Custom Configuration of DNS Server

We're using a Cloud Service for our setup, which includes two instances. The first instance is a web server, and the second one is a DNS server.

Aws Cloud Instance	Private IP	Public IP	Configured For
Web-Server	172.31.84.162	34.207.161.139	Web
DNS Server	172.31.40.241	54.198.43.87	DNS

- **Now, let's dive into the configuration of each instance.**

Setup of Web Server

Step 1: Install the Packages using yum

- Package name: **httpd**
- Command: **yum install httpd -y**

```
[root@Abhishek ~]# yum install httpd -y
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use "rhc" or "subscription-manager" to register.

Red Hat Enterprise Linux 9 for x86_64 - AppStream from RHUI (RPMs)                50 MB/s |  41 MB     00:00
Red Hat Enterprise Linux 9 for x86_64 - BaseOS from RHUI (RPMs)                   46 MB/s |  31 MB     00:00
Red Hat Enterprise Linux 9 Client Configuration                                   32 kB/s | 3.2 kB     00:00
Last metadata expiration check: 0:00:01 ago on Sat 21 Sep 2024 04:40:08 PM UTC.
Dependencies resolved.
================================================================================================================
 Package                         Architecture         Version                            Repository         Size
================================================================================================================
```

Step 2: Start and enable the apache service after installation

- Service name: **httpd.service**
- Command: **systemctl start httpd**
- Command: **systemctl enable httpd**

```
[root@Abhishek ~]# systemctl start httpd
[root@Abhishek ~]#
```

```
[root@Abhishek ~]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@Abhishek ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
```

Step 3: Create Web Pages

Navigate to the default web root directory **(/var/www/html)** and create **index.html**.

```
[root@Abhishek ~]# vim /var/www/html/index.html
```

```html
<!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>Abhishek Chomal's Portfolio</title>
    <link rel="stylesheet" href="styles.css">
    <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600&display=swap" rel="stylesheet">
</head>
<body>
    <header>
        <div class="header-container">
            <h1>Abhishek Chomal</h1>
            <p>Developer | Linux Enthusiast | Problem Solver</p>
            <nav>
                <ul>
                    <li><a href="#about">About Me</a></li>
                    <li><a href="#projects">Projects</a></li>
                    <li><a href="#skills">Skills</a></li>
                    <li><a href="#contact">Contact</a></li>
                </ul>
            </nav>
        </div>
    </header>
```

Step 4: Restart the apache service after creating web pages

- Command: **systemctl restart httpd**

```
[root@Abhishek ~]# systemctl restart httpd
[root@Abhishek ~]#
```

Setup of DNS Server

Step 1: Install the Packages using yum

- Package name: **bind, bind-utils**
- Command: **yum install bind bind-utils -y**

The bind package is used to run a **DNS server**, while bind-utils provides tools to query and **troubleshoot DNS issues** from the client side.

```
[root@Abhishek ~]# yum install bind bind-utils -y
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use "rhc" or "subscription-manager" to register.

Last metadata expiration check: 2:34:41 ago on Sun 22 Sep 2024 09:42:43 AM UTC.
Dependencies resolved.
==============================================================================================================================
 Package                      Architecture      Version                               Repository                          Size
==============================================================================================================================
Installing:
 bind                         x86_64            32:9.16.23-18.el9_4.6                 rhel-9-appstream-rhui-rpms         509 k
 bind-utils                   x86_64            32:9.16.23-18.el9_4.6                 rhel-9-appstream-rhui-rpms         214 k
```

Step 2: Verify that the packages were installed correctly

- Command: **rpm -qa bind bind-utils**

```
[root@Abhishek ~]# rpm -qa bind bind-utils
bind-utils-9.16.23-18.el9_4.6.x86_64
bind-9.16.23-18.el9_4.6.x86_64
```

Step 3: Start the DNS service after installation.

- Service name: **named**
- Command: **systemctl start named**

```
[root@Abhishek ~]# systemctl start named
[root@Abhishek ~]#
```

Step 4: To Ensure named service start automatically on boot.

