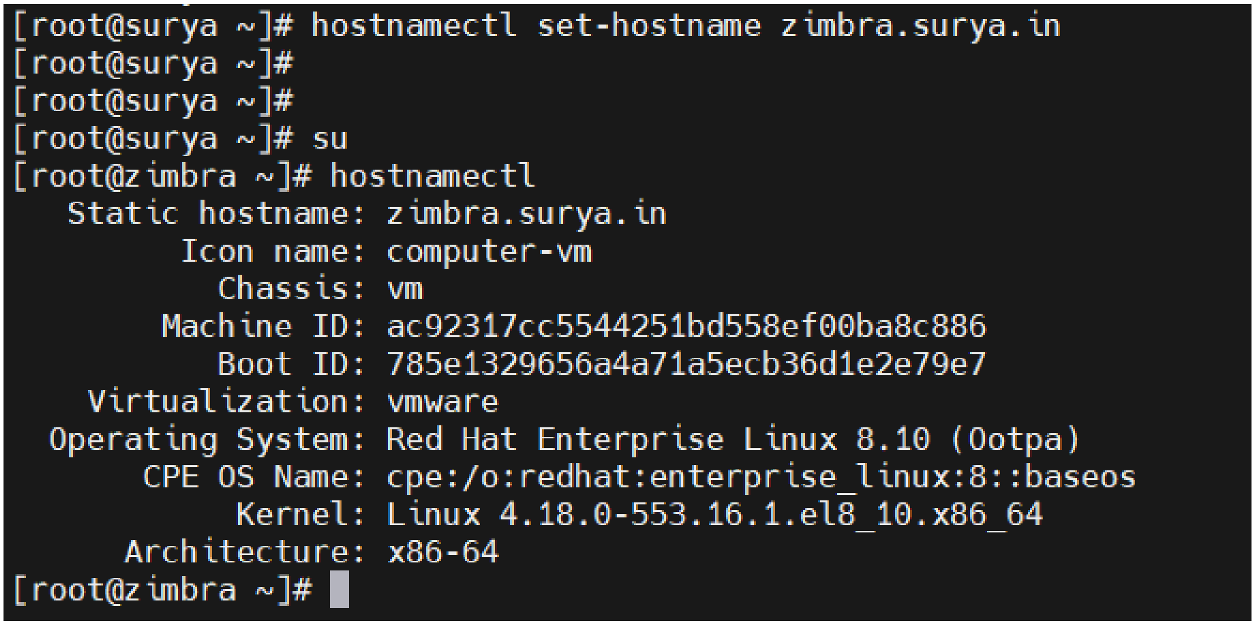
================== DNS_Configuration =====================

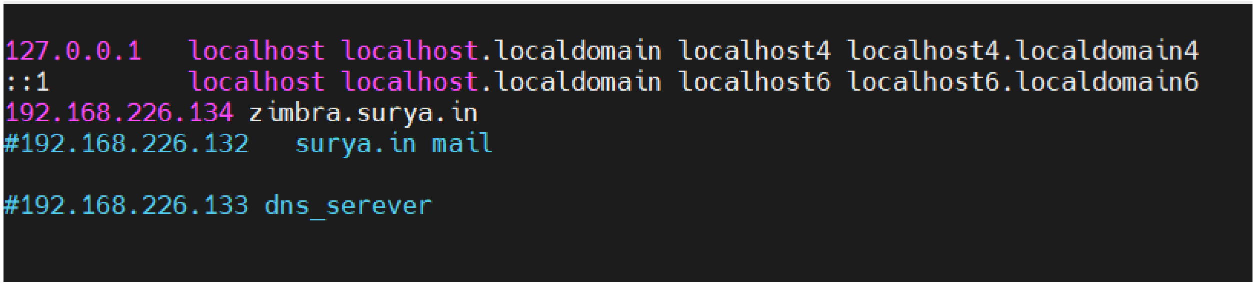
Step 1. Set hostname and verify it and give entry in /etc/hosts file

hostnamectl set-hostname surya.com
hostname



```
[root@surya ~]# hostnamectl set-hostname zimbra.surya.in
[root@surya ~]#
[root@surya ~]#
[root@surya ~]# su
[root@zimbra ~]# hostnamectl
   Static hostname: zimbra.surya.in
         Icon name: computer-vm
           Chassis: vm
        Machine ID: ac92317cc5544251bd558ef00ba8c886
           Boot ID: 785e1329656a4a71a5ecb36d1e2e79e7
    Virtualization: vmware
  Operating System: Red Hat Enterprise Linux 8.10 (Ootpa)
       CPE OS Name: cpe:/o:redhat:enterprise_linux:8::baseos
            Kernel: Linux 4.18.0-553.16.1.el8_10.x86_64
      Architecture: x86-64
[root@zimbra ~]#
```

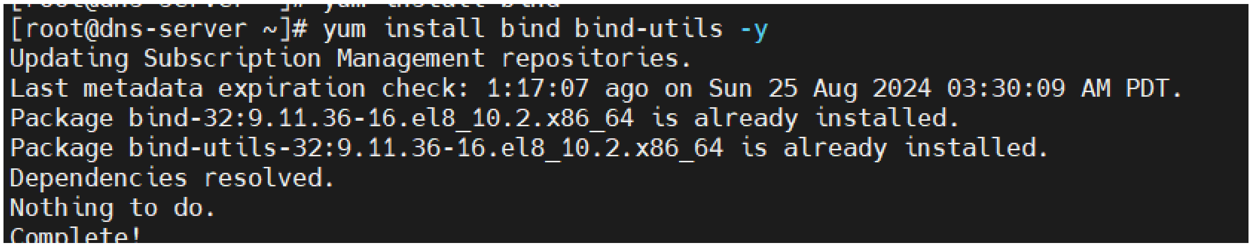
# nano /etc/hosts



```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.226.134 zimbra.surya.in
#192.168.226.132   surya.in mail

#192.168.226.133 dns_serever
```

