

University of Wolverhampton

School of Mathematics and Computer Science

5CS022 Distribute and Cloud Systems Programming

Workshop 7 Linux server on Azure

Overview


In this tutorial, you will be setting up a Linux server on Microsoft Azure cloud service.

Part 1 – Creating a Virtual Machine for Linux

- Log into Azure via <https://azureforeducation.microsoft.com/> and go the main Azure portal.
- In the Virtual machines page, create a new virtual machine.
- At the next page fill in the fields similar to below:

Create a virtual machine

×

 Changing Basic options may reset selections you have made. Review all options prior to creating the virtual machine.

- Basics
- Disks
- Networking
- Management
- Advanced
- Tags
- Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Azure for Students

Resource group * ⓘ

(New) lx1234567_group

Create new

Instance details

Virtual machine name * ⓘ

lx1234567


Region * ⓘ

(US) East US

Availability options ⓘ

No infrastructure redundancy required

Image * ⓘ

 Ubuntu Server 18.04 LTS - Gen1

See all images

Azure Spot instance ⓘ

Size * ⓘ

Standard_B1s - 1 vcpu, 1 GiB memory (£5.66/month)

See all sizes

Administrator account

Authentication type ⓘ

SSH public key

☒ Password

Username * ⓘ

jeffrey

Password * ⓘ

••••••••••

Confirm password * ⓘ

••••••••••

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.


Public inbound ports * ⓘ

None

☒ Allow selected ports

Select inbound ports *

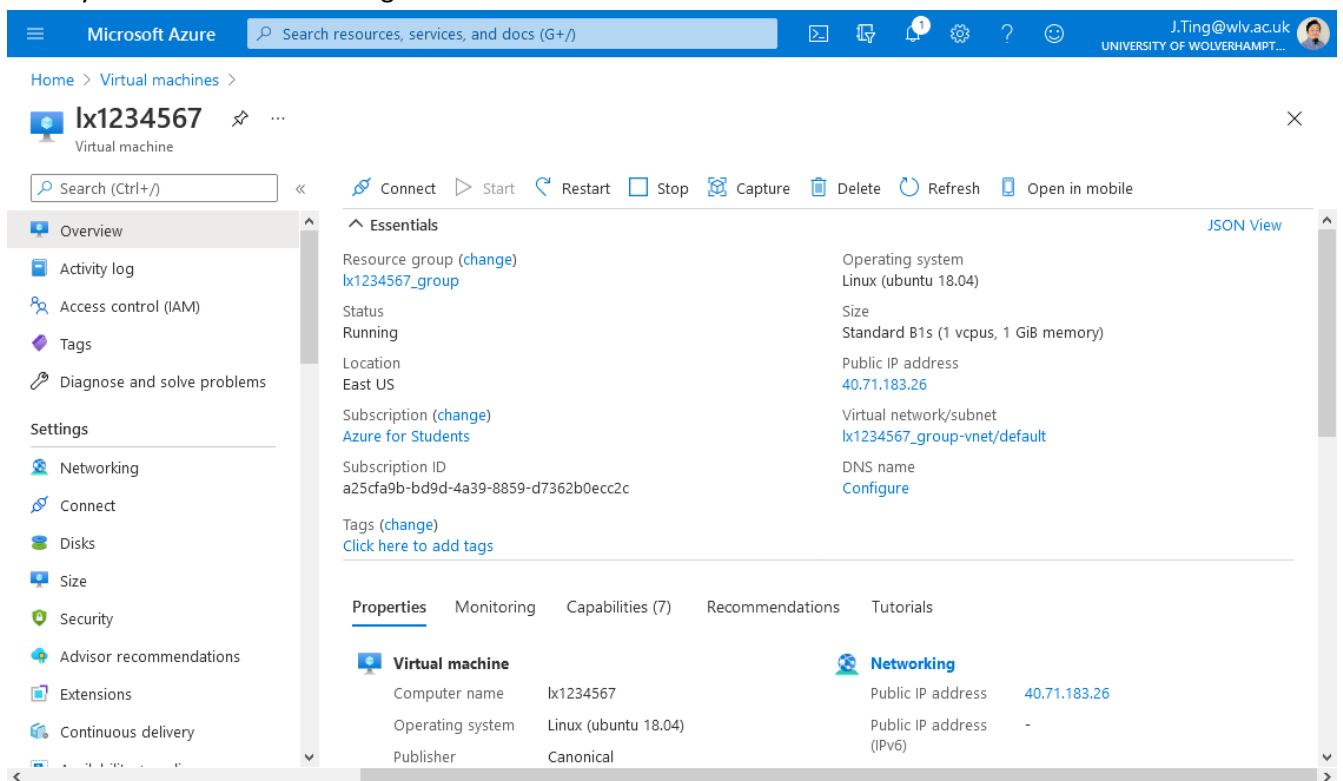
HTTP (80), HTTPS (443), SSH (22)

 **This will allow all IP addresses to access your virtual machine.** This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

- Make sure the Subscription is on "Azure for Students"
- Name the Virtual Machine name "lx"+ your student ID number, eg. "lx1234567"
- Make sure the Region is on (US) East US
- Set the Availability options to "No infrastructure redundancy required"
- Select the "Ubuntu Server 18.04 LTS – Gen 1" image
- Select the VM Size: "Standard_B1s – 1vcpu, 1GiB memory". This is an appropriate size and cost for the Linux Server as it does not require as much as Windows.
- Create your administrator user account and password. Please note this information down somewhere. If you forget, you will be locked out of your Virtual Machine and there will be nothing we can do.
- For the Inbound ports rules, select all available ones for now, and those are 80, 443, and 22.

