Manage HTTP Web Access Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

Session - 45 Agenda:

Manage HTTPD Web Access Using Ansible:

1. Configuration of name-based Apache virtual hosts.
2. Configuration of alternate document root for Apache.
3. Configuration of an alternate web access port for Apache.

1. Configuration of name-based Apache virtual hosts:

Let's create an ansible playbook to setup name-based Apache web server on the managed node(s).

```
$ vim name-based-apache-server.yml
---
- name: Name Based Apache HTTP Server Configuration Playbook
  hosts: node1
  become: true
  tasks:
    - name: Setting up the static hostname in the server machine.
      hostname:
        name: node1.nehraclasses.com
        use: systemd

    - name: Making entries in the /etc/hosts file for the server hostname & IP Address
      lineinfile:
        dest: /etc/hosts
        line: 192.168.229.129 www.nehraclasses.com node1
        insertafter: EOF

    - name: Installing Apache packages in the machine.
      dnf:
        name:
          - httpd
        state: latest

    - name: Copying the image file to the /var/www/html/ directory.
      ansible.builtin.copy:
        src: /home/vikasnehra/NehraClassesLogo.png
        dest: /var/www/html/NehraClassesLogo.png
        mode: '0644'

    - name: Creating the website index file in the /var/www/html/ directory.
      copy:
        dest: "/var/www/html/index.html"
        content: |
          <img src="/NehraClassesLogo.png" width="500" height="500" />
          <h1>Nehra Classes Are Awesome.</h1>
          <i>This page is hosted on node1 using name-based Apache virtual hosting.</i>

    - name: Creating HTTPD configuration file in the /etc/httpd/conf.d/ directory.
      copy:
        dest: "/etc/httpd/conf.d/httpd.conf"
        content: |
          <VirtualHost 192.168.229.229:80>
          ServerAdmin root@node1.nehraclasses.com
          ServerAlias nehraclasses.com node1
          DocumentRoot /var/www/html/
          ServerName www.nehraclasses.com
          ErrorLog logs/www.nehraclasses.com-error_log
          CustomLog logs/www.nehraclasses.com-access_log common
          </VirtualHost>

    - name: Making changes in the /etc/httpd/conf/httpd.conf file for the name-based hosting.
      replace:
        dest: /etc/httpd/conf/httpd.conf
        regexp: '^#NameVirtualHost'
        replace: 'NameVirtualHost 192.168.229.129:80'

    - name: Allowing HTTP traffic in the firewall.
      firewalld:
        service: http
        zone: public
        permanent: true
        immediate: true
        state: enabled

    - name: Starting & enabling the httpd service.
      service:
        name: httpd
        state: started
        enabled: yes
...
```

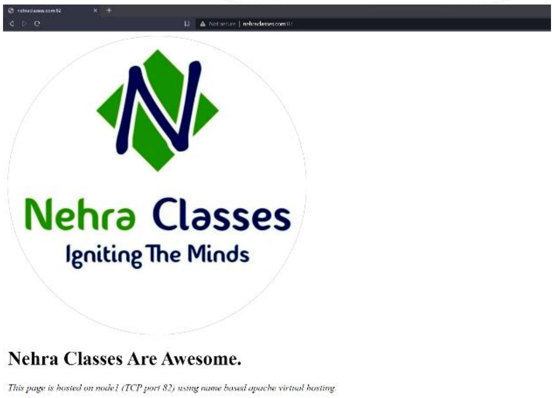
Install the Ansible posix collection from the Ansible Galaxy first before executing the playbook.

```
$ ansible-galaxy collection install ansible.posix
```

Now, execute the playbook to setup the webserver at node1 machine.

```
$ ansible-playbook name-based-apache-server.yml
```

Now, open any web browser and mention the IP address (hostname if DNS is available) of the managed node in the address bar of the web browser to access the page hosted on your managed node.



Nehra Classes Are Awesome.

