsa* ../E377-Wed-XXX/ssh/
cd ../E377-Wed-XXX/
git add .
git commit -m "keys"
git push
```

Saving your Files

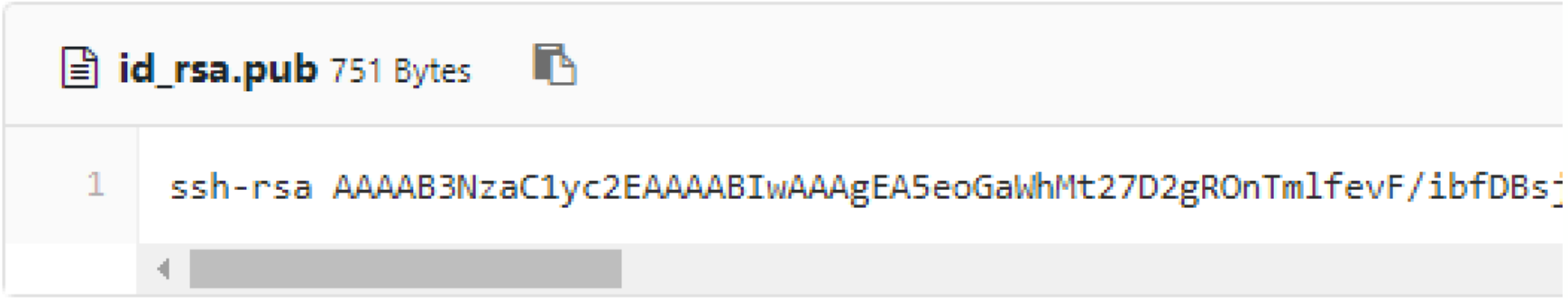
- The key files are now on the git server, need to add them to your account
- leave the vm using control-alt
- Open another tab on the chrome browser and go to https://code.engineering.queensu.ca

Saving your Files



Saving your Files

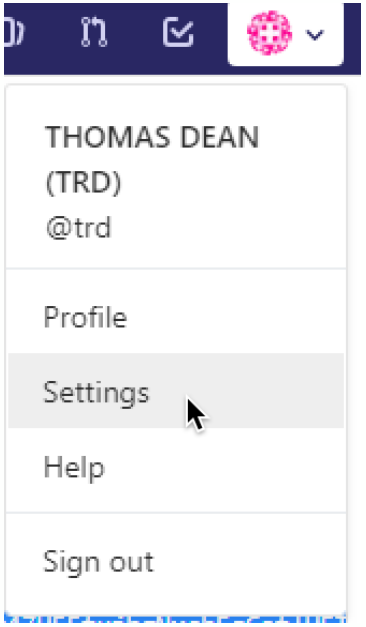
- Open your repository and scroll down. You should see a list of the items in your repository. There should be a *lab0* folder and a *ssh* folder
- Open The ssh folder by selecting *ssh*
- View the *id_rsa.pub* file by selecting *id_rsa.pub*



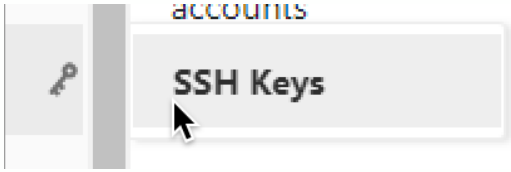
- triple click the contents to select the entire line and right click-> copy to copy the public key.