- Command: **systemctl enable named**

```
[root@Abhishek ~]# systemctl enable named
Created symlink /etc/systemd/system/multi-user.target.wants/named.service → /usr/lib/systemd/system/named.service.
[root@Abhishek ~]# systemctl status named
● named.service - Berkeley Internet Name Domain (DNS)
     Loaded: loaded (/usr/lib/systemd/system/named.service; enabled; preset: disabled)
```

Step 5: Edit the configuration file

- Configuration file: **named.conf**
- Open the named configuration file using vim **(/etc/named.conf)**
- Command: **vim /etc/named.conf**

```
[root@Abhishek ~]# vim /etc/named.conf
```

```
//
// named.conf
//
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
// server as a caching only nameserver (as a localhost DNS resolver only).
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//

options {
        listen-on port 53 { 127.0.0.1; };
        listen-on-v6 port 53 { ::1; };
        directory       "/var/named";
        dump-file       "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        memstatistics-file "/var/named/data/named_mem_stats.txt";
        secroots-file   "/var/named/data/named.secroots";
        recursing-file  "/var/named/data/named.recursing";
        allow-query     { localhost; };
```

- Clear all the existing entries by typing→**:%d**
- After clearing the entries in **named.conf**, start defining the **DNS zones**.
- Add the following entries to define the **DNS options** and **your zone**.

```
options{
        directory "/var/named";
        recursion no;
};


zone    "abhichomal.online" IN{
        type master;
        file "abhi";

};
```

- Save and close file using→**:wq**

Step 6: Verify the configuration file

- Check the syntax of configuration file.
- Command: **named-checkconf /etc/named.conf**

```
[root@Abhishek ~]# named-checkconf /etc/named.conf
[root@Abhishek ~]#
```

- If no output display in terminal it means syntax ok

Step 7: Setup the zone file

- Navigate the zone file directory **/var/named**

```
[root@Abhishek ~]# cd /var/named/
[root@Abhishek named]#
```

- The zone files are stored in the **/var/named directory.** After reaching the **/var/named directory**, list the files to check if **named.empty** exists.
- You should see something like this in the output, confirming the existence of the **named.empty** file.

```
[root@Abhishek named]# ls -l named.empty
-rw-r-----. 1 root named 152 Aug  9 11:21 named.empty
```

- This file is a template that can be used to create your custom **DNS zone file**.
- To set up a zone for your domain, copy the **named.empty** file to a new file with a name that matches what is defined in your **/etc/named.conf** configuration file.
- Use the **cp** command with the **-p** option to preserve the file permissions.
- Command: **cp –p named.empty < zonefile name >**

```
[root@Abhishek named]# cp -p named.empty abhi
[root@Abhishek named]#
```

- Navigate the zone file directory **/var/named**

Step 8: Verify the new zone file

- Once the file is copied, verify that the new zone file was created has the same permission as the **named.empty**.
- You should see your newly created zone file, such as:

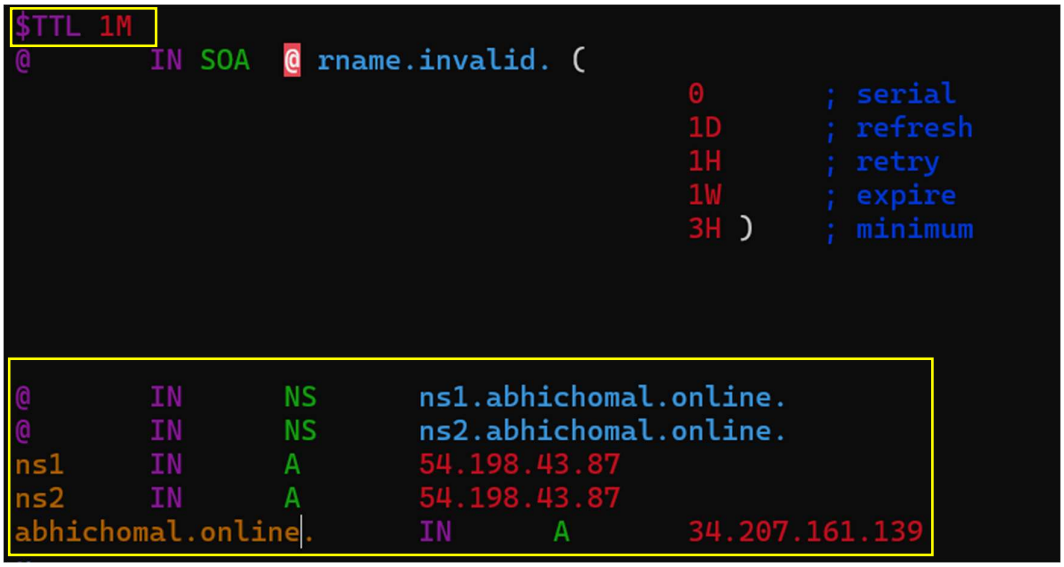
```
[root@Abhishek named]# ls -l abhi
-rw-r-----. 1 root named 152 Aug  9 11:21 abhi
```

Step 9: Edit the zone file

- Now, you need to edit this newly created zone file to **add your DNS records**.
- Open the file using **vim**.

```
[root@Abhishek named]# vim abhi
```

- Inside the zone file, add your required **DNS records**.

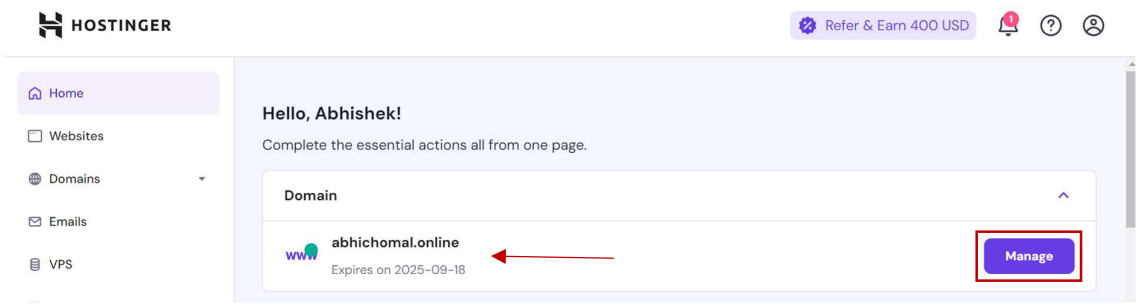


Step 10: Restart the named service after changes in file

- Service name: **named**
- Command: **systemctl restart named**

```
[root@Abhishek ~]# systemctl restart named
[root@Abhishek ~]#
```

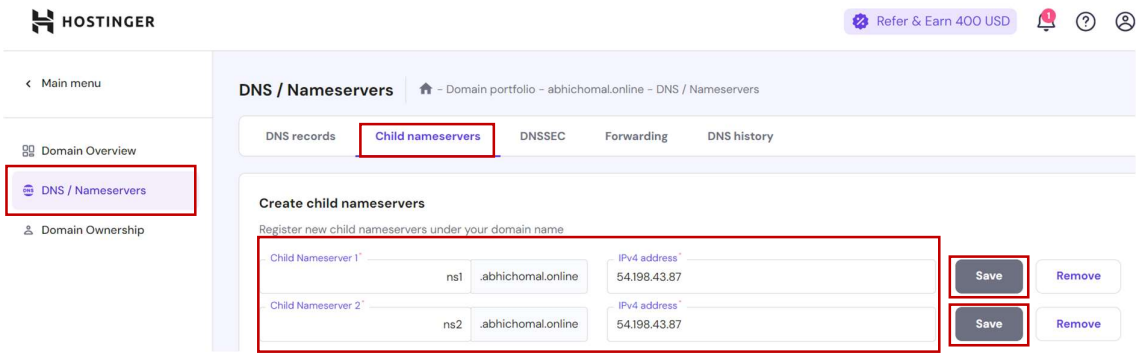
Step 11: Open Your Domain Name Registrar where your domain is registered (e.g., GoDaddy, Namecheap, Hostinger).and click on manage for your domain