This page is hosted on node1 (TCP port 82) using name based apache virtual hosting.

2. Configuration of alternate document root for Apache:
Let’s create an ansible playbook to setup an alternate document root for the Apache web server on the managed node(s).

```
$ vim apache-alternate-document-root.yml
---
- name: Alternate Document Root Apache HTTP Server Configuration Playbook
  hosts: node1
  become: true
  tasks:
    - name: Setting up the static hostname in the server machine.
      hostname:
        name: node1.nehraclasses.com
        use: systemd

    - name: Making entries in the /etc/hosts file for the server hostname & IP Address
      lineinfile:
        dest: /etc/hosts
        line: 192.168.229.129 www.nehraclasses.com node1
        insertafter: EOF

    - name: Installing Apache & SELinux Policy packages in the machine.
      dnf:
        name:
          - httpd
          - policycoreutils-python-utils
        state: latest

    - name: Creating /nehraclassesweb/ directory for the document root.
      file:
        path: /nehraclassesweb/
        mode: '0755'
        state: directory
        owner: apache
        group: apache

    - name: Copying the image file to the /nehraclassesweb/ directory.
      ansible.builtin.copy:
        src: /home/vikasnehra/NehraClassesLogo.png
        dest: /nehraclassesweb/NehraClassesLogo.png
        mode: '0644'

    - name: Creating the website index file in the /nehraclassesweb/ directory.
      copy:
        dest: "/nehraclassesweb/index.html"
        content: |
          <img src="/NehraClassesLogo.png" width="500" height="500" />
          <h1>Nehra Classes Are Awesome.</h1>
```

Manage HTTP Web Access Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
      <i>This page is hosted on node1 at /nehraclassesweb/ using Apache virtual hosting.</i>

  - name: Creating HTTPD configuration file in the /etc/httpd/conf.d/ directory.
    copy:
     dest: "/etc/httpd/conf.d/httpd.conf"
     content: |
       <VirtualHost 192.168.229.229:80>
       ServerAdmin root@node1.nehraclasses.com
       ServerAlias nehraclasses.com node1
       DocumentRoot /nehraclassesweb/
       ServerName www.nehraclasses.com
       ErrorLog logs/www.nehraclasses.com-error_log
       CustomLog logs/www.nehraclasses.com-access_log common
       <Directory /nehraclassesweb/>
       Options Indexes FollowSymLinks
       AllowOverride None
       Require all granted
       </Directory>
       </VirtualHost>

  - name: Copying /etc/httpd/conf/httpd.conf file using ansible jinja template.
    template:
      src: httpd.conf.j2
      dest: /etc/httpd/conf/httpd.conf
      force: true

  - name: Allowing Apache to modify the files in the /nehraclassesweb/ directory. (SELinux Context)
    community.general.sefcontext:
      target: '/nehraclassesweb(/.*)?'
      setype: httpd_sys_content_t
      state: present

  - name: Applying new SELinux file context to filesystem /nehraclassesweb/
    command: restorecon -Rv /nehraclassesweb/

  - name: Allowing HTTP traffic in the firewall.
    firewalld:
      service: http
      zone: public
      permanent: true
      immediate: true
      state: enabled

  - name: Starting & enabling the httpd service.
    service:
      name: httpd
      state: started
      enabled: yes
...
```

Install the Ansible posix collection from the Ansible Galaxy first before executing the playbook.