Saving your Files

- Go to the upper right corner to access settings

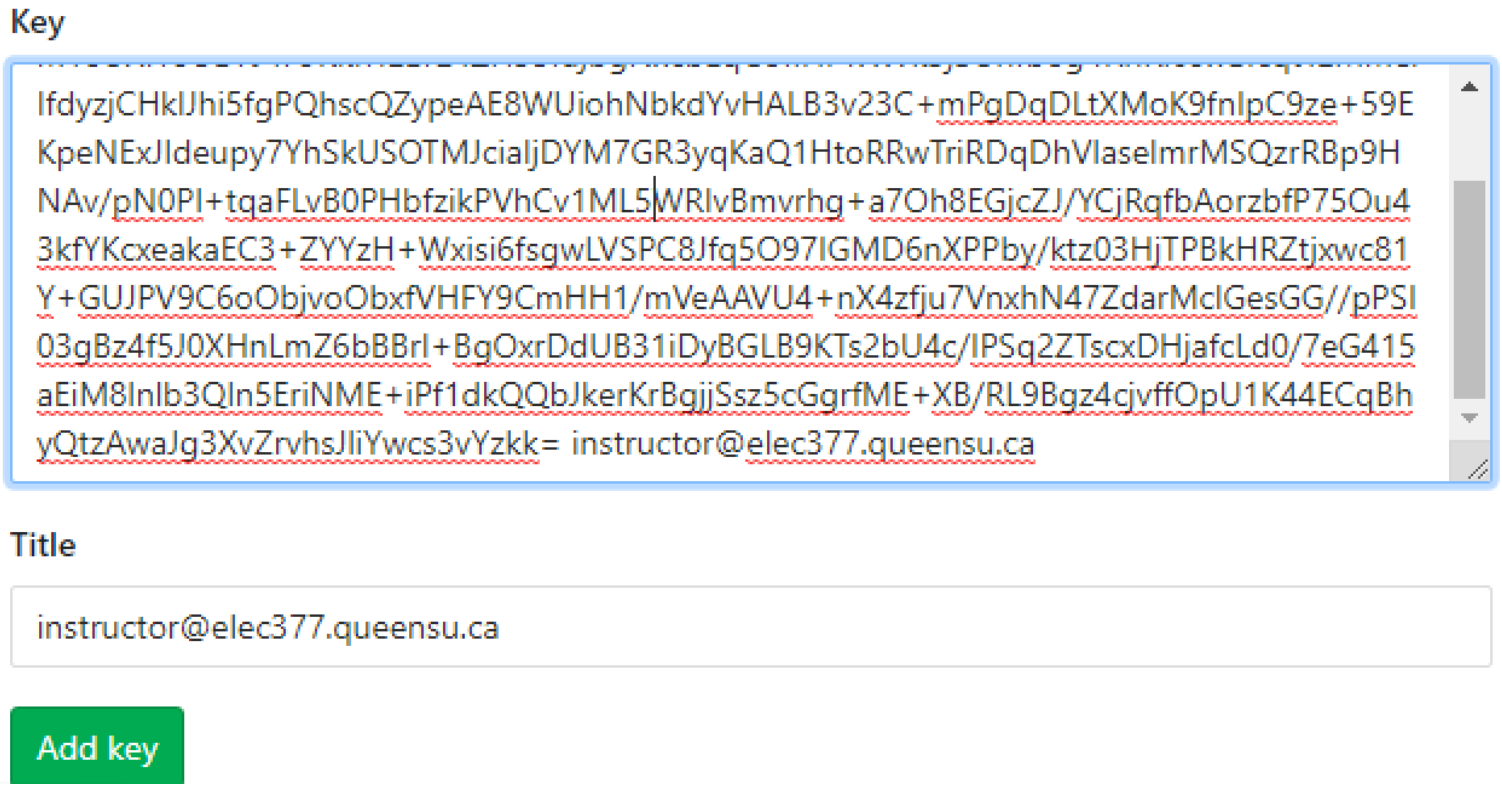


- Select ssh keys on the left side



Saving your Files

- past the key into the box. The fake email address will automatically be entered into the title. Select "Add key"



Saving your Files

- The public key is now associated with your netid
- Go back to the vm and remove the web based repository

cd
rm -rf E377-Wed-XXX
git clone git@code.engineering.queensu.ca:Elec377/E377-Wed-XXX.git
— this will ask for your pass phrase

Saving your Files

- To have your login remember your passphrase:

ssh-agent bash
source /etc/profile
ssh-add .ssh/id_rsa

— this will ask for your pass phrase. It will then be remembered until you exit that console. You will also have to exit twice (subshell which will be explained later in the class)
— your passphrase will not be remembered in other consoles

Saving your Files

- your private key is also in the git repository and can be downloaded by the web interface. This is why you make sure that it had a good passphrase!!

Managing files

- telling git which files have changed
  - If a new file is created or changed, it must be *added* before git will add it to the change list.
  - allows you to decide which changes to send to the server.

  vi foo.c
  --- change some text
  git add foo.c

- git add *directory*
  - directory must be in an git directory.
  - adds all contents of the directory
  - "git add ." easy way to add all your changes.

Managing files

- The files are not on the server yet
- Next have to tell git that the added files are to be committed together
  - git commit -m "message"
- The files are not on the server yet
  - git push
  - the files are pushed up to the server.

Managing files

- getting the latest version from the server

  git pull

  - must be run somewhere inside the directory from the first checkout
  - pulls all changes down from the server

File Management

- You can also check out the repository at home
  - git clients for windows macintosh

- TA checks in skeleton version of the assignment
- You use *git pull* to get the code
- do the assignment and execute the following before running:

  git add . (or *git add files*)
  git commit -m "a message"
  git push
- at the end of the make sure all your changes along with your test data and output is pushed to the repository

File Management

- You can also check out the repository at home
  - git clients for windows macintosh
- at home use *git pull* to get latest version from lab
- write documentation (new files)
- add them to be managed (*git add lab1doc.txt*)
- commit them (*git commit -m "message"*)
- push the changes (*git push*)

- lab windows VDI will have git, so can check out to windows too.

