ADVANCE LINUX ADMINISTRATION MARATHON

DAY3

Sumeha Jakate

➢ Apache multiple hosting

Apache multiple hosting, commonly known as **Apache virtual hosting**, is a feature of the Apache web server that allows you to host multiple websites (or domains) on a single server.

Configuration:

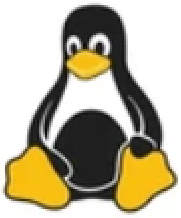
- In the Apache configuration file (httpd.conf or /etc/apache2/sites-available/), you define multiple <VirtualHost> blocks, each with a different ServerName (domain name) and DocumentRoot (directory for the site's files).
- Example:

```
<VirtualHost *:80>
   ServerName example1.com
   DocumentRoot /var/www/example1
</VirtualHost>

<VirtualHost *:80>
   ServerName example2.com
   DocumentRoot /var/www/example2
</VirtualHost>
```

Benefits of Apache Multiple Hosting:

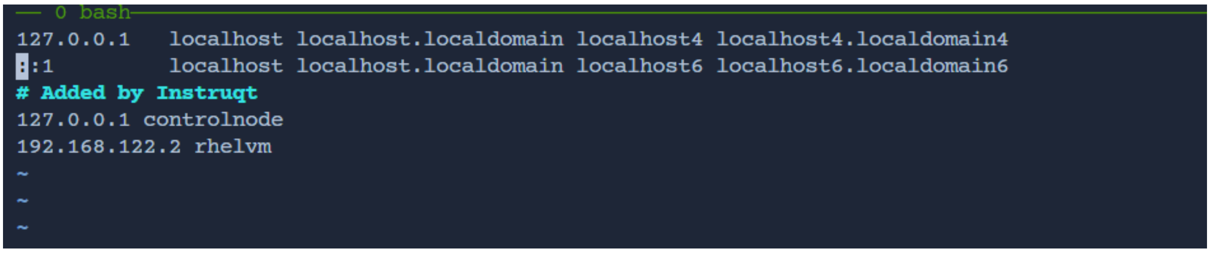
- **Cost-Effective:** You can host multiple websites on a single server, reducing infrastructure costs.
- **Efficient Management:** Easier to manage multiple sites from one server and centralize control.
- **Scalability:** You can add or remove domains easily by editing the Apache configuration.



➢ Practical

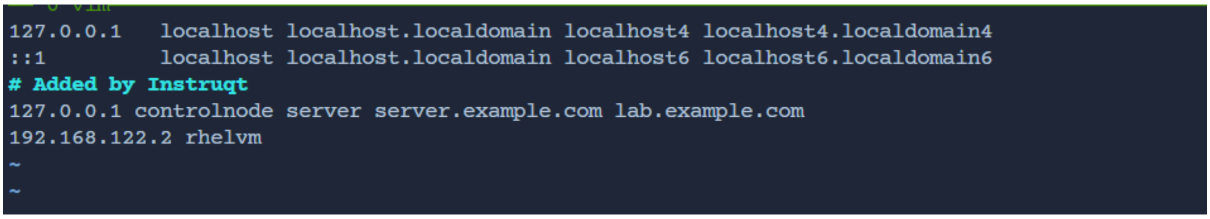
Step 1: Configuring Domains

Open /etc/hosts in test editor with root user



```
── 0 bash──────────────────────────────────────────────────────────
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
# Added by Instruqt
127.0.0.1 controlnode
192.168.122.2 rhelvm
~
~
~
```