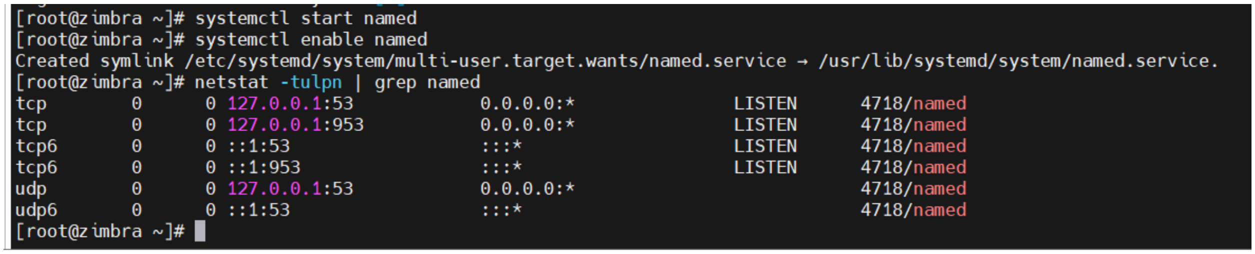
Step 2. Install DNS Package and start services and verify it

yum install bind bind-utils -y



```
[root@dns-server ~]# yum install bind
[root@dns-server ~]# yum install bind bind-utils -y
Updating Subscription Management repositories.
Last metadata expiration check: 1:17:07 ago on Sun 25 Aug 2024 03:30:09 AM PDT.
Package bind-32:9.11.36-16.el8_10.2.x86_64 is already installed.
Package bind-utils-32:9.11.36-16.el8_10.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
```

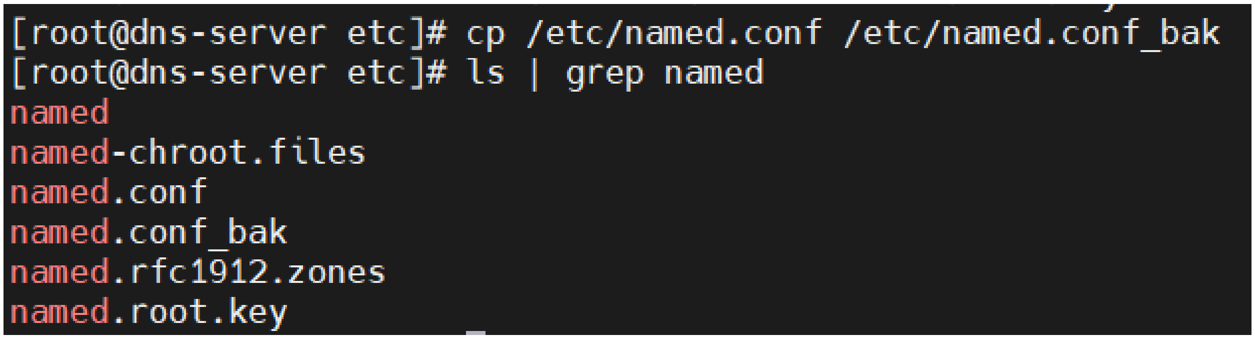
# systemctl start named
# ststemctl enable named
# systemctl status named
# netstart -tulpn | grep named



```
[root@zimbra ~]# systemctl start named
[root@zimbra ~]# systemctl enable named
Created symlink /etc/systemd/system/multi-user.target.wants/named.service → /usr/lib/systemd/system/named.service.
[root@zimbra ~]# netstat -tulpn | grep named
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN      4718/named
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN      4718/named
tcp6       0      0 ::1:53                  :::*                    LISTEN      4718/named
tcp6       0      0 ::1:953                 :::*                    LISTEN      4718/named
udp        0      0 127.0.0.1:53            0.0.0.0:*                           4718/named
udp6       0      0 ::1:53                  :::*                                4718/named
[root@zimbra ~]#
```