- There is a explanation of common git error on OnQ

Lab Pairs

- Go to OnQ now
- Both members of each pair join the group with the same name as the VM you are currently using.

Look at Lab0 now

- Already checked out
- cd E377-Wed-XXX/lab0

- vi lab0mod.c

- three parts
  - my_read (main part of kernel)
  - init_module -- used to initialize data structures
  - cleanup_module -- used to remove pointers

Compiling Modules

- Sometimes a Makefile will be provided
  - makefiles contain dependencies. Example:

```
all: lab0mod.o lab0user

lab0mod.o: lab0mod.c
        cc -c -Wall -DMODULE -D__KERNEL__ lab0mod.c

lab0user: lab0user.c
        cc -o lab0user lab0user.c
```

Use the command "make" to build

Inserting a Module

- Login as root

- use alt-fn to change consoles
  - cd /home/student/E377-Wed-XXX/lab0
- use the following command to load the module
  insmod lab0mod.o
- you will get the message:

Warning: loading lab0mod.o will taint the kernel:
no license
See http://www.tux.org/lkml/#export-tainted for information about tainted modules

- after testing, unload the module with:
  rmmod lab0mod (note: no.o)

User Programs

- Some labs will also have a user level program
  - kernel: not all libraries are available!!
  - lab0mod: ctime() function formats a date
    - module returns a number
  - user level program that formats for ease of user understanding

% lab0user
The system was started September 17th at 3:40.

Testing

- Some labs are interactive
  - ◊ Interact with your kernel module
  - – use *script* command

% script lab1_out1.txt
Script started, output file is lab1_out1.txt
% run your tests here
...
% exit
Script done, output file is lab1_out1.txt

- Some labs just have output

% cat /proc/lab1 > lab1_out1.txt

Testing

- Testing is your proof that the program works
  - ◊ TA might not be running your code
    – industry standard in many domains, give customer test suites to provide confidence it works.
- Show input and output
  – Sometimes input is a kernel data structure
  – system commands such as ps, vmstat, iostat, and files in /proc give you some idea of what the state of the kernel is
  – think about this *before* you come to the lab
  – man pages available on the internet

Documentation

- Documentation are to be presented in text files or pdf. Do *not* commit word documents to the repository.

Labs - Connecting to Linux VM

- The square icon is used to enter and exit full screen
- Type the enter key to turn off screen saver
- Login as *root*, with the password *root*
- Use the command 'shutdown -h now'
- Wait until it says power down
- use ctrl-alt to get your windows pointer back
- select shut down guest from the pull down menu beside the pause icon
- close the connection window.

Labs - Working from home

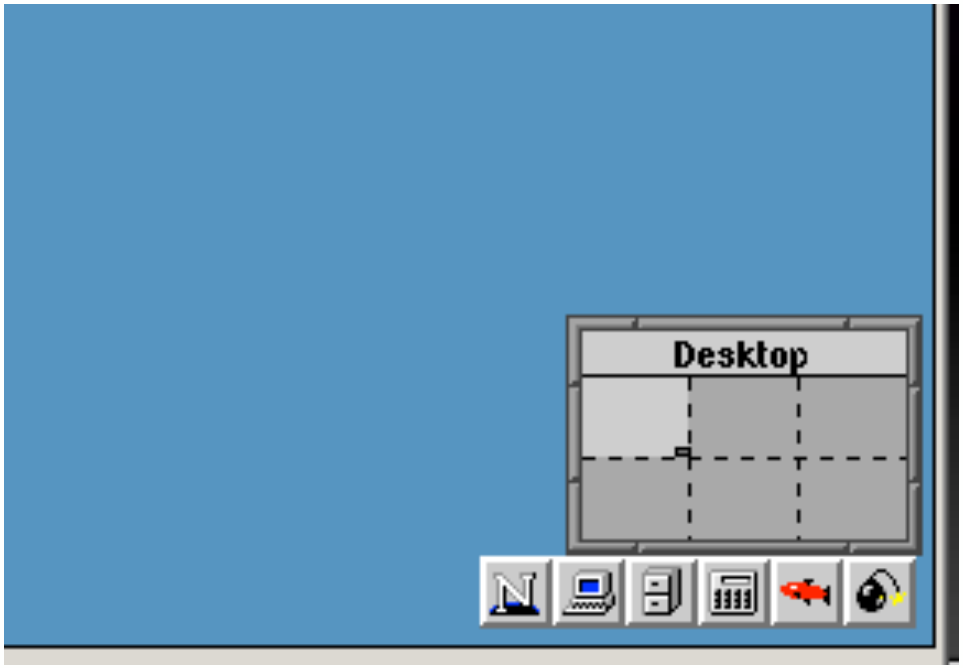
- There will be a copy of the VM that you can use at home.

