Laboration 5 – Miniproject

During this laboration we will learn how to setup a Linux server to handle web applications.

Goals

- Install and configure a web server using Apache.
- Install and configure a database server using PostgreSQL.
- Setup a simple web-based application

Required material

You are required to print out this document. It will serve as guide during the laboration.

You are going to need your student user name and the password in order to login in the lab computers.

Take with you the material from ***previous*** *laboration exercises.*

If you have the course book, please take it with you to the laboration.

Preparation

BEFORE each laboration exercise, you will be required to prepare the PREPARATION assignment (Page 2).

Doing this preparation before the exercise will help you solve the exercises and, more important, understand what you are doing in the exercise.

Preparations

1. Check the course book or the Internet about how to install and configure:
    a. Apache

    b. PostgreSQL (not present on the course book. Use Internet)

    c. Firewall

    d. Setting up static IP address for the network interface

Laboration

Scenario

Your employer has given you the assignment to set up a Linux server with Apache for web access including scripting and a Postgresql server for the database.

1. Setting up the environment

1.1. Check the course web page for information about where the virtual machine repository is located.

1.2. Copy the virtual machine folder called DO2003 from the repository to D:\VMW\DO2003.

1.3. Execute (double click) the file do2003.vmx (VMware virtual machine configuration).

1.4. Log in to the system

User:	ide
Password:	ide

2. Network configuration

2.1. Configure your network so you get Internet access (refer to the previous Labs material).

____

____

3. Apache server

You are now going to add the web server Apache and look into how to configure it.

3.1. Install the Apache server. Check the course book for more details about its installation.

____

3.2. Check the installation by opening up a web browser in the host computer (Windows) and entering the URL http://*local-ip*, where *local-ip* is the ip for the virtual machine. If everything works you will be presented with a page stating “It works!”.

3.3. Which is the Apache configuration file (default)? Write down its absolute path.

____

____

3.4. You are now going to check where we can put our own files so they show up on the web server. Using a text editor, open the configuration file (default). This configuration file contains the settings we are looking for. The keyword `DocumentRoot` indicates where the root of the web server will be located in your file structure. This means, where will the web server look if you ask for the page http://local-ip/file.html. What is the path indicated by the `DocumentRoot`?

____

____

3.5. Navigate to the folder given by `DocumentRoot`. If you are in the correct directory, you will find a file named index.html. Open up this file in a text editor and replace the text “It works!” with something of your choice. Save the file and reload the page in the web browser. Check if the displayed text on the webpage matches your text changes.

____

____

3.6. You are now going to check where the scripts are going to be stored so they can be used by the web server. Look at the Apache configuration file (default) for the keyword `ScriptAlias`. The second argument specifies where scripts should be stored. What is the path indicated by the `ScriptAlias`?

____

____

3.7. Navigate to the folder given by `ScriptAlias`. This folder will be empty. Create a file named **`script`.** Change the access rights to executable for everyone using `chmod`. Open up the file and after adding the code below, save the file.

```
#!/bin/bash
echo "Content-Type: text/html"
echo
echo "<title>Info</title><h1>Today is:</h1>"
date
echo "<br>"
echo Your IP address:
echo $REMOTE_ADDR
```

3.8. Change the url in the web browser to http://local-ip/cgi-bin/script. If it works the web page must show the current date and time as well as your IP address. If not, start checking that you typed the script correctly.

4. Database Server PostgreSQL

4.1. Install the packet postgresql.

____

4.2. Once the installation is complete, you should configure the PostgreSQL server based on your needs, although the default configuration is viable. (https://help.ubuntu.com/community/PostgreSQL)

4.3. Switch to the postgres user

```
sudo su – postgres
```

4.4. From now, your commands will be executed in the postgreSQL prompt

4.4.1. Start the internal database connection using

```
psql
```

4.4.2. From this console you can issue SQL commands directly to the database.

4.4.3. Create a table

```
CREATE TABLE visits (
time     varchar(40),
ip   varchar(40)
);
```

4.4.4. Add data to the table

```
INSERT INTO visits VALUES (now(), ‘0.0.0.0’);
```

4.4.5. Insert one or more rows.

4.4.6. Test it using SELECT * FROM visits;

4.4.7. Create a user ide (for testing scripts in the command line)

```
create user “ide”;
```

4.4.8. Create a user www-data (for CGI scripts)

```
create user “www-data”;
```

4.4.9. Give the appropriate rights to these users

```
Grant all privileges on table visits to ide;
Grant all privileges on table visits to “www-data”;
```

4.4.10. Exit the postgresql utility using `\q`

4.4.11. Exit the postgres sudo (prompt) using `exit`

5. Install python-dev

6. Install python-setuptools

7. Install libpq-dev

8. Install a Python library which will allow you to connect python script to a Postgresql database.

```
sudo easy_install psycopg2
```

9. Navigate again to the folder given by `ScriptAlias` (Question 4.5). Create a file named **`script2`.** Change the access rights to executable for everyone using `chmod`. Open up the file and after adding the code below, save the file.

```
#!/usr/bin/python
import os,time,psycopg2

print "Content-Type: text/html"
print

ip = os.environ.get("REMOTE_ADDR","local")
date = time.ctime()

conn = psycopg2.connect("dbname=postgres")
cursor = conn.cursor()
cursor.execute("INSERT INTO visits VALUES (%s,%s)",(date,ip))
cursor.execute("SELECT * FROM visits ORDER BY time DESC")
for date,ip in cursor:
       print "Connection from %s at %s<br>\n" % (ip,date)
conn.commit()
```

10. Execute the script in the command line at least two times. Describe its output.

____

____

11. Change the url in the web browser to http://local-ip/cgi-bin/script2. Check if the output is similar to the output presented in the command line.

EXTRA ASSIGNMENTS

12. Change the network settings in VMWare from NAT to Bridged. Update your /etc/interfaces and set a static IP. You will find the IP on a paper sticker on the top part of the case of your host. Now your virtual machine will be accessible from ANY computer in the lab. Test gain all installations, i.e., http://local-ip/cgi-bin/script and http://local-ip/cgi-bin/script2
13. Install/ configure a firewall and allow traffic into the server for HTTP and SSH. Block everything else.

____

14. Finishing the exercise

14.1. Ask laboration assistant to check your work.
14.2. Shut down your virtual machine.

15. After the virtual machine is closed down, REMOVE it from the host.

Congratulations