

Lab: Virtual Linux Server (LAMP) in the Amazon Cloud

1 Details

Aim: To create a useable Amazon Web Services (AWS) virtual Linux Server , using open source LAMP (Linux, Apache, MySQL and php) software. The L in LAMP is the Linux Operating System. You will be given your own Linux AWS_UBUNTU instance details from your lab instructor. Demo:

<https://www.youtube.com/watch?v=pKzRIaTUPA8>

What is the IP address of your AWS instance:

L.1 In groups of two, start the Kali instance in the Napier Cloud, and log in.

L.2 Next create a file named `ssh_private_key.pem`

```
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAhm9mVfj3uKkEpv6FO5STf5WTnAAE1IUcc//NZIVDVVtWqj1VyA++aa1w8IE6
UQZaf92BUxckhcajEmbqnte9fHAXx1w6j5ZF1wUXmpPmjWgkUQYsxeBBZTCrIUV5LD3ZmtyLjnUu
39Rwd3vL4dmuCVlNHtJuNeT0F27wBVp3rEeWSc5ItaGGiGX2xZYxWcbb15h5/uG/7JygzAVPsz/n
XYcXIR+JQa158pPde493RZEXMrGp3+M3jiBtNbwGXNbr1d7AntgYO68prbg8SkVRux7pu9eY6ja8
ENeDHSayRlKXyhAovfIxaFmqzs5JuJqZskJ5G+IreLcm9+ZMwVkmEwIDAQABAoIBABu1TYt4Qu6S
zLaueUuB5vgCU+aDFnGKB/Knlsl3CCJBySYAikYQCtDvb5Wt0GXhIQnMQ9P7CkKh65nQQQLczdqt
ZDiipFEar/AfDdtqgKsSryo/g8uxhqiMbcUiiivvIdAU5O3wrcpefq4X/8IwA+sxH1C0L38ihqWQae
jZMAWeFmTR0F0+d5HHLNuJCvMQ6rvr87z1C2J4GsAkyQiwLyHoOnVCs3H9AZivswLyT11e2sPKGF
oq4MUJejeDFXpir4cr0q5GmpcO+DsAvNLNxiqCGmqN1xJy/4ZT+f1lXSHf3pIZLNTq+jfLsNbMPH
0Fft7Qm5sjyLqGuX1AG2j+7Ij0kCgYEAVQcau2nGGkRvrkcfYDaiYar7U4AXmVn/IpdrPkykoe6r
PUPZWgdwRCJ2Fn05NlovdIEisqPaTg+CKVUCGvo2E5MdBMDMHSztr9ddESfXcnJj3YdYCMSZj53A
XRThO0M6hpUoNcVm01soBwyj1qR/nG2y47rgvt/3fX8YEZYM/u8CgYEAs91WA1bie6OAO7R8EHFC
OGar7MzH71ExvhPlSUxysJbV3Oy74H4ucWUJRPWZK/t2HOJ1QVif/rVrXEI9HxPLFHsx5IX+wES1
z7+HUt98EsRE5Uv1ObzimDNGcZ11Qz1NNnxHzUhNIAffp6pznWqGo06u7TRCbM0v7yjsKDVUCx0C
gYBgeIq9ETuIV1sE/81Uii2PWPPrRX7MsZeA/IDYcYnM3LZ55MvJvJxJNw6Suz5PuWFKVy/4QW61j
X2jXfUoIyJ/ml5McxOSnEzHvJyOZSNMc1qJ/Ab79IpeYibMODkU11H6/BsqxvqA+x+wCxQGs3uE9
zYKZ1Ns0rwsZIk7/UqwpQwKBgQCg1xQ+8wpQPdGTWIgU3iqz1NChTs2LeK8i+PyVD06vYF0eXbf5
Xm/1CrLWATp7TI9OhtsONyTS9AQhYXrVDoUZik0nroOgAyrQbD9UvPk4gejLOaB/pqfiIJXaw4zo
Fgmkh5NqSpDWQRV3RhRC1Wdxr4kR66whQ0GSdFgstpF+QKBgBiKxREdhbxdqJMb5/iMzy+PNV6y
Hm9NH92414bKamLS0dzrwpjg8GpNSJcRV9heAyTPbsnEfbjJz+0Bq13XLs6AxuWgwKlFYKjz1TGP
stWMPJP3VZqaVpVvHgPWokN/TDw4gk4jpkXdhC6/ggFBzghKiM53fGS+FpsuWMeP8Shs
-----END RSA PRIVATE KEY-----
```

