**RHCE in RHEL-7 DEMO QUESTION & SOLUTION**

**=========================**

**01. SELinux must be running in enforcing mode.**

#vim /etc/selinux/config or vim /etc/sysconfig/selinux

SELINUX=enforcing

:wq!

#reboot

#getenforce

**02. Create a repository by using the following URL http://classroom.example.com/pub/rhel7/dvd/**

#vim /etc/yum.repos.d/yum.repo

[yum]

name=yum.repo

baseurl=http://classroom.example.com/pub/rhel7/dvd/

enabled=1

gpgcheck=0

:wq!

#yum clean all

#yum repolist all

**03. SSH Configuration.**

**-Configure ssh access on your virtual hosts as follows**

**-Client within hacker.org should NOT have access to ssh on your system**

[By default it may configure ssh server, IF NOT ]

#yum install openssh-server -y

#systemctl start sshd.service

#systemctl enable sshd.service

#firewall-cmd --permanent --add-service=ssh

#firewall-cmd --reload

#firewall-cmd --list-all

#firewall-config

Will open Firewall GUI mode then Select public go to Configuration & select Permanent-->Rich Rule-->Add

Family:ipv4

Element:service ssh

Action: reject

Source: 172.16.0.0/24(Already provided this IP for hacker.org)

Press OK & OK

Option--> Reload Firewalld

or File--> Quit & #firewall-cmd–reload

#firewall-cmd --list-all

**04. Configure serverX to forward traffic incoming on port 5243/tcp from source network 172.25.x.0/255.255.255.0 to port on 80/tcp**

#firewall-config

Will open Firewall GUI mode then Select public go to Configuration & select Permanent-->Rich Rule-->Add

Family:ipv4

Element:service forward-port

Protocol:tcp, Port / Port Range:5243 select Local Forwarding

Port/Port Range:80

OK

Source: 172.16.10.10/24 (where X=StationNumer)

Ok

Option--> Reload Firewalld

or File--> Quit & #firewall-cmd–reload

**05. Customize user environment**

**-Create a command called qstat on both serverX & desktopX**

**-It should be able to execute the following command**

**-(ps-eo,pid,tid,class,rtprio,ni,pri,psr,pcpu,stat,com)**

#vim /etc/bashrc or **vim /etc/profile**

qstat ()

{

ps-eo,pid,tid,class,rtprio,ni,pri,psr,pcpu,stat,com

}

:wq!

#source /etc/bashrc or **source /etc/profile**

#qstat[For Check]

**06. Configure ipv6 network**

**-Configure eth0 with a static ipv6 address as follows**

**-Configure a static IP address in serverX as fddb:fe2a:ab1e::c0a8:64/64**

**-Configure a static IP address in serverX as fddb:fe2a:ab1e::c0a8:02/64**

**-Both machine can be communicates with fddb:fe2a:ab1e/64**

#nmcli connection show

#nmcli connection modify "System eth0" ipv6.addresses fddb:fe2a:ab1e::c0a8:64/64 ipv6.method static connection.autoconnect yes **[By default connection.autoconnect yes]**

#nmcli connection down "System eth0" ;nmcli connection up "System eth0"

#ifconfig -a [For Check]

#ping6 fddb:fe2a:ab1e::c0a8:02 [For Check]

N.B: Same way in desktopX machine also.

**07. Link aggregation**

**-Configure your systems which watches for link changes & selects an active port for data transfers.**

**-Systems should have the address 192.168.0.10/255.255.255.0**

#lab teambridge setup [Only for lab environment]

#ifconfig -a [shows how many LAN port’s are available in system]

#nmcli connection show [For Check]

#nmcli connection add con-name team0 type team ifname team0 config '{"runner": {"name": "activebackup"}}'

#nmcli connection add con-name team-port1 type team-slave ifname eno1 master team0

#nmcli connection add con-name team-port2 type team-slave ifname eno2 master team0

#teamdctl team0 state

[For Check: To know the state of **team-port1,** **team-port2 & teamdctl team-port1 state**]

#nmcli connection show [For Check]

#nmcli connection modify team0 ipv4.addresses 192.168.10.10/24 ipv4.method static connection.autoconnect yes

#nmcli restart NetworkManager [To get this ip192.168.10.10/24 live]

#nmcli connection up team-port1

#nmcli connection up team-port2

#nmcli connection up team0

#ifconfig -a [For Check]

#teamdctl team0 state [For Check]

N.B: Same way in desktopX machine also.

**08. Your ServerX system should accept new email Message over SMTP from 172.25.X.0/24 subnet. All Message not addressed to @serverX.example.com or @localhost[.localdomain] should be forwarded to the SMTP smarthost running on desktoX.example.com**

#yum install postfix –y

#firewall-cmd –permanent –add-service=smtp

#firewall-cmd –reload

#firewall-cmd –list-all

#systemctl start postfix

#systemctl enable postfix

#systemctl restart postfix

#cd /etc/postfix/

#cp main.cf main.cf.ori

#vim main.cf

myhostname= server7.example.com

mydomain = example.com

myorigin = $myhostname

or

myorigin = $mydomain

inet\_interfaces = all

inet\_protocols = all

mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain,

mynetworks = 192.168.44.132/24 127.0.0.0/8

relayhost = desktop7.example.com

wq!

