Ecole Supérieure Privée
d'Ingénierie et de Technologies

Install and configure a web server

Services and network administration

ESPRIT
Realized by: Tarek BEN YOUNES

2014/2015

Install and configure a web server

Services and network administration

Requirements:

- A newly installed CentOS 6
- Internet connection
- root privilege

Goals:

- Install Apache Web server
- Set up name based virtual host
- Set up IP based virtual host
- Set up port based virtual host

Introduction

The World Wide Web commonly known as the web is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and navigate between them via hyperlinks.

- HTTP protocol :

The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems.

HTTP functions as a request-response protocol in the client-server computing model. A web browser, for example, may be the client and an application running on a computer hosting a web site may be the server. The client submits an HTTP request message to the server. The server, which provides resources such as HTML files and other content, or performs other functions on behalf of the client, returns a response message to the client. The response contains completion status information about the request and may also contain requested content in its message body.

- Port numbers :

A port is associated with an IP address of the host, as well as the type of protocol used for communication. The purpose of ports is to uniquely identify different applications or processes running on a single computer and thereby enable them to share a single physical connection to a packet-switched network like the Internet.

By default, HTTP uses port 80 and HTTPS uses port 443, but a URL like http://www.example.com:8080/path/ specifies that the web resource be served by the HTTP server on port 8080.

Install Apache on CentOS

Your system must have Apache Web software installed. if not, install it using default package installer called yum.

```
[root@www ~]# yum install httpd
```

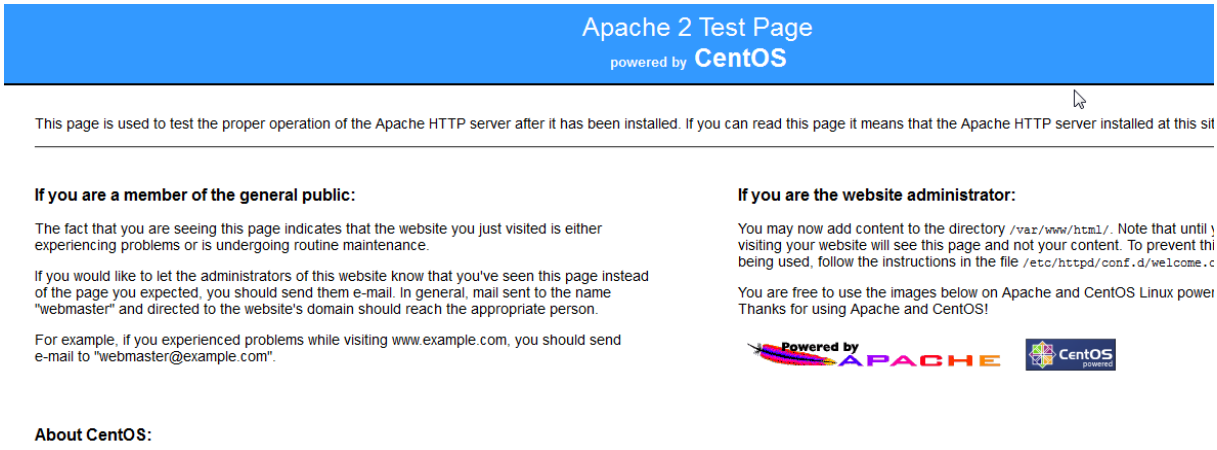
After httpd is installed we need to start the service

```
[root@www ~]# service httpd start
```

now we need to set the service to start on startup/reboot to make sure that every time we start our machine the web server will be activated .

```
[root@www ~]# chkconfig httpd on
```

To check if our server is working we can either visit the localhost on our web browser or just check the status of the httpd



Set up name-based virtual host

The term Virtual Host refers to the practice of running more than one web site (such as company1.example.com and company2.example.com) on a single machine. Virtual hosts can be "IP-based", meaning that you have a different IP address for every web site, or "name-based", meaning that you have multiple names running on each IP address. The fact that they are running on the same physical server is not apparent to the end user.

The first step in creating a virtual host is to create a directory where we will keep the new website's information. Create directories for these two virtual hosts under **/var/www/** folder. This location will be your Document Root in the Apache virtual configuration file later on.

```
[root@www ~]# mkdir /var/www/esprit1.com/

[root@www ~]# mkdir /var/www/esprit2.com/
```

Now it's time to create a test page for each website called **index.html** and add some content to the file so we will have something to check it when the IP calls the virtual host.

```
[root@www ~]# vi /var/www/esprit1.com/index.html
```

```
<html>
  <head>
    <title>www.esprit1.com</title>
  </head>
  <body>
    <h1>Hello, Welcome to www.esprit1.com.</h1>
  </body>
</html>
```

```
[root@www ~]#vi /var/www/esprit2.com/index.html
```

```
<html>
  <head>
    <title>www.esprit2.com</title>
  </head>
  <body>
    <h1>Hello, Welcome to www.esprit2.com.</h1>
  </body>
</html>
```