Step 12: Add Childnameservers entries

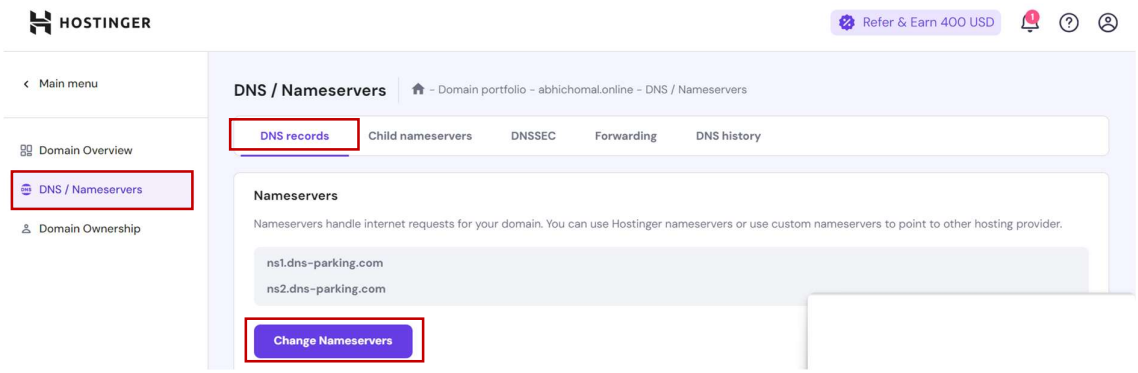
- Click on the **DNS/Nameservers** section.
- Select **Child nameservers** tab.
- Add the following entries.

Child Nameserver	DNS Server Public IP
ns1.abhichomal.online	54.198.43.87
ns2.abhichomal.online	54.198.43.87



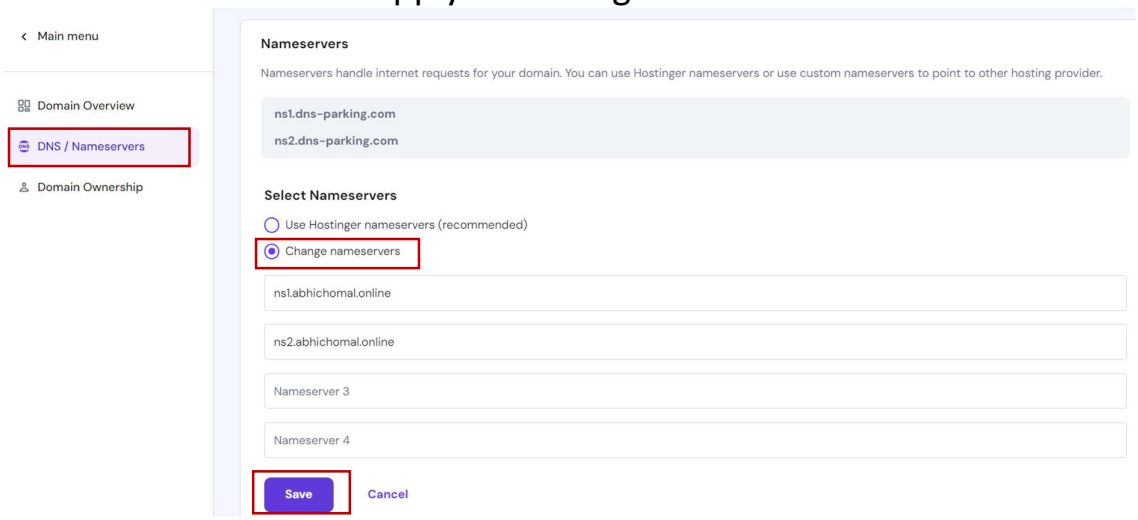
Step 13: Change Nameservers

- After adding the **Child Nameserver entries**, go to **DNS Records Section**, where click on **Change Nameservers**.