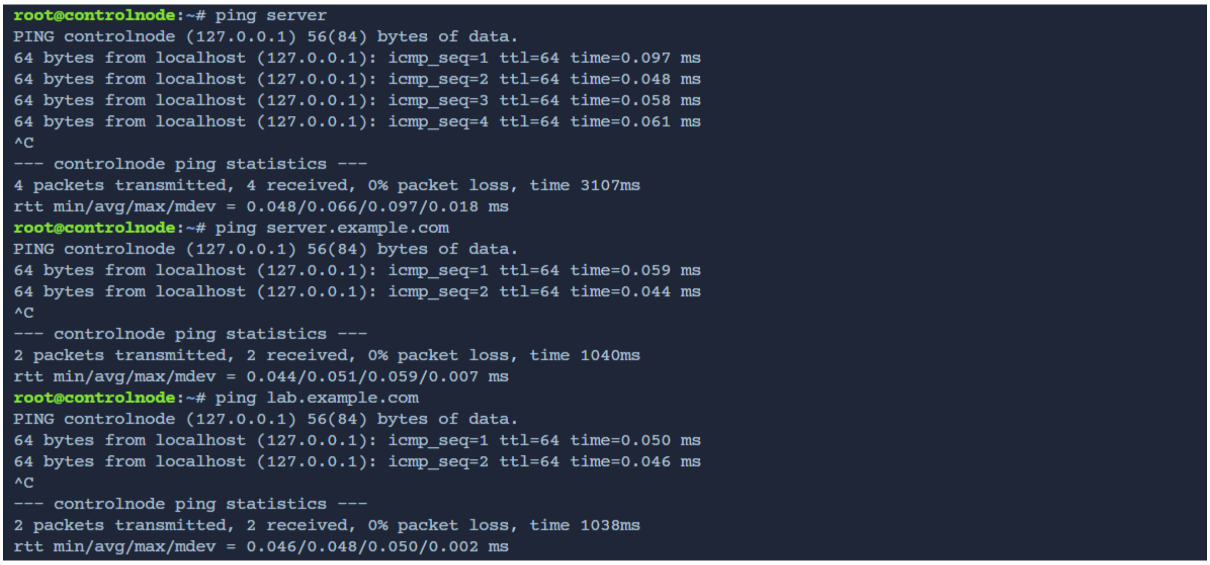
Add more domain names to your IP address



```
── 0 vim───────────────────────────────────────────────────────────
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
# Added by Instruqt
127.0.0.1 controlnode server server.example.com lab.example.com
192.168.122.2 rhelvm
~
~
```

