Linux Server for Geographic Data

By Anne Hagerman – 01 Sept 2016 to xx Dec 2016

continued by Andrew Roberts - Nov 2016 - 28 April 2017

INTRODUCTION

In order to create a flexible learning/development environment for the co-op Web Map Application project, it was decided to use a virtual machine running Ubuntu 16.04. This virtual machine was used to create a local Linux Apache server that connects to the shared Linux servers Ulysses and Daedalus, for play in the sandbox and as a development environment for the application prototype.

1 VIRTUAL MACHINE

1.1 Install Virtual Box

Oracle VirtualBox – www.virtualbox.org

VirtualBox 5.1.6 for Windows hosts

**install and record steps on personal PC

1.2 Create Virutal Machine

1.2.1 Step 1: Create New Virtual Machine

VirtualBox → Create New Virtual Machine

Name: Ubuntu
The name / full path to the virtual machine folder you are about to create
C:\Users\Hagermanan\VirtualBox VMs
VM's can be put anywhere so can leave on C: drive

Type: Linux
The type of OS

Version: Ubuntu (64-bit)
The version / distribution of the OS

Memory Size: 8192 MB
The amount of memory that will be dedicated to the VM when it is running as a maximum allocation.
8 GB
This / Work PC has 32 GB of RAM

Hard Disk: Option [Create a virtual hard disk now]
Can choose to add a virtual hard disk to the new machine now or later. Either can create a new hard disk file, choose and existing one or if you need a more complex storage set-up, do not add one now and make changes to the machine settings once it is created.

Create1.2.2 Step 2: Create Virtual Hard Disk
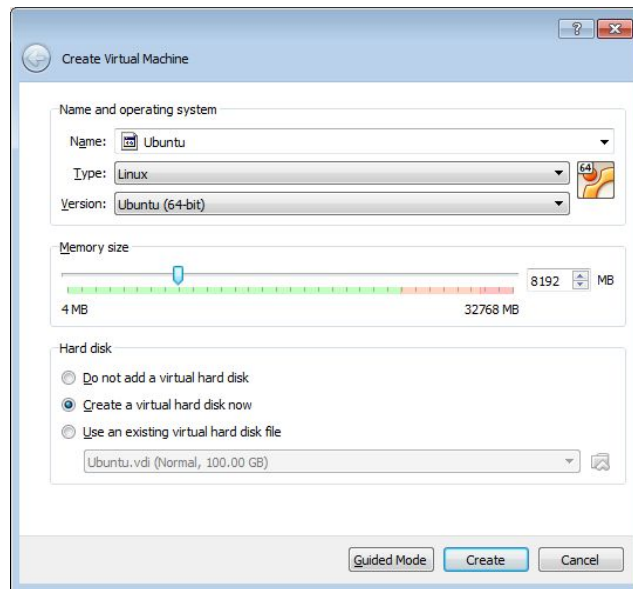
This window auto-appears when you select "Create a virtual hard disk now" above.

File Location: K:\VirtualBox_VMs\Ubuntu2.vdi
> *The file path & name of the virtual hard disk that will be created – ext .vdi*
> Auto puts the same name as the VM and in the same directory.
> We put it where we wanted the File Size to be taken from (not C: drive)

File Size: 100 GB
> *The size allocated to the VM.*

Storage on physical hard disk: Option [Dynamically allocated]
> *How the virtual hard disk should be created: grow as it is used (dynamic) or should it be created at its maximum size [File Size].*
> *Dynamic means will only take the space from the physical hard disk as needed up to the maximum indicated in the [File Size] but it will not shrink again automatically when space is used.*
> *Fixed means the full [File Size] will be allocated at the time of creation.*

Hard disk file type: Option [VDI (VirtualBox Disk Image)]
> *The type of file that should be used for the new virtual hard disk. How will the OS be installed.*
> Ubuntu was installed from a ISO image file.

Create
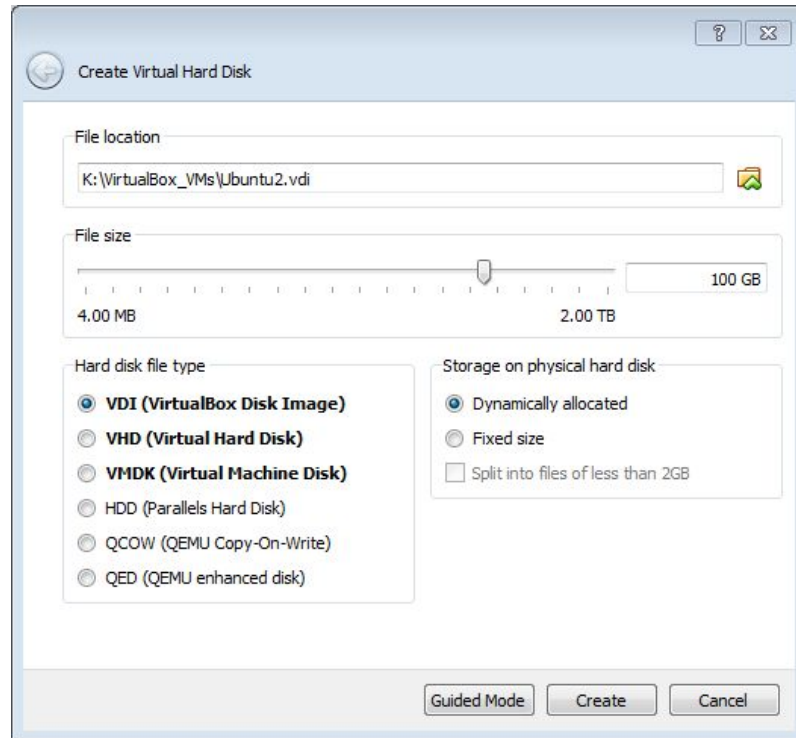
2 UBUNTU

You can download packages for the desired Linux distribution on www.virtualbox.org .

2.1 Select OS

Start the newly created VM from Oracle VM VirtualBox Manager

Select start-up disk: O:\Research\CanSISData\Open Source Tools\ubuntu-16.01.1-desktop-amd64.iso
        64 bit: ubuntu-16.01.1-desktop-amd64.iso
        32 bit: ubuntu-16.01.1-desktop-i386.iso

Start
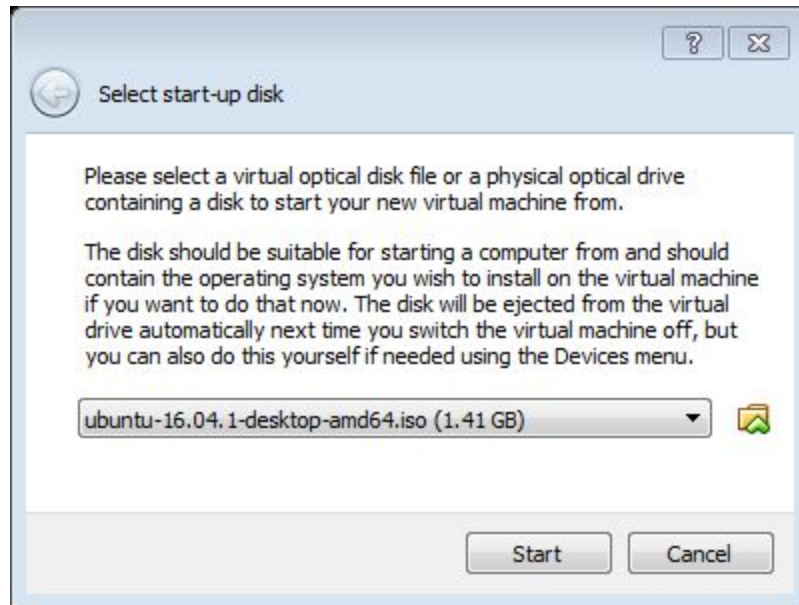
2.2 Install Ubuntu (as superuser)

Language: English

Install Ubuntu
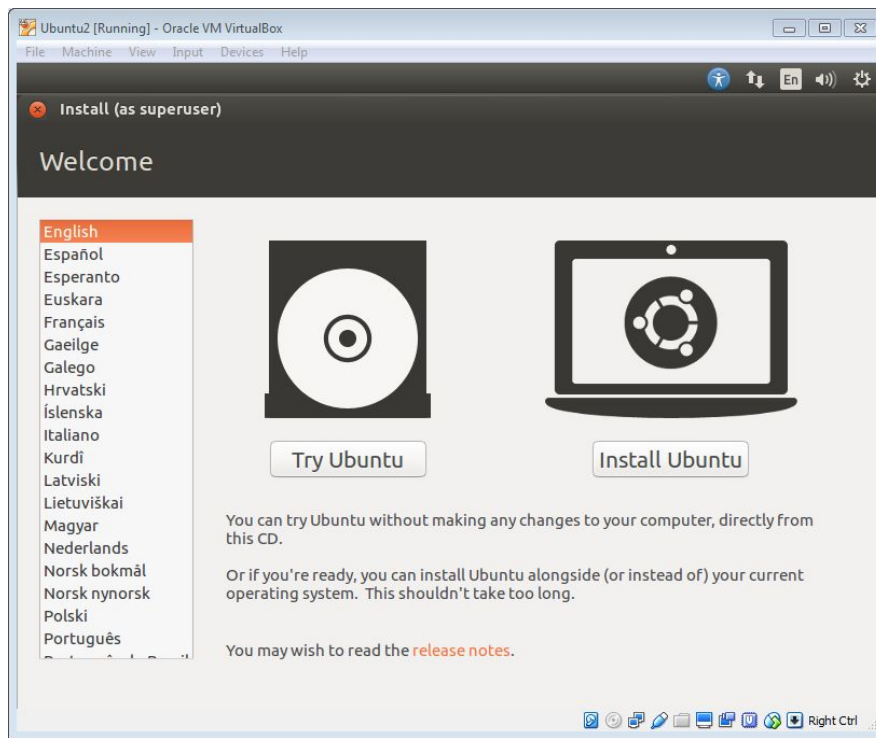
Download updates while installing Ubuntu: True
> If False, would need to execute [$ sudo apt-get update] and [$sudo apt-get upgrade] commands
> in Ubuntu to make up to date.

Install third-party software for graphics and Wi-Fi hardware, Flash, MP3 and other media: False

Installation type: Option [Erase disk and install Ubuntu]

Install Now
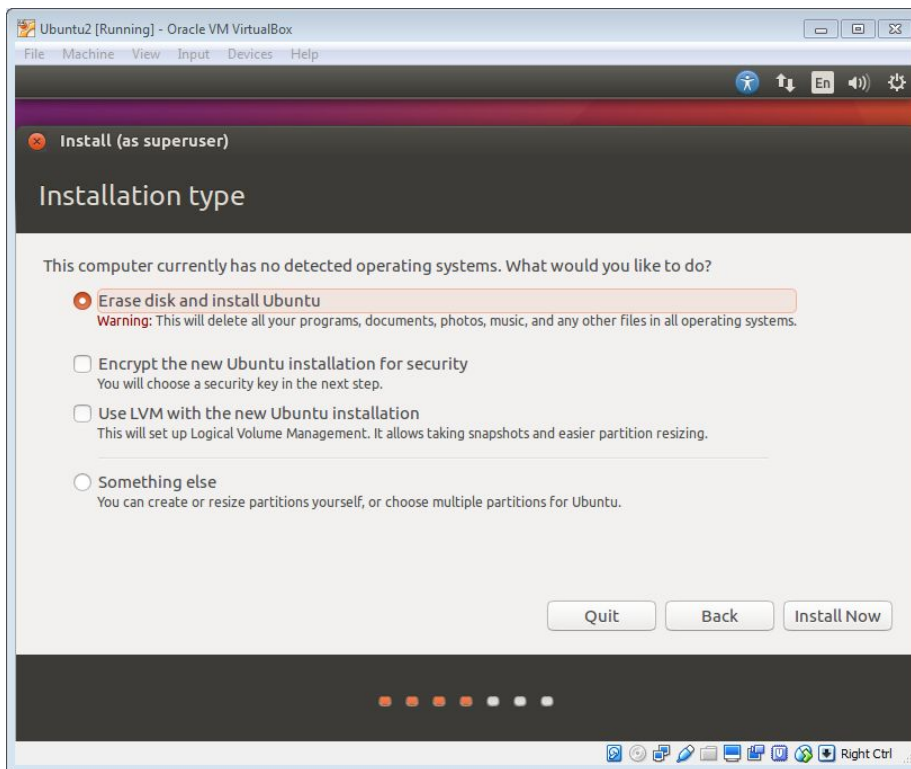
Write changes to disks?: see screenshot below

