Apache Server (http,https) Objectives Tasks in PDF file

Apache Server Objectives (http,https)

1. Configure a virtual host **http://rhce.example.com** on server machine using the default location of html file. On opening on browser, it should display "**Welcome to Virtual Host**".

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

ls -ldZ /var/www/html **(To verify SELINUX context type set on Default Document Root,It must be httpd_sys_content_t)**

vim /var/www/html/index.html **(To create index.html file under Document Root)**

Welcome to Virtual Host

:wq

restorecon -Rv /var/www/html **(To restore context on index.html file)**

firewall-cmd --add-service=http --permanent **(To configure firewall to accept inbound http traffic)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

vim /etc/httpd/conf.d/rhce.conf **(To define Virtual Host)**

```
<VirtualHost *:80>
        ServerName rhce.example.com
        ServerAdmin root@rhce.example.com
        DocumentRoot /var/www/html
        ErrorLog logs/rhce_error.log
        CustomLog logs/rhce_custom.log combined
</VirtualHost>
```

…..continued on next page

:wq

httpd -t (To check syntax of rhce.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://rhce.example.com (To access the Web page contents as defined in index.html file)

You can also open Firefox browser and type http://rhce.example.com to display the contents.

I assume you have DNS configured and all Host Names ,Web Server names can be resolved by DNS,if not make entries in /etc/hosts file on all machines to resolve the hostnames locally.

Make sure LAB machines hostname entries are above than virtual host/web server hostname entries.

In case you are using the DNS, don't configure reverse records for Virtual Hostnames/Web Server Names but you must create reverse records for LAB machine hostnames because Reverse DNS lookup is involved while checking host access configured by Require Directive.

2. Configure web server **http://web.example.com** on server , which should display the contents of html file located under **/web** directory.
   a) Only hosts on **example.com** domain should be able to access this server.
   b) Access should be denied for everyone else.

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir /web **(To Create non-default Document root directory)**

vim /web/index.html **(To create index.html file under non-default Document Root)**

Welcome to Web Server

:wq

semanage fcontext -a -t httpd_sys_content_t "/web(/.*)?" **(To set SELINUX context type Document Root Directory)**

restorecon -Rv /web **(To restore context on index.html file)**

firewall-cmd --add-service=http --permanent **(To configure firewall to accept inbound http traffic)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

vim /etc/httpd/conf.d/web.conf **(To define Web Server)**

```
<VirtualHost *:80>
        ServerName web.example.com
        ServerAdmin root@web.example.com
        DocumentRoot /web
        ErrorLog logs/web_error.log
        CustomLog logs/web_custom.log combined
</VirtualHost>
```

…..continued on next page

<Directory “/web”> (To configure access for example.com domain)

AllowOverride none

Require host example.com

</Directory>

:wq (To save and quit)

httpd -t (To check syntax of web.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://web.example.com (To access the Web page contents as defined in index.html file)

Test from different machines on example.com domain and it should work.

3. Setup a virtual host **http://vhost1.example.com/riya** on server with the alternate document root under **/web/www/vhost1**. Create a directory "**riya**" under the document root used for the website.

   a) The webpage should say **"vhost1".**

   b) The webpage must be configured for user-based authentication. Only user "**riya**" should be allowed to login with a password "**password**".

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir -p /web/www/vhost1 **(To Create non-default Document root directory)**

mkdir /web/www/vhost1/riya **(To create riya directory under document root)**

vim /web/www/vhost1/riya/index.html **(To create index.html file)**

vhost1

:wq

semanage fcontext -a -t httpd_sys_content_t "/web/www/vhost1 (/.*)?" **(To set SELINUX context type Document Root Directory)**

restorecon -Rv /web/www/vhost1 **(To restore context on index.html file)**

firewall-cmd --add-service=http --permanent **(To configure firewall to accept inbound http traffic)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

htpasswd -c /etc/htpasswd riya **(To create file for riya for basic authentication, check man page for htpasswd)**

Provide password – password **(As per task requirements)**

vim /etc/httpd/conf.d/vhost.conf (To define Virtual Host)

```
<VirtualHost *:80>
        ServerName vhost1.example.com
        ServerAdmin vhost1@vhost1.example.com
        DocumentRoot /web/www/vhost1
        ErrorLog logs/vhost1_error.log
        CustomLog logs/vhost1_custom.log combined
</VirtualHost>
<Directory "/web/www/vhost1/riya">     (To configure Authentication for user "riya" )
        AllowOverride none
        AuthType Baisc
        AuthName "Riya Private Directory"
        AuthBasicProvider file
        AuthUserFile /etc/htpasswd
        Require user riya
</Directory>
```

:wq (To save and quit)

httpd -t (To check syntax of vhost.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://vhost1.example.com/riya (it should ask for username and password and after entering riya credentials ,Web page should be shown)

4. Implement a website for **http://group.example.com/group**. Create a directory "group" under the document root used for the website. The webpage should say "**Welcome to Group**".

   a) The webpage must be configured for group-based authentication and require users to login.

   b) Users "**harry**" and "**riya**" who belong to group "**operations**" should be able to access the share with password "**access**" .