Or

#vim main.cf

#inet\_interfaces = all

#mynetworks = 172.25.7.0/24 127.0.0.0/8

#mydestination =server7.example.com, localhost, localhost.localdomain

# relayhost = desktop7.example.com

wq!

#systemctl restart postfix

[Test Mail Server}

#mail -v student(another systemuser)

Subject:.....

Mail Body:......

.

#su student (systemuser)

#tail -f /var/spool/mail/student

**09. Configure the SMTP mail service on desktopX which relay the mail only from local system through classroom.example.com, all outgoing mail has their sender domain as example.com. Ensure that mail should not store locally.**

**- Verify the mail server is working by sending mail to a natasha user.**

**- Check the mail on desktopX with the below URL** [**http://content.example.com/system2**](http://content.example.com/system2)

#yum install postfix –y

#firewall-cmd –permanent –add-service=postfix

#firewall-cmd –reload

#firewall-cmd –list-all

#systemctl start postfix

#systemctl enable postfix

#systemctl restart postfix

#cd /etc/postfix/

#cp main.cf main.cf.ori

#vim main.cf

myhostname = desktop7.example.com

mydomain = example.com

myorigin = $myhostname

or

myorigin = $mydomain

inet\_interfaces = all

inet\_protocols = all

mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain,

mynetworks = 192.168.44.133/24 127.0.0.0/8

relayhost = classroom.example.com

:wq!

[Test Mail Server}

#mail -v natasha@example.com(another domainuser)

Subject:.....

Mail Body:......

.

#mailq

**10. Implement a webserver for the site** [**http://serverX.example.com**](http://serverX.example.com) **Download the webpage from** [**http://content.example.com/pub/rhce/rhce.html**](http://content.example.com/pub/rhce/rhce.html)**. Rename the downloaded file into index.html copy the fileinto your webserver document root. Don not make any modification with the content of the index.html. Clients within hacker.org should NOT access the webserver on your systems.**

#yum install httpd –y

#firewall-cmd –permanent –add-service=http

#firewall-cmd –reload

#firewall-cmd –list-all

#systemctl start httpd [no need to run this]

#systemctl enable httpd

#systemctl restart httpd

#cd /etc/httpd/conf/

#cphttpd.confhttpd.conf.ori

#vim httpd.conf

ServerName server7.example.com:80

#cd /var/www/html

#wget<http://content.example.com/pub/rhce/rhce.html>

#mv rhce.html index.html

#systemctl restart httpd

#curl server7.example.com [for check]

#firewall-config

Will open Firewall GUI mode then Select public go to Configuration & select Permanent-->Rich Rule-->Add

Family:ipv4

Element:service http

Action: reject

Source: 172.16.0.0/24

Press OK & OK

Option--> Reload Firewalld

or File--> Quit & #firewall-cmd –reload

#firewall-cmd --list-all

**11. Configure the website** [**https://serverX.example.com**](https://serverX.example.com) **with TLS SSLCertificate file http://classroom.example.com /pub/tls/certs/server.crt**

**SSLCertificatekeyfile** [**http://classroom.example.com/pub/tls/private/serverX.key**](http://classroom.example.com/pub/tls/private/serverX.key)

**SSL CA certificate file http://classroom.example.com/pub/example-ca.crt**

#yum install mod\_ssl -y

#cd /etc/httpd/conf.d

#cp ssl.conf ssl.conf.ori

#vim ssl.conf

Find: /**SSLCertificateFile** text and copy the link location of the **SSLCertificateFile** then **wget** the provided link content on the provided location. After that rename the downloaded file same with provided link text to the station number of the **serverX.crt**

Find: **/SSLCertificateKeyFile**text and copy the link location of the **SSLCertificateKeyFile** then **wget** the provided link content on the provided location. After that rename the downloaded file same with provided link text to the station number of the **serverX.key**

Then uncomment **SSLCertificateChainFile** or **SSLCACertificateFile** and similarly copy the link location of the **SSLCertificateChainFile** or **SSLCACertificateFile** then **wget** the provided link content on the provided location. After that change the link text same with the downloaded file to station number of the **server-chain.crt** or **ca-bundle.crt**

#firewall-cmd –permanent –add-port=443/tcp

#firewall-cmd –permanent –add-service=https

#firewall-cmd –reload

#firewall-cmd –list-all

#systemctl start httpd

#systemctl enable httpd

#systemctl restart httpd

[check the webpage link started with https://serverX.example.com]

**12. Create a folder topsecret in your web server Document Root Download http://content.example.com/pub/thce/restrict.html Rename the file into index.html. The content of topsecret should be visible to everyone browsing from your localsystem but should not accessible from other location.**

#mkdir /var/www/html/topsecret

#cd /var/www/html/topsecret

#wgethttp://content.example.com/pub/thce/restrict.html

#mv restrict.html index.html

#vim /etc/httpd/conf.d/vhosts.conf

<VirtualHost serverX.example.com:80>

ServerName serverX.example.com

DocumentRoot /var/www/html

</VirtualHost>

<Directory /var/www/html/topsecret>

Require host serverX.example.com

</Directory>

:wq!

#systemctl start httpd

#systemctl enable httpd

#systemctl restart httpd