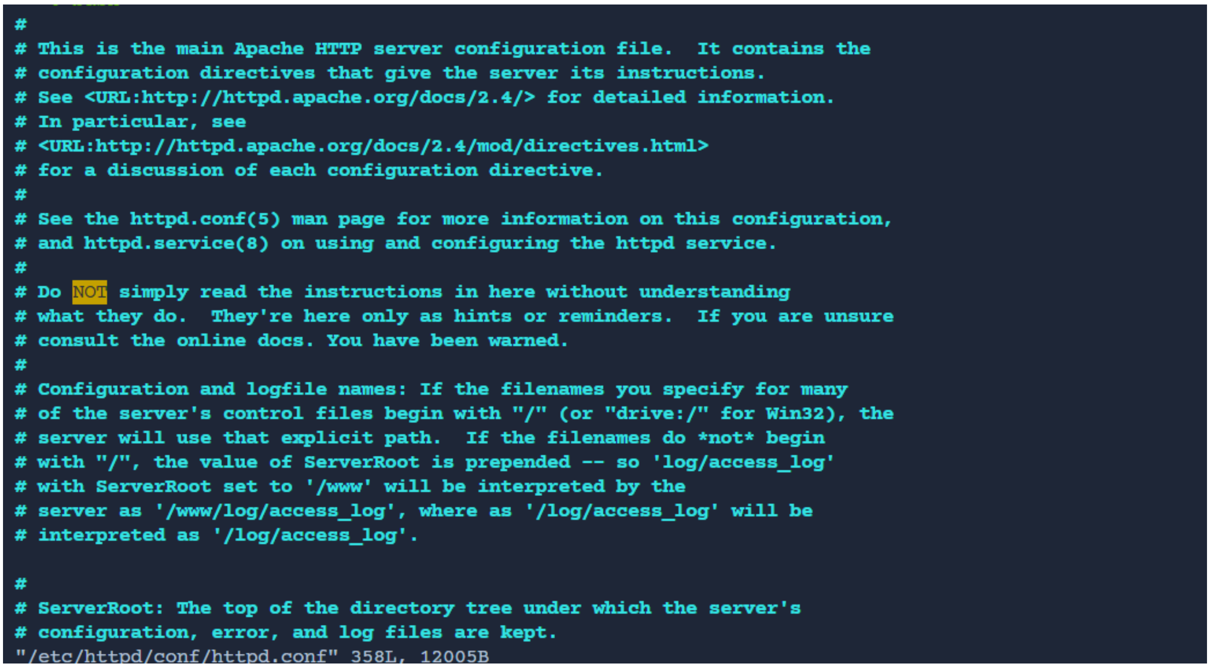
Step 2: Verify

Verify that your domains are configured properly by pinging them



```
root@controlnode:~# ping server
PING controlnode (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.097 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.048 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.058 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.061 ms
^C
--- controlnode ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3107ms
rtt min/avg/max/mdev = 0.048/0.066/0.097/0.018 ms
root@controlnode:~# ping server.example.com
PING controlnode (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.059 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.044 ms
^C
--- controlnode ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1040ms
rtt min/avg/max/mdev = 0.044/0.051/0.059/0.007 ms
root@controlnode:~# ping lab.example.com
PING controlnode (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.050 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.046 ms
^C
--- controlnode ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1038ms
rtt min/avg/max/mdev = 0.046/0.048/0.050/0.002 ms
```

When we go to our IP address it by default takes the path **/var/www/html** because it is already configured in the **/etc/httpd/conf/httpd.conf** configuration file.

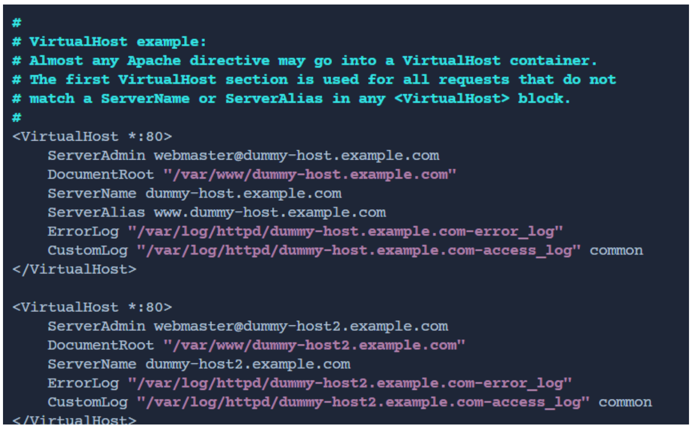


```
#
# This is the main Apache HTTP server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/docs/2.4/mod/directives.html>
# for a discussion of each configuration directive.
#
# See the httpd.conf(5) man page for more information on this configuration,
# and httpd.service(8) on using and configuring the httpd service.
#
# Do NOT simply read the instructions in here without understanding
# what they do.  They're here only as hints or reminders.  If you are unsure
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path.  If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so 'log/access_log'
# with ServerRoot set to '/www' will be interpreted by the
# server as '/www/log/access_log', where as '/log/access_log' will be
# interpreted as '/log/access_log'.

#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
"/etc/httpd/conf/httpd.conf" 358L, 12005B
```

If we want to change the default directory then we have to make a new configuration file in the **/etc/htttpd/conf.d** directory.

For the configuration format we can check **/usr/share/doc/httpd-core/httpd-vhosts.conf** file.



```
#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for all requests that do not
# match a ServerName or ServerAlias in any <VirtualHost> block.
#
<VirtualHost *:80>
    ServerAdmin webmaster@dummy-host.example.com
    DocumentRoot "/var/www/dummy-host.example.com"
    ServerName dummy-host.example.com
    ServerAlias www.dummy-host.example.com
    ErrorLog "/var/log/httpd/dummy-host.example.com-error_log"
    CustomLog "/var/log/httpd/dummy-host.example.com-access_log" common
</VirtualHost>

<VirtualHost *:80>
    ServerAdmin webmaster@dummy-host2.example.com
    DocumentRoot "/var/www/dummy-host2.example.com"
    ServerName dummy-host2.example.com
    ErrorLog "/var/log/httpd/dummy-host2.example.com-error_log"
    CustomLog "/var/log/httpd/dummy-host2.example.com-access_log" common
</VirtualHost>
```

