Parts of the guide have been updated from the previous installation documentation from Alvaro Bustos – greenes. -Thanks

This guide has been written step by step with screenshots to aid in the successful build of OM.

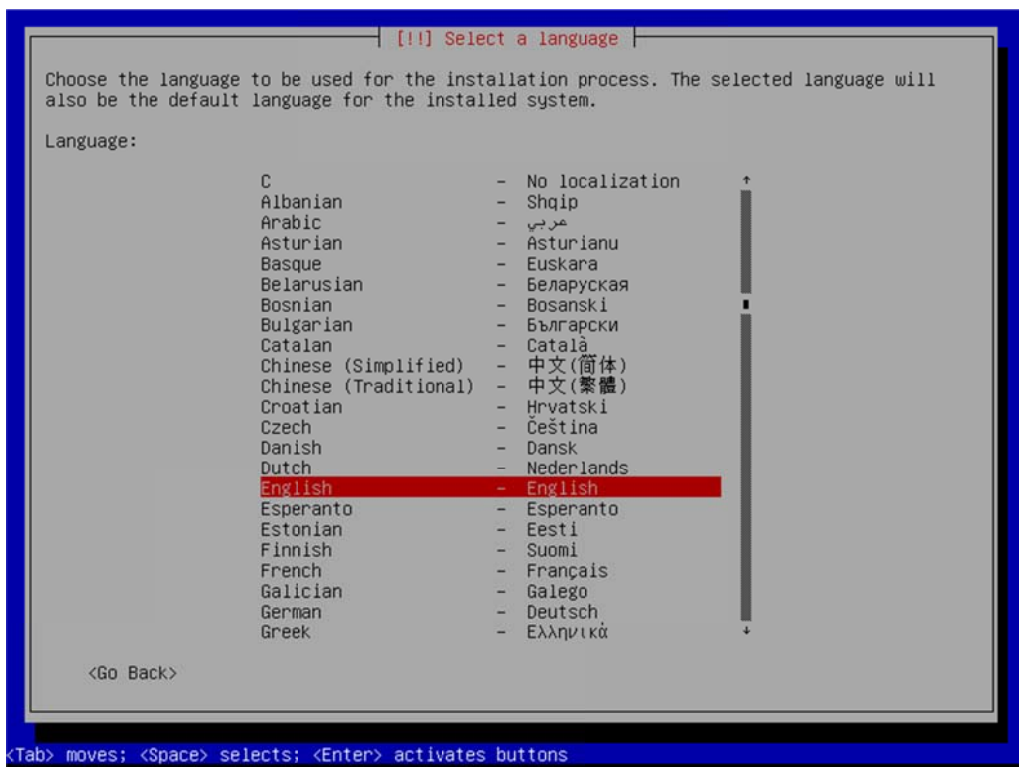# SSL and Reverse proxy steps have been added but are optional.

N.B – When copying and pasting commands please check that symbols and character returns are correctly copied across.

Stephen Cottham   **08/02/2013**
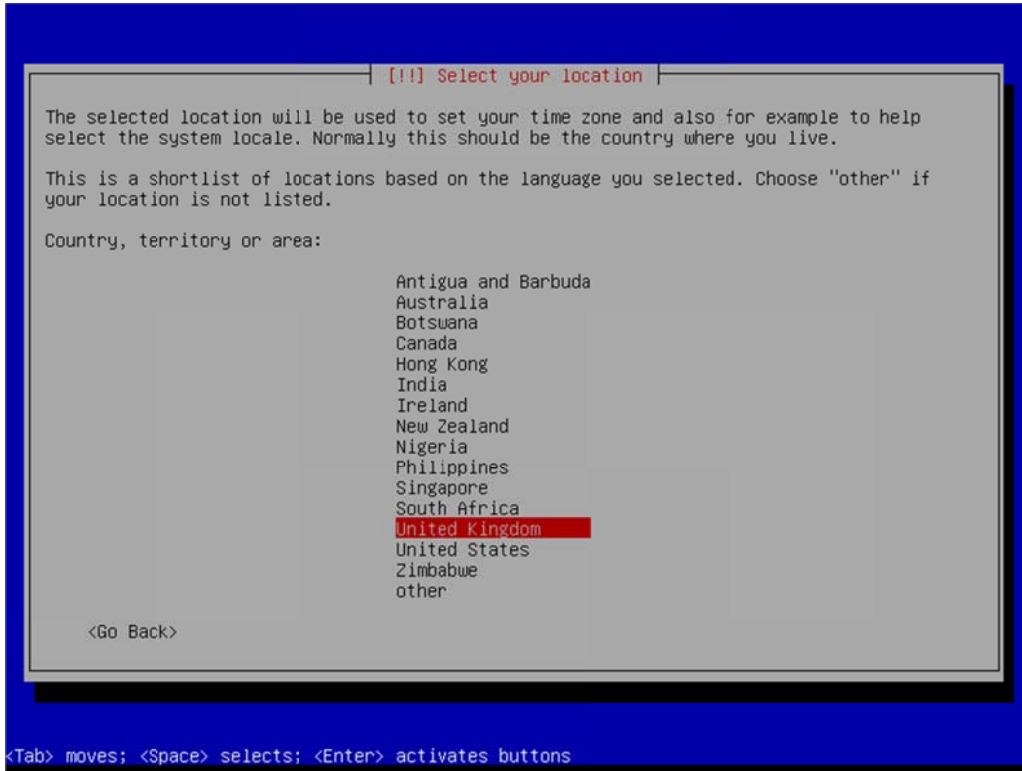
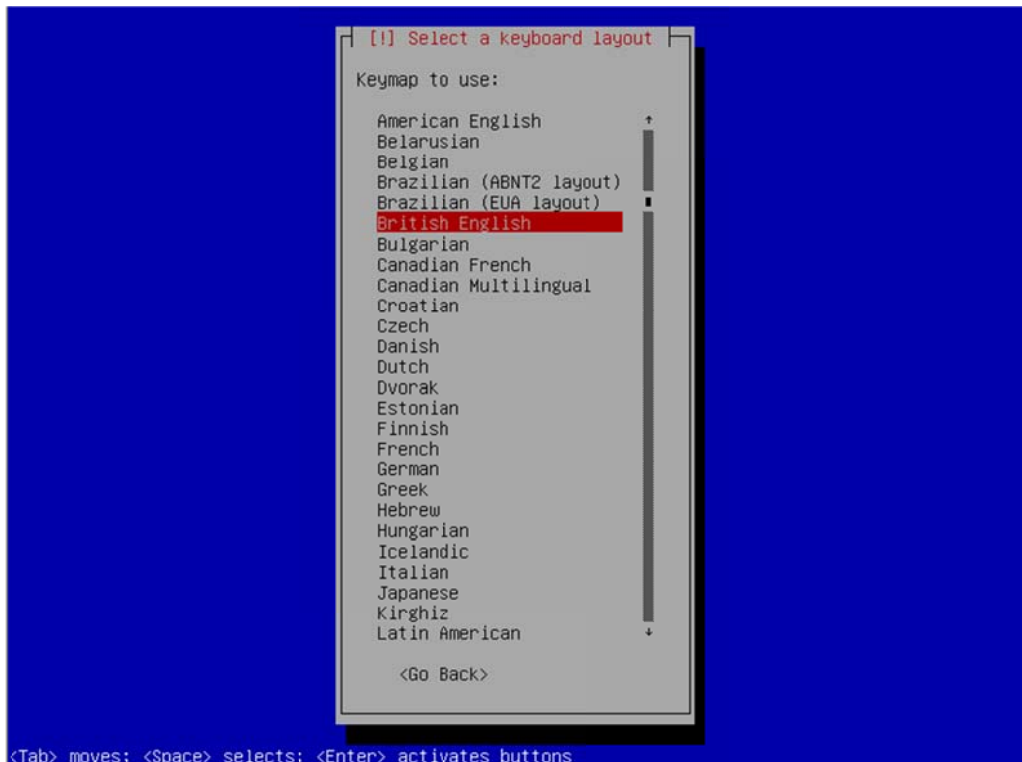## Installing Debian (Minimal Headless System)