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir /var/www/html/group **(To create group directory under document root)**

vim /var/www/html/group/index.html **(To create index.html file under group directory under Document Root)**

Welcome to Group

:wq

restorecon -Rv /var/www/html/ **(To restore context on index.html file)**

firewall-cmd --add-service=http --permanent **(To configure firewall to accept inbound http traffic)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

htpasswd -c /etc/htpasswd1 riya **(To create file for riya for basic authentication, check man page for htpasswd)**

htpasswd /etc/htpasswd1 harry **(To add user harry for basic authentication, check man page for htpasswd)**

vim /etc/group1 **(To create group file to be used to configure group authentication)**

operations : riya harry

:wq

vim /etc/httpd/conf.d/group_auth.conf (To define Web Site in Virtual Host container)

```
<VirtualHost *:80>
        ServerName group.example.com
        ServerAdmin root@group.example.com
        DocumentRoot /var/www/html
        ErrorLog logs/group_error.log
        CustomLog logs/group_custom.log combined
 </VirtualHost>
<Directory "/var/www/html/group">
        AllowOverride none
        AuthType Baisc
        AuthName "Group Directory"
        AuthBasicProvider file
        AuthUserFile /etc/htpasswd1
        AuthGroupFile /etc/group1
        Require group operations
 </Directory>
```

:wq (To save and quit)

httpd -t (To check syntax of group_auth.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://group.example.com/group (it should ask for username and password and only Group members should be allowed access)

5. Configure a Virtual host on Server **http://client12_web.rhce.com** with document root at **/web/client/html**.

a) When Browsing, It should display "**Welcome to Client Web Server**"

b) Only host **client.example.com** and **client2.example.com** should be allowed access for this site.

c) In addition **server.example.com** should also allowed access and rest all should be denied access.

d) User "**riya**" should be able to modify the contents of html file.

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir -p /web/client/html **(To Create non-default Document root directory)**

vim /web/client/html/index.html **(To create index.html file under non-default Document Root)**

Welcome to Client Web Server

:wq

semanage fcontext -a -t httpd_sys_content_t "/web/client/html(/.*)?" **(To set SELINUX context type Document Root Directory)**

restorecon -Rv /web/client/html **(To restore context on index.html file)**

firewall-cmd --add-service=http --permanent **(To configure firewall to accept inbound http traffic)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

setfacl -R -m u:riya:rwx /web/client/html/index.html **(To give riya full access on index.html file as per task requirement)**

vim /etc/httpd/conf.d/client.conf (To define Web Site in Virtual Host container)

```
<VirtualHost *:80>
        ServerName client12_web.rhce.com
        ServerAdmin root@client12_web.rhce.com
        DocumentRoot /web/client/html
        ErrorLog logs/web_client12_error.log
        CustomLog logs/web_client12_custom.log combined
 </VirtualHost>
<Directory "/web/client/html">     (To configure access for non-default document root)
        AllowOverride none
       <RequireAny>
            Require host server.example.com
            Require host client.example.com
            Require host client2.example.com
         </RequireAny>
</Directory>
```

:wq (To save and quit)

httpd -t (To check syntax of client.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://client12_web.rhce.com (Verify the access from different hosts)

6. Configure Web server on Server host **http://client1_web.rhce.com** with document root at **/client/web/html**.

a) When Browsing, It should display "**You are authorized to access this Website**".

b) **client.example.com** should be allowed access for this site and **client2.example.com** should be denied access.

c) All other hosts should be denied access for website.

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir -p /client/web/html **(To Create non-default Document root directory)**

vim /client/web/html/index.html **(To create index.html file under non-default Document Root)**

You are authorized to access this Website

:wq

semanage fcontext -a -t httpd_sys_content_t "/client/web/html(/.*)?" **(To set SELINUX context type Document Root Directory)**

restorecon -Rv /client/web/html **(To restore context on index.html file)**

firewall-cmd --add-service=http --permanent **(To configure firewall to accept inbound http traffic)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

vim /etc/httpd/conf.d/client1.conf (To define Web Site in Virtual Host container)

```
<VirtualHost *:80>
        ServerName client1_web.rhce.com
        ServerAdmin root@client1_web.rhce.com
        DocumentRoot /client/web/html
        ErrorLog logs/web_client1_error.log
        CustomLog logs/web_client1_custom.log combined
 </VirtualHost>
<Directory "/client/web/html">
        AllowOverride none
       <RequireAll>
            Require host client.example.com
            Require not host client2.example.com
         </RequireAll>
</Directory>
```

:wq (To save and quit)

httpd -t (To check syntax of client1.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://client1_web.rhce.com (Verify the access from different hosts,It should be allowed from client but denied from client2 machine and also rest world)

7. Configure access for Directory **/dir/access** with below conditions :

a) Only example.com domain should be allowed to access this.

b) Domain insecure.com should be denied access to this.

c) Rest all world should be denied access.