Now, let's configure a new file called **file.conf** in the directory

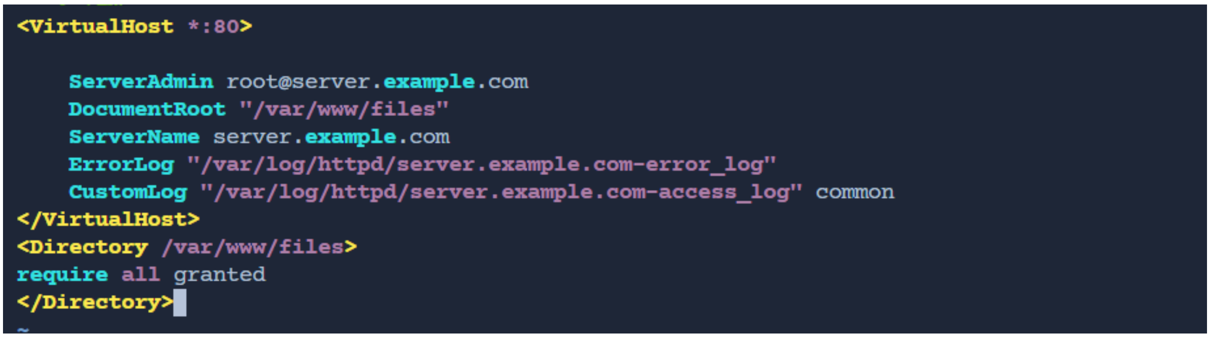
/etc/httpd/conf.d/

For that we first have to create a new directory to make it our default directory

Use command:

`mkdir /var/www/files`

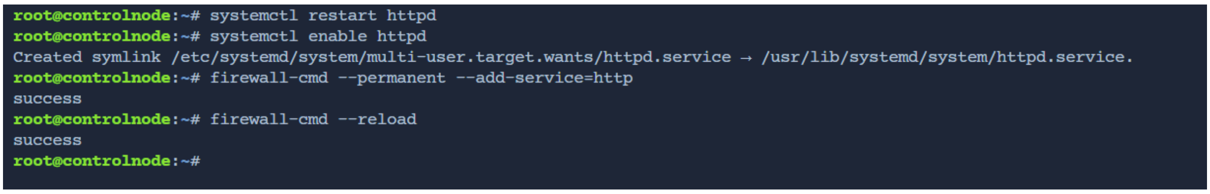
Now, open the configuration file in text editor and configure it



```
<VirtualHost *:80>

    ServerAdmin root@server.example.com
    DocumentRoot "/var/www/files"
    ServerName server.example.com
    ErrorLog "/var/log/httpd/server.example.com-error_log"
    CustomLog "/var/log/httpd/server.example.com-access_log" common
</VirtualHost>
<Directory /var/www/files>
require all granted
</Directory>
~
```

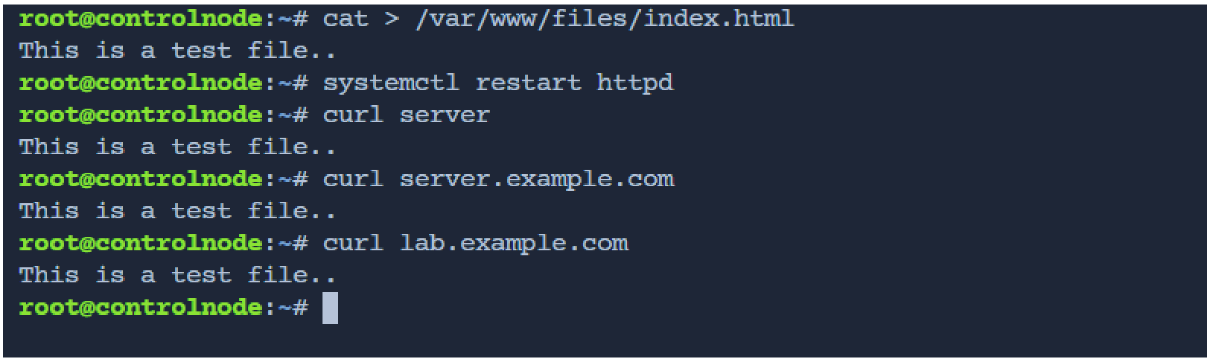
Restart and enable the httpd service and also add the service in the firewall



```
root@controlnode:~# systemctl restart httpd
root@controlnode:~# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
root@controlnode:~# firewall-cmd --permanent --add-service=http
success
root@controlnode:~# firewall-cmd --reload
success
root@controlnode:~#
```

To verify that our configuration is working create a **index.html** file in the directory **/var/www/files**

After that ping to the domain names we configure previously



```
root@controlnode:~# cat > /var/www/files/index.html
This is a test file..
root@controlnode:~# systemctl restart httpd
root@controlnode:~# curl server
This is a test file..
root@controlnode:~# curl server.example.com
This is a test file..
root@controlnode:~# curl lab.example.com
This is a test file..
root@controlnode:~#
```

We can see that it is working properly.