Then click on "Review + create" to validate your Virtual Machine, and it should after a few moments say "Validation passed"

- Then click on "Create" to create the actual Virtual Machine.
- Go to your Virtual Machines to get to:



The screenshot shows the Microsoft Azure portal interface. At the top, there's a navigation bar with the Microsoft Azure logo and a search bar. Below the navigation bar, the breadcrumb trail shows 'Home > Virtual machines > lx1234567'. The main content area displays the details of the Virtual Machine 'lx1234567'. On the left, there's a sidebar with various options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking, Connect, Disks, Size, Security, Advisor recommendations, Extensions, and Continuous delivery. The main content area is divided into two sections: 'Essentials' and 'Properties'. The 'Essentials' section shows the Resource group (lx1234567_group), Status (Running), Location (East US), Subscription (Azure for Students), Subscription ID (a25cfa9b-bd9d-4a39-8859-d7362b0ecc2c), and Tags. The 'Properties' section shows the Virtual machine details: Computer name (lx1234567), Operating system (Linux (ubuntu 18.04)), and Publisher (Canonical). The 'Networking' section shows the Public IP address (40.71.183.26) and the Public IP address (IPv6) (-).

- Please make a note of your server's Public IP address.(eg. 40.71.183.26), as you will need this to connect to it.

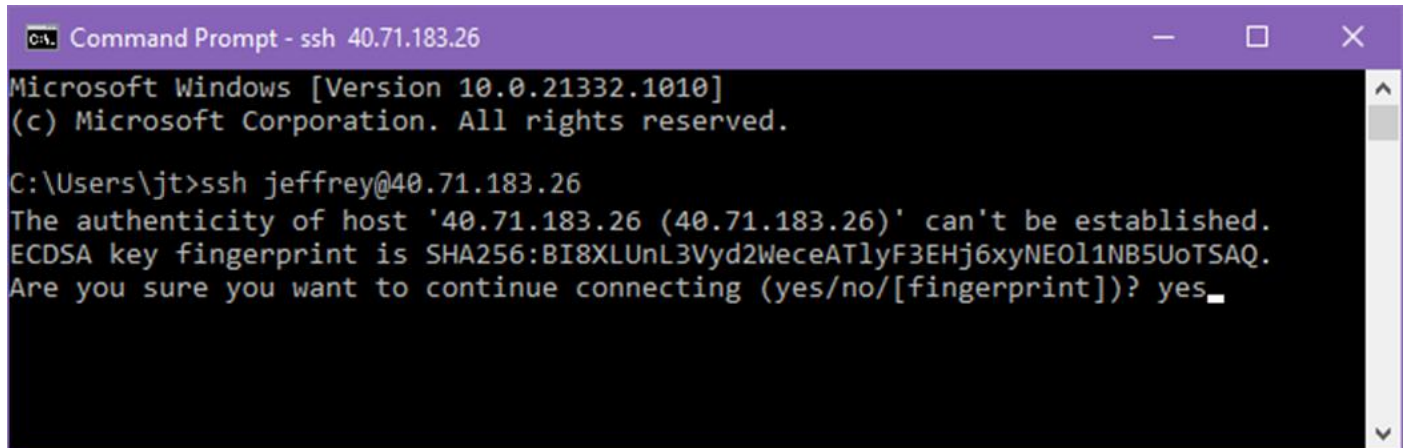
Part 2 – Connect to virtual machine

- Start a command prompt on your Windows PC.
- Connect to your Ubuntu Linux server using SSH, by enter into the command prompt :

```
ssh jeffrey@40.71.183.26
```

Obviously use your own server's public IP address, and username, rather than mine.

- You can also use ssh program Putty to connect to your server.
- The first time you connect, you will be asked to accept the authentication key fingerprint:

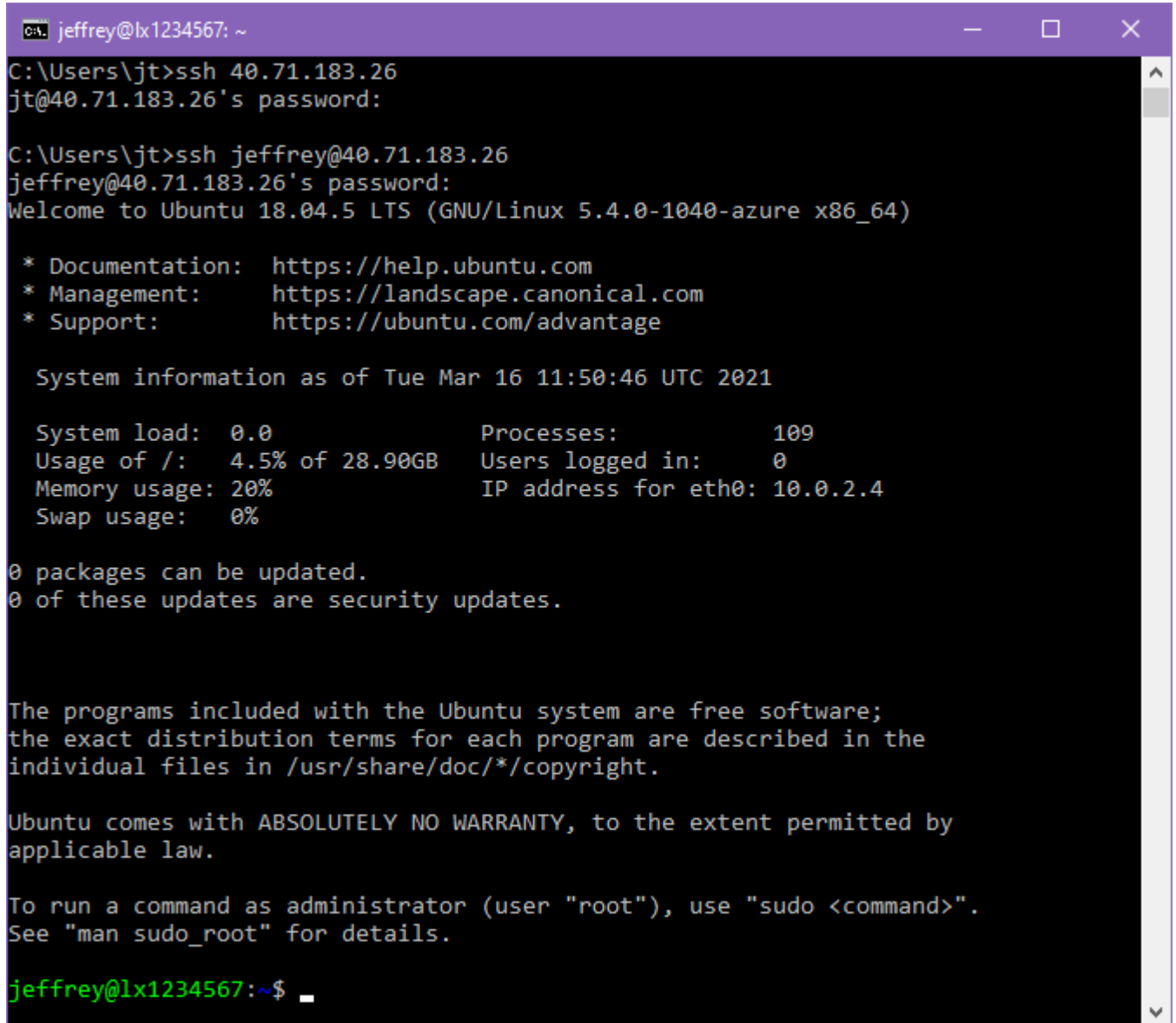


```
Command Prompt - ssh 40.71.183.26
Microsoft Windows [Version 10.0.21332.1010]
(c) Microsoft Corporation. All rights reserved.

C:\Users\jt>ssh jeffrey@40.71.183.26
The authenticity of host '40.71.183.26 (40.71.183.26)' can't be established.
ECDSA key fingerprint is SHA256:BI8XLUnL3Vyd2WeceATlyF3EHj6xyNE011NB5UoTSAQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes_
```

Enter "yes" to accept the key.

- When you have successfully logged in, you should see:

A terminal window titled 'jeffrey@lx1234567: ~' with standard window controls. It shows a user named 'jtt' at '40.71.183.26' attempting an SSH connection. After a password prompt, the connection is successful. The terminal displays the Ubuntu 18.04.5 LTS welcome message, system information (including system load, processes, memory usage, and IP address), and package update status. It also includes legal disclaimers and instructions on using 'sudo'. The prompt returns to 'jeffrey@lx1234567:~\$' with a cursor.

```
jeffrey@lx1234567: ~
C:\Users\jtt>ssh 40.71.183.26
jtt@40.71.183.26's password:

C:\Users\jtt>ssh jeffrey@40.71.183.26
jeffrey@40.71.183.26's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1040-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Mar 16 11:50:46 UTC 2021

System load:  0.0               Processes:            109
Usage of /:   4.5% of 28.90GB   Users logged in:     0
Memory usage: 20%              IP address for eth0: 10.0.2.4
Swap usage:   0%

0 packages can be updated.
0 of these updates are security updates.

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 run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

jeffrey@lx1234567:~$
```

Part 3 – Setting up Webmin on the Linux Server

- As the Linux server does not have a graphical user interface, having a web-based control dashboard to help set up the services on the server make it much easier to manage.
- We will install Webmin, a popular web-based control dashboard on this server.
- Issue the following command in terminal session to import the Webmin repository key:

```
wget http://www.webmin.com/jcameron-key.asc
```

- Next, issue the following command in Terminal to install the key:

```
sudo apt-key add jcameron-key.asc
```

- You will also add the Webmin repository into the `/etc/apt/sources.list` file. In this way, you will be able to install Webmin. Issue the following 3 commands in Terminal to do this:
 - `sudo bash`
 - `echo "deb http://download.webmin.com/download/repository sarge contrib" >> /etc/apt/sources.list`
 - `exit`

- Next, issue the following command in Terminal to update the repository index with that of the newly added Webmin repository:

```
sudo apt update
```

- Install the Webmin package as sudo via the following command:

```
sudo apt install webmin
```

- The system might prompt you for confirmation by providing you with a y/n option. Hit y to continue the installation process. Once installation is complete, you will see a message explaining how to access Webmin.

Part 4 – Configuring network firewall settings to allow access to the Webmin control panel

- In the web browser, switch back to the Overview panel for the virtual machine. You can find the VM under All Resources.
- In the Settings section, select the Networking item.
- You should see the NSG rules for the subnet in the top section and the NSG rules for the network interface in the bottom section of the same tab. In the bottom section, for the NSG rules for the network interface, select Add inbound port rule.

ix1234567 | Networking ...

Virtual machine

Search (Ctrl+/)

Attach network interface Detach network interface

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems

Settings

Networking

Connect

Disks
Size
Security
Advisor recommendations
Extensions
Continuous delivery
Availability + scaling
Configuration

ix1234567462

IP configuration ⓘ
ipconfig1 (Primary)

Network Interface: ix1234567462 Effective security rules Topology

Virtual network/subnet: ix1234567_group-vnet/default NIC Public IP: 40.71.183.26 NIC Private IP: 10.0.2.4 Accelerated networking: Disabled

Inbound port rules Outbound port rules Application security groups Load balancing

Network security group ix1234567-nsg (attached to network interface: ix1234567462)
Impacts 0 subnets, 1 network interfaces

Add inbound port rule

Priority	Name	Port	Protocol	Source	Destination	Action	
300	SSH	22	TCP	Any	Any	Allow	...
320	HTTP	80	TCP	Any	Any	Allow	...
340	HTTPS	443	TCP	Any	Any	Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	Deny	...