Step 1: - Base System

Choose 64 Bit install

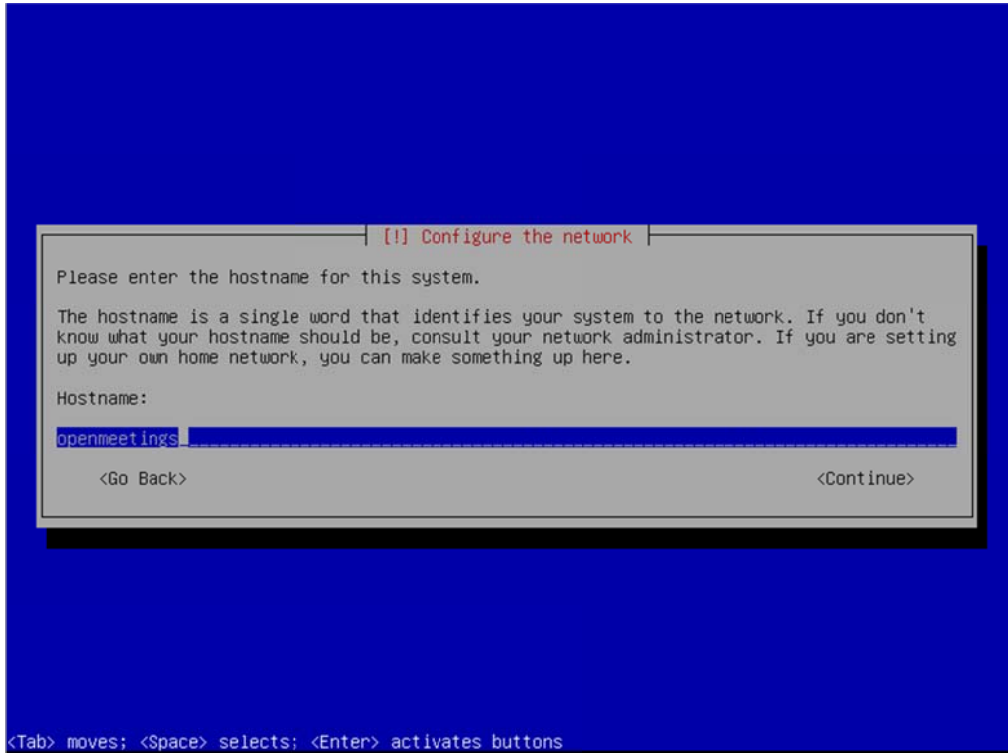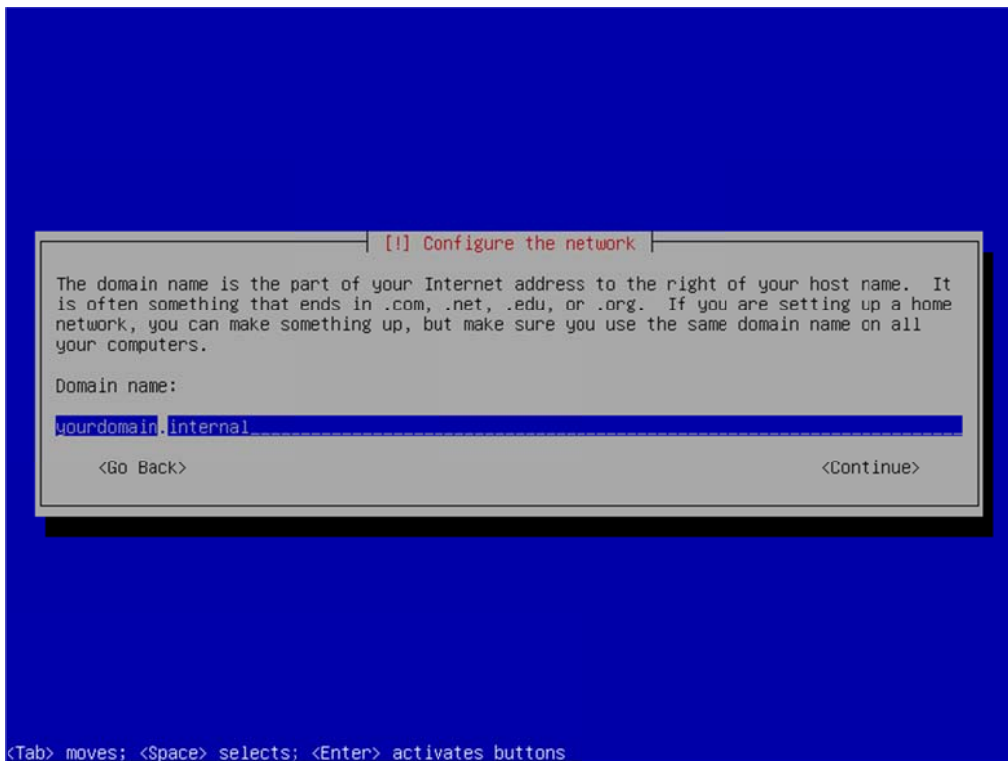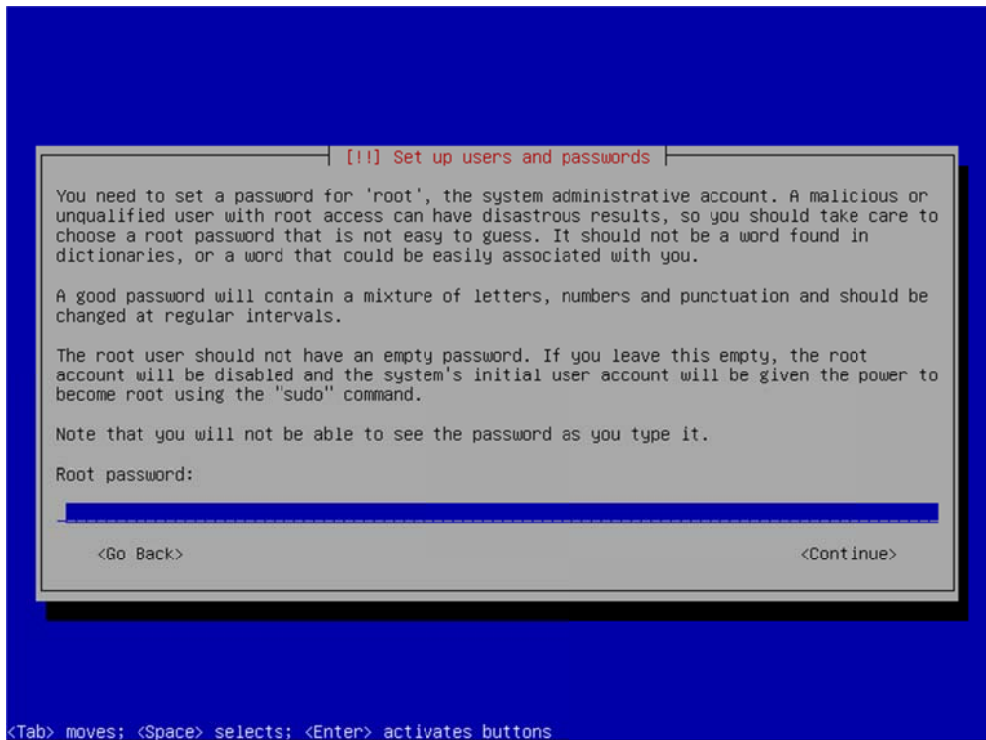Choose English

Choose "United Kingdom"
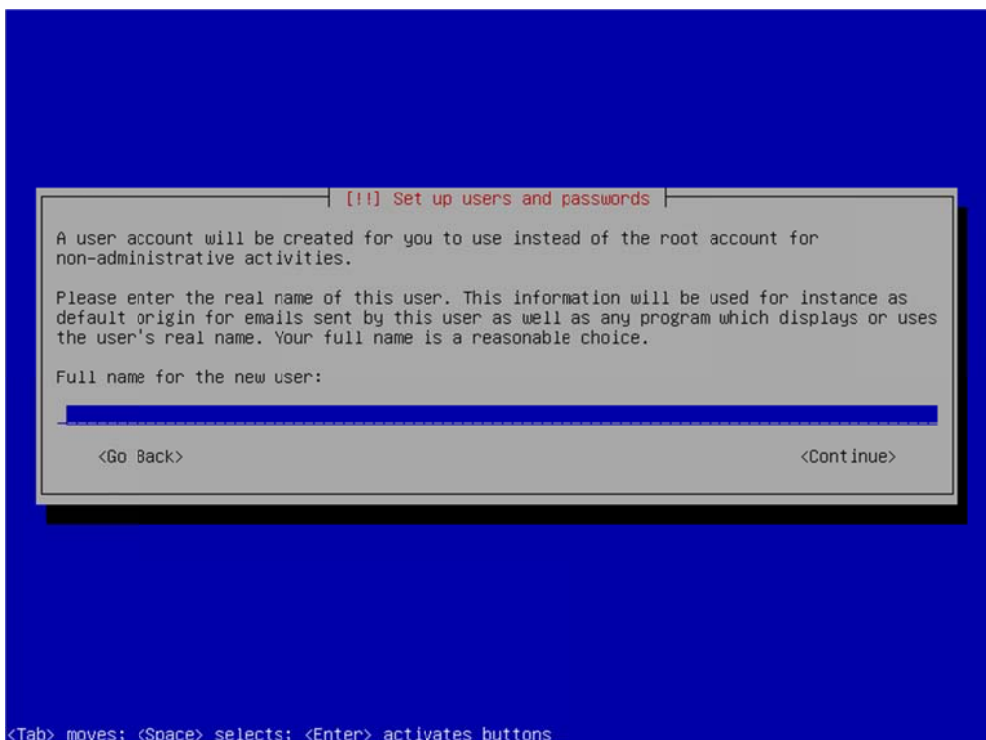

Choose "British English"

Set the hostname, in this case its "openmeetings"
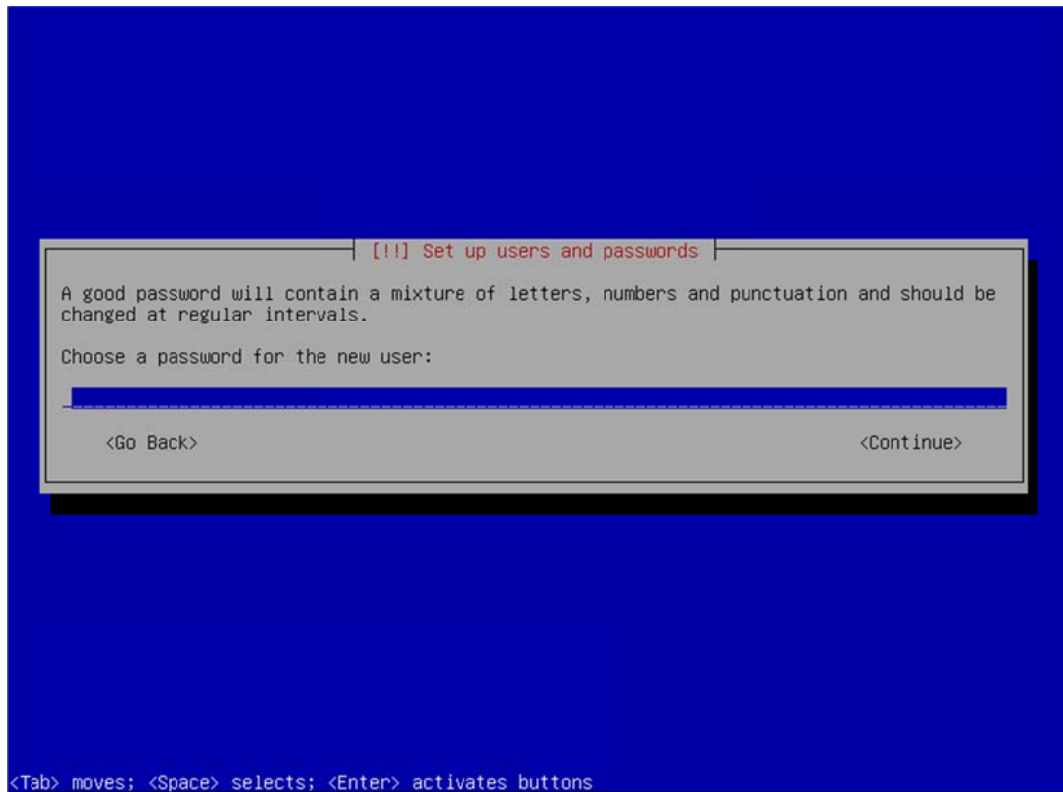

Set your domain, in this case we have used "yourdomain.internal"