Continue
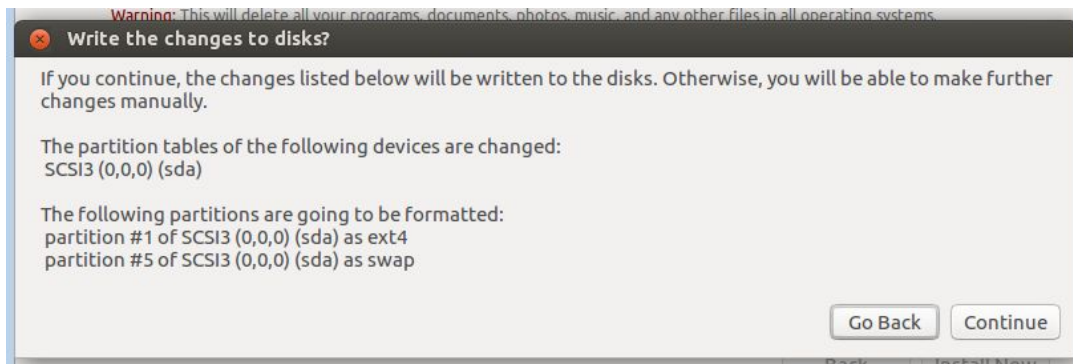
This is the default partitioning. Can do manual partitioning if choose [Something else] above.

Where are you?: Ottawa (Ontario, Canada)

Continue

Keyboard layout: English (US) – English (US)

Continue

Who are you?

Your name: anne hagerman
This is what shows up when you log in… not the username.

Your computer's name: anne-Ubuntu
Was changed after install to match Windows PC name – see section below.

Username: hagermanan
Made is the same as Windows login.

Password: user specified

Require my password to log in: True

Continue
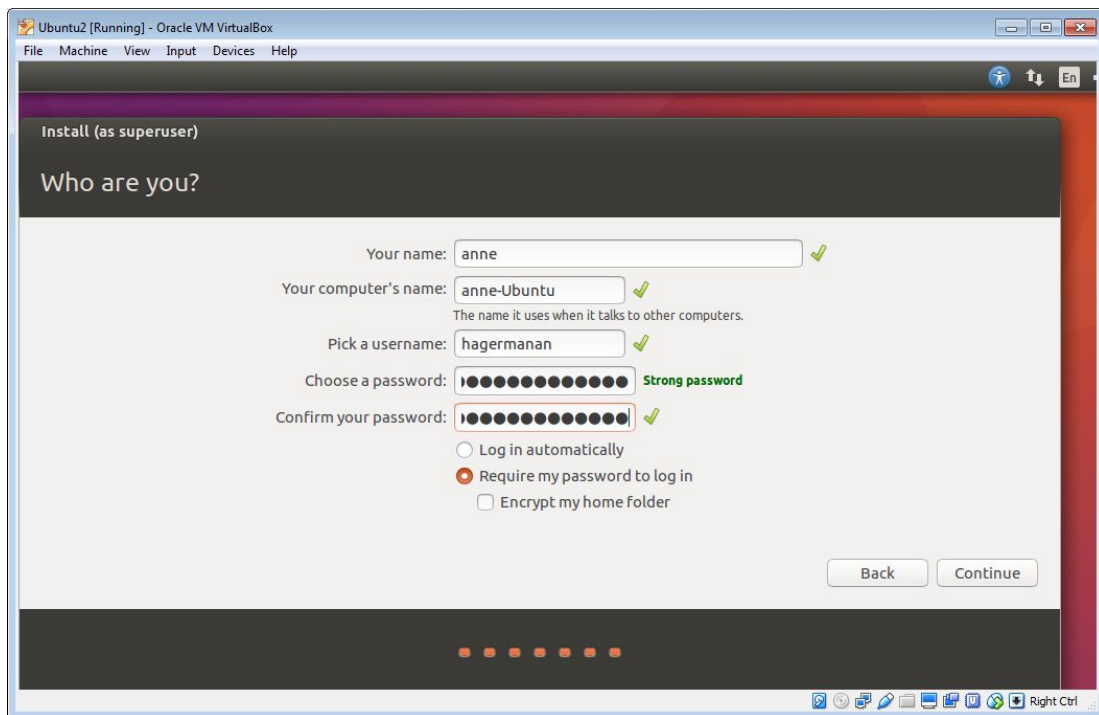
Installation Complete

Restart Now
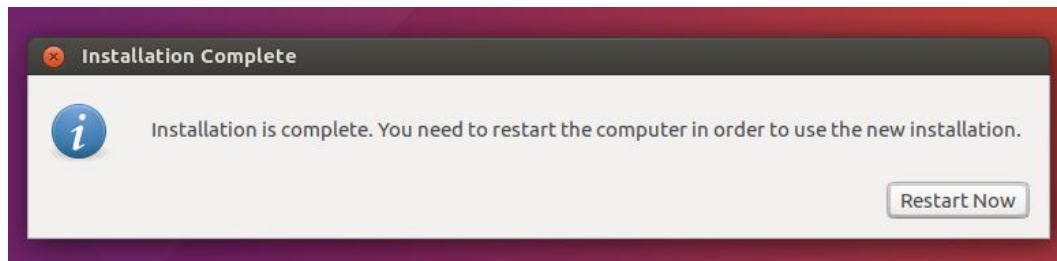
Remove Installation Medium
Since an .iso file was used, nothing required.

Enter
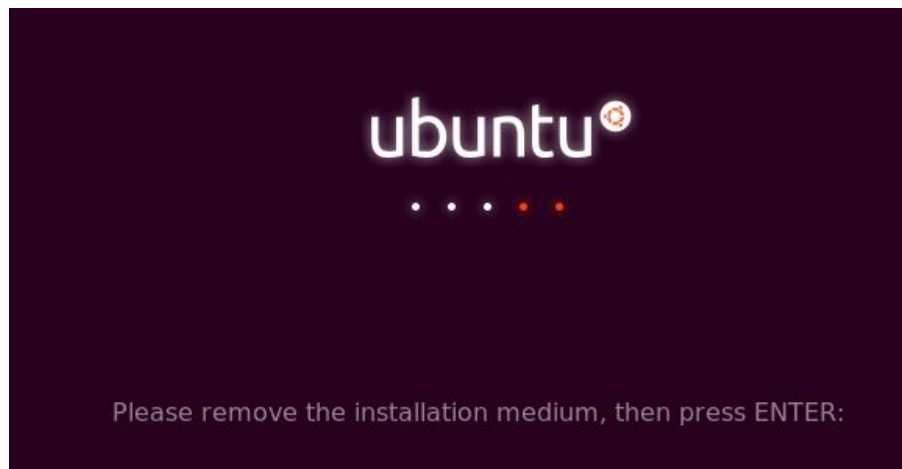
2.3 Useful Links:
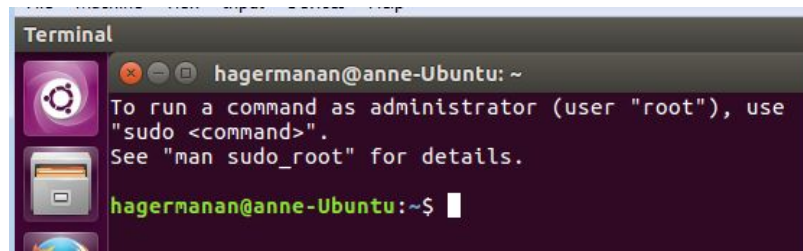
An A-Z Index of the Bash command line

Tips For Linux - Wildcards, Quotes, Back Quotes and Apostrophes in shell commands

Creating Alias Commands in .bashrc

Understanding and Changing File Permissions
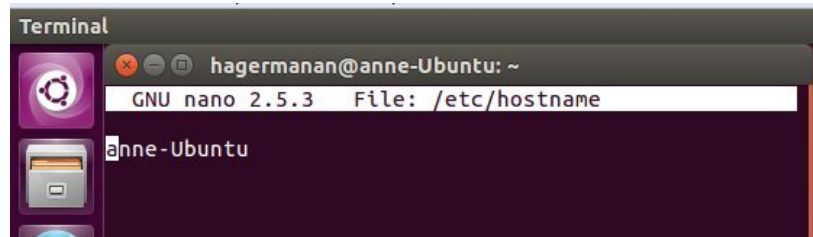
3 CHANGE COMPUTER NAME (HOSTNAME)

3.1 Step1: Edit name in 2 files: /etc/hostname and /etc/hosts.
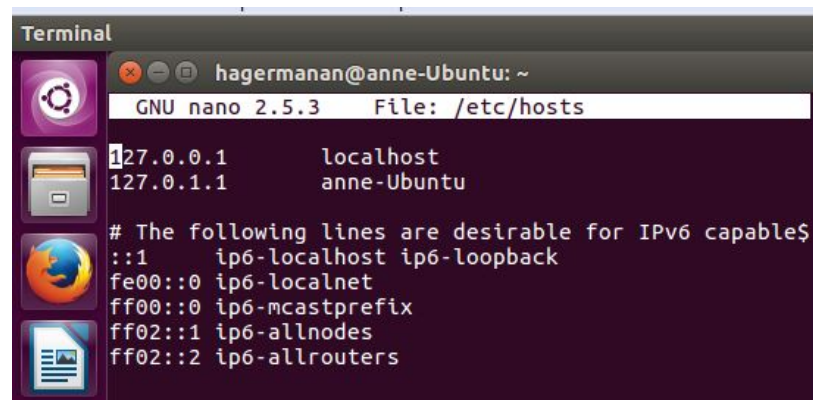
From google search [change Ubuntu computer name] – most results say this.
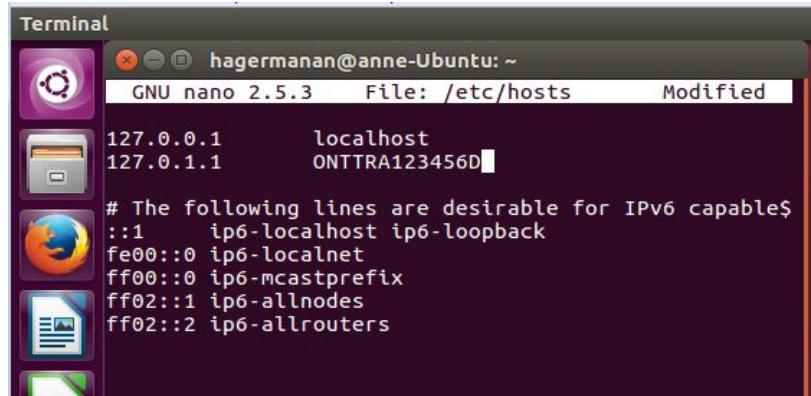
```
Terminal
  ⊗ ⊖ ⊡   hagermanan@anne-Ubuntu: ~
  GNU nano 2.5.3    File: /etc/hosts        Modified

  127.0.0.1         localhost
  127.0.1.1         ONTTRA123456D

  # The following lines are desirable for IPv6 capable$
  ::1     ip6-localhost ip6-loopback
  fe00::0 ip6-localnet
  ff00::0 ip6-mcastprefix
  ff02::1 ip6-allnodes
  ff02::2 ip6-allrouters
```

3.2 Step2: Restart Computer

$ sudo reboot


```
Terminal
  ⊗ ⊖ ⊡   hagermanan@ONOTTRA123456D: ~
  hagermanan@ONOTTRA123456D:~$
```

Example computer name is above. Actual computer name chosen for the active VM was to match the Windows PC:

ONOTTRA673312D

4 SSH WITH SSH KEY CONNECTION TO LINUX SERVER

Must already have a username and password on the computer / server trying to connect to.