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir -p /dir/access **(To Create directory)**

semanage fcontext -a -t httpd_sys_content_t "/dir/access(/.*)?" **(To set SELINUX context type on Directory)**

restorecon -Rv /dir/access **(To restore context)**

vim /etc/httpd/conf.d/directory.conf

<Directory "/dir/access"> **(To configure access for directory)**

```
        AllowOverride none
        <RequireAll>
            Require host example.com
            Require not host insecure.com
        </RequireAll>
</Directory>
```

httpd -t **(To check the syntax)**

8. Configure Virtual host **http://nondef_port.example.com** on server.example.com using the non-default document root **/nondef/html.**

   a> Virtual host must listen on **non-standard port 7654**

   b> Access must be restricted to **local host**.

   c> On browsing ,it should say " **Virtual Host is using non-standard listening port**".

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir -p /nondef/html **(To Create non-default Document root directory)**

vim /nondef/html/index.html **(To create index.html file under non-default Document Root)**

Virtual Host is using non-standard listening port

:wq

semanage fcontext -a -t httpd_sys_content_t "/nondef/html(/.*)?" **(To set SELINUX context type Document Root Directory)**

restorecon -Rv /nondef/html **(To restore context on index.html file)**

semanage port -a -t http_port_t -p tcp 7654 **(To set Correct SELINUX context type on non-default port)**

firewall-cmd --add-port=7654/tcp --permanent **(To configure firewall to accept inbound http traffic on non-standard port)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

vim /etc/httpd/conf.d/nondef.conf (To define Virtual Host in Virtual Host container)

Listen 7654 (To define Apache server to listen on non-standard port)

```
<VirtualHost *:7654>
        ServerName nondef_port.example.com
        ServerAdmin root@nondef_port.example.com
        DocumentRoot /nondef/html
        ErrorLog logs/nondef_port_error.log
        CustomLog logs/nondef_port_custom.log combined
</VirtualHost>
<Directory "/nondef/html">     (To configure Group Authentication)
        AllowOverride none
        Require local
</Directory>
```

:wq (To save and quit)

httpd -t (To check syntax of client.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://nondef_port.example.com:7654 (To Display the web page severed by Document root, Don't forget to specify non default port here)

9. Configure dynamic content on website **http://dynamic.example.com:7655/cgi-bin/hello.pl** to execute **hello.pl** CGI script.

a> Directory **/dynamic/content** should serve the script.

b> Web Server should listen on **port 7655**

c> Copy the hello.pl script from **/root/script**.

d> Configure the server for static contents with document root at **/static/content**.

e>Use the **lara.html** file for static content and it should display " **This web server is configured for dynamic contents too!!!!!"**

f> Firewall should accept connection for this web server from remote hosts.

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir -p /dynamic/content **((To Create non-default Document root to service dynamic contents)**

cp /root/script/hello.pl /dynamic/content **(To copy CGI script from mentioned path to /dynamic/content Directory)**

semanage fcontext -a -t httpd_sys_script_exec_t "/dynamic/content(/.*)?" **(To set correct SELINUX context type on Directory)**

restorecon -Rv /dynamic/content **(To restore context on hello.pl file)**

chmod a+x /dynamic/content/hello.pl **(To give execution right on script)**

getsebool -a | grep cgi **(To verify httpd_enable_cgi Boolean is on,if not set it to on with setsebool -P command)**

mkdir -p /static/content **(To Create non-default Document root to serve static contents)**

vim /static/content/lara.html

This Web site is used for dynamic contents too!!!

:wq

semanage fcontext -a -t httpd_sys_content_t /static/content(/.*)?” (To set correct SELINUX context type Document Root Directory)

restorecon -Rv /static/content (To restore context on lara.html file)

semanage port -a -t http_port_t -p tcp 7655 (To set Correct SELINUX context type on non-default port)

firewall-cmd --add-port=7655/tcp --permanent (To configure firewall to accept inbound http traffic on non-standard port)

firewall-cmd --reload (To reload the firewall to make the changes effective)

…….Continued on next page

vim /etc/httpd/conf.d/dynamic.conf (To define Virtual Host in Virtual Host container)

Listen 7655 (To define Apache server to listen on non-standard port)

```
<VirtualHost *:7655>
        ServerName dynamic.example.com
        DocumentRoot /static/content
        ScriptAlias /cgi-bin/ "/dynamic/content/"
<Directory "/static/content">
        AllowOverride none
        DirectoryIndex lara.html
        Require all granted
</Directory>
 <Directory "/dynamic/content">
        AllowOverride none
        Require all granted
</Directory>
 </VirtualHost>
```

:wq (To save and quit)

httpd -t (To check syntax of dynamic.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://dynamic.example.com:7655 (To Display the static contents)

elinks http://dynamic.example.com:7655/cgi-bin/hello.pl (To Display the dynamic contents)

10. Configure dynamic content on website **http://dynamicwsgi.example.com/mywsgi/hello.wsgi** to serve wsgi script.

a> Directory **/wsgi/content** should serve the script.

b> Copy the **hello.wsgi** script from **/root/script**.