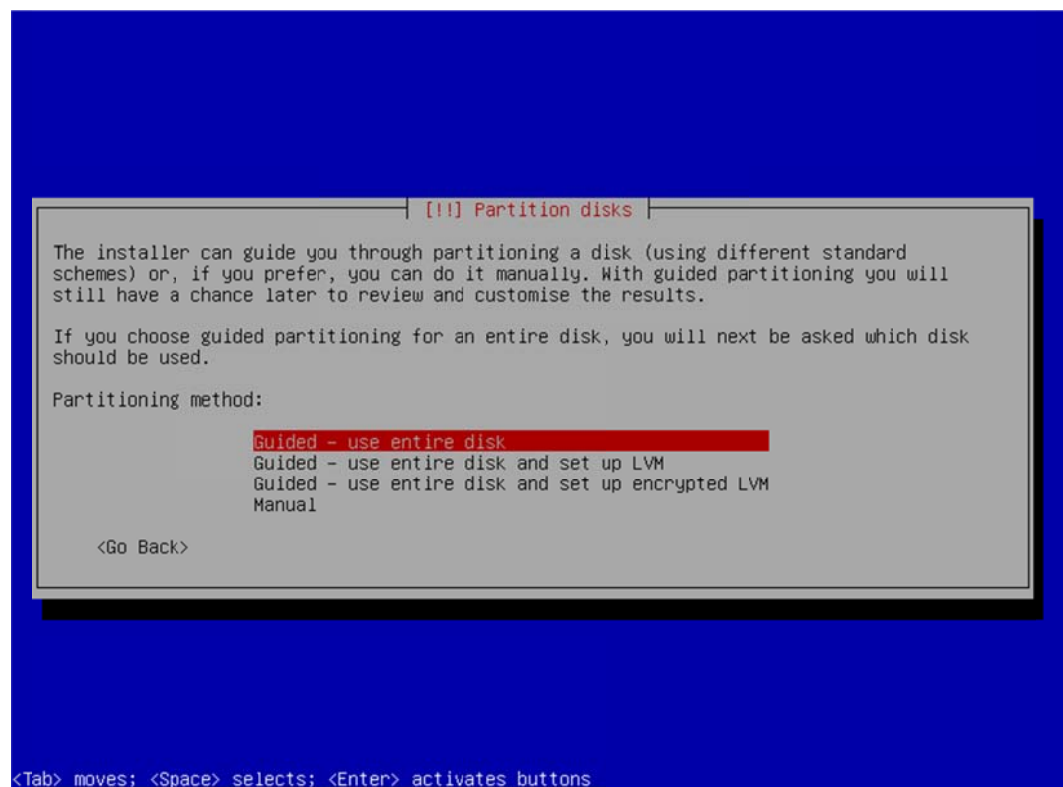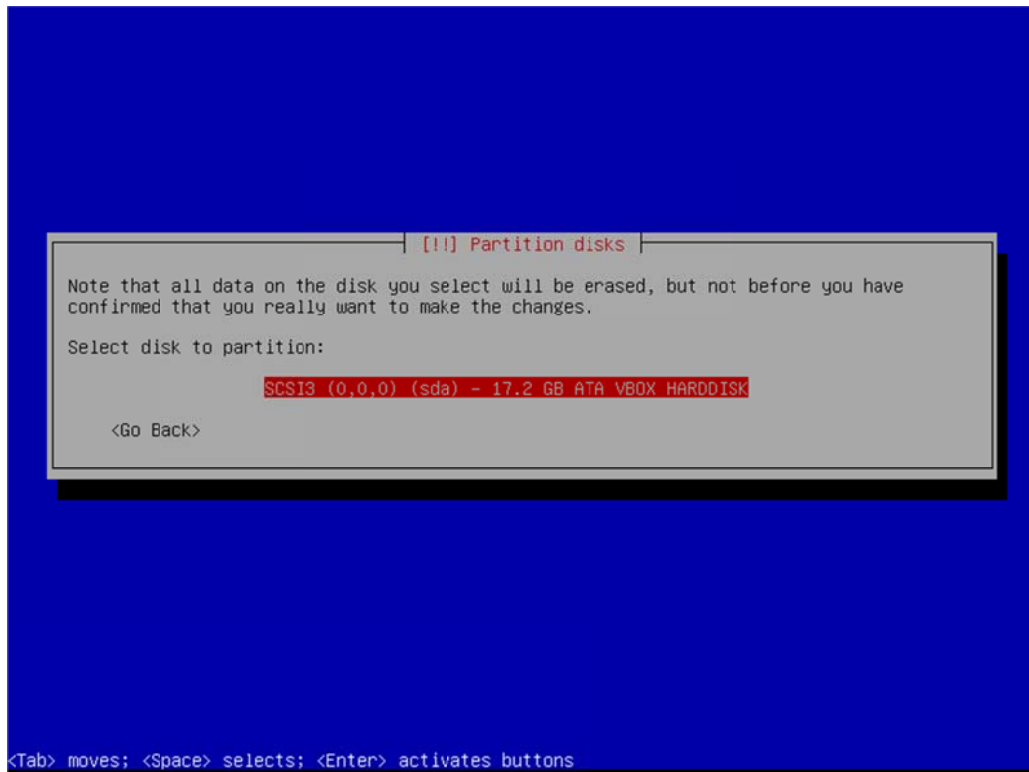Set the root password.

Create new user for server (Non-priv)

```
┤ [!!] Set up users and passwords ├

A good password will contain a mixture of letters, numbers and punctuation and should be
changed at regular intervals.

Choose a password for the new user:

_____

    <Go Back>                                                    <Continue>
```
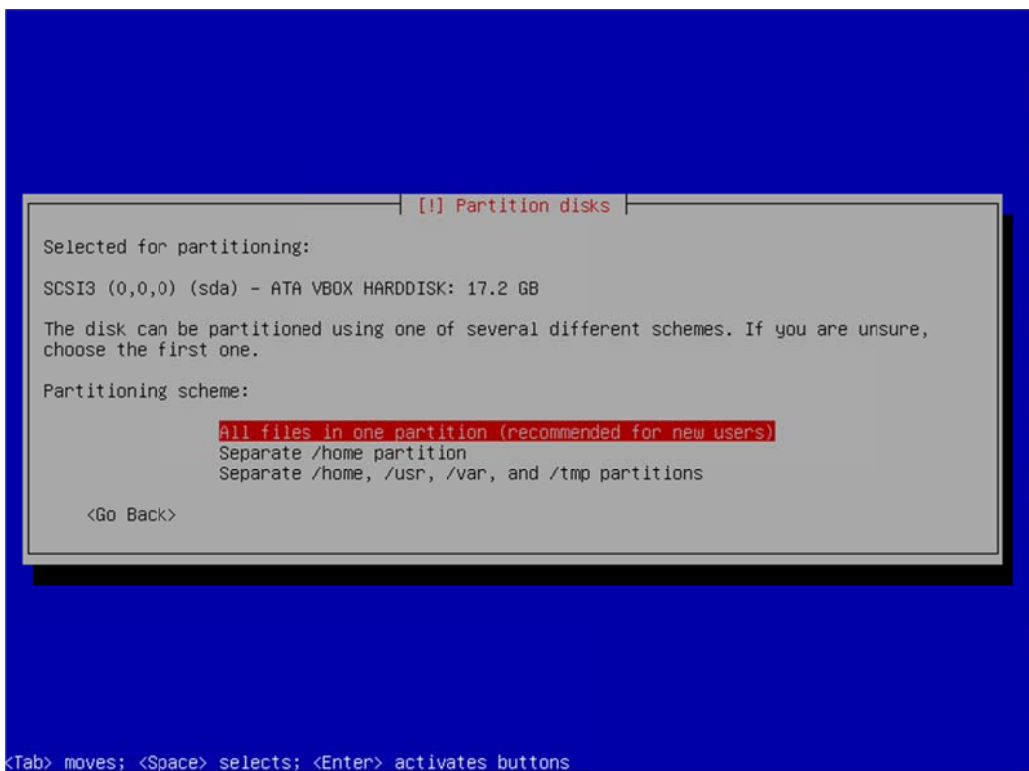
```
<Tab> moves; <Space> selects; <Enter> activates buttons
```

Set password for new user


```
┤ [!!] Partition disks ├

The installer can guide you through partitioning a disk (using different standard
schemes) or, if you prefer, you can do it manually. With guided partitioning you will
still have a chance later to review and customise the results.

If you choose guided partitioning for an entire disk, you will next be asked which disk
should be used.

Partitioning method:

                Guided - use entire disk
                Guided - use entire disk and set up LVM
                Guided - use entire disk and set up encrypted LVM
                Manual

    <Go Back>
```

```
<Tab> moves; <Space> selects; <Enter> activates buttons
```
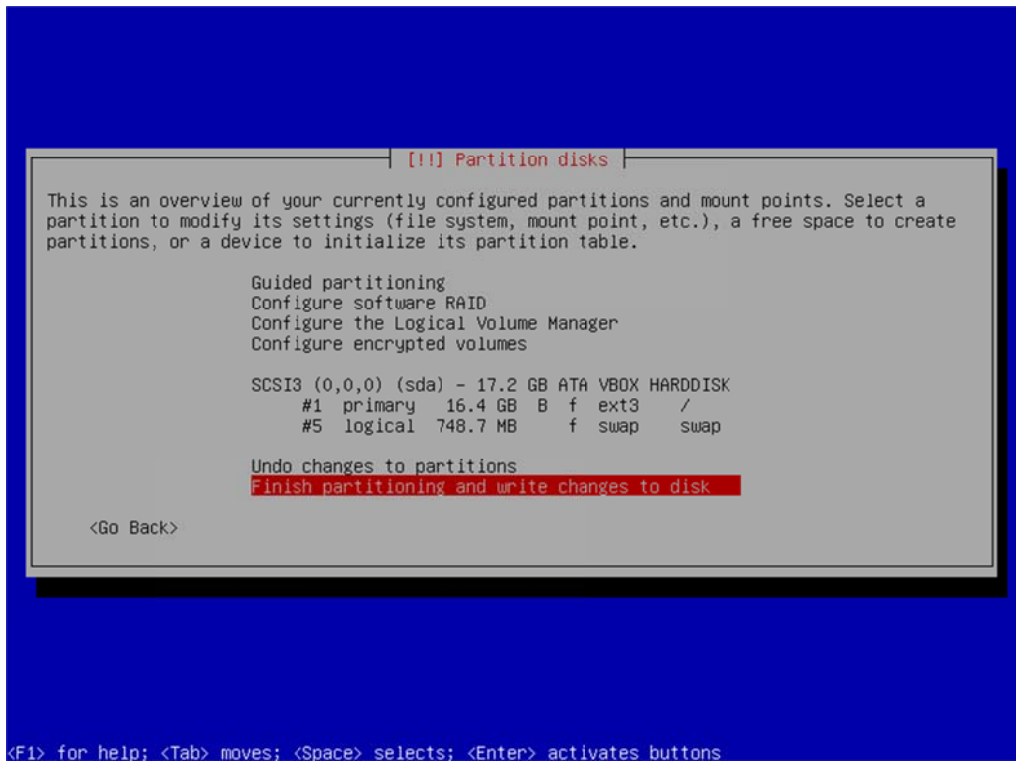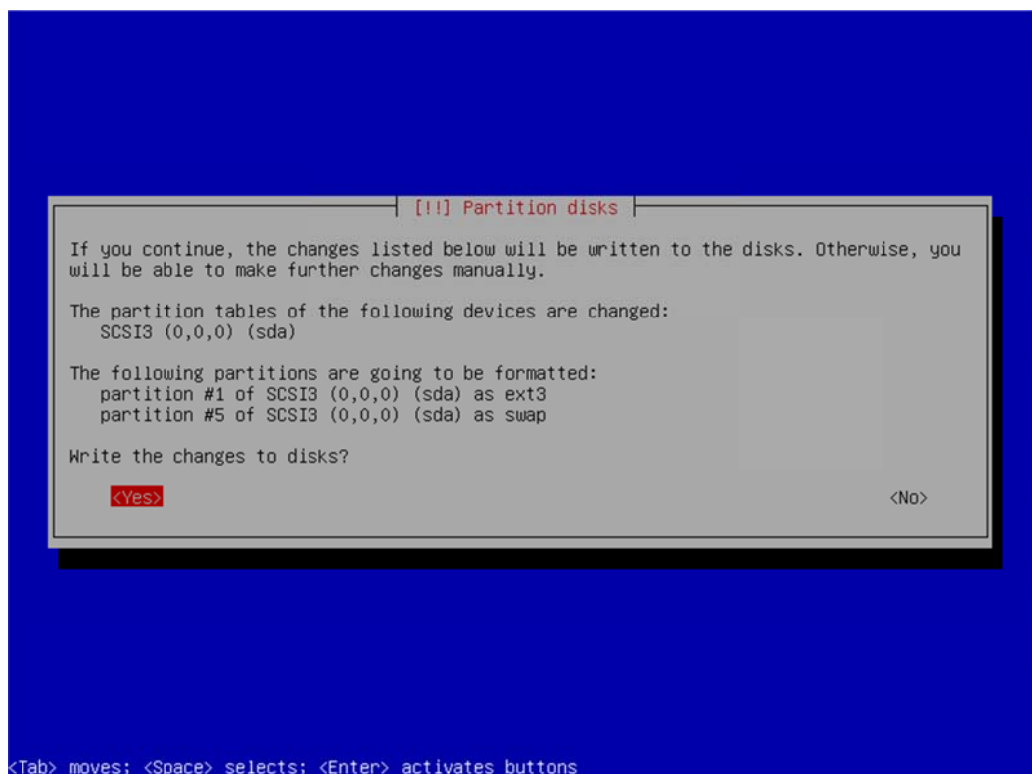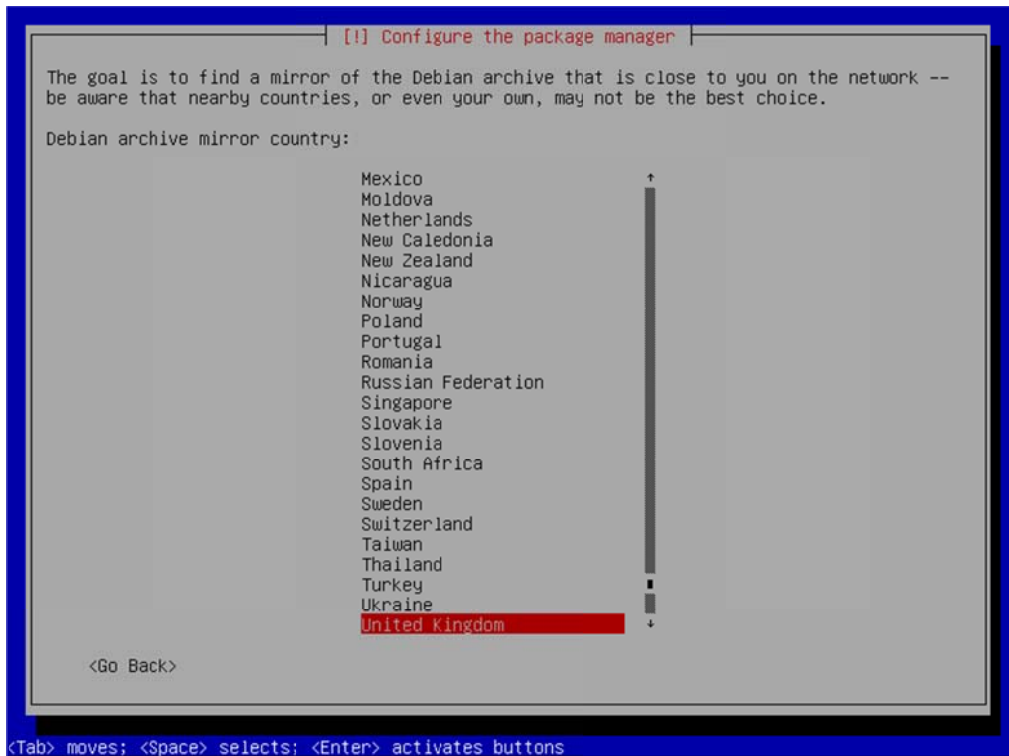
Use guided – entire disk