Step 3. Take backup of conf file

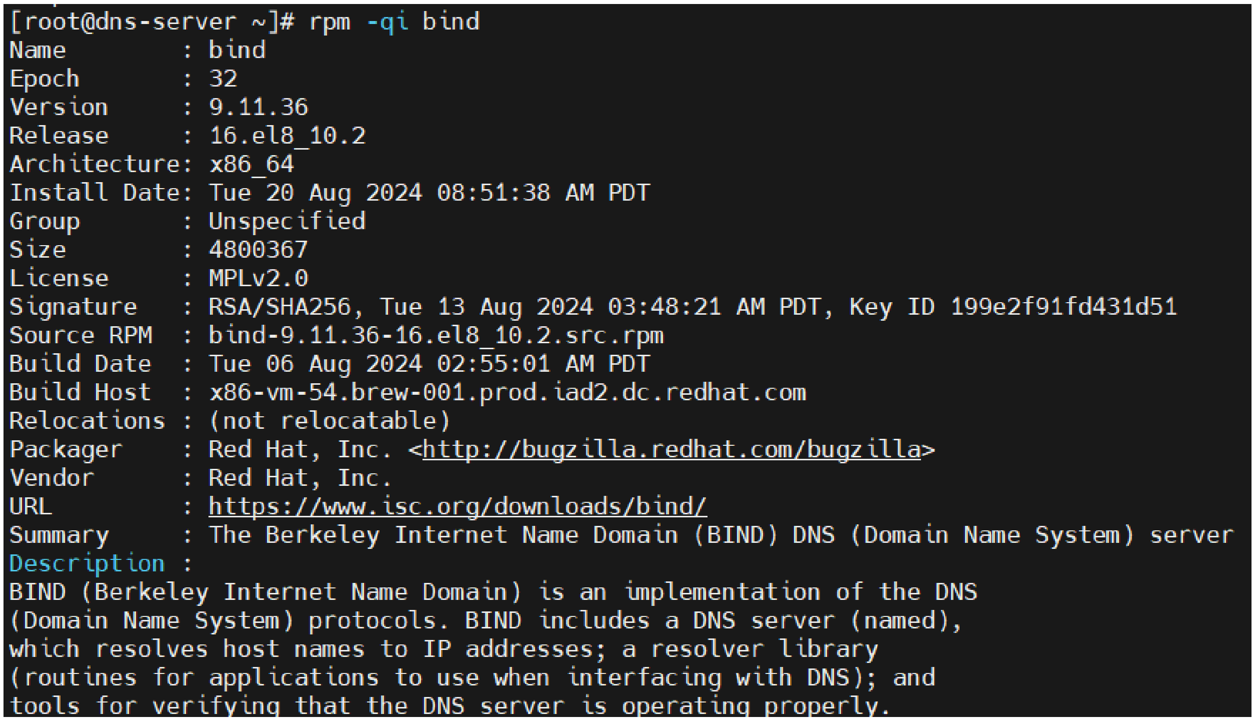
# cp -p /etc/named.conf /etc/named.conf_bak



```
[root@dns-server etc]# cp /etc/named.conf /etc/named.conf_bak
[root@dns-server etc]# ls | grep named
named
named-chroot.files
named.conf
named.conf_bak
named.rfc1912.zones
named.root.key
```

→ Check info of installed package

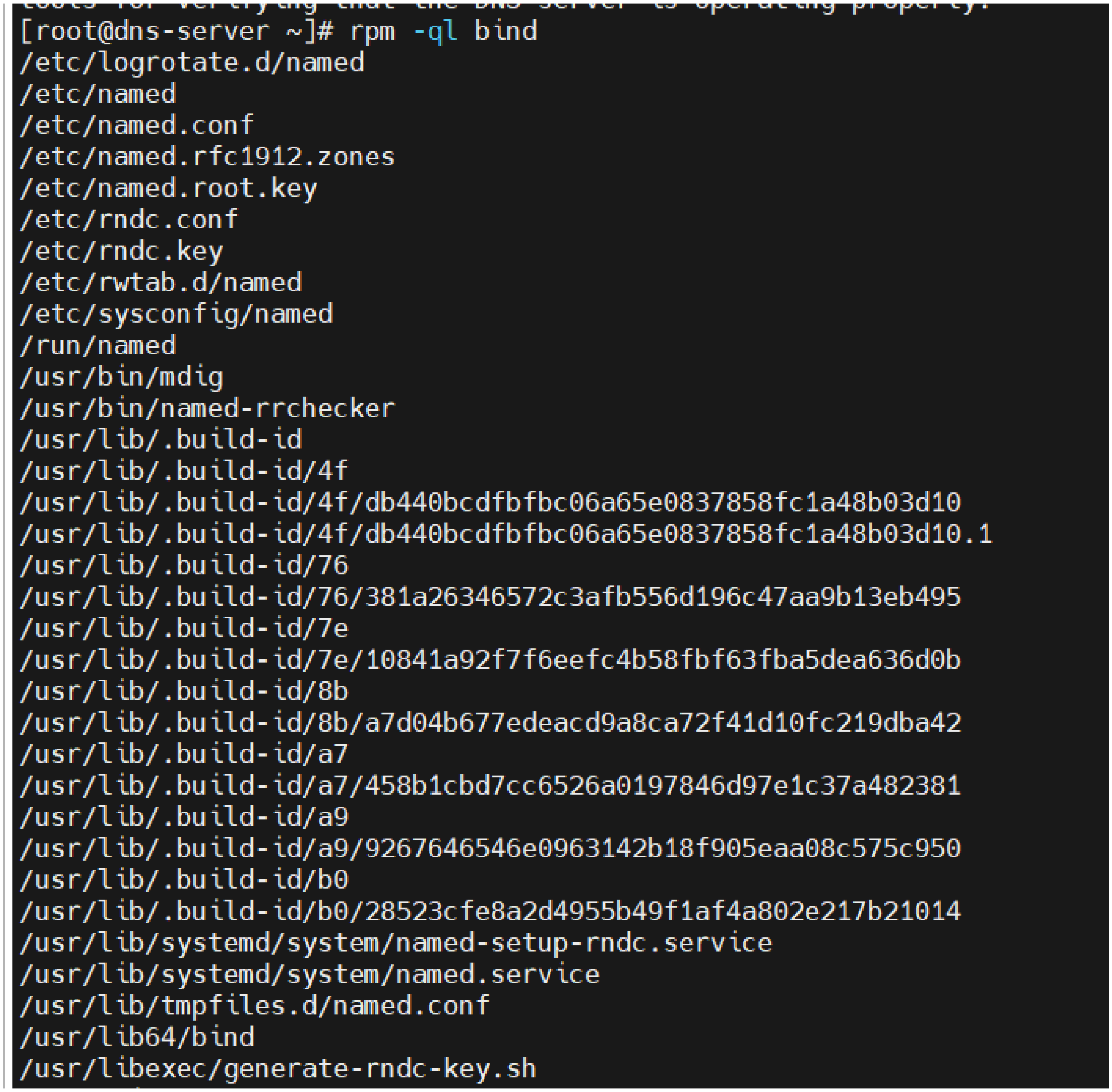
rpm -qi bind



```
[root@dns-server ~]# rpm -qi bind
Name        : bind
Epoch       : 32
Version     : 9.11.36
Release     : 16.el8_10.2
Architecture: x86_64
Install Date: Tue 20 Aug 2024 08:51:38 AM PDT
Group       : Unspecified
Size        : 4800367
License     : MPLv2.0
Signature   : RSA/SHA256, Tue 13 Aug 2024 03:48:21 AM PDT, Key ID 199e2f91fd431d51
Source RPM  : bind-9.11.36-16.el8_10.2.src.rpm
Build Date  : Tue 06 Aug 2024 02:55:01 AM PDT
Build Host  : x86-vm-54.brew-001.prod.iad2.dc.redhat.com
Relocations : (not relocatable)
Packager    : Red Hat, Inc. <http://bugzilla.redhat.com/bugzilla>
Vendor      : Red Hat, Inc.
URL         : https://www.isc.org/downloads/bind/
Summary     : The Berkeley Internet Name Domain (BIND) DNS (Domain Name System) server
Description :
BIND (Berkeley Internet Name Domain) is an implementation of the DNS
(Domain Name System) protocols. BIND includes a DNS server (named),
which resolves host names to IP addresses; a resolver library
(routines for applications to use when interfacing with DNS); and
tools for verifying that the DNS server is operating properly.
```