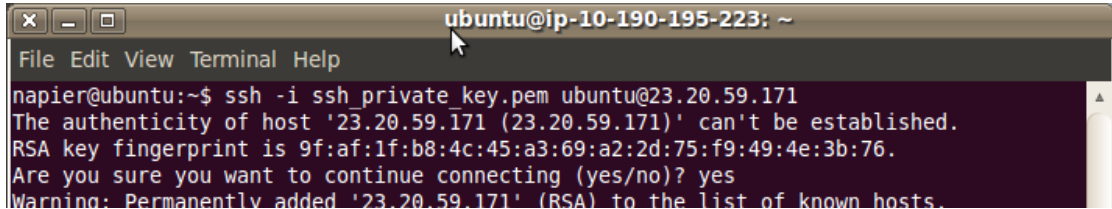
(<http://asecuritysite.com/key.zip>)

L.3 Change the permissions of the RSA Private Key file to be read only, using the `chmod` command, such as:

```
chmod 400 ssh_private_key.pem
```

Next from Kali access your AWS instance with (where *ip-instance* is your allocated AWS IP address):

```
ssh -i ssh_private_key.pem ubuntu@ip-instance
```



You should now be remotely logged into the virtual Linux server **AWS_UBUNTU** in the Amazon Cloud. Use the **ifconfig**, **pwd**, **ls** commands to check the details of the current virtual machine.

What is the IP address of the AWS instance (use ifconfig):

Can you ping your AWS instance from your Kali machine using this address: Yes/No

Can you ping your AWS instance from your Kali machine using the public IP address: Yes/No (If not, ask your tutor to open up the firewall on AWS for this).

L.4 Change User Account Passwords

Creating your own password is the first thing you should do on your server, as a basic security measure.

On **AWS_UBUNTU** Change the **root** user password to **napier123** with:

```
sudo passwd
```

Change the **ubuntu** user password to **napier123** using:

```
sudo passwd ubuntu
```

L.5 Services Running

Check services running and the associated processes on **AWS_UBUNTU** with commands such as:

```
netstat -au
netstat -at
```

or

```
netstat -a | grep udp
netstat -a | grep tcp
```

☞ Are there any established connections to services? YES/NO

☞ Which service is connected to a client?

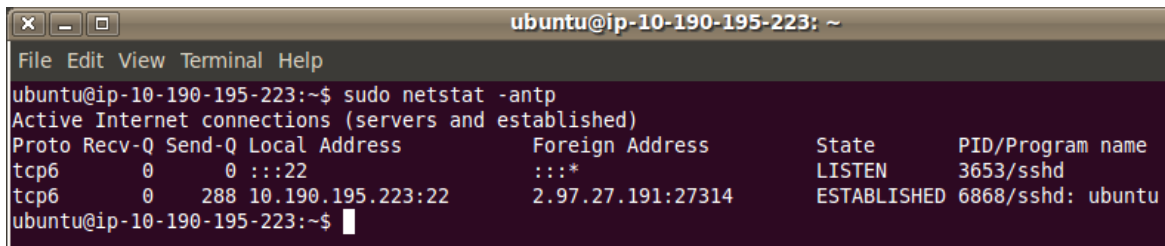
☞ List the IP address of the client?

