

Port Forwarding

Server Machine:

```
[root@server ~]# vim /etc/httpd/conf/httpd.conf
[root@server ~]#
```

```
root@sakshi:~ — vim /etc/httpd/conf/httpd.conf
# Mutex directive, if file-based mutexes are used.  If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default.  See also the <VirtualHost>
# directive.
#
# Change this to Listen on a specific IP address, but note that if
# httpd.service is enabled to run at boot time, the address may not be
# available when the service starts.  See the httpd.service(8) man
# page for more information.
#
#Listen 12.34.56.78:80
Listen 8080
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by 'httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf

-- INSERT --
```

Add Service:

```
[root@server ~]# systemctl restart httpd
[root@server ~]# firewall-cmd --add-service=http --permanent
success
[root@server ~]# firewall-cmd --reload
success
[root@server ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client http ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
[root@server ~]#
```

Add Port:

```
[root@server ~]# firewall-cmd --add-port=8080/tcp --permanent
success
[root@server ~]# firewall-cmd --reload
success
[root@server ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client http ssh
  ports: 8080/tcp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
[root@server ~]#
```

Client Machine: check with port number but not with ip address of server.

```
[root@client ~]# curl 192.168.0.3:8080
<h1> Hello, This is client machine
</h1>

[root@client ~]# curl 192.168.0.3
curl: (7) Failed to connect to 192.168.0.3 port 80: Connection refused
[root@client ~]#
```

Server Machine:

```
[root@server ~]# firewall-cmd --add-forward-port=port=80:proto=tcp:toport=8080 --permanent
success
[root@server ~]# firewall-cmd --reload
success
[root@server ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3 enp0s8
  sources:
  services: cockpit dhcpv6-client http ssh
  ports: 8080/tcp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  port=80:proto=tcp:toport=8080:toaddr=
  source-ports:
  icmp-blocks:
  rich rules:
[root@server ~]#
```

Client Machine: Able to check with port as well as Ip address.

```
curl: (7) Failed to connect to 192.168.0.3 port 80: Connection refused
[root@client ~]# curl 192.168.0.3
<h1> Hello, This is client machine
</h1>

[root@client ~]# curl 192.168.0.3:8080
<h1> Hello, This is client machine
</h1>

[root@client ~]#
```

Server Machine:

```
[root@server ~]# firewall-cmd --remove-forward-port=port=80:proto=tcp:toport=8080 --permanent
success
[root@server ~]# firewall-cmd --reload
success
[root@server ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client http ssh
  ports: 8080/tcp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
[root@server ~]#
```