- Then add the following rule:

J.Ting@wlv.ac.uk
UNIVERSITY OF WOLVERHAMPT...

1

?

😊

lx1234567-nsg

×

Source ⓘ

Any

Source port ranges * ⓘ

*

Destination ⓘ

Any

Service ⓘ

Custom

Destination port ranges * ⓘ

10000

Protocol

☒ Any

☐ TCP

☐ UDP

☐ ICMP

Action

☒ Allow

☐ Deny

Priority * ⓘ

350

Name *

Webmin

Description

Webmin Control Panel

Add

Cancel

- You should then end up with these rules for your VM:

Inbound port rules

Outbound port rules

Application security groups

Load balancing

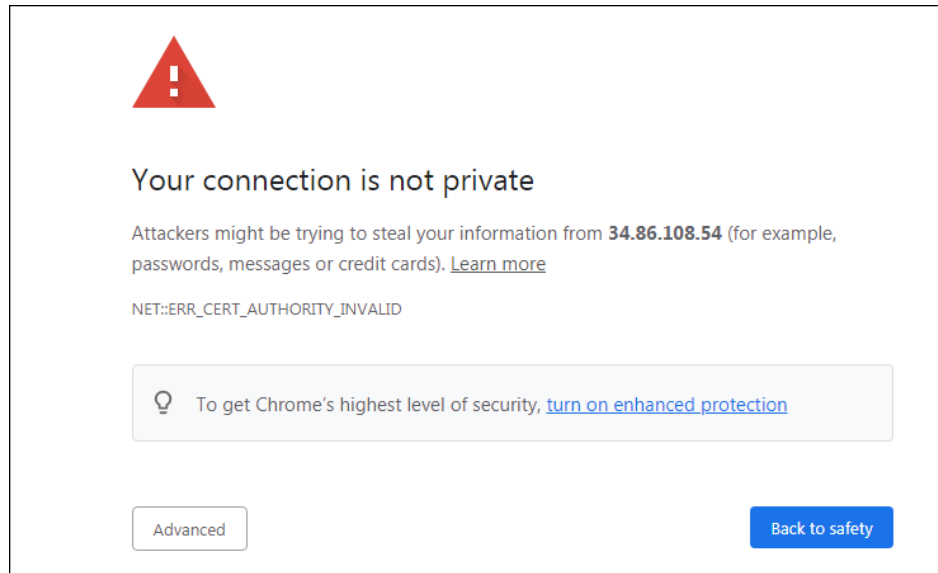
🛡️ Network security group lx1234567-nsg (attached to network interface: lx1234567462)

Impacts 0 subnets, 1 network interfaces

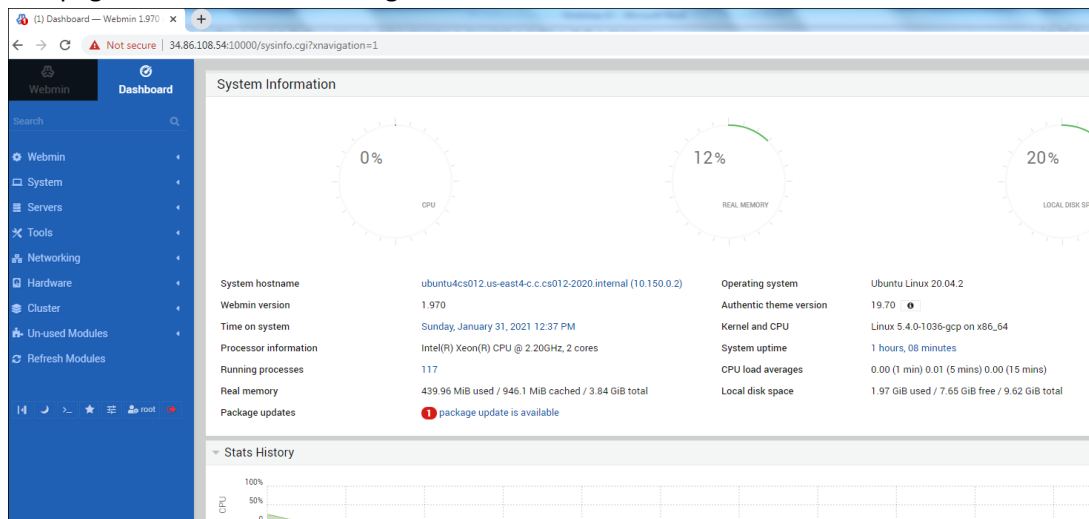
Add inbound port rule

Priority	Name	Port	Protocol	Source	Destination	Action	
300	🚩 SSH	22	TCP	Any	Any	🟢 Allow	...
320	HTTP	80	TCP	Any	Any	🟢 Allow	...
340	HTTPS	443	TCP	Any	Any	🟢 Allow	...
350	Webmin	10000	Any	Any	Any	🟢 Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	🟢 Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	🟢 Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	🔴 Deny	...

