

Apache HTTP Server Configuration on Non-Default Port

Overview

This project documents the configuration of the Apache HTTP Server to run on a non-default port (8001) in a secured Linux environment. The workflow demonstrates how application configuration, SELinux enforcement, firewall rules, and service validation work together to safely expose a web service.

What Was Done

Modified the Apache configuration file to change the listening port from the default (80) to 8001

Verified and configured SELinux to allow Apache to bind to the new port using the appropriate `http_port_t` context

Updated firewall rules to permit inbound TCP traffic on port 8001

Restarted the Apache service to apply configuration changes

Confirmed the service was active and running without errors

Validated successful access to the web service using curl on the new port

Result

Apache was successfully reconfigured to listen on port 8001 while maintaining system security controls. The service started cleanly, passed all validation checks, and served web content as expected on the new port.

Configuration Explanation

This screen shows the Apache HTTP Server configuration file being edited on a Linux system. The `httpd.conf` file is opened in a terminal text editor, and the `Listen` directive is being modified to change the port on which Apache accepts incoming connections. By default, Apache listens on port 80; in this configuration, the service is updated to listen on an alternate port (8001).

Changing the listening port allows the web server to run alongside other services or comply with environment-specific requirements without conflicting with the default HTTP port. This change requires corresponding updates to system security settings, such as firewall rules and SELinux port labeling, to allow traffic on the new port. After saving the configuration, the Apache service must be restarted for the new settings to take effect.

```
root@dev-performance-eg3: ~  
# 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.  
#  
# 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.  
#  
# Do not add a slash at the end of the directory path. If you point  
# ServerRoot at a non-local disk, be sure to specify a local disk on the  
# 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 specific IP addresses as shown below to  
# prevent Apache from glomming onto all bound IP addresses.  
#  
#Listen 12.34.56.78:80  
#Listen 80  
Listen 8001  
-- INSERT --
```

Apache Port Configuration and Verification

This screen shows the process of configuring the Apache HTTP Server to listen on a non-default port (8001) and validating that the service is running correctly. The Apache configuration file (/etc/httpd/conf/httpd.conf) is edited to update the Listen directive, allowing Apache to bind to the new port.

SELinux is then configured to permit Apache to use port 8001 by associating it with the http_port_t context using semanage port. The output confirms that the port is already properly labeled, ensuring SELinux enforcement does not block the service. The system firewall is also updated to allow inbound traffic on TCP port 8001, and the firewall configuration is reloaded to apply the change.

After applying these security updates, the Apache service is restarted to load the new configuration. The service status is checked to confirm that httpd is active and running without errors. Finally, local access is verified using curl http://localhost:8001, which returns the expected HTML output, confirming that Apache is successfully serving content on the new port.

```
root@dev-performance-eg3: ~  
[root@dev-performance-eg3 ~]# vi /etc/httpd/conf/httpd.conf  
[root@dev-performance-eg3 ~]# semanage port -m -t http_port_t -p tcp 8001  
[root@dev-performance-eg3 ~]# semanage port -l | grep http_port_t  
http_port_t          tcp      8001, 80, 81, 443, 488, 8008, 8009, 8443, 9000  
pegasus_http_port_t  tcp      5988  
[root@dev-performance-eg3 ~]# firewall-cmd --add-port=8001/tcp --perm  
Warning: ALREADY_ENABLED: 8001:tcp  
success  
[root@dev-performance-eg3 ~]# firewall-cmd --reload  
success  
[root@dev-performance-eg3 ~]# systemctl restart httpd  
[root@dev-performance-eg3 ~]# systemctl status httpd  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)  
   Active: active (running) since Wed 2025-09-17 20:01:33 EDT; 7s ago  
     Docs: man:httpd(8)  
           man:apachectl(8)  
  Process: 31613 ExecStop=/bin/kill -WINCH ${MAINPID} (code=exited, status=0/SUCCESS)  
  Process: 10704 ExecReload=/usr/sbin/httpd $OPTIONS -k graceful (code=exited, status=0/SUCCESS)  
 Main PID: 31618 (httpd)  
   Status: "Processing requests ..."  
    CGroup: /system.slice/httpd.service  
            └─31618 /usr/sbin/httpd -DFOREGROUND  
              └─31621 /usr/sbin/httpd -DFOREGROUND  
                └─31622 /usr/sbin/httpd -DFOREGROUND  
                  └─31623 /usr/sbin/httpd -DFOREGROUND  
                    └─31624 /usr/sbin/httpd -DFOREGROUND  
                      └─31625 /usr/sbin/httpd -DFOREGROUND  
  
Sep 17 20:01:27 dev-performance-eg3.procore.prod1 systemd[1]: Stopped The Apache HTTP Server.  
Sep 17 20:01:27 dev-performance-eg3.procore.prod1 systemd[1]: Starting The Apache HTTP Server...  
Sep 17 20:01:33 dev-performance-eg3.procore.prod1 systemd[1]: Started The Apache HTTP Server.  
[root@dev-performance-eg3 ~]# curl http://localhost:8001  
<!DOCTYPE html>  
<html lang="en">  
  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <meta http-equiv="X-UA-Compatible" content="ie=edge">  
  <title>Product Admin - Dashboard HTML Template</title>  
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto:400,700">
```

Summary

Apache was reconfigured to listen on a non-default port (8001) by updating the HTTP server configuration. SELinux was validated and configured to allow Apache to bind to the new port, and firewall rules were applied to permit inbound traffic. The Apache service was restarted and verified as running successfully. Connectivity was confirmed by accessing the web service locally using curl, validating that Apache is serving content correctly on the new port.