It works with vmware fusion pro (Mac) vmware workstation pro and with VirtualBox
vmware products are commercial with a cost
VirtualBox is free.

- there is an issue with the mouse tracking that is being fixed… But the mouse tracking works fine for home use. So don't use X in the lab (yet).
- the command *startx* from the student login starts the X environment.

Labs - X Windows

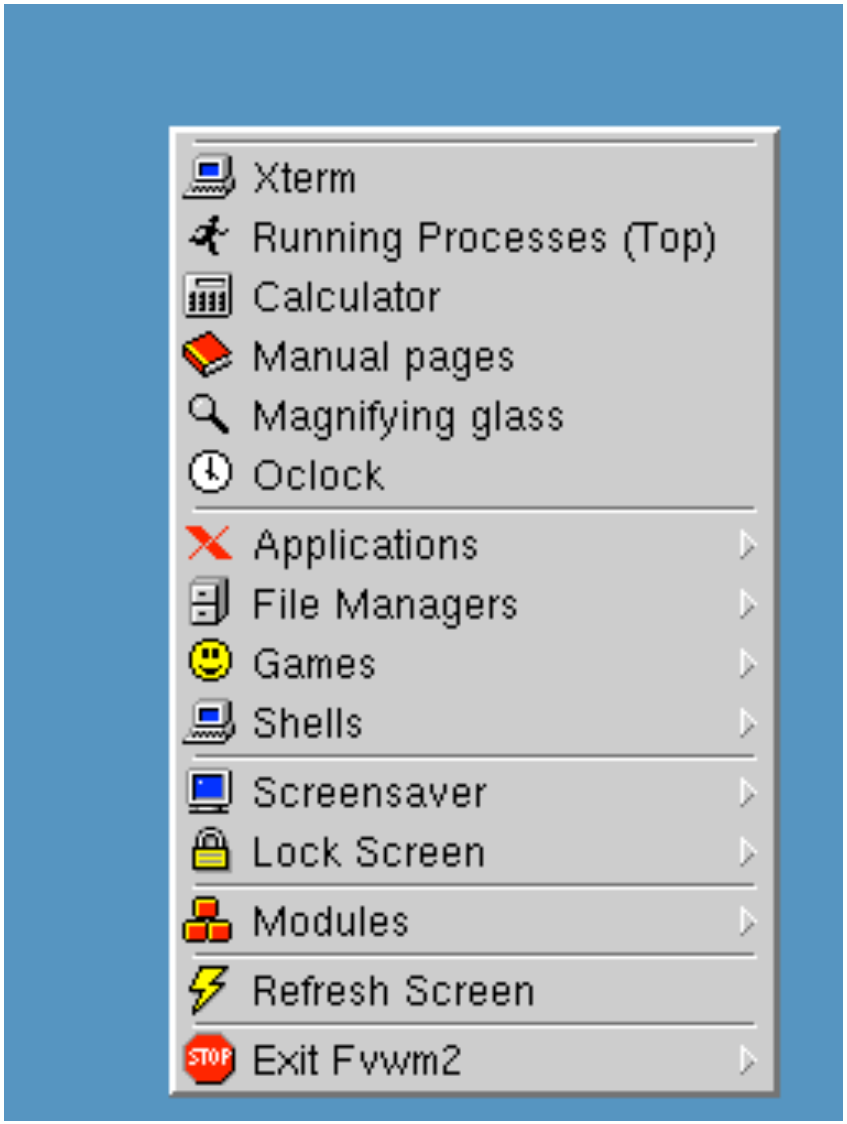
- The graphics environment has multiple desktops, and some shortcut icons



Labs - X Windows

- Left click on the background brings up the menu.

- Use the xterm window to start a shell.



Labs - Working from home

- you don't have to generate another ssh key.
- checkout using https

mkdir .ssh to create the .ssh directory

cd E377-group-XXX/ssh

cp id_rsa* ~/.ssh/

- The keys will be on the home version of linux.
- You can also download the keys through the web interface and add to windows or mac clients.