→ check list of installed packages

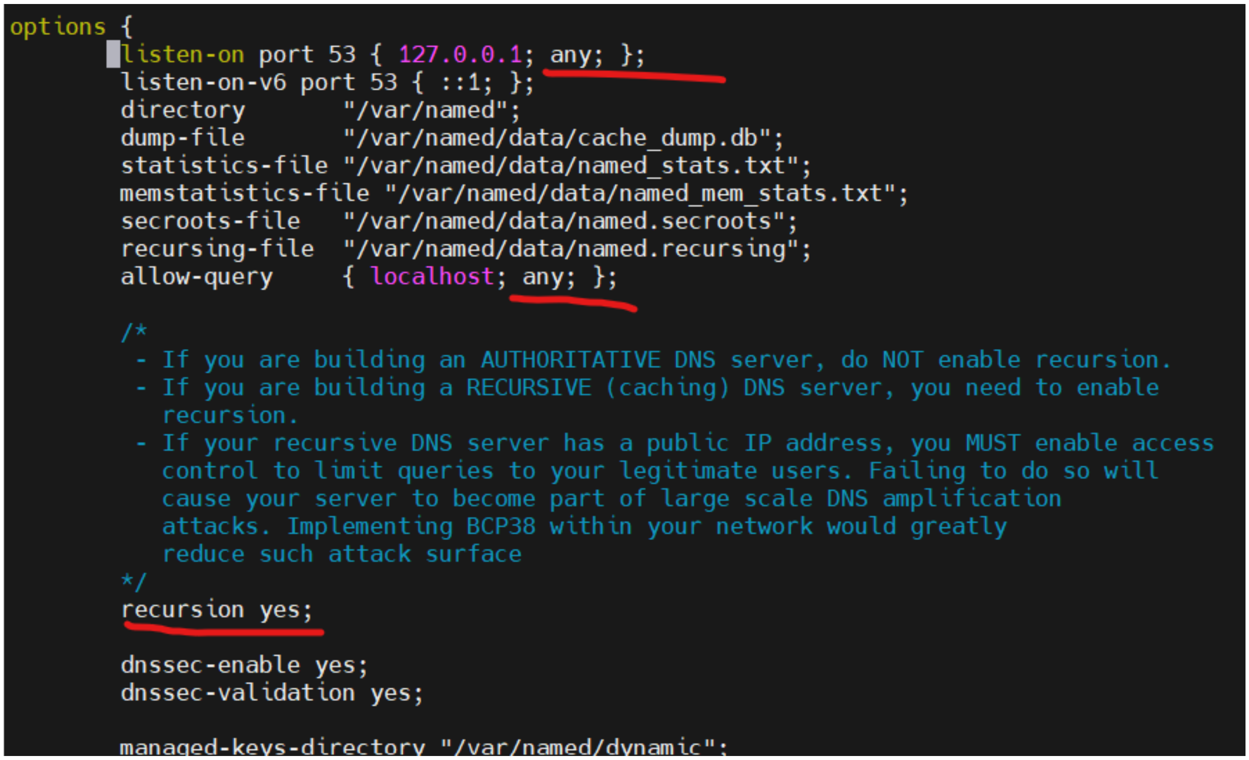
# rpm -ql bind



```
tools for verifying that the DNS server is operating properly.
[root@dns-server ~]# rpm -ql bind
/etc/logrotate.d/named
/etc/named
/etc/named.conf
/etc/named.rfc1912.zones
/etc/named.root.key
/etc/rndc.conf
/etc/rndc.key
/etc/rwtab.d/named
/etc/sysconfig/named
/run/named
/usr/bin/mdig
/usr/bin/named-rrchecker
/usr/lib/.build-id
/usr/lib/.build-id/4f
/usr/lib/.build-id/4f/db440bcdfbfbc06a65e0837858fc1a48b03d10
/usr/lib/.build-id/4f/db440bcdfbfbc06a65e0837858fc1a48b03d10.1
/usr/lib/.build-id/76
/usr/lib/.build-id/76/381a26346572c3afb556d196c47aa9b13eb495
/usr/lib/.build-id/7e
/usr/lib/.build-id/7e/10841a92f7f6eefc4b58fbf63fba5dea636d0b
/usr/lib/.build-id/8b
/usr/lib/.build-id/8b/a7d04b677edeacd9a8ca72f41d10fc219dba42
/usr/lib/.build-id/a7
/usr/lib/.build-id/a7/458b1cbd7cc6526a0197846d97e1c37a482381
/usr/lib/.build-id/a9
/usr/lib/.build-id/a9/9267646546e0963142b18f905eaa08c575c950
/usr/lib/.build-id/b0
/usr/lib/.build-id/b0/28523cfe8a2d4955b49f1af4a802e217b21014
/usr/lib/systemd/system/named-setup-rndc.service
/usr/lib/systemd/system/named.service
/usr/lib/tmpfiles.d/named.conf
/usr/lib64/bind
/usr/libexec/generate-rndc-key.sh
```