- Use your web browser and then go to the URL: **https:// 35.86.108.54:10000**. Substitute **35.86.108.54** with your Linux server's Public IP address. Also please note it's **https**, not **http**.
- The web browser may complain about the website's security certificate:



- Click on "Continue to this webpage (not recommended)."
- Login to Webmin with your Linux username and password.
- You should see a page similar to the following:

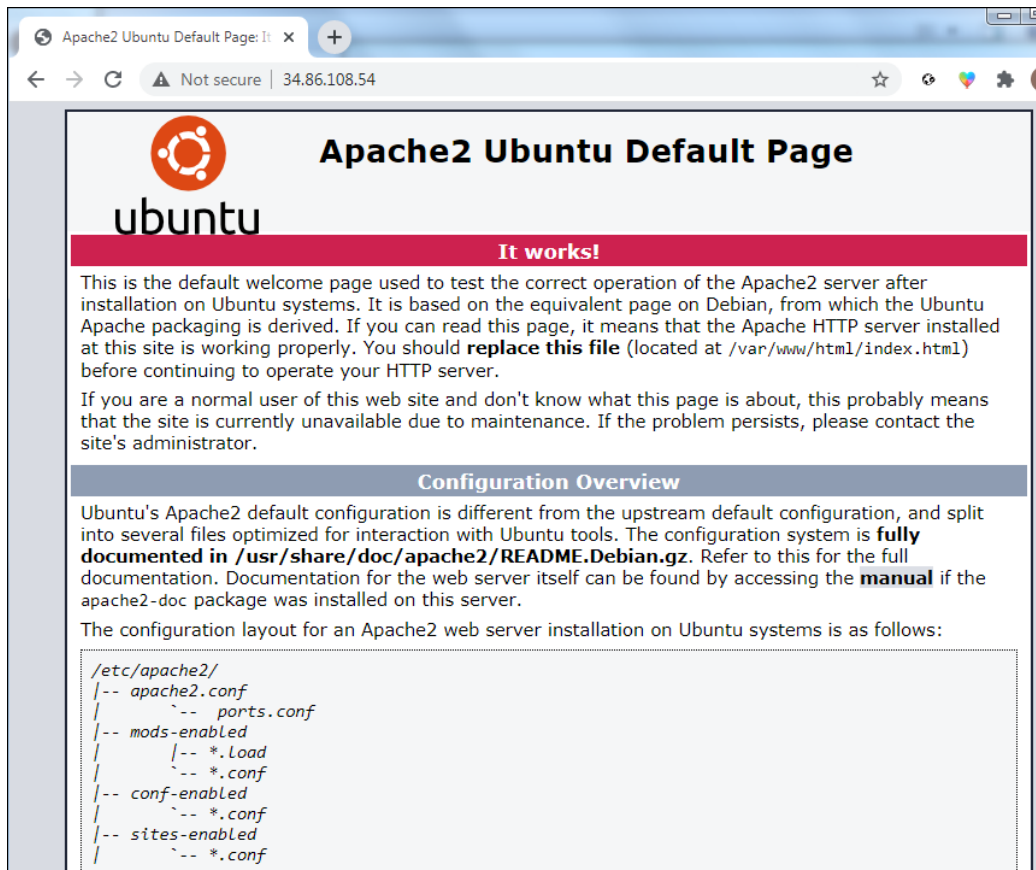


Part 5 – Installing Apache web server

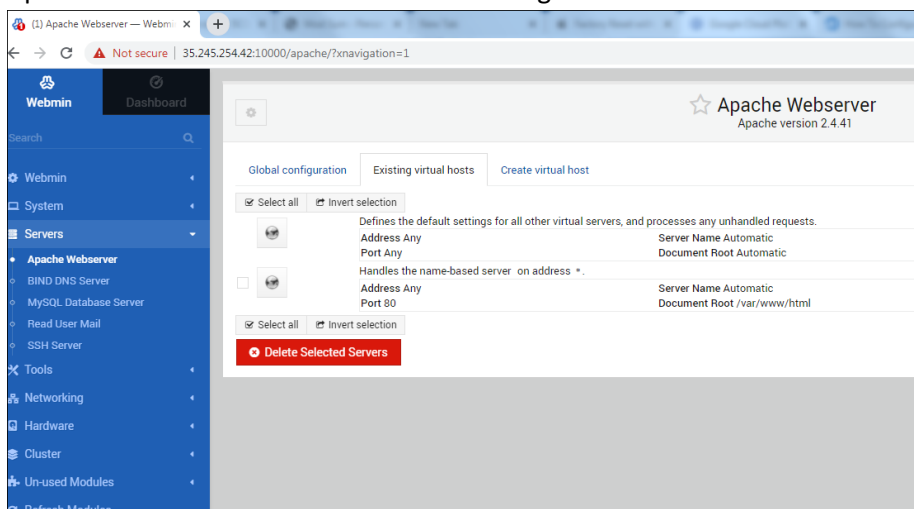
- The Apache web server is currently the most popular web server in the world, which makes it a great default choice for hosting a website. Type this in your Windows Command window:

sudo apt install apache2

- Verify if apache was installed as planned by visiting your server's IP address in your web browser, for example **http://35.204.55.123** (it's **http** this time, not **https**)
- You will see the default Apache web page, which is there for informational and testing purposes. It should look something like this:



- Go back to Webmin. On the left hand side of Webmin, click on Servers. Apache server should figure on the list. If not click on “Refresh Modules” at the bottom of the list to refresh these modules. Then reload webmin. Click on Apache Webserver. You will see the following screen:



Note: Significant parts of this document were sourced from Dr Consolee Mbarushimana

Part 6 – Installing MySQL Database Server

- Now that you have a web server up and running, you need to install the database system to be able to store and manage data for your site. MySQL is a popular database management system used within PHP environments.
- Again, use apt to acquire and install this software:

```
sudo apt install mysql-server
```