```
$ ansible-galaxy collection install ansible.posix
```

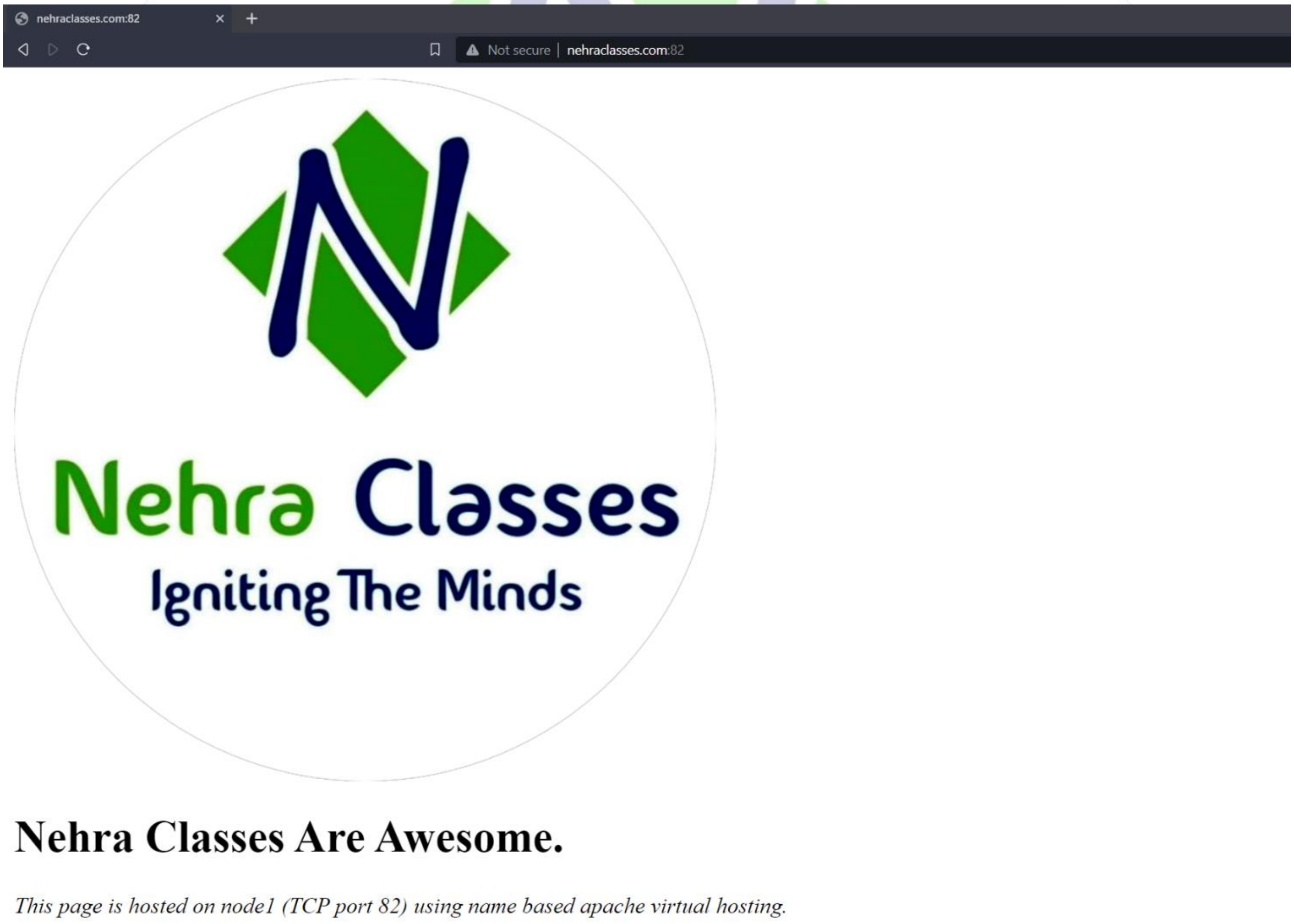
Install the community general collection from the Ansible Galaxy first before executing the playbook.

```
$ ansible-galaxy collection install community.general
```

Now, execute the playbook to setup the webserver at node1 machine.

```
$ ansible-playbook apache-alternate-document-root.yml
```

Now, open any web browser and mention the IP address (hostname if DNS is available) of the managed node in the address bar of the web browser to access the page hosted on your managed node.



Nehra Classes Are Awesome.

This page is hosted on node1 (TCP port 82) using name based apache virtual hosting.

3. Configuration of an alternate web access port for Apache:

Let's create an ansible playbook to setup an alternate web access port (say TCP port 82 instead of 80) for the Apache web server on the managed node(s).