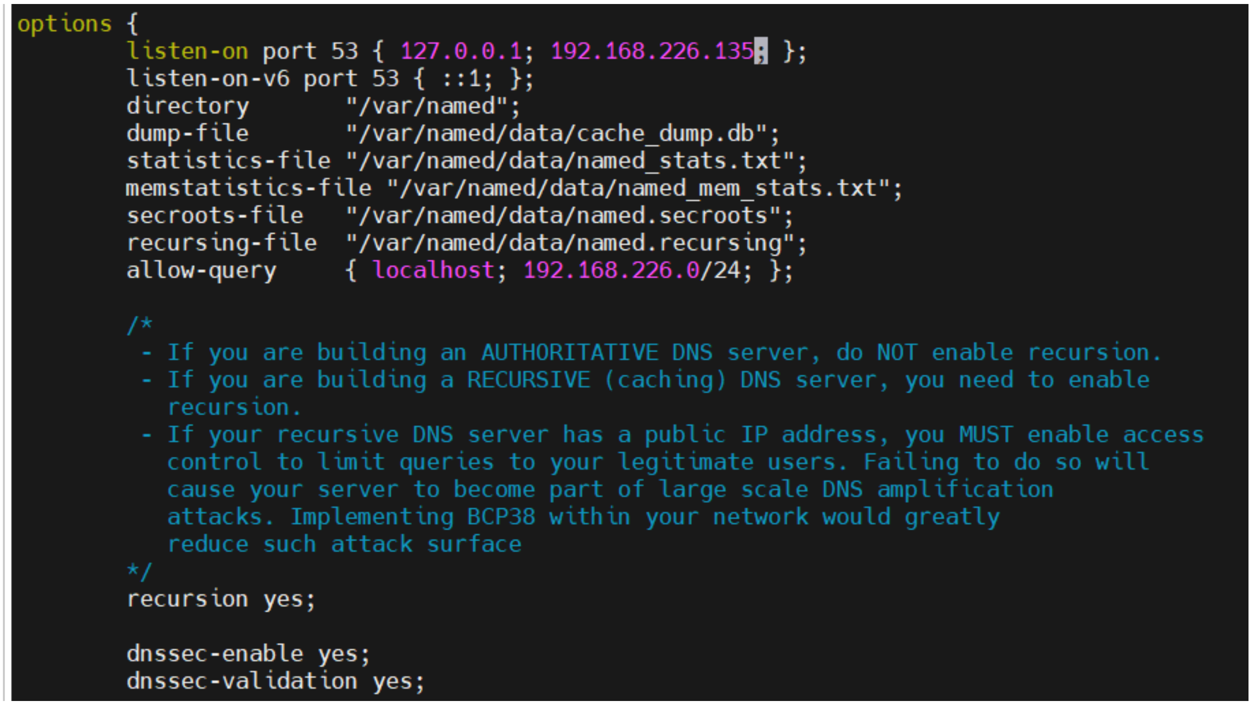
Step 4. Edit conf file in line no - 11,19,31 like below

# nano /etc/named.conf



```
options {
        listen-on port 53 { 127.0.0.1; any; };
        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; any; };

        /*
         - If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.
         - If you are building a RECURSIVE (caching) DNS server, you need to enable
           recursion.
         - If your recursive DNS server has a public IP address, you MUST enable access
           control to limit queries to your legitimate users. Failing to do so will
           cause your server to become part of large scale DNS amplification
           attacks. Implementing BCP38 within your network would greatly
           reduce such attack surface
        */
        recursion yes;

        dnssec-enable yes;
        dnssec-validation yes;

        managed-keys-directory "/var/named/dynamic";
```