4.1 SSH Connection

Connect to Server
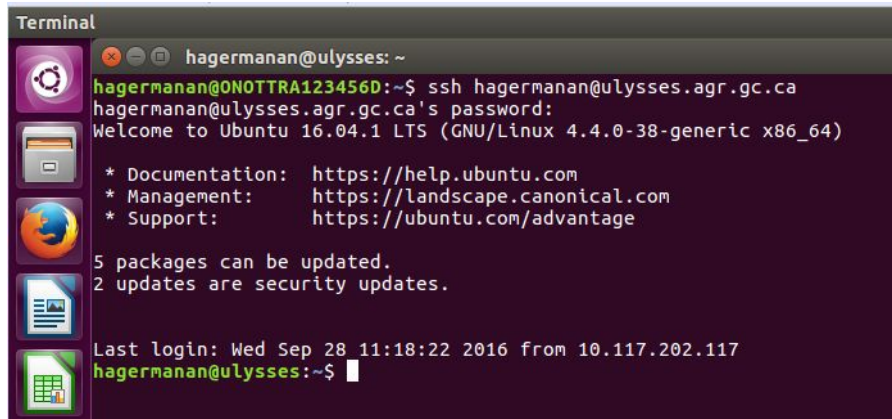
$ ssh hagermanan@ulysses.agr.gc.ca
        + password on connecting computer

$ ssh [username]@[computer IP / domain]

Or
$ ssh [computer IP / domain]
If username on host computer and connecting computer are =
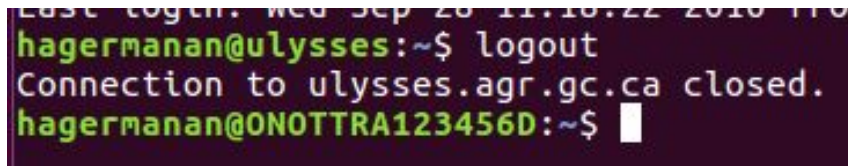
From IP is the IP of the host computer that the VM is running on

Break Connection / Exit

`$ logout`

4.2 SSH Keys

More secure than password to access the external computer / server. Creates a key pair algorithm with the IP of the host computer and the IP of the external computer / server. Then, anytime you connect from that host computer, it automatically checks the key pair and logs on.

Linux Tutorial #15: SSH Key Authentication
https://youtu.be/5Fcf-8LPvws?list=PL6gx4Cwl9DGCkg2uj3PxUWhMDuTw3VKjM

4.2.1 Step1: Generate public / private rsa key pair
Private key stays on host computer and identifies it
Public key is uploaded to server

NOT connected to server yet

`$ ssh-keygen –t rsa`

Enter → keep default
Enter → no passphrase
Enter → no passphrase

4.2.2 Step 2: Copy public key and put it on the server

$ ssh-copy-id hagermanan@ulysses.agr.gc.ca
+ password on server

$ ssh-copy-id [username]@[server / computer IP / domain]

4.2.3 Step 3: Check Connection

$ ssh hagermana@ulysses.agr.gc.ca

4.3 Allow X Windows requests to run on your computer from the server

4.3.1 Step 1: Configuration

Update *ssh_config* file on host compute to allow X Windows on the host computer:

$ sudo nano /etc/ssh/ssh_config

Under *Host ***, uncomment the 1st 2 lines and change them from *no* to *yes*:

```
Host *
#    ForwardAgent no
#    ForwardX11 no
#    ForwardX11Trusted yes
#    RhostsRSAAuthentication no
#    RSAAuthentication yes
#    PasswordAuthentication yes
#    HostbasedAuthentication no
#    GSSAPIAuthentication no
#    GSSAPIDelegateCredentials no
#    GSSAPIKeyExchange no
```

```
Host *
ForwardAgent yes
ForwardX11 yes
#    ForwardX11Trusted yes
#    RhostsRSAAuthentication no
#    RSAAuthentication yes
```

<u>Note:</u> line 3 *ForwardX11Trusted yes* also has to do with allowing X Windows, but the default is already *yes* so don't need to uncomment / change.

4.3.2 Step 2: Check configuration

Connect to ulysses and run xeyes:

$ ssh ulysses.agr.gc.ca

$ xeyes

The xeyes X Window program is now running from ulysses onto the host computer – the eyes follow you mouse movements.

To close:

Ctrl + C

4.4 Useful Links:

How To Set Up SSH Keys | DigitalOcean

How To Use SFTP to Securely Transfer Files with a Remote Server | DigitalOcean

5 WEB SERVER SOFTWARE: APACHE

5.1 Install Apache
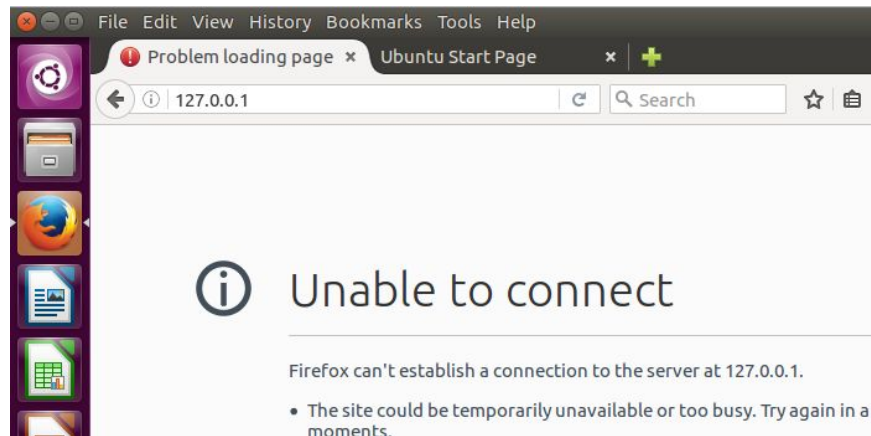
$ sudo apt-get install apache2

5.2 Make Ubuntu VM visible to outside users (Port Forwarding)

How to Forward Ports to a Virtual Maching and Use It as a Server:
http://www.howtogeek.com/122641/how-to-forward-ports-to-a-virtual-machine-and-use-it-as-a-server/

5.2.1 Step1: Network Settings

VM Settings → Network Tab → expand Advanced → click Port Forwarding
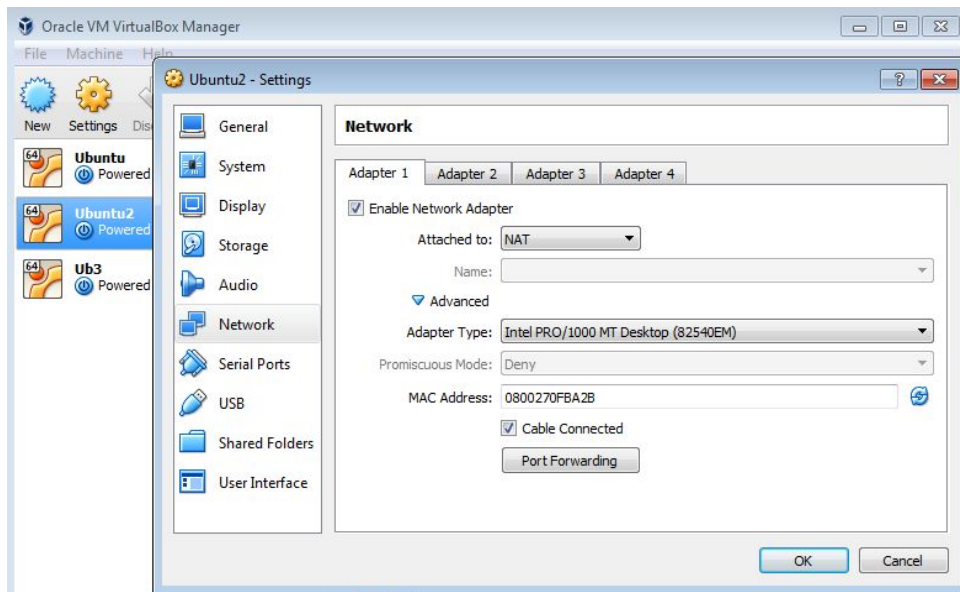
5.2.2 Step 2: Forward Port

Add new rule (green + sign)

Name: Ubuntu Apache

Host Port: 80

Guest Port: 80
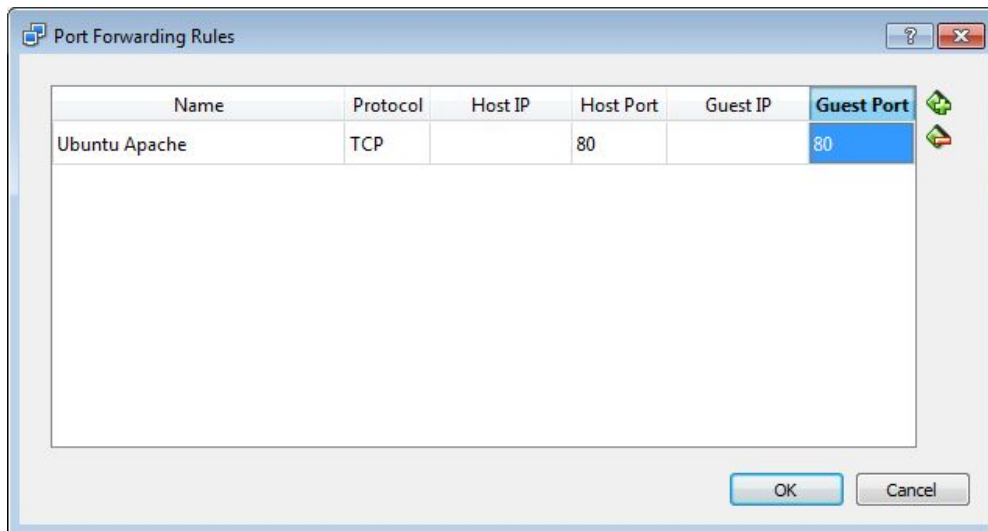
This makes it so when port 80 of the host computer is requested, the Apache server on the virtual machine will be accessed.

Port 80 is called for the Windows host IP and the Ubuntu directory is accessed.
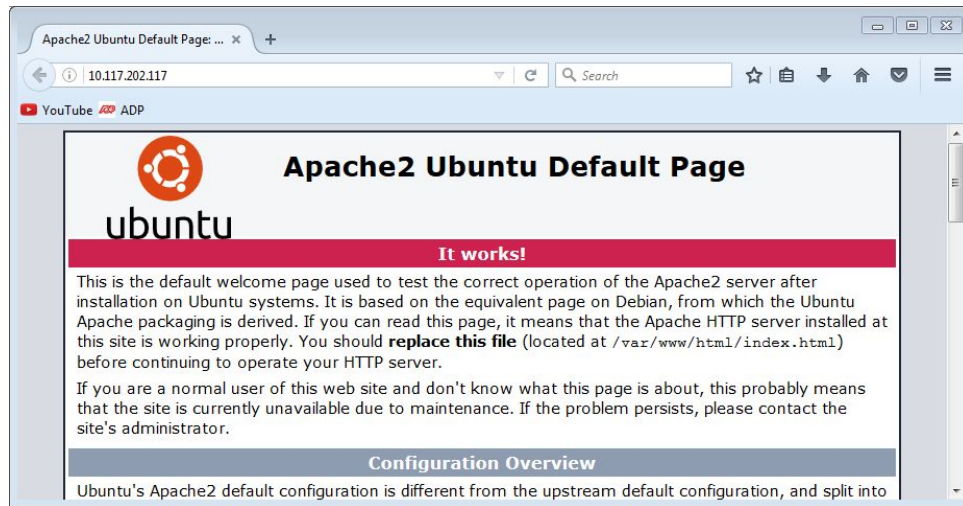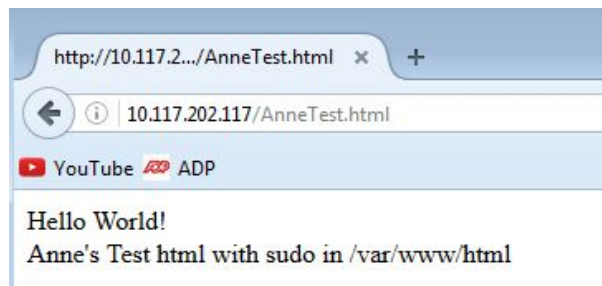
5.3 Useful Links:

[How to install Linux, Apache, MySQL, PHP (LAMP) stack on Ubuntu](#) (follow Apache only)

6 JAVA WEB SERVER: TOMCAT

A web server application that handles Java (Apache does not). This includes Java Server Pages (JSP's) coding and Java servlets. Can be used as stand-alone web server, or in addition to Apache to only handle the Java pieces.