To view the associated processes for services and the port numbers they are on, on **AWS_UBUNTU** use `netstat` command such as:

```
sudo netstat -antp
```

☞ List the services, port numbers and the processes?

You should get output similar to the following:



```
ubuntu@ip-10-190-195-223: ~  
File Edit View Terminal Help  
ubuntu@ip-10-190-195-223:~$ sudo netstat -antp  
Active Internet connections (servers and established)  
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name  
tcp6      0      0 :::22                  :::*                     LISTEN      3653/sshd  
tcp6      0      0 10.190.195.223:22      2.97.27.191:27314      ESTABLISHED 6868/sshd: ubuntu  
ubuntu@ip-10-190-195-223:~$
```

L.6 Use Advanced Packaging Tool (APT) to Update OS

Next update your **AWS_UBUNTU** image, so you can see the latest **packages** available to install, using the Advanced Packaging Tool (APT) `apt-get update` command:

```
sudo apt-get update && sudo apt-get upgrade -y
```

Use the `man` command to find out what the `-y` flag is used for.

☞ What does the `-y` switch do?

L.7 The A in LAMP is the **Apache Web Server**. Install the Apache Web server using the `apt-get install` command:

```
sudo apt-get install -y apache2
```

To start the Apache web server, use a command such as:

```
sudo /etc/init.d/apache2 start
```

☞ Using `netstat`, check the Apache web server is running?

YES/NO

From the **AWS_UBUNTU** command line, test that you can access the Web server from inside the Linux Cloud VM, using `telnet`:

```
telnet localhost 80
```

and then:

```
GET /index.html
```

☞ What message does the web page contain?

- L.8 Test the access to the remote Web Server from your local Linux VM **UBUNTU**, using a web browser, such as shown below



Figure 1 - UBUNTU to AWS_UBUNTU Apache Server

- L.9 Test the access to the remote Web Server from your Kali instance and also from your DESKTOP (Figure 2).

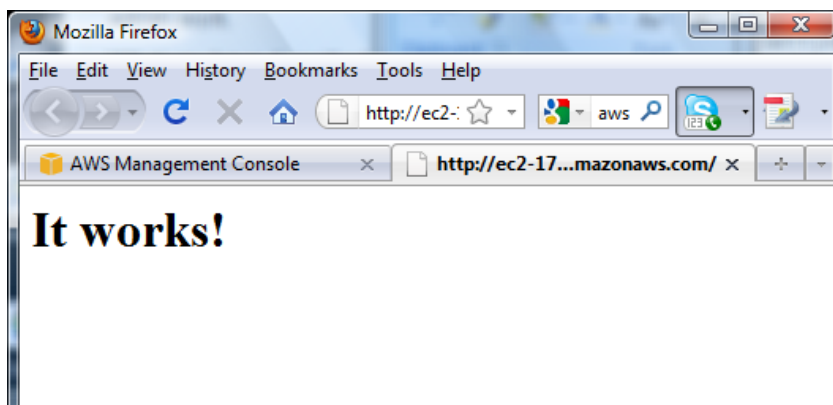


Figure 2 – DESKTOP to remote Web Server

- L.10 Go to the `/var/www` folder, and modify the default web page, so that it has a new greeting.

- L.11 Go to the `/var/log/apache2` folder, and monitor the Apache log file, for the last 5 Web Server accesses, using the `tail` command:

```
tail -n 5 -f access.log
```

While watching the window running the tail command, test the new greeting, using a Web browser from **Kali**.

- ☞ Can you access the updated web page, with the new message?
YES/NO
- ☞ Can you see the web page accesses being added to the access log file?
YES/NO
- ☞ What Information can you determine from the accesses, and what is different for the two accesses from different machines?

L.12 The P in LAMP is typically the **PHP** scripting language (can also be **Perl** or **Python**).