Select Disk to partition

Choose "All files in one partition"

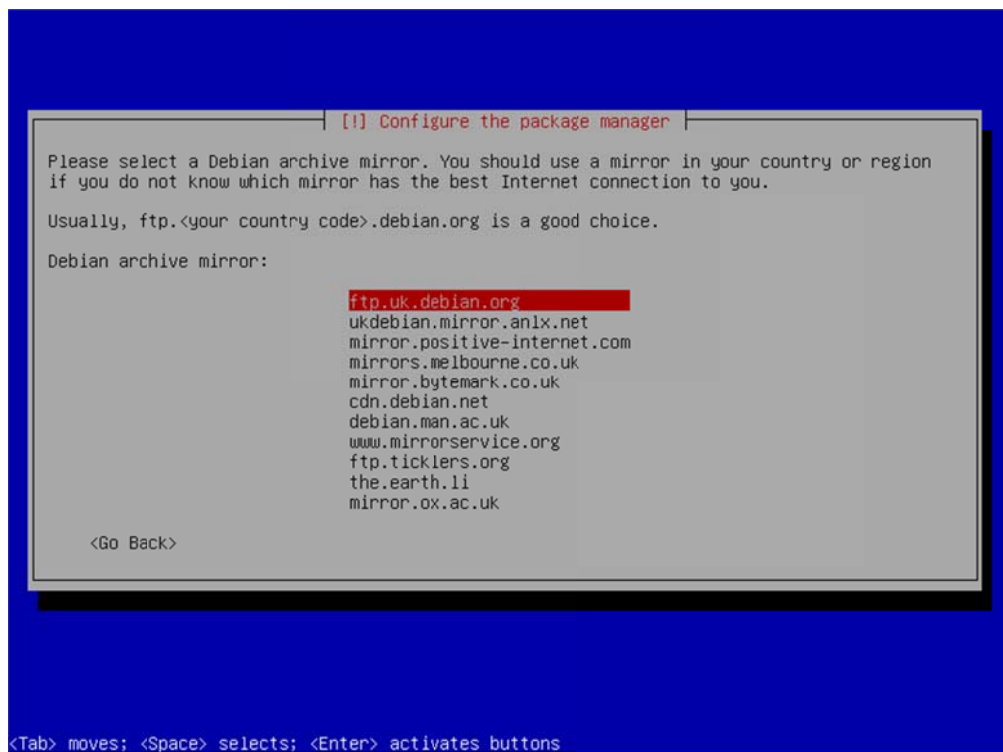Choose "Finish partitioning and write changes to disk"

And finally choose "yes"

Choose Debian archive – in this case we are using "United Kingdom"