6.1 Install Tomcat on Ubuntu

6.1.1 Step 1: Install Java

Tomcat requires Java to be installed on the server so that Java web application code can be executed. Install OpenJKD (Java Development Kit) with apt-get.

Update the apt-get package index:

$ sudo apt-get update

Install the Java Development Kit package with apt-get:

$ sudo apt-get install default-jdk
 Do you want to continue? [Y/n]: y + Enter

6.1.2 Step 2: Create Tomcat user and Tomcat group which will be used to run the Tomcat service as an unpriviledged user (i.e. not root)

Create new *tomcat* group:

$ sudo groupadd tomcat

Create new *tomcat* user. This user will be made a member of the *tomcat* group, with a home directory of */opt/tomcat* (where Tomcat will be installed), and with a shell of */bin/false* (so nobody can log into the account):

$ sudo useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat

6.1.3 Step 3: Download and Install Tomcat

On the Tomcat 8 Downloads page (http://tomcat.apache.org/download-80.cgi), find the latest stable version. Under *Binary Distributions* and *Core*, copy the link to the *tar.gz* file (right-click on the file and choose *Copy Link Location*).

Change to the */tmp* directory on your server. This is good directory to download phemeral items, like the Tomcat tarball, which we won't need after extracting the Tomcat contents:

$ cd /tmp

Use *curl* and paste the link to download the Tomcat archive:

> $ curl -O
> http://apache.parentingamerica.com/tomcat/tomcat-8/v8.5.5/bin/apache-tomcat-8.5.5
> .tar.gz

```
hagermanan@ONOTTRA123456D:/tmp$ curl -O http://apache.parentingamerica.com/tomca
t/tomcat-8/v8.5.5/bin/apache-tomcat-8.5.5.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 9070k  100 9070k    0     0   840k      0  0:00:10  0:00:10 --:--:-- 1099kk
```

Install Tomcat to the */opt/tomcat* directory. Create the directory, then extract the archive to it:

> $ sudo mkdir /opt/tomcat
> $ sudo tar xzvf apahce-tomcat-8*tar.gz -C /opt/tomcat --strip-components=1

6.1.4 Step 4: Update Permissions

Set up proper user permissions for the installation. The *tomcat* user needs to have access to the Tomcat installation.

Change the directory where we unpacked the Tomcat installation:

> $ cd /opt/tomcat

Give the *tomcat* user *write* access to the *conf* directory, and *read* access to the files in that directory:

> $ sudo chgrp -R tomcat conf
> $ sudo chmod g+rwx conf
> $ sudo chmod -R g+r conf/
> > *Not $ sudo chmod g+r conf/* as per the instructions → Ubuntu error conf/* DNE*

Make the *tomcat* use the owner of the *webapps, work, temp*, and *logs* directories:

> $sudo chown -R tomcat webapps/ work/ temp/ logs/

6.1.5 Step 5: Create a systemd Service File to Manage the Tomcat Process

To run Tomcat as a service, we will set up a systemd service file. In this file, we need to put the *JAVA_HOME* path which is where Java is installed (So Tomcat knows where to look for it).

To find it:

> $ sudo update-java-alternatives -l

The output provides 3 coloums of information:

```
java-1.8.0-openjdk-amd64       1081       /usr/lib/jvm/java-1.8.0-openjdk-amd64
```

The *JAVA_HOME* variable can be constructed from the last coloumn and appending */jre* to the end:

JAVA_HOME:
/usr/lib/jvm/java-1.8.0-openjdk-amd64/jre

Create a *tomcat.service* file in the *systemd/system* directory (below) and paste the following into the file:

```
$ sudo nano /etc/systemd/system/tomcat.service
```

/etc/systemd/system/tomcat.service → paste the following in this file

```
[Unit]
Description=Apache Tomcat Web Application Container
After=network.target

[Service]
Type=forking

Environment=JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64/jre
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
Environment='CATALINA_OPTS=-Xms512M -Xmx1024M -server
-XX:+UseParallelGC'
Environment='JAVA_OPTS=-Djava.awt.headless=true
-Djava.security.egd=file:/dev/./urandom'

ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh

User=tomcat
Group=tomcat
RestartSec=10
Restart=always

[Install]
WantedBy=multi-user.target
```

The *JAVA_HOME* variable indicated in **bold** would need to be changed if the output of the Java location is different.

Save and close the file.

Reload the systemd daemon so that it knows about the new service file:

```
$ sudo systemctl daemon-reload
```

Start the Tomcat service:

$ sudo systemctl start tomcat

Double check that it started without errors by typing:

$ sudo systemctl status tomcat

If Tomcat does not start and you are getting an error like this:

tomcat.service - Apache Tomcat Web Application Container Loaded: loaded (/etc/systemd/system/tomcat.service; enabled; vendor preset: enabled) Active: activating (auto-restart) (Result: exit-code) since Sat 2016-07-30 12:15:27 EDT; 1s ago Process: 13807 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=203/EXEC) Jul 30 12:15:27 xxxxxxx systemd[1]: Failed to start Apache Tomcat Web Application Container. Jul 30 12:15:27 xxxxxxx systemd[1]: tomcat.service: Unit entered failed state. Jul 30 12:15:27 xxxxxxxxx systemd[1]: tomcat.service: Failed with result 'exit-code'.

then you might need to give the tomcat user ownership of the whole tomcat directory:

$ cd /opt && sudo chown –R tomcat tomcat/

The status should output something like this:

```
● tomcat.service - Apache Tomcat Web Application Container
   Loaded: loaded (/etc/systemd/system/tomcat.service; disabled; vendor preset:
   Active: active (running) since Thu 2016-09-29 17:10:58 EDT; 20s ago
  Process: 10109 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=0/SUC
 Main PID: 10118 (java)
   CGroup: /system.slice/tomcat.service
           └─10118 /usr/lib/jvm/java-1.8.0-openjdk-amd64/jre/bin/java -Djava.uti
```

6.1.6 Step 6: Configure Tomcat
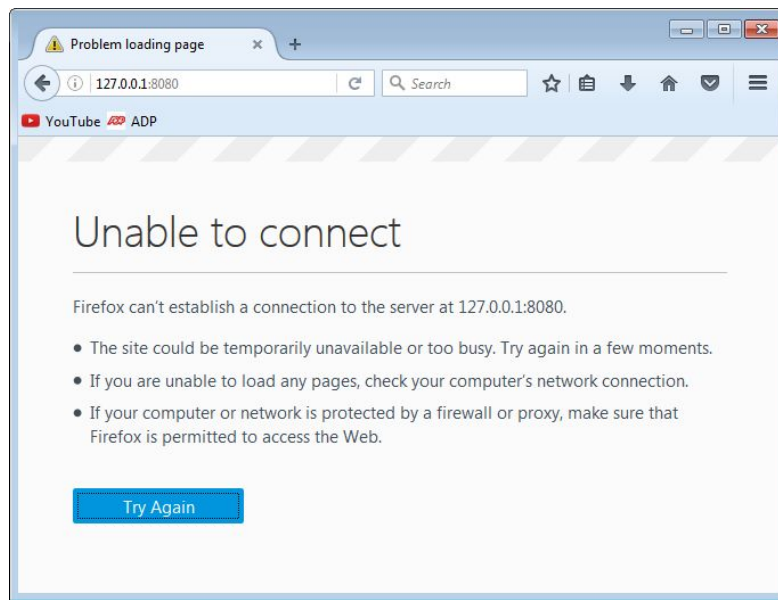
Tomcat uses port *8080* to accept conventional requests.

6.1.6.i Firewall

Adjust the firewall to allow traffic to that port:

$ sudo ufw allow 8080

6.1.6.ii Test Tomcat LocalHost

Test Tomcat in server / Ubuntu with *127.0.0.1:8080 (localhost)*
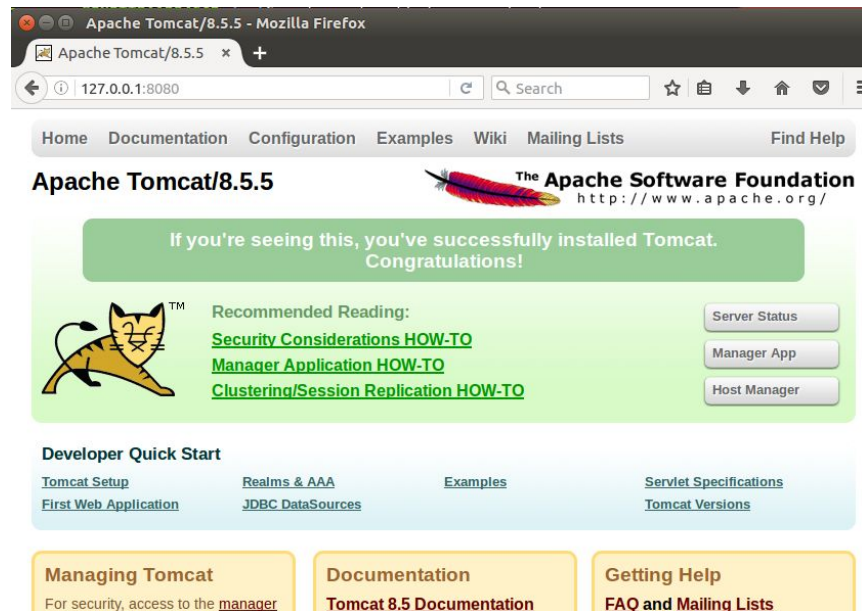
6.1.6.iii Set Tomcat to Start Automatcially on System Boot

Enable the service file so Tomcat automatically starts at boot (if server is not booted, Tomcat will not run – Apache will still run on the web even if the server is off):

$ sudo systemctl enable tomcat

6.1.6.iv Forward Port 8080

Port 8080 will need to be forwarded the same way that port 80 was for Apache. Shutdown Ubuntu, change Ubuntu VM settings and then restart Ubuntu:

Name: Ubuntu Tomcat

Host Port: 8080

Guest Port: 8080
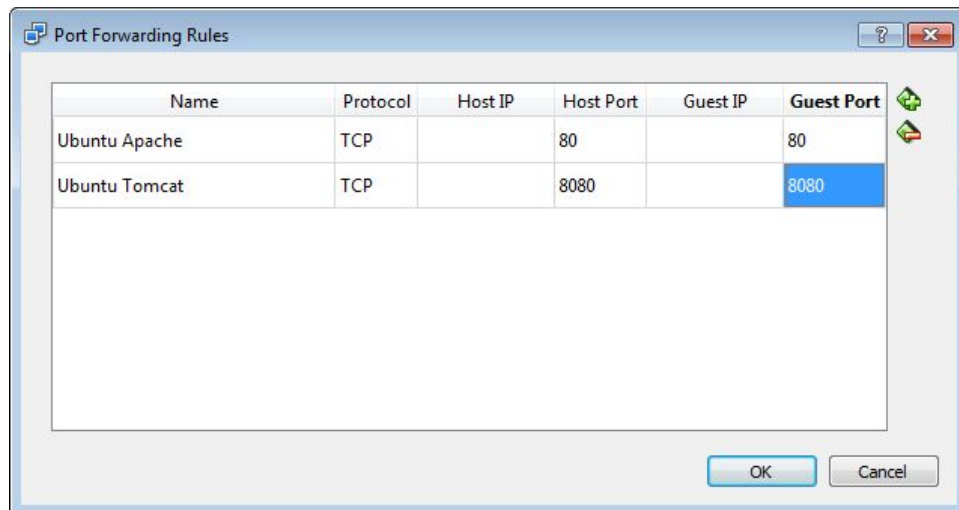
6.1.6.v Test Tomcat Externally
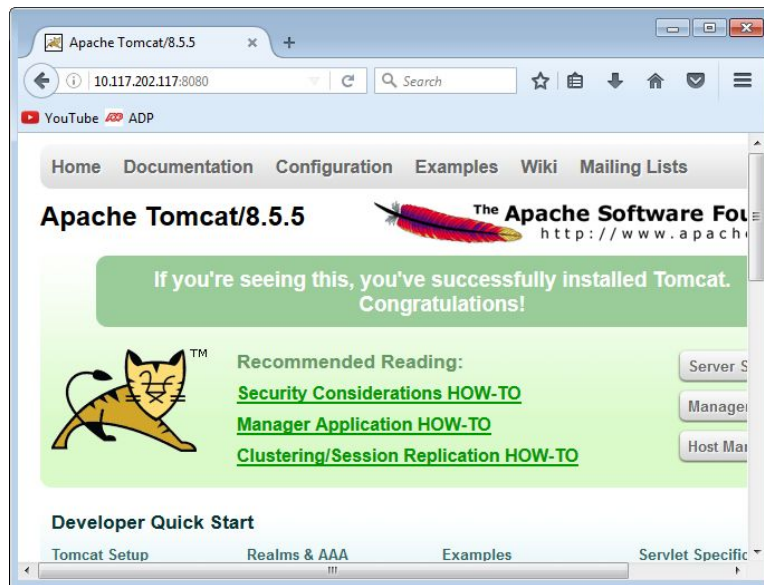
Test Tomcat in external Windows with *IP:8080* :

You will see the default Tomcat splash page, in addition to other information. However, if you click the links for the Manager App, for instance, you will be denied access. We can configure that access next.

6.1.7 Step 7: Configure Tomcat Web Management Interface

You must add a login to the Tomcat server in order to use the manager web app that comes with Tomcat. This is done by editing the *tomcat-users.xml* file:

```
$ sudo nano /opt/tomcat/conf/tomcat-user.xml
```

A user who can access the *manager-gui* and *admin-gui* (web apps that come with Tomcat) will need to be added. This can be done by defining a user between the *tomcat-user* tags with a secure username and password:

```
<user username="manadmin" password="password1" roles="manager-gui,admin-gui"/>
```

```
<user username="manadmin" password="password1" roles="manager-gui,admin-gui"/>
</tomcat-users>
```
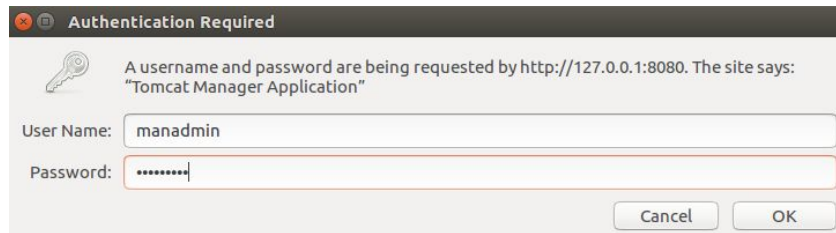
Save and close the file.

To put the changes in effect, restart the Tomcat service:
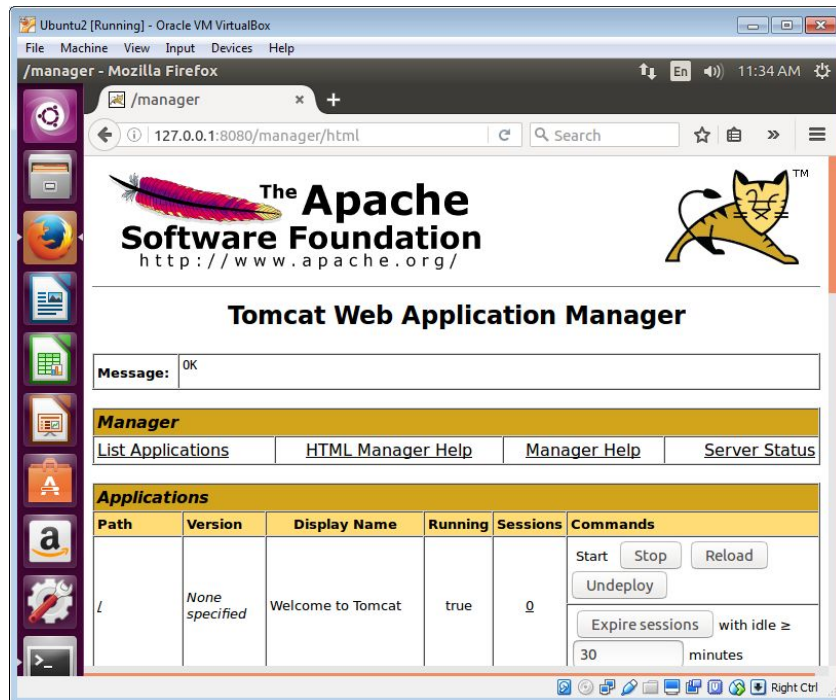
$ sudo systemctl restart tomcat

6.1.8 Step 8: Access the Tomcat Web Interface – Manager and Host Manager

From the Apache Tomcat homepage *127.0.0.1:8080*, click on *Manager App* or *Host Manager*, or add */manager/html* or */manager-host/html* to the address. Enter the account credentials previously added to the *tomcat-users.xml* file:

Tomcat is now fully installed.

6.1.9 Remote Access to the Manager App.

By default, the Manager App is only accessible via the computer that Tomcat sits on. If, for example, data is stored on a server with no graphic interface, accessing the Manager App requires some configuration and the addition of a Remote Address Valve to the manager.xml conf files.

Open manager.xml -> $ sudo nano /opt/tomcat/webapps/host-manager/manager.xml

Next, nested within the <context> tags, copy:
<Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="^.*$" />

Save and close.
Next, open context.xml -> $ sudo nano /opt/tomcat/webapps/manager/META-INF/context.xml
Change this file to comment out the Valve, so it looks like this:

<Context antiResourceLocking="false" privileged="true" >
  <!--
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
      allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />
  -->
</Context>

Restart Tomcat, $ systemctl restart tomcat, then in your browser navigate to Tomcat home, open Manager App, and enter the user credentials you used previously for step 6.1.8

This method worked for this setup, though other solutions exist. Follow these links for additional information and resources.

Manager App Remote Access
Manager App Setup Options
Tomcat Help Docs - Remote Address Valve

6.2 Server ERROR 500

In the case that an ERROR 500 is thrown when launching the tomcat index page, a possible fix may be removal of the .jsp index pages from the tomcat root directory. Instructions on how to do so were found here, StackOverFlow

" *If you are running JSPs simply go into your TOMCAT_HOME/work/Catalina/localhost/ directory, look for the name of your webapp, cd into that and go all the way down the directory tree from there until you find the JSP class files.. sudo rm -f * and try again.* "

Appears that the jsp index pages can be corrupted at compilation and throw errors. This is a good place to start if problems arise.

6.2 Allow Apache and Tomcat to work together so Apache handles static pages and Tomcat handles java pages:

6.2.1 Step 1: Install and Configure *mod_jk*

The Apache web server uses the *mod_jk* module to communicate with Tomcat using the Apache JServ Protocol.

Instal *mod_jk* from Ubuntu's default repositories:

$ sudo apt-get update
$ sudo apt-get install libapache2-mod-jk

The module will be enabled automatically upon installation.

Configure the module via the main configuration file:

$ sudo nano /etc/libapache2-mod-jk/workers.properties

Change the *workers.tomcat_home* to the Tomcat installation home directory */opt/tomcat*:

/etc/libapache2-mod-jk/workers.properties

workers.tomcat_home=/opt/tomcat

```
workers.tomcat_home=/usr/share/tomcat8
```

```
workers.tomcat_home=/opt/tomcat
```

Save and close the file.

6.2.2 Step 2: Configure the URLs that Apache should pass through Tomcat

Edit the *<VirtualHost *:80>* directive in Apache to mount the directories you want to pass through Tomcat with workers:

$ sudo nano /etc/apache2/sites-available/000-default.conf

JkMount /manager/* ajp13_worker
JkMount /manager ajp13_worker
JkMount /examples/* ajp13_worker
JkMount /examples ajp13_worker

```
        JkMount /manager/* aj13p_worker
        JkMount /manager ajp13_worker
        JkMount /examples/* ajp13_worker
        JkMount /examples ajp13_worker
</VirtualHost>
```

Add another pair for each folder you create in the *opt/tomcat/webapps* directory that you want to put java-enabled webpages in.

Restart Apache:

$ sudo systemctl restart apache2

Now anything in the manager or examples directory will go through Tomcat and the rest will run through Apache and be looked for in the /var/www/html folder.

6.3 Useful Links:

How to install and setup mod_jk module (ignore encryption steps)

Digital Ocean – How to Install Apache Tomcat 8 on Ubuntu 16.04

How to Install Java on Ubuntu with Apt-Get

How to setup Tomcat 8 with Apache 2.4 and mod_jk on Ubuntu

7 GEOSERVER

GoeServer User Manual – Installation:
http://docs.geoserver.org/latest/en/user/installation/war.html#installation-war

7.1 Step 1: Java

Currently, GeoServer requires Java 8. A Java Runtime Environment (JRE) must be installed on the system. Oracle JRE is preferred, but OpenJDK can work.

This system already has OpenJDK JRE installed.

7.2 Step 2: Download GeoServer

Go to http://geoserver.org/release/stable/

Select the version of GeoServer. This system has 2.9.1.

Stable GeoServer 2.9.1 ⊕
The recommended release of GeoServer,
tested and supported by the community.

GeoServer 2.9 releases:
2.9.1   2.9.0   2.9-RC1
2.9-beta2   2.9-beta   2.9-M0

Select *Web Archive* in the *Packages* section.

Packages

Platform Independent Binary
Operating system independent runnable binary.

Windows Installer
Installer for Windows platforms.

Mac OSX Installer
DMG for OSX platforms.

Web Archive
Web Archive (war) for servlet containers.