- When prompted, confirm installation by typing Y, and then ENTER.
 - Next, start MySQL:

```
sudo mysql
```

```
root@ubuntu4cs012:/home/cmbarushimana# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.22-0ubuntu0.20.04.3 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

Enter **quit** to exit mysql

Part 7 – Installing PHP

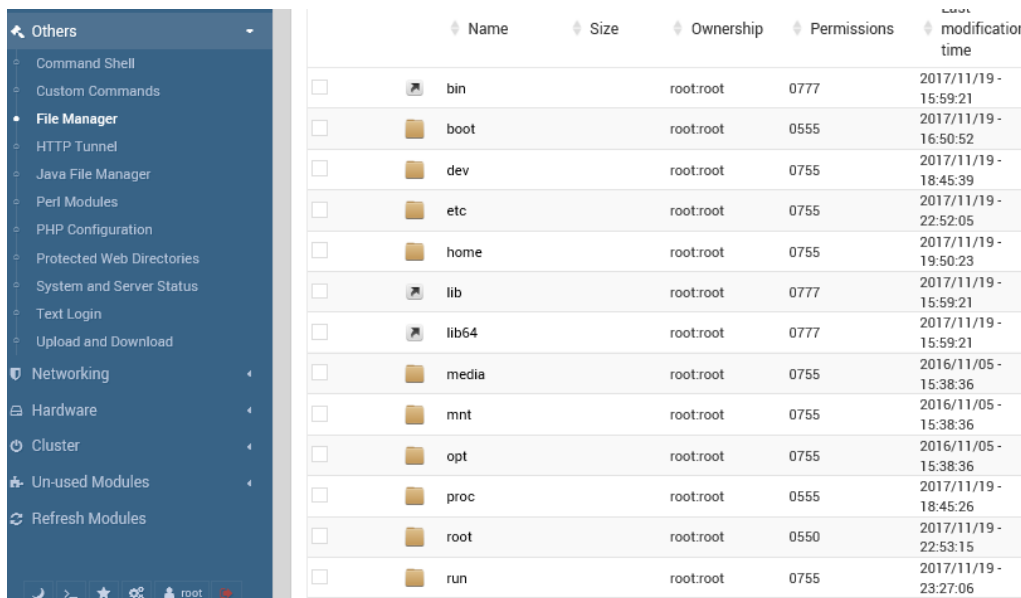
- PHP is the component of our setup that will process code to display dynamic content. It can run scripts, connect to our MySQL databases to get information, and hand the processed content over to our web server to display:

sudo apt install php libapache2-mod-php php-mysql

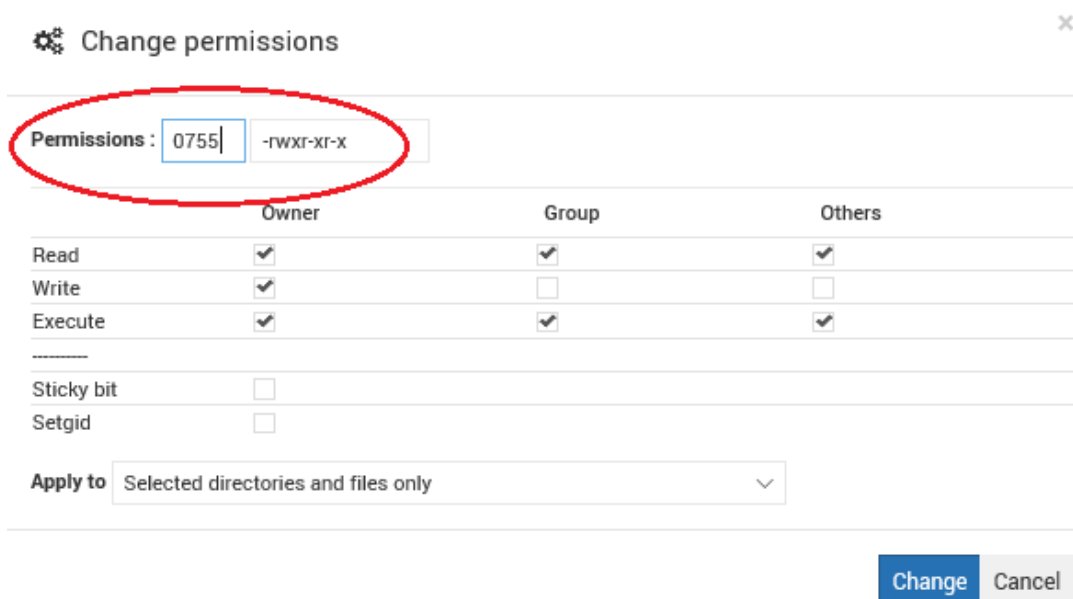
```
root@ubuntu4cs012:/home/cmbarushimana# php -v
PHP 7.4.3 (cli) (built: Oct  6 2020 15:47:56) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.3, Copyright (c), by Zend Technologies
root@ubuntu4cs012:/home/cmbarushimana#
```

Part 8 - Manipulating files on the Linux Server via Webmin

- In Webmin, click Tools and then File Manager.



- Navigate the File Manager by double-clicking the folders, to the directory `"/var/www/html"`.
- Click on the "File" and Create new file. Name the file `testpage.html`
- Right click the `testpage.html` file you have just created, and select Edit
- "Hello yourstudentnumber" or similar text into the editor box
- Save your file, and close it.
- Select the `testpage.html` file that you have just created, click on "Tools" then "Change permissions". Change the permissions to 0755.



- Now verify that your new website is working. Start a web browser and go to:

<http://40.71.183.26/testpage.html>

Use your own IP address, of course!

Note: Significant parts of this document were sourced from Dr Consolee Mbarushimana

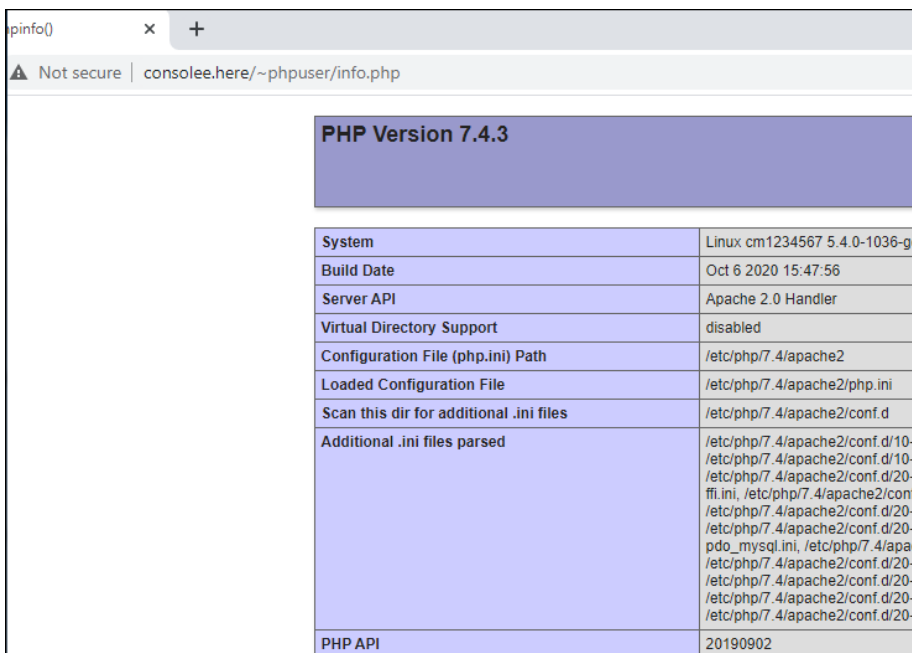
Part 9 - Setting up a PHP info page.

- 1) PHP is one of the most common web application programming languages in use today. We are now going to create a PHP web page.
- 2) Using Webmin's File Manager, create a file in the “/var/www/html” directory, called “info.php”.
- 3) Insert the following code into the “info.php” file:



```
1 <html>
2 <?php
3 phpinfo();
4 ?>
5 </html>
```

- 4) The change the “info.php” file permissions to 0755.
- 5) Now with your web browser, go to <http://35.204.55.123//info.php> . Make sure you use your own server’s IP address! You should see your version of the following page:



PHP Version 7.4.3	
System	Linux cm1234567 5.4.0-1036-gc
Build Date	Oct 6 2020 15:47:56
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d
Additional .ini files parsed	/etc/php/7.4/apache2/conf.d/10- /etc/php/7.4/apache2/conf.d/10- /etc/php/7.4/apache2/conf.d/20- ffi.ini, /etc/php/7.4/apache2/conf. /etc/php/7.4/apache2/conf.d/20- /etc/php/7.4/apache2/conf.d/20- pdo_mysql.ini, /etc/php/7.4/apac /etc/php/7.4/apache2/conf.d/20- /etc/php/7.4/apache2/conf.d/20- /etc/php/7.4/apache2/conf.d/20- /etc/php/7.4/apache2/conf.d/20-
PHP API	20190902

Part 10 - Setting up the MySQL Database Server

- 1) On the left hand side of Webmin, under Servers and then click on “MySQL Database Server”.
- 2) Then in the right hand pane, click “Start MySQL Server” if it’s not running.
- 3) When the MySQL Database Server has started, you should see the right hand pane of Webmin change to display “MySQL Databases” containing the “information_schema”, “mysql” and “performance_schema” databases , as well as the “Global Options”.
- 4) Click on the “mysql” database and Webmin will display the database tables within the “mysql” database.
- 5) Then click “Return to database list” to return to the previous screen.

Creating a new MySQL database and table

- 1) On the left hand side of Webmin, under Servers and then click on “MySQL Database Server”.
- 2) Click “Create a new database”
- 3) Give your new database the name of your student ID prefixed with the letters “db”. Thus if your student ID is 1234567, creating a new database with the name of “**db1234567**”. Make sure you use your student ID. Name the initial table “**Contacts**” and populate it with the information shown below

Field name	Data type	Type width	Key?	Auto-increment?	Allow nulls?	Unsigned?	Default value
Fullname	varchar	80	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> Yes	
Address	varchar	250	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> Yes	
Phone	varchar	20	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> Yes	
Email	varchar	30	<input type="checkbox"/> Yes	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> Yes	

- 4) Click “Create” to create the table.
- 5) On the next screen, click on your new database and your Contacts table, to display its fields or columns.

Part 11 - Entering data into the Contacts table

- 1) Go on "MySQL Database Server" and install the missing PERL modules as shown below (at the bottom)

Backup Databases

Click this button to setup the backup of all MySQL databases, either immediately or on a configured schedule.

Warning: The Perl modules DBI and DBD::mysql are not installed on your system, so Webmin will not be able to reliably access your MySQL database. [Click here to install them now.](#)

- 2) Click "Return to table list" to go back to the "Edit Database" view.
- 3) Click on the "Execute SQL" button.
- 4) Enter the following SQL statement into the box and then click the "Execute" button. Make sure you get the spaces, commas, and single-quotes correct:

INSERT INTO Contacts VALUES ('Firstname Surname', 'Your address', 'Yourphone number', 'email@wlv.ac.uk');

- 5) Then click "Return to table list"
- 6) Double Click on the Contacts table and then "View Data".
- 7) You should see the data that you have just entered into the "Contacts" table.
- 8) Repeat the procedure to create 4 more different contacts, using different values, and then capture the screen show all the contacts data, and paste it below:

★ Table Data			
Table <i>Contacts</i> in database <i>db1234567</i>			
Fullname ▼	Address ▼	Phone ▼	Email ▼
John Doe	1 London Road	11111111	johndoe@wlv.ac.uk
Jane Doe	1 London Road	11111111	janedoe@wlv.ac.uk
Peter Smith	2 Liverpool Road	22222222	petersmith@wlv.ac.uk
Mary Smith	2 Liverpool Road	22222222	marysmith@wlv.ac.uk
Results generated by SQL query: <i>select * from `Contacts` limit 0,25</i>			
Data in this table cannot be edited because it has no primary key.			

Part 12 - Connecting Apache/PHP to MySQL

- Go to the MySQL main screen showing all databases.
- Click on **"User Permissions"**
- Click on **"Create new user"**
- Create a new database user called **"phpuser"** with a password of **"Password123!"**, set the Hosts to **"localhost"**, and select all the permissions:

The screenshot shows the 'MySQL user details' form. The 'Username' field is set to 'phpuser'. The 'Password' field is set to 'Set to..' with a masked password. The 'Hosts' field is set to 'localhost'. The 'Permissions' dropdown menu is open, showing the following options: 'Select table data', 'Insert table data', 'Update table data', 'Delete table data', 'Create tables', and 'Drop tables'. The 'Maximum concurrent logins' field is set to 'Unlimited'. The 'Maximum connections per hour' field is set to 'Unlimited'. The 'Maximum queries per hour' field is set to 'Unlimited'. The 'Maximum updates per hour' field is set to 'Unlimited'. The 'Required certificate type' field is set to 'None'. The 'SSL cipher' field is empty. A green 'Create' button is at the bottom left.

- Click **"Create"**.
- Click "Return to database list"
- Click on **"Database Permissions"**, and then click on **"Create new database permissions"**
- Select the Database of **"db1234567"** (make sure you select **your own database**), and set the rest of the details as shown below:

☆ Create Database Permissions

Database permission options

Databases

☐ Any ☒ Selected

db1234567

Matching pattern

Username

☐ Anonymous user ☒

phpuser

Hosts

☐ From host permissions ☐ Any ☒

localhost

Permissions

Show View
Create Routine
Alter Routine
Execute
Create Event
Create Trigger

Create

- Click "Create".
- Click "Return to database list"
- Stop and then re-start the MySQL Database server.
- Go back to the Webmin Filemanager, and browse to the /var/www/html/ directory.
- Create a new file called "**contacts.php**".
- Enter the following code into the PHP file. Make sure you use **your own database name**:

```

<?php
$servername = "localhost";
$username = "phpuser";
$password = "Password123!";
$dbname = "db12345678";

try {
    // Attempt to establish a connection to the database
    $conn = mysqli_connect($servername, $username, $password);

    // Check connection
    if (!$conn) {
        throw new Exception("Connection failed: " . mysqli_connect_error());
    }

    // Select the database
    $dbconnect = mysqli_select_db($conn, $dbname);

    if (!$dbconnect) {
        throw new Exception("Database selection failed: " . mysqli_error($conn));
    }

    // Define the SQL query
    $sqlquery = "SELECT Fullname, Address, Phone, Email FROM Contacts";

    // Execute the query
    $myresults = mysqli_query($conn, $sqlquery);

    // Check if the query executed successfully
    if (!$myresults) {
        throw new Exception("Query execution failed: " . mysqli_error($conn));
    }

    // Fetch and display results
    while ($row = mysqli_fetch_array($myresults, MYSQLI_NUM)) {
        echo($row[0]. " ");
        echo($row[1]. " ");
        echo($row[2]. " ");
        echo($row[3]. "<br>");
    }

    // Close the connection
    mysqli_close($conn);
} catch (Exception $e) {
    // Handle any exceptions that occurred
    echo "Error: " . $e->getMessage();
}
?>

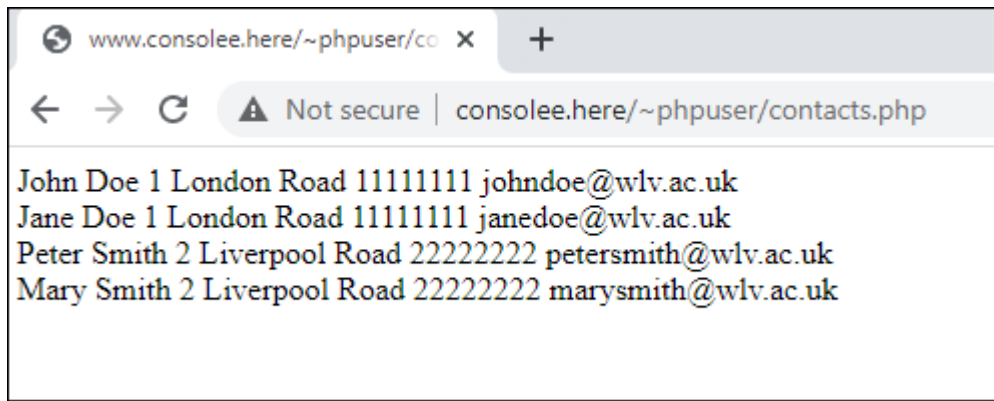
```

Find the code in the drive.

Or use the code from this example: https://www.w3schools.com/php/php_mysql_select.asp

- Make sure you don't get the curly brackets {} mixed up with the round ones ()!
- Save the file and then set its permissions to **0755** and both ownership fields to **"phpuser"**
- Now with your web browser, go to <http://35.204.55.123/contacts.php> . Make sure you use your own server's IP address! You should see your version of the following page:

Note: Significant parts of this document were sourced from Dr Consolee Mbarushimana



This is the end of this workshop.