```
$ vim apache-alternate-port.yml
---
- name: Alternate Port Apache HTTP Server Configuration Playbook
  hosts: node1
  become: true
  tasks:
    - name: Setting up the static hostname in the server machine.
      hostname:
        name: node1.nehraclasses.com
        use: systemd

    - name: Making entries in the /etc/hosts file for the server hostname & IP Address
      lineinfile:
        dest: /etc/hosts
        line: 192.168.229.129 www.nehraclasses.com node1
        insertafter: EOF

    - name: Installing Apache & SELinux Policy packages in the machine.
      dnf:
        name:
          - httpd
          - policycoreutils-python-utils
        state: latest

    - name: Creating /nehraclassesweb/ directory for the document root.
      file:
        path: /nehraclassesweb/
        mode: '0755'
        state: directory
        owner: apache
        group: apache

    - name: Copying the image file to the /nehraclassesweb/ directory.
      ansible.builtin.copy:
        src: /home/vikasnehra/NehraClassesLogo.png
        dest: /nehraclassesweb/NehraClassesLogo.png
        mode: '0644'

    - name: Creating the website index file in the /nehraclassesweb/ directory.
      copy:
        dest: "/nehraclassesweb/index.html"
        content: |
          <img src="/NehraClassesLogo.png" width="500" height="500" />
          <h1>Nehra Classes Are Awesome.</h1>
         <i>This page is hosted on node1 using port 82 for name based apache virtual
hosting.</i>
```

```
  - name: Creating HTTPD configuration file in the /etc/httpd/conf.d/ directory.
    copy:
     dest: "/etc/httpd/conf.d/httpd.conf"
     content: |
        <VirtualHost 192.168.229.229:82>
        ServerAdmin root@node1.nehraclasses.com
        ServerAlias nehraclasses.com node1
        DocumentRoot /nehraclassesweb/
        ServerName www.nehraclasses.com
        ErrorLog logs/www.nehraclasses.com-error_log
        CustomLog logs/www.nehraclasses.com-access_log common
        <Directory /nehraclassesweb/>
        Options Indexes FollowSymLinks
        AllowOverride None
        Require all granted
        </Directory>
        </VirtualHost>

  - name: Copying /etc/httpd/conf/httpd.conf file using ansible jinja template.
    template:
      src: httpd2.conf.j2
      dest: /etc/httpd/conf/httpd.conf
      force: true

  - name: Allowing Apache to modify the files in the /nehraclassesweb/ directory. (SELinux Context)
    community.general.sefcontext:
      target: '/nehraclassesweb(/.*)?'
      setype: httpd_sys_content_t
      state: present

  - name: Applying new SELinux file context to filesystem /nehraclassesweb/
    command: restorecon -Rv /nehraclassesweb/

  - name: Adding TCP port 82 in SELinux
    command: semanage port -a -t http_port_t -p tcp 82

  - name: Allowing HTTP traffic in the firewall.
    firewalld:
      port: 82/tcp
      zone: public
      permanent: true
      immediate: true
      state: enabled

  - name: Starting & enabling the httpd service.
    service:
      name: httpd
      state: started
      enabled: yes
```

...

Install the Ansible posix collection from the Ansible Galaxy first before executing the playbook.