Install PHP and the Apache module with the **apt-get install** command:

```
sudo apt-get install php5
sudo apt-get install libapache2-mod-php5
```

Enable the PHP Apache module, and restart the Apache Web Server using:

```
sudo a2enmod php5
sudo /etc/init.d/apache2 restart
```

Create a new PHP file in the `/var/www` folder using a text editor such as with:

```
sudo vi webpage.php
```

and add:

```
<html><body><h1>It works!</h1>
<?php
print("Hello World");
?>
</body></html>
```

Test the access from your host system **DESKTOP**, as shown in

Figure 3.

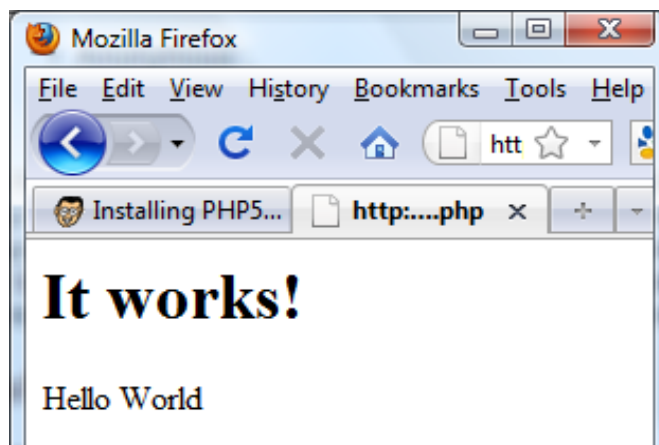


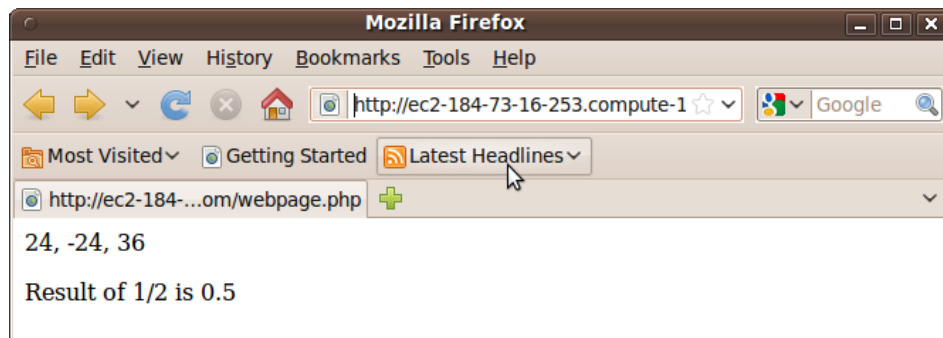
Figure 3 - Browsing new.php on remote Web Server

Next create web pages to investigate the following PHP code:

```
<?php
/* hyperlink */
print("<a href='index.html'>Index Page</a>"); // ; terminates php statements
print("<br>This is an <b>important</b> example");
?>
```

```
<?php
$a = 24; // decimal number
$b = -24; // a negative number
$c = 0x24; // hexadecimal number
print "<P>$a, $b, $c";
$val1=1;
$val2=2;
$result=$val1/$val2;
print "<P>Result of $val1/$val2 is $result";
?>
```

L.13 The PHP code can be tested from a Web Browser on the local UBUNTU VM, as shown below:



```
<?php
/* Strings and String Operators */
$name1 = "Fred ";
$name2 = "Smith";
print "Hello \"Fred\" . How are you?"; // . Concatenates strings
print "<br>Your full name is " . $name1 . " " . $name2;
$name1 .= $name2; // .= Appends strings
print "<br>Your full name is " . $name1
?>
```

```
<?php
/* Dates */
$today_us = date("Y-m-d");
$today_uk = date("d-m-Y");
$d_and_t = date("Y-m-d H:i:s (T)");
PRINT "Date (UK): $today_uk";
PRINT "<br>Date (US): $today_us";
PRINT "<br>Date and time: $d_and_t";
$today = getdate();
$month = $today['month'];
$mday = $today['mday'];
$year = $today['year'];
echo "<br>US Date is: $month $mday, $year";
?>
```

```
<?php
/* Loops */
```

```

for ($i=0;$i<11;$i++)
{
    $sqr_value=$i*$i;
    print "<BR>$i $sqr_value";
}

$i=0;
do
{
    $sqr_value=$i*$i;
    print "<BR>$i $sqr_value";
    $i++;
} while ($i<11);

$i=0;
while ($i<11)
{
    $sqr_value=$i*$i;
    print "<BR>$i $sqr_value";
    $i++;
}
?>