:- This configuration use for globelly

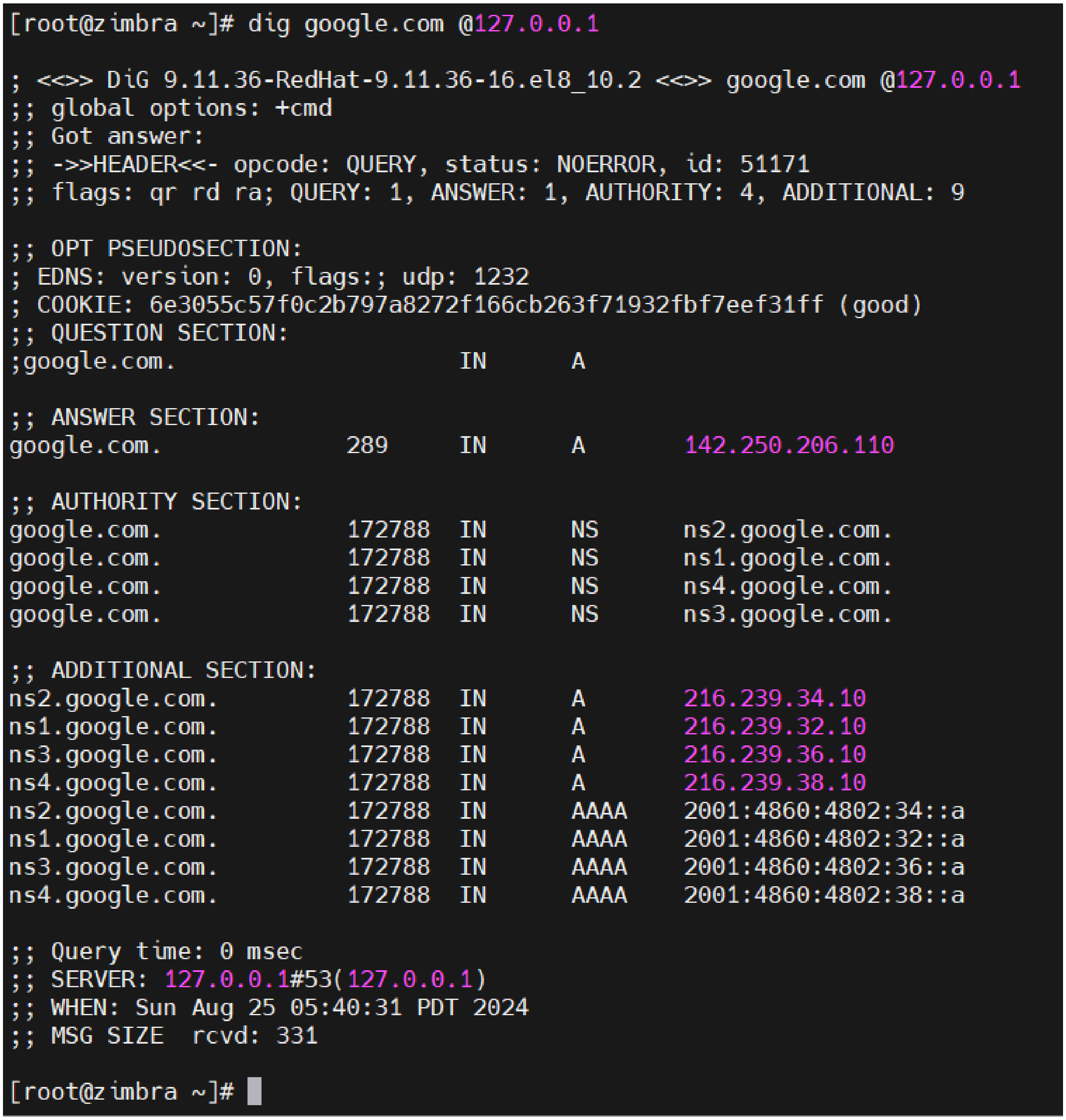


```
options {
        listen-on port 53 { 127.0.0.1; 192.168.226.135; };
        listen-on-v6 port 53 { ::1; };
        directory       "/var/named";
        dump-file       "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        memstatistics-file "/var/named/data/named_mem_stats.txt";
        secroots-file   "/var/named/data/named.secroots";
        recursing-file  "/var/named/data/named.recursing";
        allow-query     { localhost; 192.168.226.0/24; };

        /*
         - If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.
         - If you are building a RECURSIVE (caching) DNS server, you need to enable
           recursion.
         - If your recursive DNS server has a public IP address, you MUST enable access
           control to limit queries to your legitimate users. Failing to do so will
           cause your server to become part of large scale DNS amplification
           attacks. Implementing BCP38 within your network would greatly
           reduce such attack surface
        */
        recursion yes;

        dnssec-enable yes;
        dnssec-validation yes;
```

:- This configuration is use for your local network

Step 5. Do quarry with google your dns server

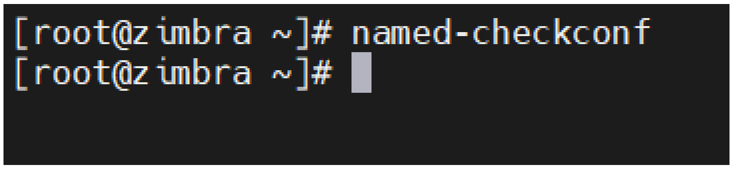
# dig google.com @127.0.0.1



```
[root@zimbra ~]# dig google.com @127.0.0.1

; <<>> DiG 9.11.36-RedHat-9.11.36-16.el8_10.2 <<>> google.com @127.0.0.1
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51171
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 9

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 6e3055c57f0c2b797a8272f166cb263f71932fbf7eef31ff (good)
;; QUESTION SECTION:
;google.com.                    IN      A

;; ANSWER SECTION:
google.com.             289     IN      A       142.250.206.110

;; AUTHORITY SECTION:
google.com.             172788  IN      NS      ns2.google.com.
google.com.             172788  IN      NS      ns1.google.com.
google.com.             172788  IN      NS      ns4.google.com.
google.com.             172788  IN      NS      ns3.google.com.

;; ADDITIONAL SECTION:
ns2.google.com.         172788  IN      A       216.239.34.10
ns1.google.com.         172788  IN      A       216.239.32.10
ns3.google.com.         172788  IN      A       216.239.36.10
ns4.google.com.         172788  IN      A       216.239.38.10
ns2.google.com.         172788  IN      AAAA    2001:4860:4802:34::a
ns1.google.com.         172788  IN      AAAA    2001:4860:4802:32::a
ns3.google.com.         172788  IN      AAAA    2001:4860:4802:36::a
ns4.google.com.         172788  IN      AAAA    2001:4860:4802:38::a

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Aug 25 05:40:31 PDT 2024
;; MSG SIZE  rcvd: 331

[root@zimbra ~]#
```