Save the file.
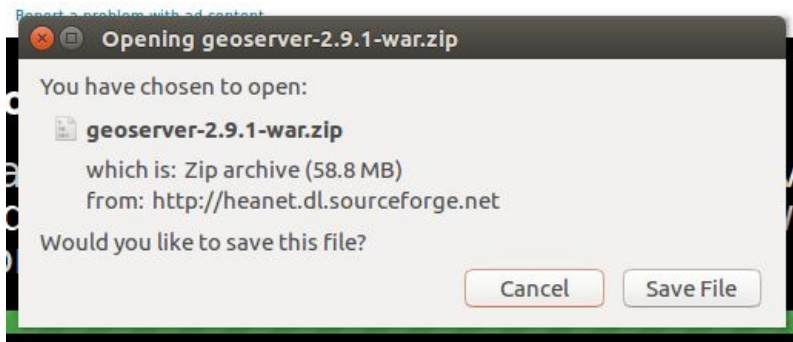
7.3 Step 3: Unpack the archive

Locate the downloaded .zip file and unpack it to the */opt* directory (/opt = optional application software packages):

```
hagermanan@ONOTTRA123456D:~$ ll Downloads
total 60256
drwxr-xr-x  2 hagermanan hagermanan     4096 Sep 30 14:09 ./
drwxr-xr-x 17 hagermanan hagermanan     4096 Sep 30 11:39 ../
-rw-rw-r--  1 hagermanan hagermanan 61693319 Sep 30 14:08 geoserver-2.9.1-war.zip
```

$ sudo mkdir /opt/geoserver

$ sudo ~/Downloads/geoserver-2.9.1-war.zip –d /opt/geoserver

```
Archive:  /home/hagermanan/Downloads/geoserver-2.9.1-war.zip
  inflating: /opt/geoserver/geoserver.war
  inflating: /opt/geoserver/LICENSE.txt
  inflating: /opt/geoserver/GPL.txt
   creating: /opt/geoserver/target/
  inflating: /opt/geoserver/target/VERSION.txt
hagermanan@ONOTTRA123456D:~$ ll /opt/geoserver
total 66104
drwxr-xr-x 3 root root     4096 Sep 30 14:23 ./
drwxr-xr-x 4 root root     4096 Sep 30 14:20 ../
-rw-r--r-- 1 root root 67626672 Aug  3 23:37 geoserver.war
-rw-r--r-- 1 root root    18008 Sep 18  2013 GPL.txt
-rw-r--r-- 1 root root    26610 Aug  3 18:28 LICENSE.txt
drwxr-xr-x 2 root root     4096 Aug  3 23:44 target/
hagermanan@ONOTTRA123456D:~$
```

7.4 Step 4: Deploy the GeoServer application

Often, all that is necessary is to copy the *geoserver.war* file to the application server's *webapps* directory, and then the GeoServer application will be deployed.

```
hagermanan@ONOTTRA123456D:~$ sudo ls -l /opt/tomcat/webapps
sudo: unable to resolve host ONOTTRA123456D: Connection timed out
total 20
drwxr-x--- 14 tomcat root 4096 Sep 29 15:58 docs
drwxr-x---  6 tomcat root 4096 Sep 29 15:58 examples
drwxr-x---  5 tomcat root 4096 Sep 29 15:58 host-manager
drwxr-x---  5 tomcat root 4096 Sep 29 15:58 manager
drwxr-x---  3 tomcat root 4096 Sep 29 15:58 ROOT
```

$ sudo cp /opt/geoserver/geoserver.war /opt/tomcat/webapps

This copies the *geoserver.war* file and a *geoserver* directory.

```
hagermanan@ONOTTRA123456D:~$ sudo ls -l /opt/tomcat/webapps
sudo: unable to resolve host ONOTTRA123456D: Connection timed out
total 66068
drwxr-x--- 14 tomcat root        4096 Sep 29 15:58 docs
drwxr-x---  6 tomcat root        4096 Sep 29 15:58 examples
drwxr-x---  5 tomcat tomcat      4096 Sep 30 14:32 geoserver
-rw-r--r--  1 root   root    67626672 Sep 30 14:32 geoserver.war
drwxr-x---  5 tomcat root        4096 Sep 29 15:58 host-manager
drwxr-x---  5 tomcat root        4096 Sep 29 15:58 manager
drwxr-x---  3 tomcat root        4096 Sep 29 15:58 ROOT
```

Restart tomcat

$ sudo systemctl restart tomcat

7.5 Access GeoServer

GeoServer has a browser-based web administration interface application used to configure all aspects of GeoServer, from adding & publishing data to changing server settings.

http://localhost:8080/geoserver

Log In
>   Username: admin
>   Password: geoserver

7.6 GeoServer Data Directory

Data can be either added to the default GeoServer data directory -> */opt/tomcat/webapps/geoserver/data,* or by creating a new store, a local data repository can be linked to and that data published to GeoServer.

For more information on GeoServer data management, follow this link to the GeoServer user manual.

7.7 Allow Cross-Origin Request Config

To allow cross origin resource sharing and avoid "Same Origin Policy" errors when requesting server pages through OpenLayers, configurations are made to the Geoserver web.xml config file to include "Allow Cross Origin" headers.

Open:

/tomcat/webapps/geoserver/WEB-INF/web.xml

Add the following:

```
<filter>
  <filter-name>CorsFilter</filter-name>
  <filter-class>org.apache.catalina.filters.CorsFilter</filter-class>
</filter>
<filter-mapping>
  <filter-name>CorsFilter</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
```

Then restart Tomcat:

$ systemctl restart tomcat

7.8 Enabling JSONP

As a workaround to the Same Origin Policy, JSONP (see link for a good explanation of JSONP) can be enabled on GeoServer as it is a common solution for CORS. However, it seems the most sustainable step is still to modify the CORS headings as explained in the previous section. This step is not critical.

Add to this file: /tomcat/webapps/geoserver/WEB-INF/web.xml

```
<context-param>
   <param-name>ENABLE_JSONP</param-name>
   <param-value>true</param-value>
</context-param>
```

How To Enable JSONP

GeoServer Reference to JSONP

7.9 Increasing Heap Memory

Running certain WPS processing on Geoserver can throw an "Out Of Memory" error. To solve this, we make modifications to the tomcat context.xml file and add a custom environment setting file.

In opt/tomcat/conf/context.xml add the following lines above </Context>

```
< Resources cachingAllowed="true" cacheMaxSize="100000" />
```

Next create custom environment setting file (may need to enter $ sudo su for access)

go to opt/tomcat/bin and enter command $ gedit setenv.sh

Once open, paste the following block of code:

```
export CATALINA_OPTS="$CATALINA_OPTS -Xms512m"

export CATALINA_OPTS="$CATALINA_OPTS -Xmx8192m"

export CATALINA_OPTS="$CATALINA_OPTS -XX:MaxPermSize=256m"
```

Save and exit. Finally, go up to opt/tomcat/bin directory, and run command $ ./catalina.sh run

7.10 Enabling Web Processing Services (WPS)

Web Processing Service (WPS) is an OGC service for the publishing of geospatial processes, algorithms, and calculations. The WPS service is available as an extension for geoserver providing an execute operation for data processing and geospatial analysis.

Installation steps for the WPS module can be found through the following link;

Installing WPS Module

To download WPS module, navigate through the following link and find the WPS extension version which matches that of your Geoserver version. (see Production, Development, and Archived tabs for more GS versions). WPS service is found under Extensions.

Download -> Geoserver Downloads Page

Next, extract the zip file into your WEB-INF/lib directory using the following command (modify command if local path and/or version is different)

$ sudo unzip /home/<user>/Downloads/geoserver-2.10.0-wps-plugin.zip -d /opt/tomcat/webapps/geoserver/WEB-INF/lib/

Next, restart Geoserver through the Tomcat Web Application Manager;

http://127.0.0.1:8080/manager/html

Once reloaded, visit Geoservers web administration interface and under Service Capabilities, WPS 1.0.0 should be visible.

7.11 Enabling REST services for GET request

The REST interface provides quick and simple access to geoserver settings and services. Enabling anonymous GET requests allows for the SoilMap application to access workspaces and available data stores without authentication prompts.

Open rest.properties in /security:

root@ulysses:/opt/tomcat/webapps/geoserver/data/security# gedit rest.properties

Modify the current GET security setting to this:

/**;GET=IS_AUTHENTICATED_ANONYMOUSLY

Save, exit, and restart GeoServer from console or manager app

GeoServer Docs - Security options for REST interface

7.12 Useful Links

[Modifying web.xml to allow CORS](#)

[Modifying JVM Heap Memory Settings](#)

[Increase Tomcat Cache Size](#)

[GeoServer User Manual](#)

[Publishing a WMS in GeoServer with QGIS](#)

8 OPENLAYERS

OpenLayers 3: http://openlayers.org/download/

Download the *v3.18.2.zip* (or latest release). This archive includes the distribution plus examples, API docs and sources.

Follow link to goto latest release and click to download

In console, goto /home/<user>/Downloads and unpack the newly downloaded zip into Apaches web root folder using the following command:

```
$ sudo unzip /home/<user>/Downloads/<openlayers.zip> -d /var/www/html/ol
```

When building a web page, we can reference either our local script source, or the available web source.

```
Web:
<script src="https://openlayers.org/en/v3.19.1/build/ol.js"></script>
<link rel="stylesheet" href="https://openlayers.org/en/v3.19.1/css/ol.css" type="text/css">

Local:
<script src="../../ol/v3.19.1-dist/ol.js"></script>
<link rel="stylesheet" href="../../ol/v3.19.1-dist/ol.css" type="text/css">
```

9 QGIS

Instructions on the QGIS Installer website were followed:
http://www.qgis.org/en/site/forusers/alldownloads.html#debian-ubuntu

9.1 Step 1: Remove Previous QGIS & Grass Packages

Ensure there are no previous repositories or any residual and broken packages if any. This is advised on the QGIS Installer website. The following steps were found on:
http://askubuntu.com/questions/351899/qgis-quantum-gis-install-fails-unmet-dependencies

$ sduo software-properties-gtk

In the Software & Updates , navigate to the Other Software tab. Highlight the QGIS repositories. Look for names similar to *http://qgis.org/debian* and *ubuntugis-unstable*.

Click Remove, then Close the window.

$ sudo apt-get autoremove
$ sudo apt-get autoclean
$ sudo apt-get -f install
$ sudo apt-get autoremove qgis
$ sudo apt-get --purge remove qgis

9.2 Step 2: Choose QGIS Release

From the QGIS website, choose the desired release.

Release: Latest Release
Version: 2.18.x Las Palmas
Description: Release for Debian and Ubuntu
Repository: http://qgis.org/debian
Ubuntu v16.04 Codename: xenial

9.3 Step 3: Add QGIS Repository

Add the following lines for the chosen repository to the */ect/apt/sources.list* file:
deb   http://qgis.org/debian xenial main
deb-src   http://qgis.org/debian xenial main

9.4 Step 4: Keyserver

Update keyserver by adding the qgis.org repository public key to your apt keyring

$ wget -O - http://qgis.org/downloads/qgis-2016.gpg.key | gpg --import
$ gpg --fingerprint 073D307A618E5811

This should output:
pub   2048R/618E5811 2016-08-17 [expires: 2017-08-17]
      Key fingerprint = 942D 6AD5 DF3E 75DE A9AF 72B2 073D 307A 618E 5811
uid   QGIS Archive Automatic Signing Key (2016) <qgis-developer@lists.osgeo.org>
sub   2048R/D34A963D 2016-08-17

```
$ gpg --export --armor 073D307A618E5811 | sudo apt-key add -
```

9.5 Step 5: Install QGIS

```
$ sudo apt-get update
$ sudo apt-get install qgis python-qgis qgis-plugin-grass
```

9.6 Helpful Links
http://www.qgis.org/en/site/forusers/alldownloads.html

http://gis.stackexchange.com/questions/133033/installing-the-latest-qgis-version-on-ubuntu-14-04

10 MySQL

10.1 Step 1: Add MySQL APT to the Software Repository List

From the MySQL APT repository (http://dev.mysql.com/downloads/repo/apt/), download the release package for the Linux distribution, Ubuntu 16.04.

Install the downloaded release package with the by going to the directory where the download was saved and then initiate the install config app:

```
$ cd ~/Downloads
$ sudo dpkg -i mysql-apt-config_0.8.0-1_all.deb
```

During the installation, an options dialogue appears. Leave the default selection, press Right Arrow & Enter on both screens, then select Ok & Right Arrow Enter to complete.

```
Configuring mysql-apt-config
MySQL APT Repo features MySQL Server along with a variety of MySQL
components. You may select the appropriate product to choose the version
that you wish to receive.

Once you are satisfied with the configuration then select last option
'Apply' to save the configuration. Advanced users can always change the
configurations later, depending on their own needs.

Which MySQL product do you wish to configure?

        MySQL Server (Currently selected: mysql-5.7)
        MySQL Tools & Connectors (Currently selected: Enabled)
        MySQL Preview Packages (Currently selected: Disabled)
        Ok


                    <Ok>
```

```
Configuring mysql-apt-config
This configuration program has determined that  is installed on your
system, and has highlighted the most appropriate repository package. If
you are not sure which version to install, do not change the
auto-selected version. Advanced users can always change the version as
needed later.

Which server version do you wish to receive?

                mysql-5.6
                mysql-5.7
                mysql-8.0 preview
                None


                <Ok>
```

Note: This screenshot was taken after mysql-5.7 was already added. When it is the first time, the dialogue will read: "… determined that it is not installed on your system…".

```
Package configuration

                 ┤ Configuring mysql-apt-config ├
  MySQL APT Repo features MySQL Server along with a variety of MySQL
  components. You may select the appropriate product to choose the version
  that you wish to receive.

  Once you are satisfied with the configuration then select last option
  'Apply' to save the configuration. Advanced users can always change the
  configurations later, depending on their own needs.

  Which MySQL product do you wish to configure?

          MySQL Server (Currently selected: mysql-5.7)
          MySQL Tools & Connectors (Currently selected: Enabled)
          MySQL Preview Packages (Currently selected: Disabled)
          Ok

                          <Ok>
```

Update package information from the MySQL APT repository:

$ sudo apt-get update

Note: Once the MySQL APT repository is enabled on your system, you will no longer be able to install any MySQL packages from your platform's native software repositories until the MySQL APT repository is disabled.