Any archive will do closest to you; in this case we are using ftp.uk.debian.org
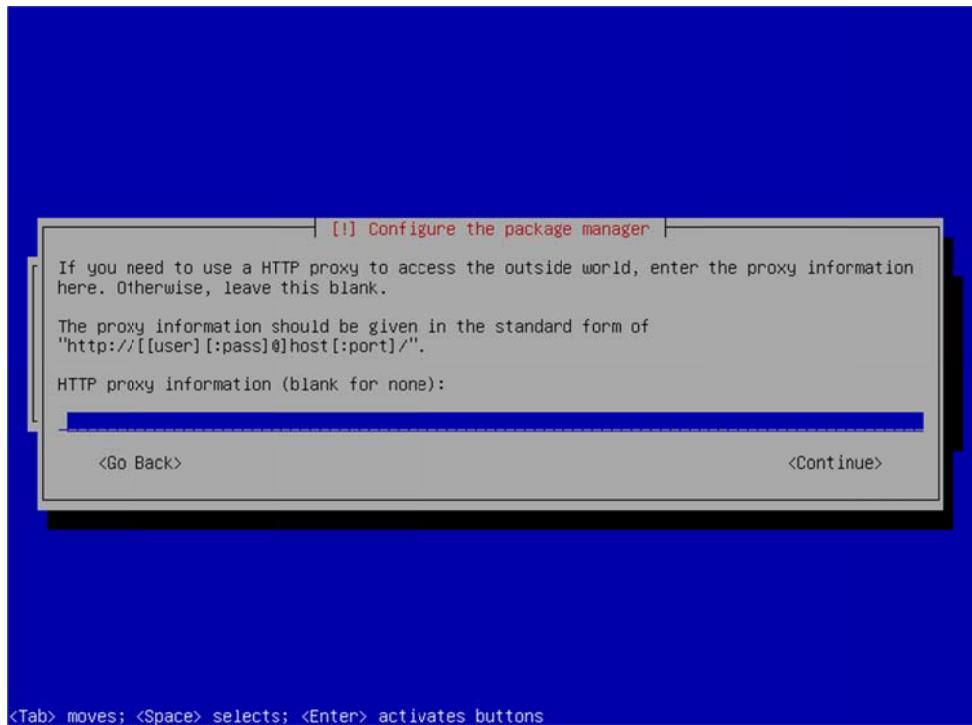
If you use a proxy server then add the details here, if you have full outbound access then just choose continue.
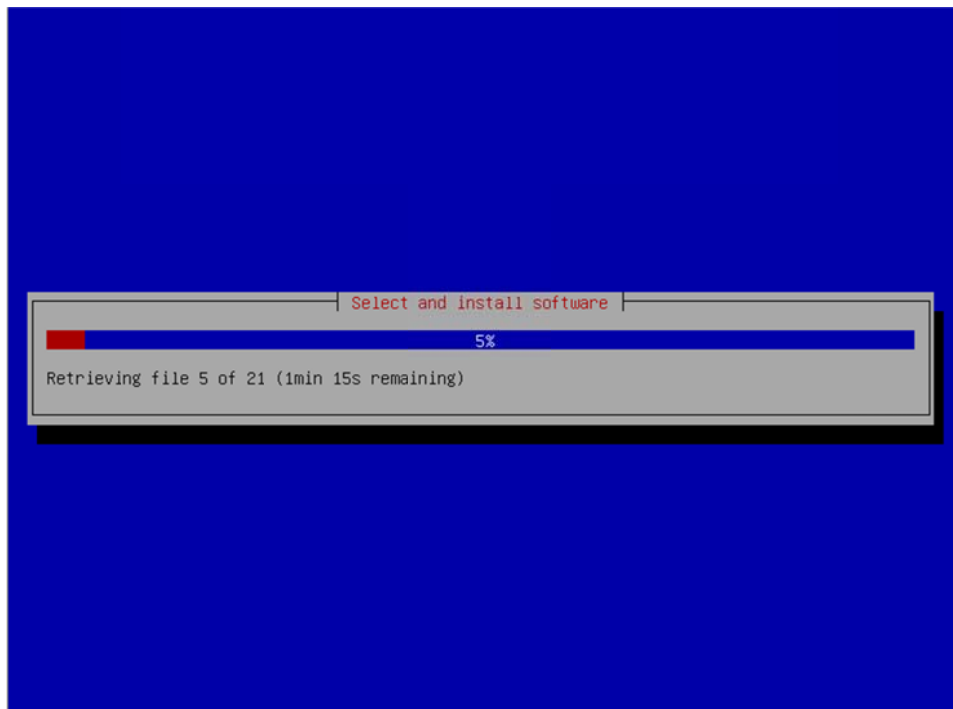
"apt" will now update the local repository information.
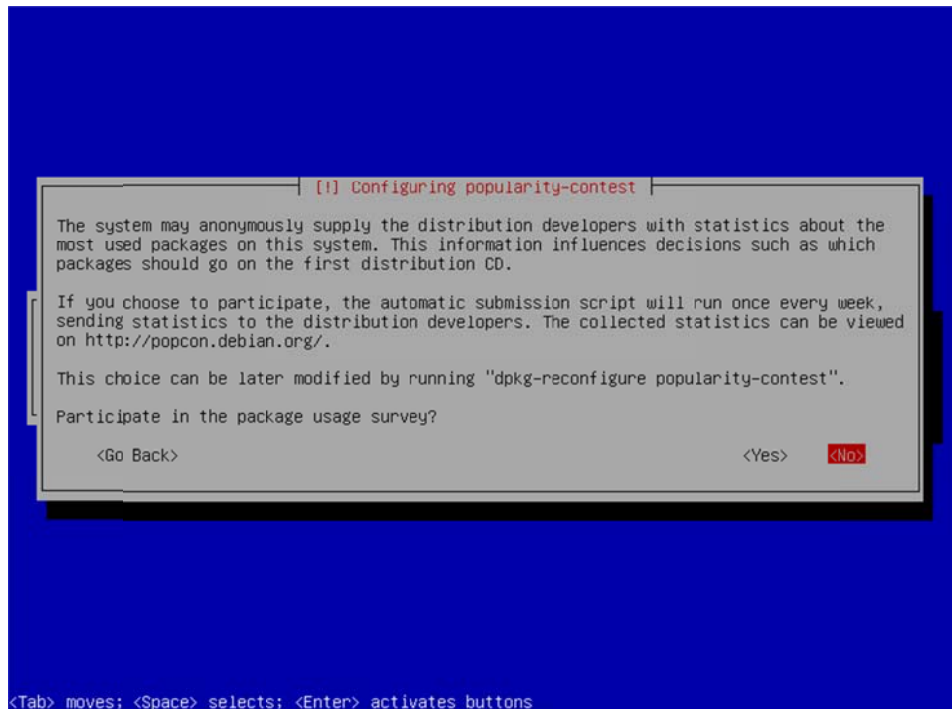
Choose not to participate in the survey.
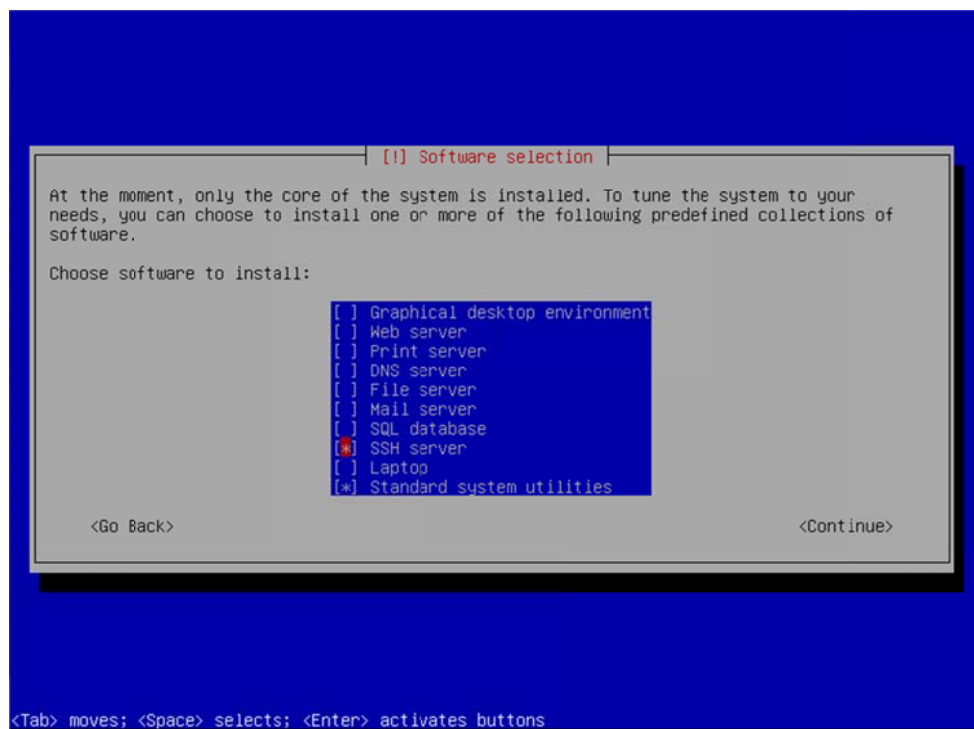
Choose **only** SSH Server and Standard System utilities.
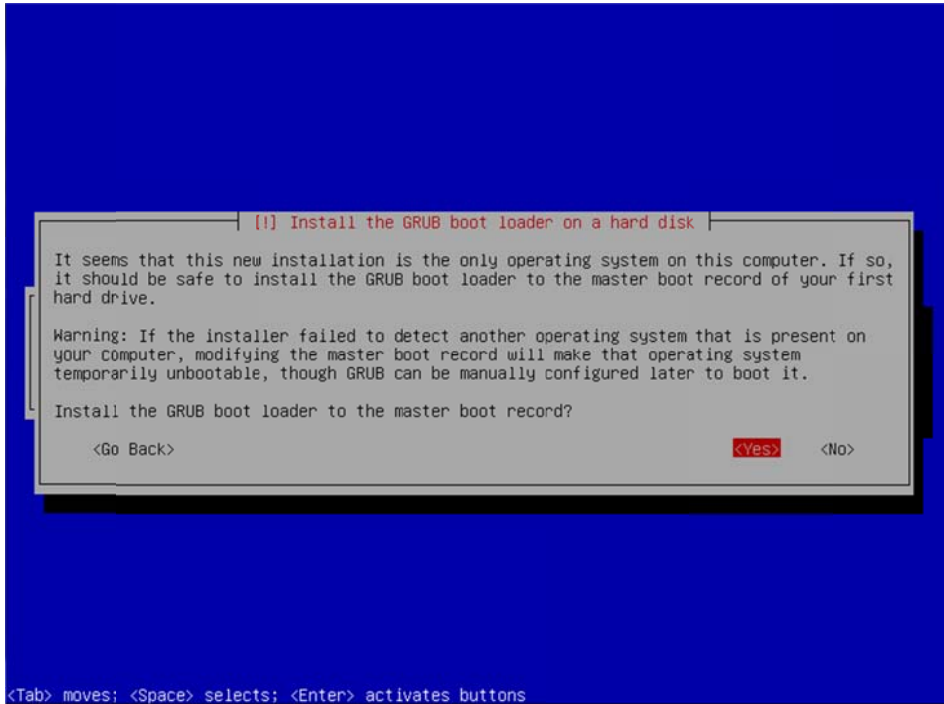
Choose Yes to install Grub.
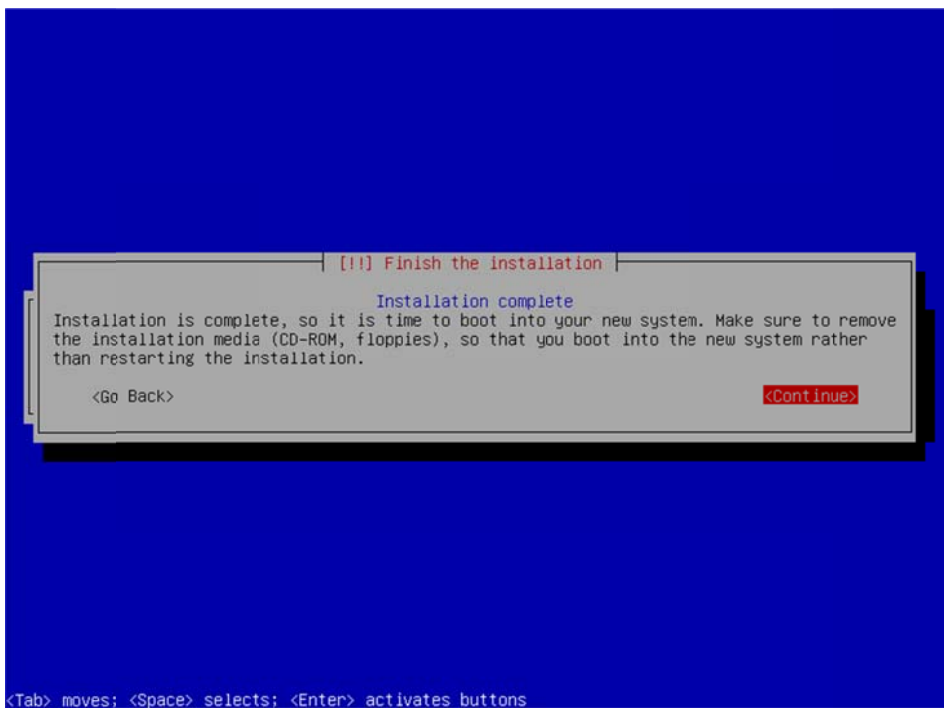
Base install has now completed, choose continue to reboot into your new system.

Step 2: - Setup SSH Environment

```
Setting kernel variables ...done.
Configuring network interfaces...done.
Starting portmap daemon....
Starting NFS common utilities: statd.
Cleaning up temporary files....
Setting console screen modes.
Skipping font and keymap setup (handled by console-setup).
Setting up console font and keymap...done.
INIT: Entering runlevel: 2
Using makefile-style concurrent boot in runlevel 2.
Starting NFS common utilities: statd.
Starting portmap daemon...Already running..
Starting enhanced syslogd: rsyslogd.
Starting VirtualBox AdditionsVBoxService: 3.2.10_OSE r66523 started. Verbose lev
el = 0
.
Starting ACPI services....
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.
Starting OpenBSD Secure Shell server: sshd.
Starting MTA: exim4.

Debian GNU/Linux 6.0 openmeetings tty1

openmeetings login: _
```

You should now be at the following screen, the next steps are easier done from a remote desktop using an SSH client such as putty. – But first we need to know our IP address, in most cases this was issued by your DHCP server (unless you specified manual network setup during install)

To find your IP address, first logon to your physical machine using root, then issue the following command:

ifconfig

This will show the following screen:

```
root@openmeetings:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:22:1d:a1
          inet addr:10.17.23.3  Bcast:10.17.23.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe22:1da1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:433 errors:0 dropped:0 overruns:0 frame:0
          TX packets:30 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:41590 (40.6 KiB)  TX bytes:2756 (2.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:560 (560.0 B)  TX bytes:560 (560.0 B)

root@openmeetings:~# _
```

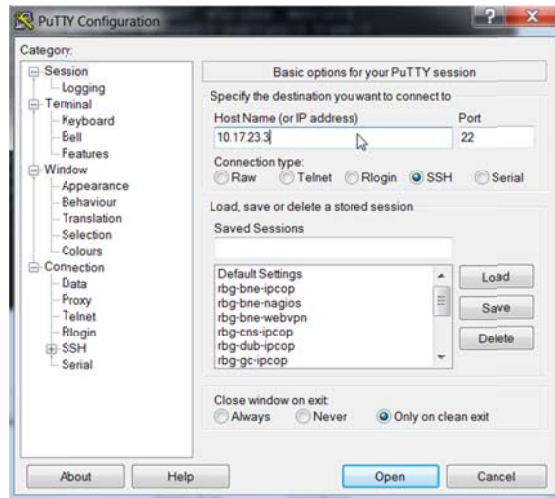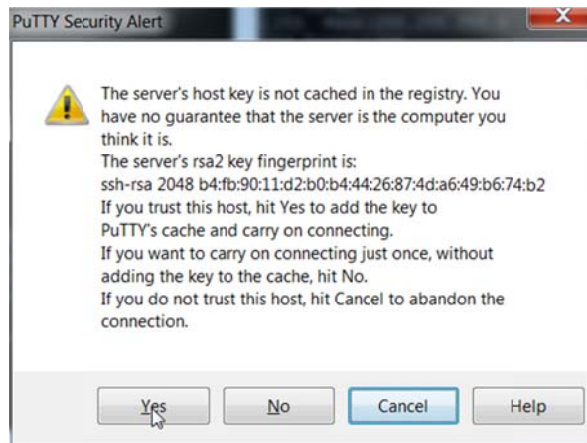You can see the IP Address in this case is 10.17.23.3 (Interface eth0)
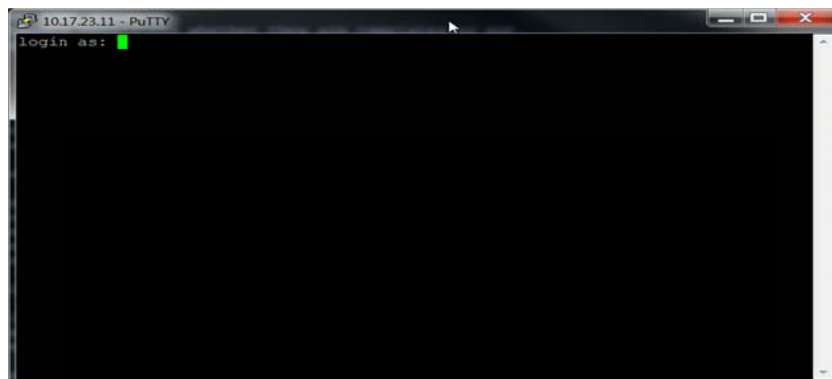
You can now log off of the server.