```

```

<?php
/* Arrays */
$colors = array('red','blue','green','yellow');
for ($i=0;$i<4;$i++)
{
    print "<BR>Color : $colors[$i]";
}

print "<BR>\n";
print "<BR>\n";
$mths = array(1=>'January', 'February', 'March');
print_r($mths);
?>

```

```

<?php
/* Calling Std Funtions */
print "<table border='1'>";
print "<tr><td>Value</td><td>Hex</td><td>Binary</td></tr>";
for ($val=0; $val<=16; $val++)
{
    print "<tr><td>" . $val . "</td><td>" . dechex($val);
    print "</td><td>" . decbin($val) . "</td></tr>";
}
print "</table>";
?>

```

```

<?php
/* Calling Funtions & Global Var's */
$val1 = 11;
$val2 = 21;

function Add ()
{
    global $val1, $val2;
    $val2 = $val1 + $val2;
}

function add_with_args($a,$b)
{
    return($a+$b);
}

$result=add_with_args($val1,$val2);

```

```

print"<BR>Result is $result";
Add ();
print "<BR>Result is $val2";
?>

<?php
print("Browser: $HTTP_USER_AGENT <br />\n");
print("IP Address: " . gethostbyname("amazon.co.uk"));
?>

```

```

<?php
/* File System Functions */
<?php
$fname = "./webpage.php";
$dname=".";
print "<BR>Real path: " . realpath($fname);
print "<BR>File name: " . basename($fname);
print "<BR>Dirname: " . dirname(realpath($fname));
print "<BR>Disk free space (MB): " .
round(disk_free_space($dname)/1024/1024);
print "<BR>Total disk space (GB): " .
round(disk_total_space($dname)/1024/1024/1024);
print "<BR>File group: " . filegroup($fname);
print "<BR>File mode: " . fileinode($fname);
print "<BR>File owner: " . fileowner($fname);
print "<BR>File permissions: " . decoct(fileperms($fname));
print "<BR>File size (Bytes): " . filesize($fname);
print "<BR>File type: " . filetype($fname);

$str= strftime("%H:%M, Date: %d-%m-%Y ", fileatime($fname));
print "<BR>File last accessed: " . $str ;
$str= strftime("%H:%M, Date: %d-%m-%Y ", filemtime($fname));
print "<BR>File last modified: " . $str ;

if (is_readable($fname))
{
    print "<BR>File is readable";
}
else
{
    print "<BR>File is not readable";
}
if (is_writable($fname))
{
    print "<BR>File is writable";
}
else
{
    print "<BR>File is not writable";
}
if (is_executable($fname))
{
    print "<BR>File is executable";
}
else
{
    print "<BR>File is not executable";
}
?>

```

L.14 The M in LAMP is the **MySQL** database. Install MySQL using:

```

sudo apt-get install mysql-server
sudo apt-get install php5-mysql

```


and then restart the Apache Web Server with:

```
sudo /etc/init.d/apache2 stop
sudo /etc/init.d/apache2 start
```

L.15 Next login to the MySQL Server, using the command line MySQL client application `mysql`. This allows us to connect to the MySQL Server, create and modify databases, and execute SQL queries and view the results.