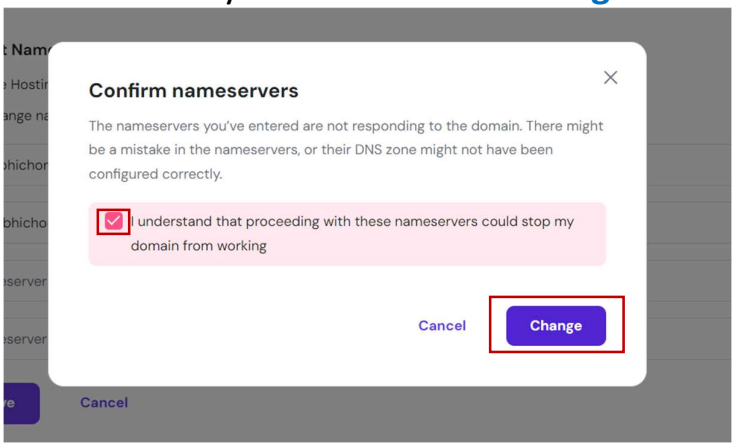


- Replace the existing nameservers with the newly created ones (**ns1.abhichomal.online** and **ns2.abhichomal.online**).

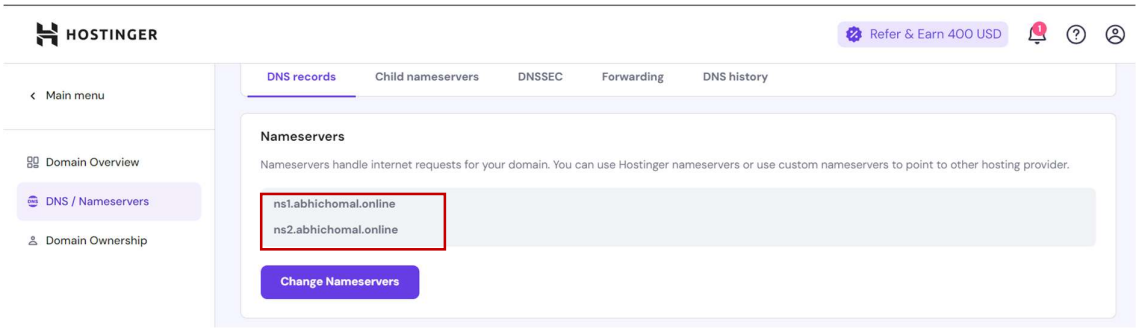
- Click "**Save**" to apply the changes.



- **Confirm** that your **nameserver changes** have been saved.



- Now **your name server** change.

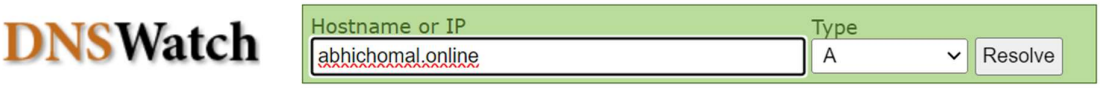


- wait for **DNS propagation**.

  Note: Wait for some time to update the DNS cache with the updated configuration.

Step 14: Verify the DNS changes

- After Propagation, verify the DNS configuration using tools like DNSWatch (**https://dnswatch.info**) or by running the **nslookup command** in a terminal.



- Ensure the **nameservers** reflect the **updated information**.

A record found: 34.207.161.139

Domain	Type	TTL	Answer
abhichomal.online.	NS	3600	ns1.abhichomal.online.
abhichomal.online.	NS	3600	ns2.abhichomal.online.
abhichomal.online.	A	60	34.207.161.139

- Check using **nslookup** Command.

```
[root@Abhishek ~]# nslookup abhichomal.online
Server:         172.31.0.2
Address:        172.31.0.2#53

Non-authoritative answer:
Name:   abhichomal.online
Address: 34.207.161.139
```

Step 15: Verify Website Accessibility Using Domain Name