```
$ ansible-galaxy collection install ansible.posix
```

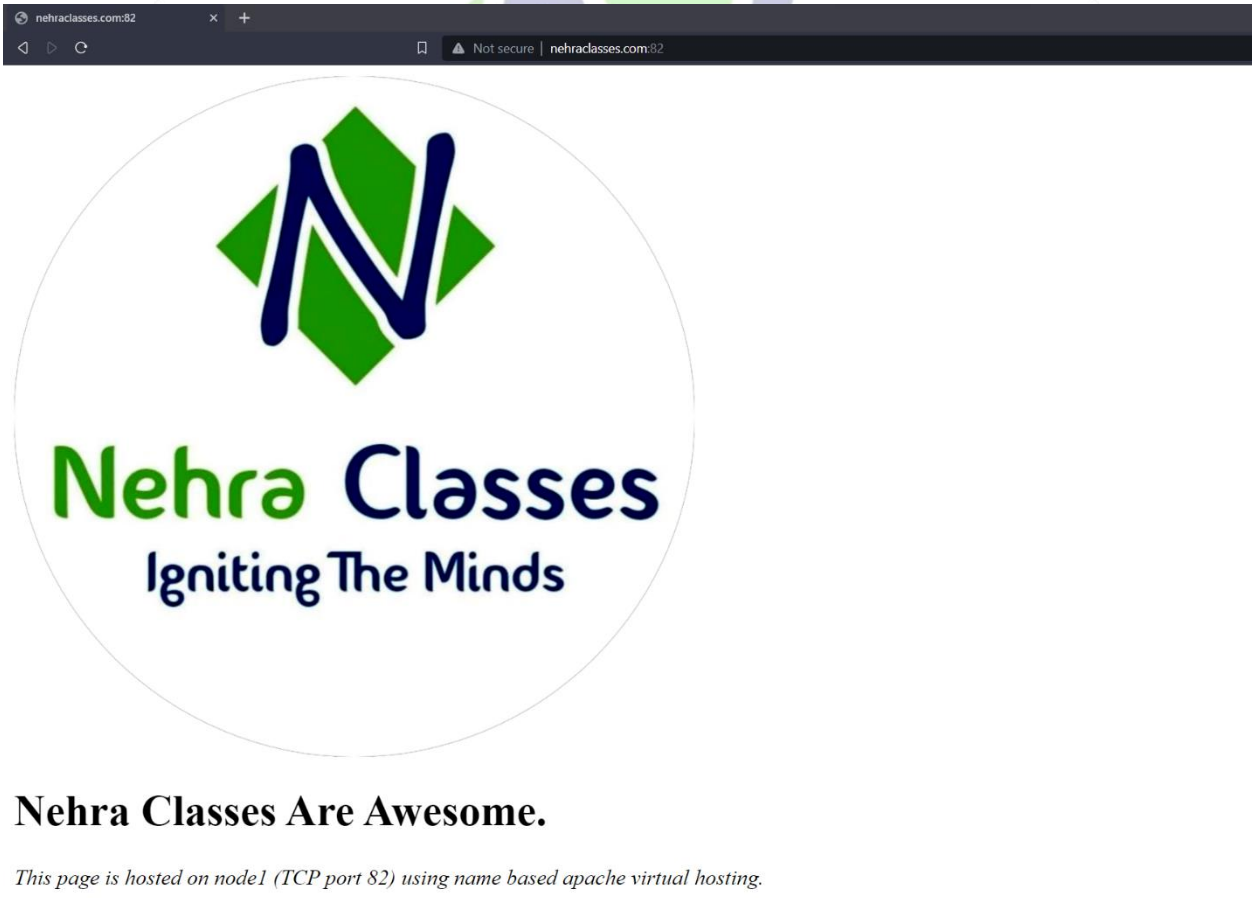
Install the community general collection from the Ansible Galaxy first before executing the playbook.

```
$ ansible-galaxy collection install community.general
```

Now, execute the playbook to setup the webserver at node1 machine.

```
$ ansible-playbook apache-alternate-port.yml
```

Now, open any web browser and mention the IP address (hostname if DNS is available) of the managed node in the address bar of the web browser to access the page hosted on your managed node.



Nehra Classes Are Awesome.

This page is hosted on node1 (TCP port 82) using name based apache virtual hosting.

All the webservers are working as expected.

Thank You