From your desktop machine open your SSH client, in this case we will be using the putty client to connect to our new Server.

Enter the details and choose open

The first log on you will receive this message; you can choose yes here and accept the key.

Now log in with your root credentials.

Step 3: Install Dependent software

Firstly we need to add some repos to apt to get the required Java version, so first issue this command:

vi /etc/apt/sources.list

This will open the following file in vi: (you can use whatever file editor you are comfortable with)

Now at the top of this file add the following entries:

deb http://security.debian.org/ squeeze/updates main contrib non-free
deb-src http://security.debian.org/ squeeze/updates main contrib non-free
deb http://ftp.debian.org/debian/ squeeze main contrib non-free
deb-src http://ftp.debian.org/debian/ squeeze main contrib non-free
deb http://ftp.debian.org/debian/ squeeze-updates main contrib non-free
deb-src http://ftp.debian.org/debian/ squeeze-updates main contrib non-free
deb http://ftp2.de.debian.org/debian squeeze main non-free
deb http://deb-multimedia.org squeeze main

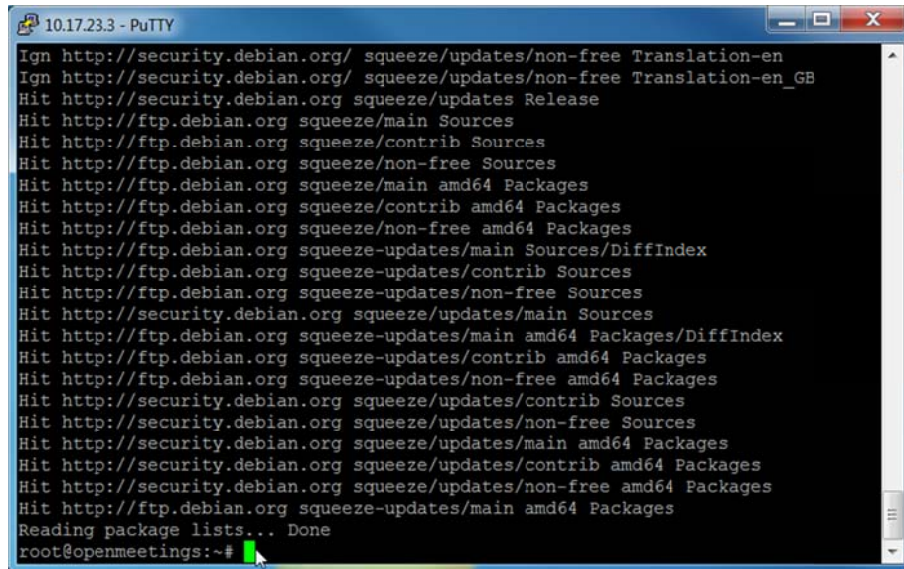Example above – now save the file

To update the repos we need to issue the following command:

apt-get update

Once that has completed you will be here:

Let's install the needed software by issuing the following commands: (Please accept the **sun-java6-jre** license agreement during install)

apt-get install deb-multimedia-keyring
apt-get install sun-java6-jdk -y
apt-get install openoffice.org-writer openoffice.org-calc openoffice.org-impress -y
apt-get install openoffice.org-draw openoffice.org-math imagemagick gs-gpl -y
apt-get install libgif-dev xpdf libfreetype6 libfreetype6-dev libjpeg8 libjpeg62 libjpeg8-dev -y
apt-get install g++ libjpeg-dev libdirectfb-dev libart-2.0-2 libt1-5 zip unzip bzip2 -y
apt-get install subversion git-core checkinstall yasm texi2html libfaac-dev libfaad-dev -y
apt-get install libmp3lame-dev libsdl1.2-dev libx11-dev libxfixes-dev -y
apt-get install libogg-dev sox libvorbis0a libvorbis-dev libgsm1 libgsm1-dev -y
apt-get install libxvidcore4-dev zlib1g-dev libfaad2 flvtool2 lame -y

Step 4: - Create mysql DB for OM

Now we need to install MYSQL, issue this command (In this case username and password are openmeetings : ompassword)

apt-get install mysql-server -y
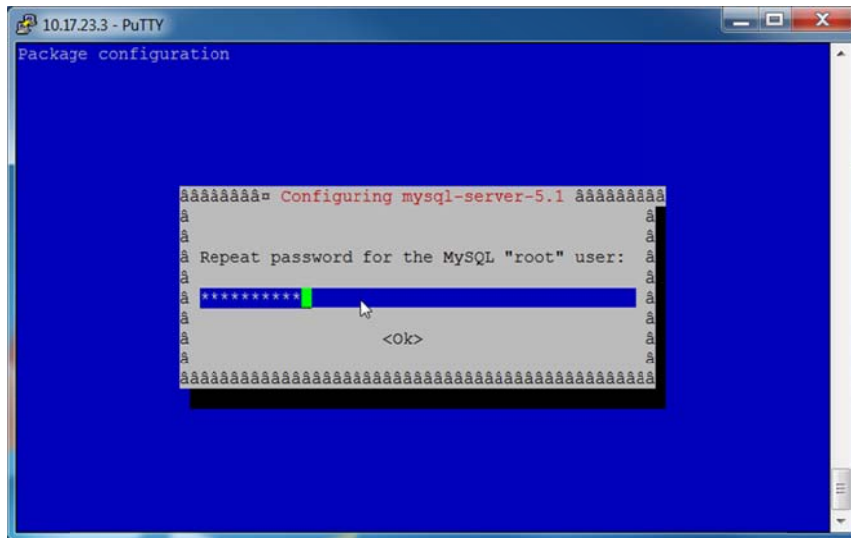
Enter the password as before "ompassword" and choose ok.

Now let's crate the needed DB's for OM 2.x

Issue these commands:
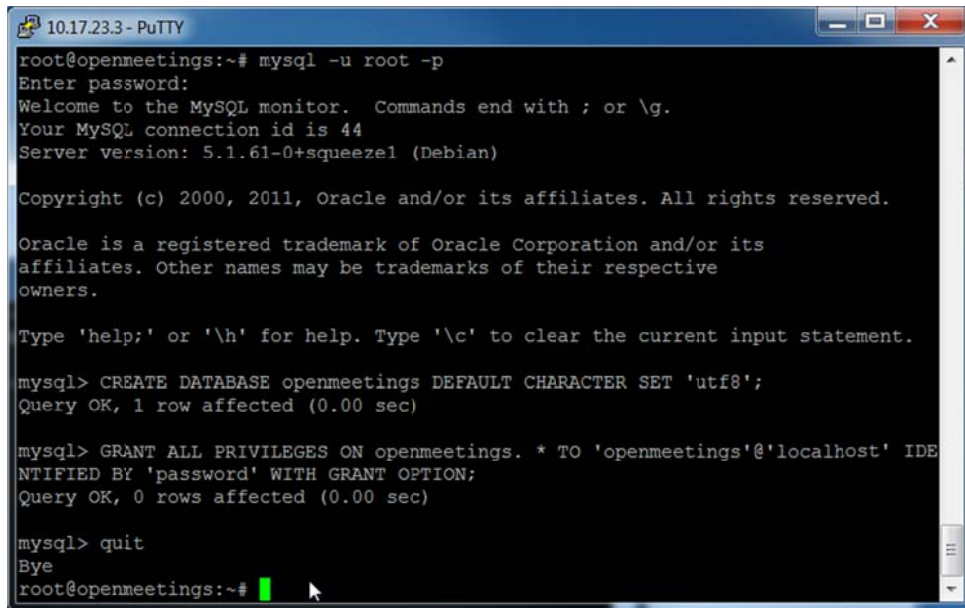
mysql -u root -p
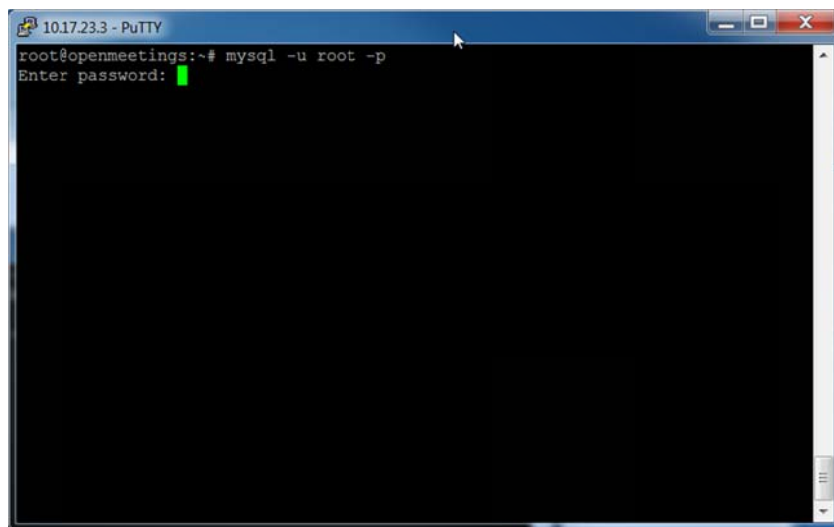
Enter password "**ompassword**"

Now issue these: (Assuming username **openmeetings** and password = **password**)

CREATE DATABASE openmeetings DEFAULT CHARACTER SET 'utf8';
GRANT ALL PRIVILEGES ON openmeetings.* TO 'openmeetings'@'localhost' IDENTIFIED BY
'password' WITH GRANT OPTION;
quit

Successful DB creation shown above.

Step 5: Compile Install SWFTools (2012-10-15-1307)

Now let's create a temporary working area by issuing these commands:

mkdir -p /usr/adm
cd /usr/adm

Download, compile and install swftools by issuing these commands:

wget http://www.swftools.org/swftools-2012-10-15-1307.tar.gz
tar -zxvf swftools-2012-10-15-1307.tar.gz
cd swftools-2012-10-15-1307
./configure
make
checkinstall

N.B - You will be asked a series of questions at the beginning of the install, press return for each to continue.

Once that has completed you can now test it by issuing the following:

pdf2swf --version

Which should give you the following output:

pdf2swf - part of swftools 2012-10-15-1307

Successful swftools build.