Connect to the MySQL Server as the **root** user (use the root password from earlier):

```
root@ip-10-212-230-15:/var/www# sudo mysql -p
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.0.51a-3ubuntu5.5 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

Mysql>
```

You are now connected to the MySQL Server as **root**. Next enter the following commands to create a database, and then add a table, and some data:

```
create database mydatabase1;
use mydatabase1;
create table mynames (id INTEGER NOT NULL, name VARCHAR(50) NOT NULL);
insert into mynames values(1, 'fred smith');
select * from mynames;
```

and show that the output is in the form of:

```
+----+-----+
| id | name   |
+----+-----+
|  1 | fred smith |
+----+-----+
2 rows in set (0.00 sec)
```

L.16 Create a new table named **Products**, and add the following data:

Item	Description	Price
XT311	CISSP Certification	10
XG312	CCNA Security	22
OT821	CCNP ONT	33
XP411	CCNP Route	44

L.17 Next modify `webpage.php` to access your database (from the mynames table) with:

```

<html><body>
<?
$user="root";
$password="napier123";
mysql_connect(localhost,$user,$password);
mysql_select_db('mydatabase1')
    or die('Could not select a database.');
```

```

$sql = "SELECT name FROM mynames";

$result=mysql_query($sql);
$row = mysql_fetch_array($result);
print "Showing $row[name]<hr/>";

mysql_close();

?>
</body></html>
```

L.18 Next modify `webpage.php` so that it displays the Products table.

TELNET Service

L.19 Next install a **Telnet** server, and start the service using:

```

sudo apt-get install telnetd
sudo apt-get install inetutils-inetd

sudo /etc/init.d/inetutils-inetd restart
```

L.20 Next test that you can login in with Telnet from **DESKTOP / UBUNTU**, such as with:

```

ip-10-212-230-15 login: ubuntu
Password: napier123
Last login: Tue Mar 23 10:12:11 UTC 2010 from 5ac77477.bb.sky.com on pts/0
Linux ip-10-212-230-15 2.6.24-10-xen #1 SMP Tue Sep 8 19:06:53 UTC 2009 i686
```

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in `/usr/share/doc/*/copyright`.

Ubuntu comes with **ABSOLUTELY NO WARRANTY**, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
<http://help.ubuntu.com/>

```

System information as of Tue Mar 23 10:30:01 UTC 2010

System load: 0.0          Memory usage: 5%    Processes:      54
Usage of /:  3.7% of 14.76GB  Swap usage:  0%    Users logged in: 1

Graph this data and manage this system at https://landscape.canonical.com/
-----
At the moment, only the core of the system is installed. To tune the
system to your needs, you can choose to install one or more
predefined collections of software by running the following
command:

    sudo tasksel
-----
3 failures since last login.
```

☞ Have you successfully logged in to the remote Telnet Server from both **DESKTOP** and **UBUNTU**?

YES/NO

FTP Service

L.21 Install an FTP server with:

```
sudo apt-get install vsftpd
```

and connect to the FTP server from **DESKTOP / UBUNTU**:

```
ftp ec2-xx-xx-xx-xx.compute-1.amazonaws.com
```

Login with an anonymous login:

```
220 (vsFTPD 2.0.6)
530 Please login with USER and PASS.
USER ubuntu
530 This FTP server is anonymous only.
USER anonymous
331 Please specify the password.
PASS w.buchanan@napier.ac.uk
230 Login successful.
PWD
257 "/"
```

☞ Have you successfully logged in to the remote FTP Server from **UBUNTU** and **DESKTOP**?

YES/NO

Appendix A

Linux packages/commands:

```
sudo apt-get update && sudo apt-get upgrade -y
sudo apt-get install -y apache2
sudo apt-get install php5
sudo apt-get install libapache2-mod-php5
sudo apt-get install vsftpd
sudo apt-get install telnetd
sudo apt-get install inetutils-inetd

sudo apt-get remove -y apache2
sudo apt-get remove php5
sudo apt-get remove libapache2-mod-php5
sudo apt-get remove vsftpd
sudo apt-get remove telnetd
sudo apt-get remove inetutils-inetd
```