10.2 Step 2: Install MySQL with APT

The following command installs the package for the MySQL server as well as the packages for the client and for the database common files.

$ sudo apt-get install mysql-server

Supply a password for the root user for the MySQL installation.

freedatabase

```
Package configuration




                       ┌─────────┤ Configuring mysql-server-5.7 ├─────────┐
                       │ While not mandatory, it is highly recommended that you set a password │
                       │ for the MySQL administrative "root" user.                            │
                       │                                                                      │
                       │ If this field is left blank, the password will not be changed.       │
                       │                                                                      │
                       │ New password for the MySQL "root" user:                              │
                       │                                                                      │
                       │ ***********                                                          │
                       │                                                                      │
                       │                              <Ok>                                    │
                       │                                                                      │
                       └──────────────────────────────────────────────────────────────────────┘
```
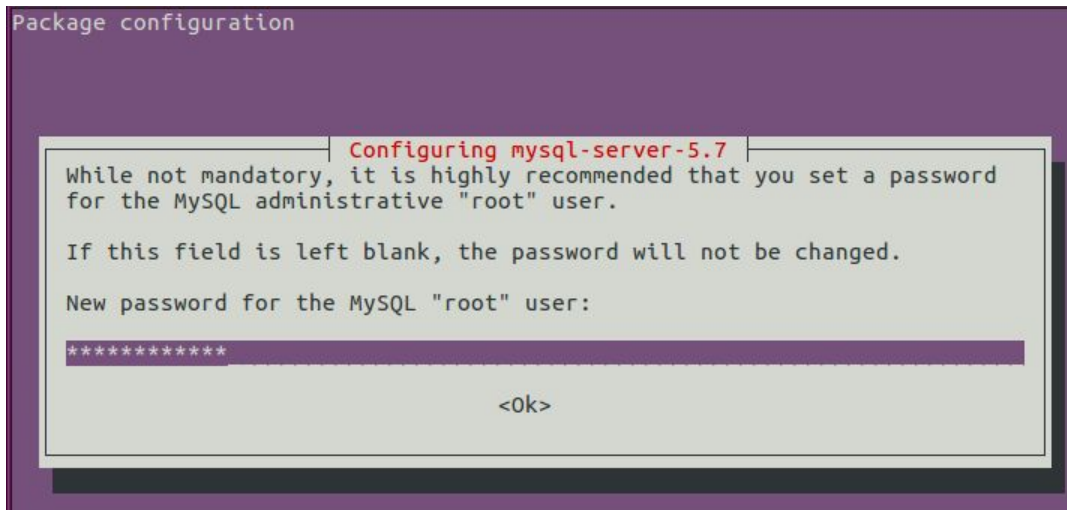
Note: Make sure you remember the root password you set. Users who want to set a password later can leave the password field blank in the dialogue box and just press OK; in that case, root access to the server will be authenticated by The Socket Peer-Credential Authentication Plugin for connections using a Unix socket file. You can set the root password later using the program mysql_secure_installation.

10.3 Starting and Stopping the MySQL Server

MySQL server is started automatically after the installation. You can check the status of the MySQL server, stop the MySQL server and restart the MySQL server as follows:

$ sudo service mysql status

```
hagermanan@ONOTTRA673312D:~/Downloads$ sudo service mysql status
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: en
   Active: active (running) since Mon 2016-11-21 16:16:07 EST; 5min ago
 Main PID: 9890 (mysqld)
   CGroup: /system.slice/mysql.service
           └─9890 /usr/sbin/mysqld

Nov 21 16:16:06 ONOTTRA673312D systemd[1]: Starting MySQL Community Server...
Nov 21 16:16:07 ONOTTRA673312D systemd[1]: Started MySQL Community Server.
lines 1-9/9 (END)
```

Ctrl + C

$ sudo service mysql stop
$ sudo service mysql status

```
hagermanan@ONOTTRA673312D:~/Downloads$ sudo service mysql stop
hagermanan@ONOTTRA673312D:~/Downloads$ sudo service mysql status
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: en
   Active: inactive (dead) since Mon 2016-11-21 16:22:19 EST; 4s ago
 Main PID: 9890 (code=exited, status=0/SUCCESS)

Nov 21 16:16:06 ONOTTRA673312D systemd[1]: Starting MySQL Community Server...
Nov 21 16:16:07 ONOTTRA673312D systemd[1]: Started MySQL Community Server.
Nov 21 16:22:17 ONOTTRA673312D systemd[1]: Stopping MySQL Community Server...
Nov 21 16:22:19 ONOTTRA673312D systemd[1]: Stopped MySQL Community Server.
lines 1-9/9 (END)
```

Ctrl + C

$ sudo service mysql start
$ sudo service mysql status

```
hagermanan@ONOTTRA673312D:~/Downloads$ sudo service mysql start
hagermanan@ONOTTRA673312D:~/Downloads$ sudo service mysql status
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: en
   Active: active (running) since Mon 2016-11-21 16:23:31 EST; 3s ago
  Process: 10096 ExecStartPost=/usr/share/mysql/mysql-systemd-start post (code=e
  Process: 10092 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exi
 Main PID: 10095 (mysqld)
   CGroup: /system.slice/mysql.service
           └─10095 /usr/sbin/mysqld

Nov 21 16:23:30 ONOTTRA673312D systemd[1]: Starting MySQL Community Server...
Nov 21 16:23:31 ONOTTRA673312D systemd[1]: Started MySQL Community Server.
lines 1-11/11 (END)
```

Ctrl + C

10.4 Installing Additional MySQL Products and Components

APT can be used to install individual components of MySQL from the MySQL APT repository.

Get the latest package information from the MySQL repository and then install any packages of choice replacing   *package-name*   with the name of the package (list of available packages: http://dev.mysql.com/doc/mysql-apt-repo-quick-guide/en/#repo-qg-apt-available)

$ sudo apt-get update

$sudo apt-get install *package-name*

This is for informational purposes only as no additional components were added to the system at the time of system build.

11 PHP

11.1 Step 1: Install PHP

Install PHP 7 and Apache PHP module.

$ sudo apt-get -y install php7.0 libapache2-mod-php7.0

Restart Apache

$ sudo systemctl restart apache2

11.2 Step 2: Test PHP & Get Details the PHP Installation

Create a PHP file in the root of the default website to provide details about the PHP installation.

$ sudo nano /var/www/html/info.php

```
<?php
phpinfo();
?>
```

Change the owner of the info.php file from root to the www-data user and group.

```
-rw-r--r-- 1 root root      20 Nov 21 14:09 info.php
```

$ sudo chown www-data:www-data /var/www/html/info.php

```
-rw-r--r-- 1 www-data www-data      20 Nov 21 14:09 info.php
```

Call the file in a web browser.

localhost/info.php   C   Q Search   ☆ | 自   ↓

Most Visited ✓ Getting Started OpenLayers 3 - Welco... 02_WeatherWS 02_ApacheWS 01_OL3Exs GeoServer SoilGrids FTP

PHP Version 7.0.8-0ubuntu0.16.04.3 php

System	Linux ONOTTRA673312D 4.4.0-47-generic #68-Ubuntu SMP Wed Oct 26 19:39:52 UTC 2016 x86_64
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.0/apache2
Loaded Configuration File	/etc/php/7.0/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.0/apache2/conf.d
Additional .ini files parsed	/etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-iconv.ini, /etc/php/7.0/apache2/conf.d/20-json.ini, /etc/php/7.0/apache2/conf.d/20-phar.ini, /etc/php/7.0/apache2/conf.d/20-posix.ini, /etc/php/7.0/apache2/conf.d/20-readline.ini, /etc/php/7.0/apache2/conf.d/20-shmop.ini, /etc/php/7.0/apache2/conf.d/20-sockets.ini, /etc/php/7.0/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.0/apache2/conf.d/20-sysvsem.ini, /etc/php/7.0/apache2/conf.d/20-sysvshm.ini, /etc/php/7.0/apache2/conf.d/20-tokenizer.ini
PHP API	20151012
PHP Extension	20151012
Zend Extension	320151012
Zend Extension Build	API320151012,NTS
PHP Extension Build	API20151012,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:
Zend Engine v3.0.0, Copyright (c) 1998-2016 Zend Technologies
    with Zend OPcache v7.0.8-0ubuntu0.16.04.3, Copyright (c) 1999-2016, by Zend Technologies   zend engine

Configuration

apache2handler

NOTE: Steps 3 & 4 may not be required. Check info.php website before adding the modules below

11.3 Step 3: Install PHP Modules

The following code provides a list of available PHP modules.

```
$ apt-cache search php7.0
```

The following modules were selected for this system configuration:
- php7.0-mysql
  - MySQL support in PHP
- php7.0-common
  - Documentation, Examples and Common Module
- php7.0-json
  - Implements the JavaScript Object Notation (JSON) data interchange format

- o http://php.net/manual/en/book.json.php
- php7.0-xml
  - o DOM, SimpleXML, WDDX, XML & XLS Module
  - o Create XML parsers (with adjustable parameters) then define handlers for different XML events
  - o http://php.net/manual/en/book.xml.php

`$ sudo apt-get -y install php7.0-mysql php7.0-common php7.0-json php7.0-xml`

- In order to manipulate and clean invalid XML objects, Tidy PHP was selected for installation as well. Tidy is a binding for the Tidy HTML clean and repair utility which allows you to not only clean and otherwise manipulate HTML documents, but also traverse the document tree.

  [PHP - Tidy documentation manual](#)

  `$ sudo apt-get install php-tidy`

- In order to connect between servers in PHP the libcurl package was installed. "PHP supports libcurl, a library created by Daniel Stenberg, that allows you to connect and communicate to many different types of servers with many different types of protocols. libcurl currently supports the http, https, ftp, gopher, telnet, dict, file, and ldap protocols. libcurl also supports HTTPS certificates, HTTP POST, HTTP PUT, FTP uploading (this can also be done with PHP's ftp extension), HTTP form based upload, proxies, cookies, and user+password authentication."

   [PHP - Libcurl package documentation manual](#)

  `$ sudo apt-get install php-curl`

- In order to create a zipped archive of user downloaded datasets, .zip library had to be installed

   [PHP - Zip package documentation manual](#)

  `$ sudo apt-get install php7.0-zip`

After installations, restart Apache

`$ sudo service apache2 restart`

11.4 Step 4: Install APCu PHP Cache to speed up PHP

APCu is a free PHP opcode cacher for caching and optimizing PHP intermediate code.
- http://php.net/manual/en/book.apcu.php

Install APCu

`$ sudo apt-get -y install php-apcu`

Restart Apache

$ sudo systemctl restart apache2

11.5 Step 5: View PHP Details & Delete info.php

Load *localhost/info.php* in your web browser to view all of the modules that have been added.

Delete the *info.php* when you don't need it anymore as it provides sensitive details of your server.

$ sudo rm -f /var/www/html/info.php

11.6 Step 6: Enable SSL Website in Apache (Security) (Beware! - caused failure in Andrews setup)

SSL / TLS is a security layer to encrypt the connection between the web browser and the server.

The following code enables *https://* by enabling the ssl module and adding a symlink in the */etc/apache2/sites-enabled* to the file */etc/apache2/sites-available/default-ssl.conf* to include it into the active apache configuration.