To set up Name based virtual hosting you must need to tell Apache to which IP you will be using to receive the Apache requests for all the websites or domain names. We can do this with **NameVirtualHost** directive. Open Apache main configuration file with your chosen editor.

```
[root@WWW ~]# vi /etc/httpd/conf/httpd.conf
```

Search for **NameVirtualHost** and uncomment this line

Next add the IP with possible in which you want to receive Apache requests. After the changes, your file should look like this:

```
NameVirtualHost 192.168.0.100:80
```

We are setting up virtual host sections for two domains:

- www.esprit1.com
- www.esprit2.com

Add the following two virtual directives at the bottom of the file. Save and close the file.

```
<VirtualHost 192.168.0.100:80>
    ServerAdmin webmaster@esprit1.com
    DocumentRoot /var/www/eprit1.com
    ServerName www.example1.com
ErrorLog logs/www.esprit1.com-error_log
CustomLog logs/www.esprit1.com-access_log common
</VirtualHost>

<VirtualHost 192.168.0.100:80>
    ServerAdmin webmaster@esprit2.com
    DocumentRoot /var/www/esprit2.com
    ServerName www.esprit2.com
ErrorLog logs/www.esprit2.com-error_log
CustomLog logs/www.esprit2.com-access_log common
</VirtualHost>
```

It is recommended to check the syntax of the file after making some changes and before restarting the Web server because if any syntax goes wrong Apache will refuse to work with some errors and eventually affect your existing web server go down for a while. If syntax is OK. Please restart your Web server and add it to chkconfig to make your web server start at the boot time

```
[root@www ~]# httpd -t

[root@www ~]# service httpd restart

[root@www ~]#chkconfig httpd on
```

❖ Note: Creating virtual host configurations on your Apache server does not magically cause DNS entries to be created for those host names. You must have the names in DNS, resolving to your IP address, or nobody else will be able to see your web site. You can put entries in your **hosts** file for local testing, but that will work only from the machine with those hosts entries.

Edit the hosts file and add the two hosts that you created

```
[root@www ~]#vi /etc/hosts

192.168.0.100 www.esprit1.com

192.168.0.100 www.esprit2.com
```

Once you're done with it, you can test the setup by accessing both the domains in a browser.

Set up IP-Based virtual host

To setup IP based virtual hosting, you must have more than one IP address/Port assigned to your server or your Linux machine.

It can be on a single NIC card , For example: eth0:1, eth0:2, eth0:3 … so forth. Multiple NIC cards can also be attached. If you don't know how to create multiple IP's on a single NIC, check the previous TP (BASIC CONFIG)

Purpose of implementing IP based virtual hosting is to assign implementing for each domain and that particular IP will not be used by any other domain.

Use the ifconfig command to visualize your NICs:

```
eth0      Link encap:Ethernet  HWaddr 08:00:27:4C:EB:CE
          inet addr:192.168.0.100  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe4c:ebce/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1


eth0:1    Link encap:Ethernet  HWaddr 08:00:27:4C:EB:CE
          inet addr:192.168.0.101  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

As you can see in above output, two IPs 192.168.0.100 (eth0) and 192.168.0.101 (eth0:1) is attached to the server, both IPs are assigned to the same physical network device (eth0).

Now, assign a specific IP/Port to receive http requests, you can simply do it by changing the directive in httpd.conf file.

```
<VirtualHost 192.168.0.101:80>
    ServerAdmin webmaster@esprit3.com
    DocumentRoot /var/www/esprit3
    ServerName www.esprit3.com
ErrorLog logs/www.esprit3.com-error_log
TransferLog logs/www.esprit3.com-access_log
</VirtualHost>
```

- Note : don't forget to create a new directory and the testing page(index.html) for the new web site www.esprit3.com

Set up port-based virtual host

The default port number for HTTP is 80. However, most webservers can be configured to operate on almost any port number, provided the port number is not in use by any other program on the server.

For example, a server may host the website www.esprit2.com. However, if the owner wishes to operate a second site, and does not have access to the domain name configuration for their domain name, and/or owns no other IP addresses which could be used to serve the site from, they could instead use another port number, for example, www.esprit.com:8080 for port 8080.

Open Apache main configuration file with your chosen editor and add listen 8080

```
[root@www ~]# vi /etc/httpd/conf/httpd.conf

Listen 8080
```

Add the following virtual directive at the bottom of the file. Save and close the file.

```
<VirtualHost 192.168.0.100:8080>
    ServerAdmin webmaster@esprit4.com
    DocumentRoot /var/www/esprit4.com
    ServerName www.esprit4.com
ErrorLog logs/www.esprit4.com-error_log
TransferLog logs/www.esprit4.com-access_log
</VirtualHost>
```

- Note : Don't forget to create a new directory and the testing page(index.html) for the new web site www.esprit4.com