:- It means your server is up and working fine

→ Run this command for check error of your configuration
# named-checkconf

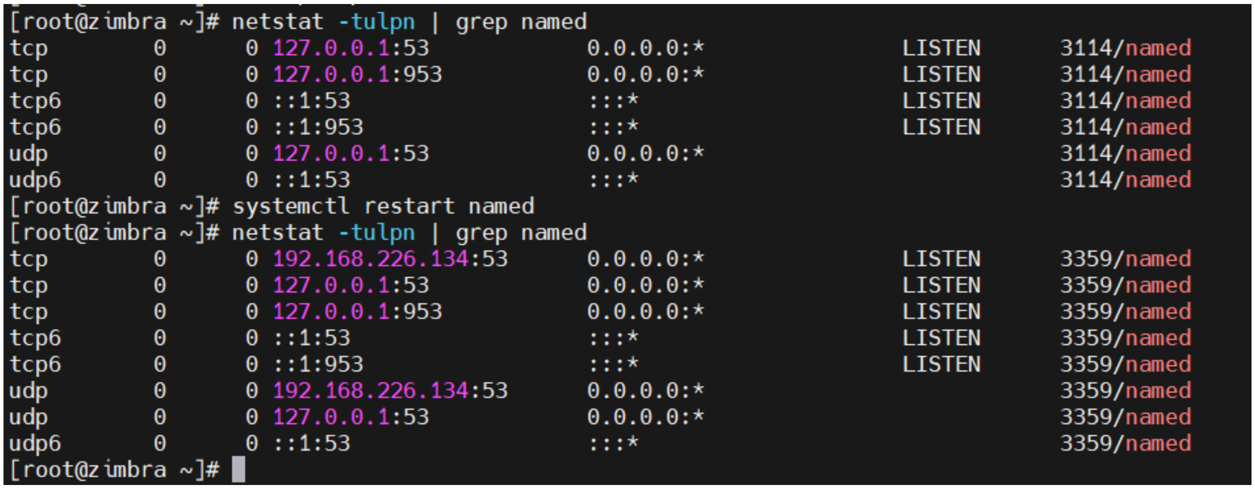


```
[root@zimbra ~]# named-checkconf
[root@zimbra ~]#
```

Step 6. Now restart the service and check listening from hosts

netstat -tulpn | grep named

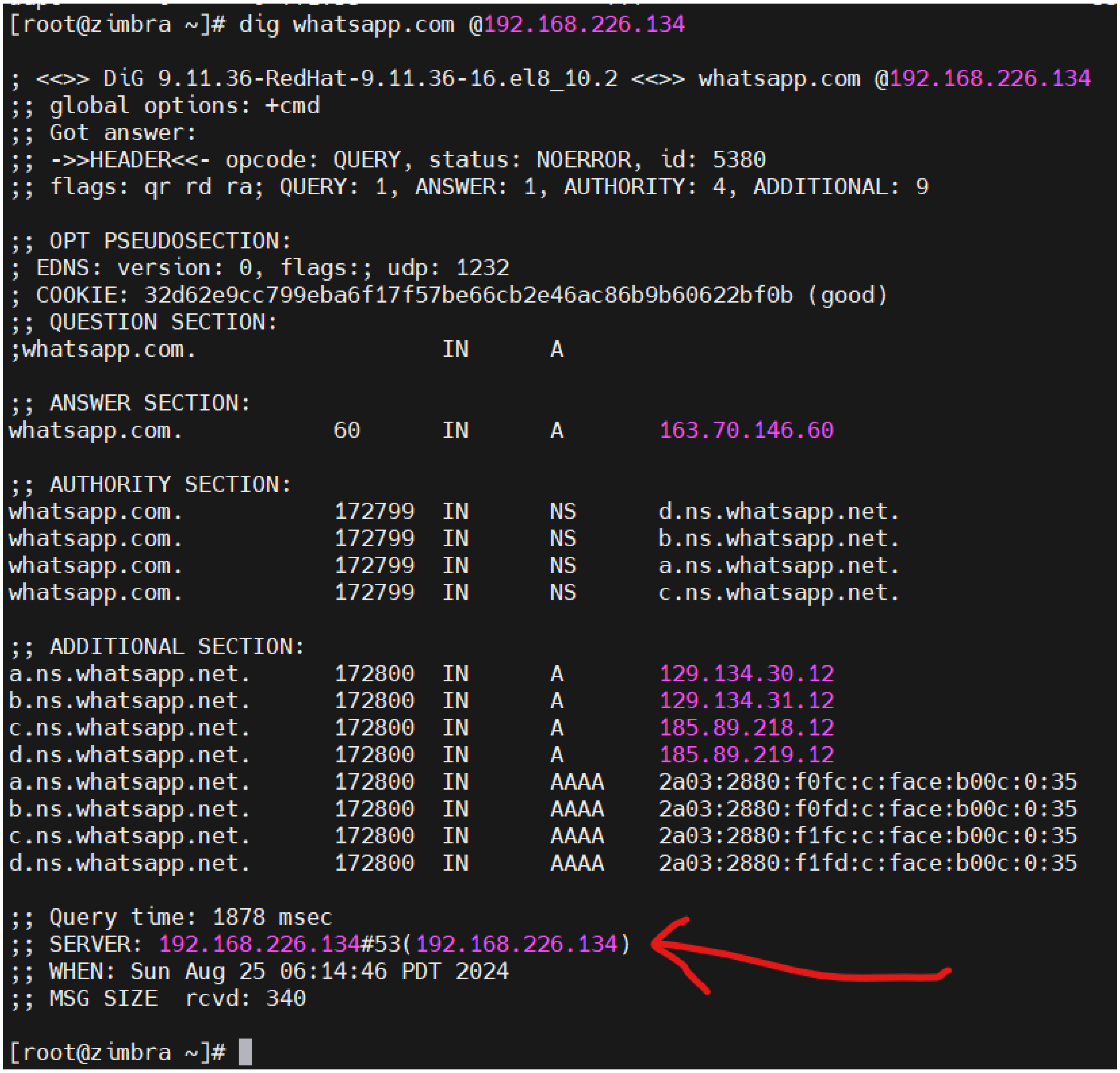
systemctl restart named



```
[root@zimbra ~]# netstat -tulpn | grep named
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN      3114/named
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN      3114/named
tcp6       0      0 ::1:53                  :::*                    LISTEN      3114/named
tcp6       0      0 ::1:953                 :::*                    LISTEN      3114/named
udp        0      0 127.0.0.1:53            0.0.0.0:*                           3114/named
udp6       0      0 ::1:53                  :::*                                3114/named
[root@zimbra ~]# systemctl restart named
[root@zimbra ~]# netstat -tulpn | grep named
tcp        0      0 192.168.226.134:53      0.0.0.0:*               LISTEN      3359/named
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN      3359/named
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN      3359/named
tcp6       0      0 ::1:53                  :::*                    LISTEN      3359/named
tcp6       0      0 ::1:953                 :::*                    LISTEN      3359/named
udp        0      0 192.168.226.134:53      0.0.0.0:*                           3359/named
udp        0      0 127.0.0.1:53            0.0.0.0:*                           3359/named
udp6       0      0 ::1:53                  :::*                                3359/named
[root@zimbra ~]#
```