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir /wsgi/content **(To Create Directory to service dynamic contents)**

cp /root/script/hello.wsgi /wsgi/content/ **(To copy the WSGI script from mentioned path to /wsgi/content directory)**

semanage fcontext -a -t httpd_sys_content_t "/wsgi/content(/.*)?" **(To set correct SELINUX context type on Directory)**

restorecon -Rv /wsgi/content **(To restore context on hello.wsgi script file)**

chmod a+x /wsgi/content/hello.wsgi **(To give execution right on script)**

Note for this case context type httpd_sys_content_t is set on wsgi script not like cgi script !!!! First difference

vim /etc/httpd/conf.d/wsgi.conf (To define Virtual Host in Virtual Host container)

<VirtualHost *:80>

ServerName dynamicwsgi.example.com

WSGIScriptAlias /mywsgi "/wsgi/content" (Second difference from CGI script)

<Directory "/wsgi/content">

AllowOverride none

Require all granted

</Directory>

</VirtualHost>

:wq (To save and quit)

yum install mod_wsgi (Install this package otherwise WSGIScriptAlias directive will not be recognised –Third difference!!!)

httpd -t (To check syntax of wsgi.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

yum install elinks (To install Text browser)

elinks http://dynamicwsgi.example.com/mywsgi/hello.wsgi (To Display the dynamic contents)

11. Configure website **https://ssl.example.com** with TLS. Generate a self-signed certificate, the only requirement for the certificate is to match the webserver name **ssl.example.com**.

a> Make sure that SSLv2 and SSLv3 protocols are disabled.

b> The content of the websites should be visible to everyone browsing from the localhost but should not be accessible from any other location.

c> The webpage should say "**Welcome to Secure Server**". Use directory **/web/secure** as document root.

Commands: (On server.example.com)

yum groups install "Basic Web Server" **(To install Apache Web Service)**

systemctl start httpd.service **(To start Apache Web Service)**

systemctl enable httpd.service **(To Enable Apache Web Service to Start at Boot)**

mkdir -p /web/secure **(To Create non-default Document root)**

vim /web/secure/index.html

Welcome to Secure Server

:wq

semanage fcontext -a -t httpd_sys_content_t "/web/secure(/.*)?" **(To set correct SELINUX context type on Directory)**

restorecon -Rv /web/secure **(To restore context on index.html file)**

firewall-cmd --add-service=https --permanent **(To configure firewall to accept inbound https traffic)**

firewall-cmd --reload **(To reload the firewall to make the changes effective)**

keygen ssl.example.com **(To generate server certificate and private key)**

```
The key will be stored in
    /etc/pki/tls/private/ssl.example.com.key
The certificate stored in
    /etc/pki/tls/certs/ssl.example.com.crt
```

Press next ,choose medium security and press next. It will take some time for Certificate file and private file key generation which will be stored on path as shown in snapshot above.

vim /etc/httpd/conf.d/sslweb.conf **(To define Virtual Host in Virtual Host container)**

```
<VirtualHost *:443>
        ServerName ssl.example.com
        ServerAdmin root@ssl.example.com
        DocumentRoot /web/secure
        SSLEngine on
        SSLProtocol all -SSLv2 -SSLv3
        SSLCertificateFile /etc/pki/tls/certs/ssl.example.com.crt
        SSLCertificateKeyfile /etc/pki/tls/private/ssl.example.com.key
    <Directory "/web/secure">
        AllowOverride none
        Require local
   </Directory>
</VirtualHost>
```

:wq **(To save and quit)**

httpd -t (To check syntax of sslweb.conf file and make sure everything is fine)

systemctl restart httpd.service (To restart Web server to make the changes effective)

Open the firefox and access the secure server using below url :

https://ssl.example.com

Don’t use text browser for this task as it does not work with secure sites.

12. Configure TLS encryption for the web server “http://ssl.example.com”

a) A signed certificate for the web server is server.crt file in /root directory

b) The required key for this certificate file is server.key in /root directory

c) The certificate for signing authority is example-ca.crt in /root directory

Task is same as last task ,only one extra directive **SSLCertificateChainFile** is used in this task (Refer to video)