$ sudo a2enmod ssl

```
hagermanan@ONOTTRA673312D:/var/www/html$ sudo a2enmod ssl
[sudo] password for hagermanan:
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create s
elf-signed certificates.
To activate the new configuration, you need to run:
  service apache2 restart
```

$ sudo a2ensite default-ssl

```
hagermanan@ONOTTRA673312D:/var/www/html$ sudo a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
```

Restart Apache

$ sudo systemctl restart apache2

Test the SSL connection by loading *https://localhost/* in a web browser.
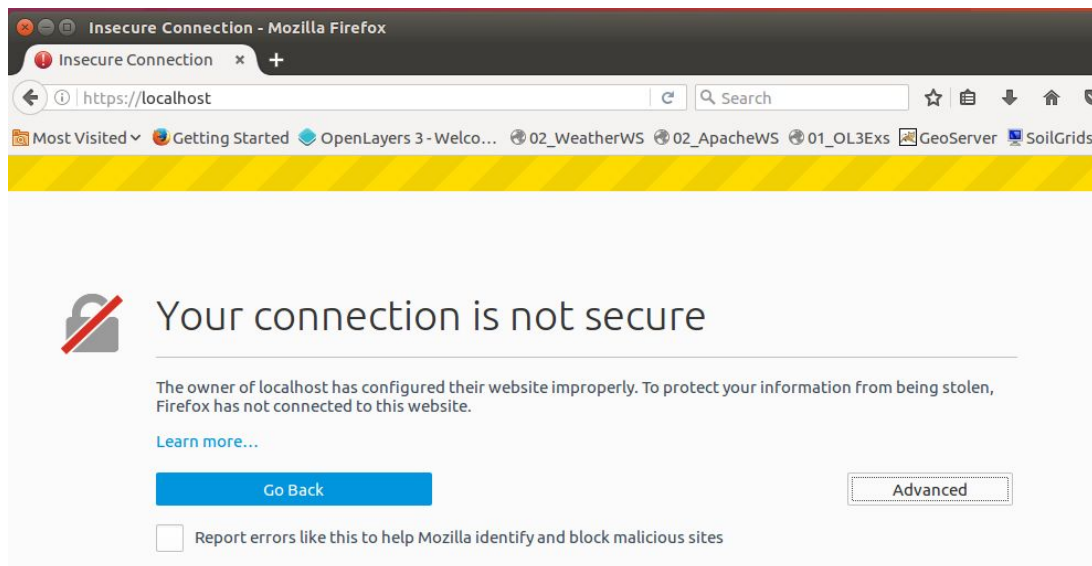
You will receive an SSL warning as the SSL certificate of the server is a "self-assigned" SSL certificate which means that the browser does not trust this certificate by default. You first have to accept the warning by adding the site as an exception.

Click Advanced

Click Add Exception
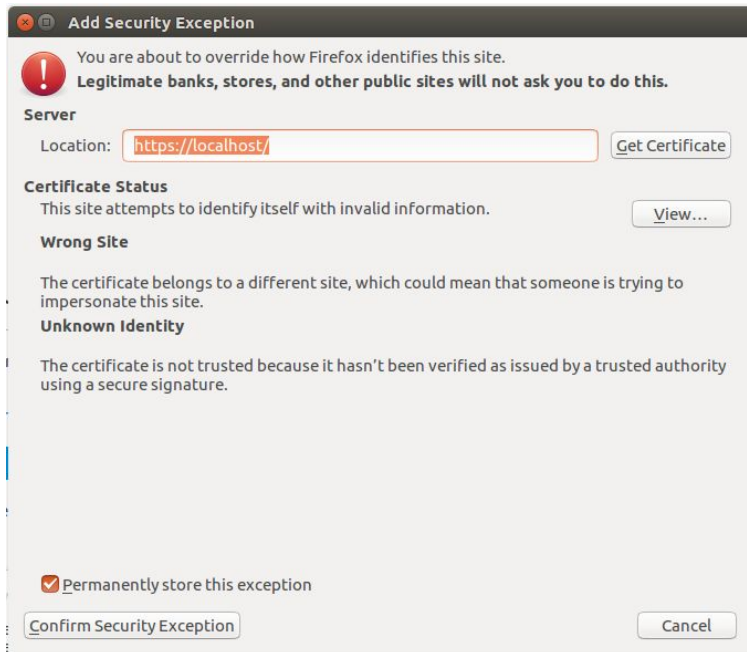
Click Confirm Security Exception

This will bring you to the apache default page.

The lock symbol in front of the URL in the browser shows that the connection is encrypted but it not a verified SSL connection and that you have added an exception for this site.

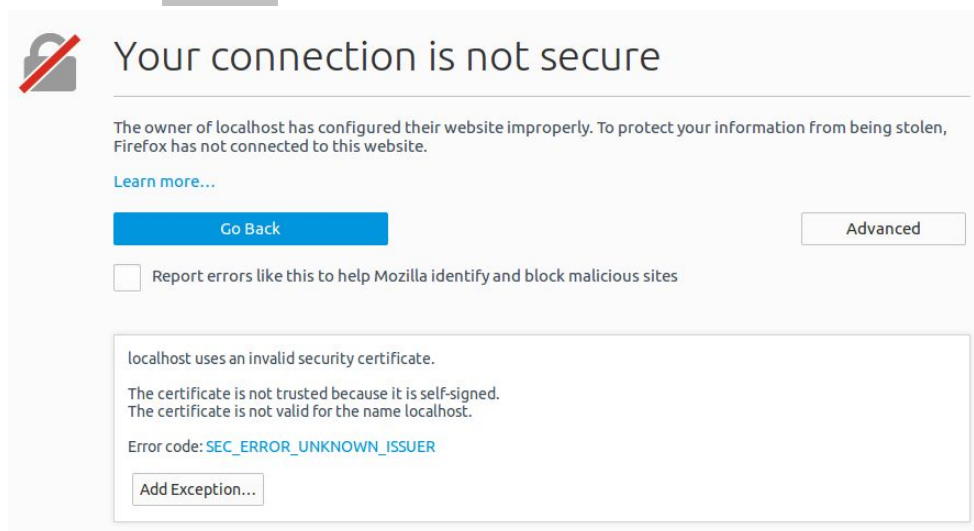This was not performed for this system configuration, but in order to get rid of the warning, replace the self-signed SSL certificate */etc/ssl/certs/ssl-cert-snakeoil.pem* with an officially signed SSL certificate from an SSL Authority.

11.7 Step 7: Allow File Uploads

Update the *php.ini* files so that the *file_uploads* directive is set to *On* (http://www.w3schools.com/php/php_file_upload.asp) .

$sudo nano /etc/php/7.0/apache2/php.ini

$sudo nano /etc/php/7.0/cli/php.ini

```
;;;;;;;;;;;;;;;;;
; File Uploads ;
;;;;;;;;;;;;;;;;;

; Whether to allow HTTP file uploads.
; http://php.net/file-uploads
file_uploads = On
```

11.8 Step 8 : Give Apache Permission to Specified Directories

If a webpage is to interact with directories on the server, the Apache user needs to have the appropriate permissions to those directories. Instead of making the entire web root folder available to Apache (very dangerous), provide specified directories for web pages to interact with the server.

The Apache user and group are both *www-data* . Any directories that need to allow access to Apache should be changed from *root* to *www-data* .

```
drwxr-xr-x 2 root root   4096 Dec 12 16:31 ClientData
```

```
drwxr-xr-x 2 www-data www-data  4096 Dec 12 16:31 ClientData
```

The next 2 lines of code are examples of allowing a webpage to read and write files to the *ClientData* directory as well as having a separate *uploads* directory for client file uploading

```
$sudo chown -R www-data /var/www/html/ClientData
$sudo chown -R www-data /var/www/html/ClientData/uploads
```

11.9 Option to Install phpMyAdmin

phpMyAdmin is a web interface through which MySQL databases can be managed.

Follow intructions under *Install phpMyAdmin* on this site:
https://www.howtoforge.com/tutorial/install-apache-with-php-and-mysql-on-ubuntu-16-04-lamp/#get-mysql-mariadb-support-in-php

This was not done for this system configuration.

11.10 Debugging PHP In Browser

Debugging PHP code is notoriously difficult without an integrated debugging environment in your IDE. Without that, a solution had to be found to enable view of error logs, variable contents, traces, etc.

The solution found was FirePHP, an extension to FireFox web browser that logs to the extension, FireBug, web developer console.

1. Install FireBug web developer tools extension to FireFox, Add-Ons -> Extensions -> FireBug. This is discontinued since the built in Dev Tools do the trick, but FirePHP is not yet updated to the new browser integrated dev toolkit.
2. Install FirePHP extension and enable all panels, Add-Ons -> Extensions -> FirePHP.
3. Download the FirePHPCore library from the FirePHP Github repo and include the fb.php file in your application page. Only the fb.php file is needed. To keep things clean, I put this alongside other required scripts in /var/www/html/tests/Script. Simply download the FirePHPCore directory from the repo (listed below), place in your local script folder, and include in your application using *require('<script path>');* at the top of the page.

Syntax for console logging, as well as a well rounded introduction to FirePHP, can be found at the following links.

Links:
> Debugging with FirePHP - Intro
> FirePHPCore at GitHub

12 BRACKETS

Brackets is an open sourced code editor that allows for quick viewing and access of your full project directory, dual page editing, and organized flow. More information can be found at the following site, and installed with the following 3 commands.

website: http://sourcedigit.com/19895-how-to-install-brackets-1-7-editor-in-ubuntu-16-04-via-ppa/

commands:

$ sudo add-apt-repository ppa:webupd8team/brackets

$ sudo apt-get update

$ sudo apt-get install brackets

Some Handy Extensions include:

– Beautify (auto-format html, css, javascript)
– PHP SmartHints (code hints for PHP)
– Emmet (super quick html shortcuts)

13 Migrating Documents To Application Server

13.0 Transferring working files to Daedalus

You will likely want to develop the application locally as running a GUI off of Daedalus presents some lag issues. Migrating your working files over presents a new set of issues.

1. After upload, all URLs must be changed from local 127.0.0.1 to Ulysses.agr.gc.ca
2. After upload, all data layers must be updated to direct to data stored on Ulysses.
   a. Layer references made in PHP or HTML must reflect layers found on Ulysses. GeoServer Workspace and Layer creation is not covered under this setup document. See GeoServer useful links
3. Upload of the local development directories into Daedalus's html root does not work, so a workaround has to be made:
   a. Compress your html development directory in a .zip, including all script dependencies and style sheets. Zip everything required for transfer.
   b. Open an sftp connection to Daedalus:

   $ sftp <your user name>@daedalus.agr.gc.ca

   c. Upload the .zip file into your home directory on Daedalus:

   Check current directory -> $ pwd

   Move to home directory -> $ cd /home/<your user name>

   Upload .zip -> $ put <.zip file>

   d. Exit sftp, then log onto Daedalus into your home directory and unzip your local .zip directory into Daedalus's HTML root:

   Log onto server -> $ ssh <your user name>@daedalus.agr.gc.ca

   unzip to server web root. In this case, we want the unzipped files in the index directory, but change to suite your case. * The "-o" sets to overwrite with no prompt

    -> $ sudo unzip -o <your zip file> -d /var/www/html/index

   *** Note that overwriting files with the new directory will revert any changes you made to URLs and layer names.

13. 1 Changing permissions for editing

When developing locally, it's likely that the owner of your files will be 'root'. When we log onto the server as a user, we do so under our user name. In order to make edits to application code on the server, you will likely have to change the owner of the file which you are trying to save under your user name. You can do so by entering this command (from same level as directory you wish to change)

sudo chown -R www-data:<your user name> <directory path>

13.2 Automating temp file cleanup

The temp file with get filled up with test files and logs unless it is cleaned out regularly. To do so, we use Ubuntu's CronJob functions. From here, we can set up regular running of commands or scripts. In this case, we clean out the temp file everyday at noon.

create new cronjob -> $ sudo crontab -e

enter command on first open line -> 0 12 * * * rm -f /var/www/html/<path to your temp file>