Step 6: Compile and Install ffmpeg (1.1.2)

Let's go back to our temporary working area

Let's make our temporary working area

cd /usr/adm

Download, compile and install ffmpeg by issuing these commands:

wget http://ffmpeg.org/releases/ffmpeg-1.1.2.tar.gz
tar -zxvf ffmpeg-1.1.2.tar.gz
cd ffmpeg-1.1.2
./configure --enable-libmp3lame --enable-libxvid --enable-libvorbis --enable-libgsm \
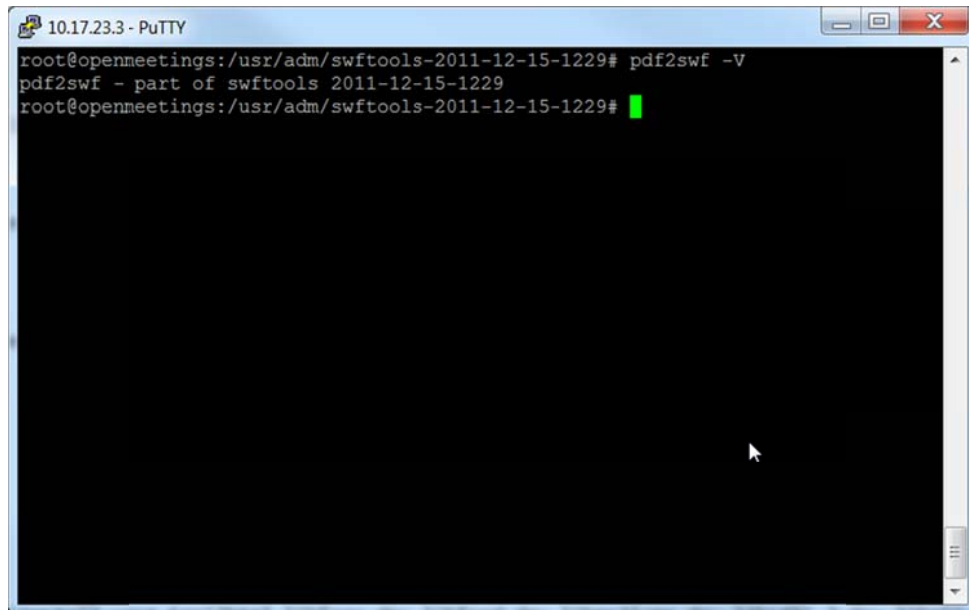--enable-libfaac --enable-gpl --enable-nonfree
make
checkinstall

N.B - You will be asked a series of question towards the end of the install, press return for each to continue.

Once that has completed you can now test it by issuing the following:

ffmpeg -version

Which should give you the following output:

ffmpeg-1.1.2

Step 7: Install JOD Converter

Let's go back to our temporary working area

cd /usr/adm

Download, extract JOD by issuing these commands: **(We will move the JOD location after the installation of OM 2.x)**

wget http://jodconverter.googlecode.com/files/jodconverter-core-3.0-beta-4-dist.zip
unzip jodconverter-core-3.0-beta-4-dist.zip

Step 8: Install ANT 1.8.4 for compiling latest OM 2.x

Let's go back to our temporary working area

cd /usr/adm

Download, extract ANT by issuing these commands:

wget http://mirror.catn.com/pub/apache//ant/binaries/apache-ant-1.8.4-bin.tar.gz
tar -zxvf apache-ant-1.8.4-bin.tar.gz

Once that has completed you can test it by issuing the following commands:

cd /usr/adm/apache-ant-1.8.4/bin
./ant -version

This should output the following:

Apache Ant(TM) version 1.8.4 compiled on May 22 2012

Stephen Cottham   08/02/2013

Step 9a: Download and compile latest OM 2.1

Again back to our working area:

cd /usr/adm

Then check out the latest source code using the following:

svn checkout https://svn.apache.org/repos/asf/openmeetings/trunk/singlewebapp/

Once that has completed we can then build the source by issuing the following:

cd /usr/adm/singlewebapp
/usr/adm/apache-ant-1.8.4/bin/ant clean.all
/usr/adm/apache-ant-1.8.4/bin/ant -Ddb=mysql

This will take a little while depending on your system, once it has finished you should be left the following message:

BUILD SUCCESSFUL

Now we move the compiled sourced into the correct location.

/usr/adm/singlewebapp/dist
mv red5/ /usr/lib/

Step 9a: Install pre-built OM 2.x (Alternative to Step 9)

To Download the latest builds we use the following links:

2.0

https://builds.apache.org/view/M-R/view/OpenMeetings/job/OpenMeetings%202.0/

Or

2.1

https://builds.apache.org/job/openmeetings/lastSuccessfulBuild/artifact/singlewebapp/dist/

The file will be something like the following "apache-openmeetings-incubating-2.xxxxx.tar.gz: (Where xxx is the date and build version)

We can do this using wget, so first we need to go back to our build area like so:

cd /usr/adm
mkdir -p prebuilt/red5
cd prebuilt/red5

Then grab the file and extract it:

wget https://builds.apache.org/job/openmeetings/lastSuccessfulBuild/artifact/singlewebapp/
dist/apache-openmeetings-incubating-2.xxxxx.tar.gz

tar -zxvf apache-openmeetings-incubating-2.xxxxx.tar.gz

The precompiled deployment doesn't include the mysql connector so we need to download it:

cd /usr/adm/prebuilt/red5/webapps/openmeetings/WEB-INF/lib

wget http://ftp.up.ac.za/pub/linux/mysql/Downloads/Connector-J/mysql-connector-java-5.1.20.zip

unzip mysql-connector-java-5.1.20.zip

cd mysql-connector-java-5.1.20

mv mysql-connector-java-5.1.20-bin.jar /usr/adm/prebuilt/red5/webapps/openmeetings/WEB-INF/lib

Now we need to move the compiled source into the correct location, in this system we are using /usr/lib/red5.

cd /usr/adm/prebuilt/
mv red5/ /usr/lib/

Step 10: Install compiled\Pre-Built OM 2.x

Let's move the JOD into place now

cp -R /usr/adm/jodconverter-core-3.0-beta-4 /usr/lib/red5/webapps/openmeetings

And set some permissions and ownerships

chown -R nobody /usr/lib/red5
chmod +x /usr/lib/red5/red5.sh
chmod +x /usr/lib/red5/red5-debug.sh

Set the start-up script for OM 2.x by issuing the following:

vi /etc/init.d/red5

and adding the following:

```
#! /bin/sh
### BEGIN INIT INFO
# Provides:        red5
# Required-Start:      $remote_fs $syslog
# Required-Stop:    $remote_fs $syslog
# Default-Start: 2 3 4 5
# Default-Stop:     0 1 6
# Short-Description: Starts red5 server for Openmeetings.
### END INIT INFO
# For RedHat and cousins:
# chkconfig: 2345 85 85
# description: Red5 flash streaming server for OpenMeetings
# processname: red5
# Created By: Sohail Riaz (sohaileo@gmail.com)
# Modified by Alvaro Bustos
PROG=red5
RED5_HOME=/usr/lib/red5
DAEMON=$RED5_HOME/$PROG.sh
PIDFILE=/var/run/$PROG.pid
[ -r /etc/sysconfig/red5 ] && . /etc/sysconfig/red5
RETVAL=0
case "$1" in
start)
        cd $RED5_HOME
    start-stop-daemon --start -c nobody --pidfile $PIDFILE \
        --chdir $RED5_HOME --background --make-pidfile \
        --exec $DAEMON >/dev/null 2>/dev/null &
```

```
            RETVAL=$?
            if [ $RETVAL -eq 0 ]; then
                    echo $! > $PIDFILE
            fi
            echo
;;
stop)
        start-stop-daemon --stop --quiet --pidfile $PIDFILE \
            --name java
        rm -f $PIDFILE
            echo
            [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/$PROG
;;
restart|force-reload)
            $0 stop
            $0 start
;;
status)
            # Debian and Ubuntu 10 status check
            ps aux | grep -f $PIDFILE  >/dev/null 2>/dev/null  && RETVAL=0 || RETVAL=3
            # Ubuntu 12 status check using improved "start-stop-daemon" status query
            # (use the above command, or comment out above command and uncomment the two
below commands.
            # start-stop-daemon --status --pidfile $PIDFILE
            # RETVAL=$?
            [ $RETVAL -eq 0 ]  && echo "$PROG is running"
            [ $RETVAL -eq 1 ]  && echo "$PROG is not running and the pid file exists"
            [ $RETVAL -eq 3 ]  && echo "$PROG is not running"
            [ $RETVAL -eq 4 ]  && echo "$PROG  - unable to determine status"
;;
checkports)
            netstat -anp | grep soffice
            netstat -anp | grep java
;;
*)
            echo $"Usage: $0 {start|stop|restart|force-reload|status|checkports}"
            RETVAL=1
esac
exit $RETVAL
```

Save the file and then set the permissions like below:

```
chmod +x /etc/init.d/red5
update-rc.d red5 defaults
```

Now we need to move the persistence files so we can connect to mysql, so issue the following:

Make backup copy
**mv /usr/lib/red5/webapps/openmeetings/WEB-INF/classes/META-INF/persistence.xml **
/usr/lib/red5/webapps/openmeetings/WEB-INF/classes/META-INF/persistence.xml-ori

Rename mysql template to persistence.xml

**mv /usr/lib/red5/webapps/openmeetings/WEB-INF/classes/META-INF/mysql_persistence.xml **
/usr/lib/red5/webapps/openmeetings/WEB-INF/classes/META-INF/persistence.xml

Edit the persistence file and add out mysql details, in this case we used "**openmeetings**" and
"**password**" – so issue the following:

vi /usr/lib/red5/webapps/openmeetings/WEB-INF/classes/META-INF/persistence.xml

Then change the following

, Username=openmeetings
, Password=password"/>
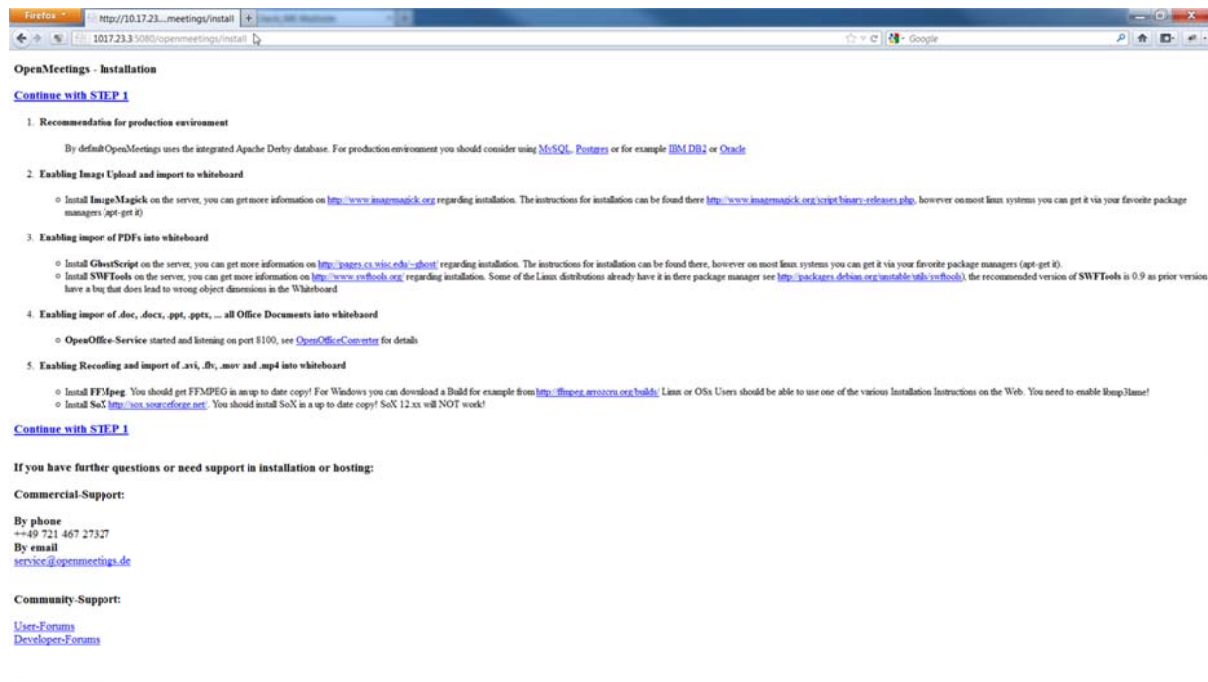
At this stage we are ready to start up OM 2.x for the first time.

/etc/init.d/red5 start

Now open the browser and go to the following link. **N.B remember to change the IP address to your
OM2.x server, the one below 10.17.23.3 is just for this example. Also make sure to clear your
browser cache.**

http://10.17.23.3:5080/openmeetings/install

If all went well you should now see this page:

Choose the "Continue with STEP 1" link

The only section we need to fill out at this stage is the following:

Username: **omadmin**
Userpass: **ompassword**
Email: **something@something.com**
TimeZone: **United Kingdom**
Domain Name: somedomain

Now click on INSTALL at the bottom of the page, this will then create all the needed tables etc.. - it can take a little while but be patient.

Once that has completed you can now enter the application by clicking on the "**Enter the Application**" link

You should see the following logon screen:

Enter these details to sign in.

Username: **omadmin**
Userpass: **ompassword**

Step 11: Add relevant paths to the configuration

Once logged in go to **Administration > Configuration**

You will see on the left hand pane a list of keys and values, the ones we are interested in are

SWFTools Path	**/usr/local/bin**
ImageMagick Path	**/usr/bin**
FFMPEG Path	**/usr/local/bin**
SoX Path	**/usr/bin**
JOD Path	**/usr/lib/red5/webapps/openmeetings/jodconverter-core-3.0-beta-4/lib**

Click on the left hand pane option and then enter the value as above, click on the save button to apply the changes, once you have done each key you should see the following:

JOD will find open office in this case so we do not need to set the path.

Step 12: Securing OpenMeetings using encryption (Optional)

12.1 - Generating CSR:

We can do this in a few ways, the first way I will show here is simply by generating a CSR and inserting these into OpenMeetings.

Create a new keystore and key, use the same password for both: (Taken from OM Website http://incubator.apache.org/openmeetings/RTMPSAndHTTPS.html)

keytool -keysize 2048 -genkey -alias red5 -keyalg RSA -keystore red5/conf/keystore
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: <your hostname, e.g demo.openmeetings.de>
What is the name of your organizational unit?
[Unknown]: Dev
What is the name of your organization?
[Unknown]: OpenMeetings
What is the name of your City or Locality?
[Unknown]: Henderson
What is the name of your State or Province?
[Unknown]: Nevada
What is the two-letter country code for this unit?
[Unknown]: US
Is CN=demo.openmeetings.de, OU=Dev, O=OpenMeetings, L=Henderson, ST=Nevada, C=US correct?
[no]: yes
Enter key password for <red5>

Generate a CSR:

keytool -certreq -keyalg RSA -alias red5 -file red5.csr -keystore red5/conf/keystore

Submit CSR to your CA of choice and receive a signed certificate
Import your chosen CA's root certificate into the keystore (may need to download it from their site - make sure to get the root CA and not the intermediate one)

keytool -import -alias root -keystore red5/conf/keystore -trustcacerts -file root.crt

(note: you may receive a warning that the certificate already exists in the system wide keystore - import anyway)

Import the intermediate certificate(s) you normally receive with the certificate:

keytool -import -alias intermed -keystore red5/conf/ keystore -trustcacerts -file intermediate.crt

Import the certificate you received:

keytool -import -alias red5 -keystore red5/conf/keystore -trustcacerts -file demo.openmeetings.de.crt

12.2 – Using Existing certs such as wild card certificates instead of generating a new CSR.

First let's go back to our work area:

cd /usr/adm/
mkdir certs
cd certs/

Using WinSCP or equivalent copy your wild card key and cert files: yourdomain.key.pem and yourdomain.cert.pem - **(These should be in PEM format)**

Now issue the following to convert the files to DER format

openssl pkcs8 -topk8 -nocrypt -in apache.key.pem -inform PEM -out key.der -outform DER
openssl x509 -in apache.cert.pem -inform PEM -out cert.der -outform DER

Now we need a couple of files to help us import the DER files into the keystore, so issue the following:

wget http://www.agentbob.info/agentbob/80/version/default/part/AttachmentData/data/ImportKey.java
wget http://www.agentbob.info/agentbob/81/version/default/part/AttachmentData/data/ImportKey.class

Then use these commands to import:

java ImportKey key.der cert.der

Finally move the keystore to the correct location and create a copy for the Screen Sharing

mv /root/keystore.ImportKey /usr/lib/red5/conf/keystore
cp /usr/lib/red5/conf/keystore /usr/lib/red5/conf/keystore.screen

N.B = Alias:importkey Password:importkey (When using the java import key files, you can change the password afterwards)

Now that we have either a new Cert of the wild card cert inside our Keystore we need to make some changes to OM 2.x to use these certificates and thus encrypt communications using HTTPS and RTMPS.

To use RTMPS do the following:
First make some changes to the red5-core.xml file by issuing the following:

cd /usr/lib/red5/conf
vi red5-core.xml

now uncomment **<!-- RTMPS -->** section by removing the **<!--** and the **-->** leaving this:

```
<bean id="rtmpsMinaIoHandler"
    class="org.red5.server.net.rtmps.RTMPSMinaIoHandler">
    <property name="handler" ref="rtmpHandler" />
    <property name="codecFactory" ref="rtmpCodecFactory" />
    <property name="rtmpConnManager" ref="rtmpMinaConnManager" />
    <property name="keyStorePassword" value="${rtmps.keystorepass}" />
    <property name="keystoreFile" value="conf/keystore" />
</bean>

<bean id="rtmpsTransport" class="org.red5.server.net.rtmp.RTMPMinaTransport" init-method="start" destroy-method="stop">
    <property name="ioHandler" ref="rtmpsMinaIoHandler" />
    <property name="connectors">
      <list>
        <bean class="java.net.InetSocketAddress">
          <constructor-arg index="0" type="java.lang.String" value="${rtmps.host}" />
          <constructor-arg index="1" type="int" value="${rtmps.port}" />
        </bean>
      </list>
    </property>
        <property name="ioThreads" value="${rtmp.io_threads}" />
    <property name="jmxPollInterval" value="1000" />
    <property name="tcpNoDelay" value="${rtmp.tcp_nodelay}" />
</bean>
```

Save this file and then do the following:

cd /usr/lib/red5/conf
vi red5.properties

set rtmps.port=**5443**
rtmps.keystorepass=**password (password = password you set on your new keystore)**

Now edit config.xml by doing the following:

cd /usr/lib/red5/webapps/openmeetings/
vi config.xml

Set these following values:

<rtmpsslport>**5443**</rtmpsslport>
<useSSL>**yes**</useSSL>
<proxyType>**best**</proxyType>

To use HTTPS do the following:

First make a backup of the original jee-container file by doing the following:

cd /usr/lib/red5/conf
mv jee-container.xml jee-container.xml.orig

Then rename the SSL jee template

mv jee-container-ssl.xml jee-container.xml

Now edit the config.xml

cd /usr/lib/red5/webapps/openmeetings/
vi config.xml

set

<protocol>**https**</protocol>
<red5httpport>**443**</red5httpport>

Lastly edit red5.properties by doing the following:

cd /usr/lib/red5/conf
vi red5.properties

set

https.port=**443**
http.port=**443**

Now restart OM using the following:

/etc/init.d/red5 restart

We can now connect using the following link:

https://yourdomain/openmeetings

Step 13: Installing Reverse Proxy using Apache Web Server (Optional)

Another way to secure the OpenMeetings service is to use Apache as a reverse proxy, to do this we need to do the following:

First install Apache2 and enabling relevant modules by running the following commands:

```
apt-get install apache2
a2enmod proxy
a2enmod proxy_http
a2enmod ssl
a2enmod headers
a2enmod rewrite
a2enmod cache
/etc/init.d/apache2 restart
```

We can now redirect port 80 (less secure) or port 443 (secure) to port 5080, to do this we need to create a virtual host, to do this do the following:

```
cd /etc/apache2/sites-enabled/
```

Now for SSL redirect **(using a Cert on Apache instead of keystore)** do the following

```
vi om.yourdomain.com-ssl
```

and add the following

```
<IfModule mod_ssl.c>
#NameVirtualHost *:443
ProxyRequests Off
<VirtualHost *:80>
  ServerAdmin hostmaster@domain.com
  ServerName om.yourdomain.com

  ProxyPreserveHost On
  RewriteEngine on
  # Redirect http traffic to https
  RewriteRule ^/(.*)$       https://om.yourdomain.com/$1 [L,R]
</VirtualHost>
```

Stephen Cottham   08/02/2013

```
<VirtualHost *:443>
  ServerAdmin hostmaster@domain.com
  ServerName om.yourdomain.com

  SSLEngine on
  SSLProxyEngine On
  RequestHeader set Front-End-Https "On"
  ProxyPreserveHost On
  RewriteEngine on
  CacheDisable *

  # Reverse proxy all requests
  RewriteRule ^/(.*)      http://om.yourdomain.com:5080/$1 [P]

  SSLCertificateFile /etc/ssl/certs/yourdomain.pem
  SSLCertificateKeyFile /etc/ssl/private/yourdomain.key

  SetEnvIf User-Agent ".*MSIE.*"   \
  nokeepalive ssl-unclean-shutdown  \
  downgrade-1.0 force-response-1.0
</VirtualHost>
```

You will need SSL certs for this to work, so copy your Key and Cert to the following **locations (use WinSCP or equiv)**

```
/etc/ssl/certs/ = yourdomain.pem
/etc/ssl/private/ = yourdomain.key
```

Now restart apache2

```
/etc/init.d/apache2 restart
```

You can now go to https://om.yourdomain.com/openmeetings which will encrypt ONLY the HTTPS components and re-write the address so it doesn't show the 5080 port; it still uses RTMP for flash.

And finally for HTTP redirect and re-write do the following: **(assuming no SSL don't use this in conjunction with the other config – both can be incorporated but this is just for example)**

```
vi om.yourdomain.com-http
```

Add the following:

```
ProxyRequests Off
<VirtualHost *:80>
  ServerAdmin hostmaster@domain.com
```

ServerName om.yourdomain.com

ProxyPreserveHost On
RewriteEngine on
CacheDisable *

 # Reverse proxy all requests
RewriteRule ^/(.*)   http://om.yourdomain.com:5080/$1 [P]
</VirtualHost>

Then restart Apache with

/etc/init.d/apache2 restart

Now you can access OM with

http://om.yourdomain.com/