Step 7. Now do quarry to whatsapp through your local_server

# dig whatspp.com @19.168.226.134

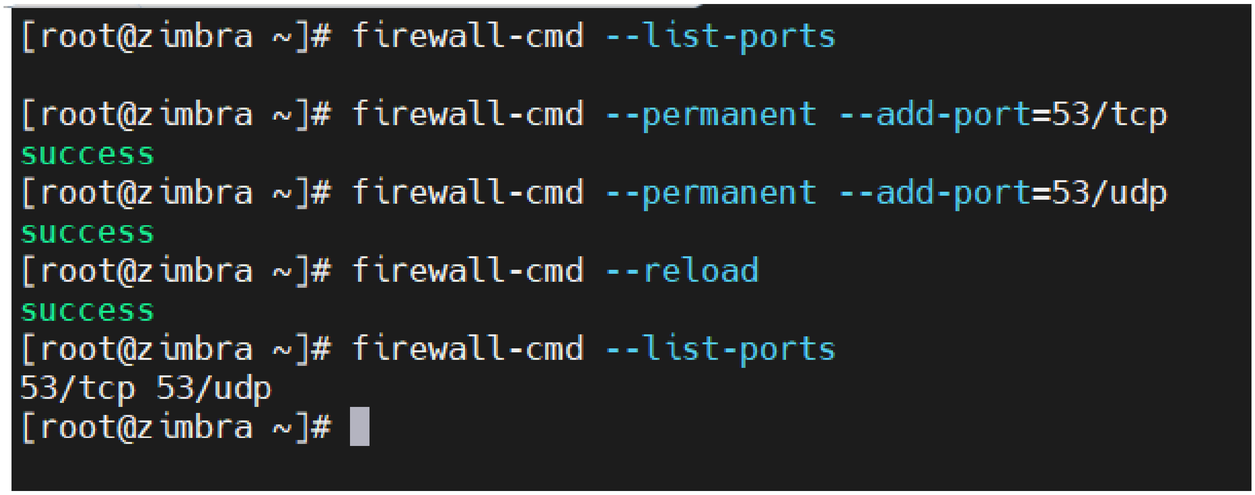


```
[root@zimbra ~]# dig whatsapp.com @192.168.226.134

; <<>> DiG 9.11.36-RedHat-9.11.36-16.el8_10.2 <<>> whatsapp.com @192.168.226.134
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5380
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 9

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 32d62e9cc799eba6f17f57be66cb2e46ac86b9b60622bf0b (good)
;; QUESTION SECTION:
;whatsapp.com.                  IN      A

;; ANSWER SECTION:
whatsapp.com.           60      IN      A       163.70.146.60

;; AUTHORITY SECTION:
whatsapp.com.           172799  IN      NS      d.ns.whatsapp.net.
whatsapp.com.           172799  IN      NS      b.ns.whatsapp.net.
whatsapp.com.           172799  IN      NS      a.ns.whatsapp.net.
whatsapp.com.           172799  IN      NS      c.ns.whatsapp.net.

;; ADDITIONAL SECTION:
a.ns.whatsapp.net.      172800  IN      A       129.134.30.12
b.ns.whatsapp.net.      172800  IN      A       129.134.31.12
c.ns.whatsapp.net.      172800  IN      A       185.89.218.12
d.ns.whatsapp.net.      172800  IN      A       185.89.219.12
a.ns.whatsapp.net.      172800  IN      AAAA    2a03:2880:f0fc:c:face:b00c:0:35
b.ns.whatsapp.net.      172800  IN      AAAA    2a03:2880:f0fd:c:face:b00c:0:35
c.ns.whatsapp.net.      172800  IN      AAAA    2a03:2880:f1fc:c:face:b00c:0:35
d.ns.whatsapp.net.      172800  IN      AAAA    2a03:2880:f1fd:c:face:b00c:0:35

;; Query time: 1878 msec
;; SERVER: 192.168.226.134#53(192.168.226.134)
;; WHEN: Sun Aug 25 06:14:46 PDT 2024
;; MSG SIZE  rcvd: 340

[root@zimbra ~]#
```

:- Now it replying from your local_server

Step 8. Check port in firewall if not added then add first

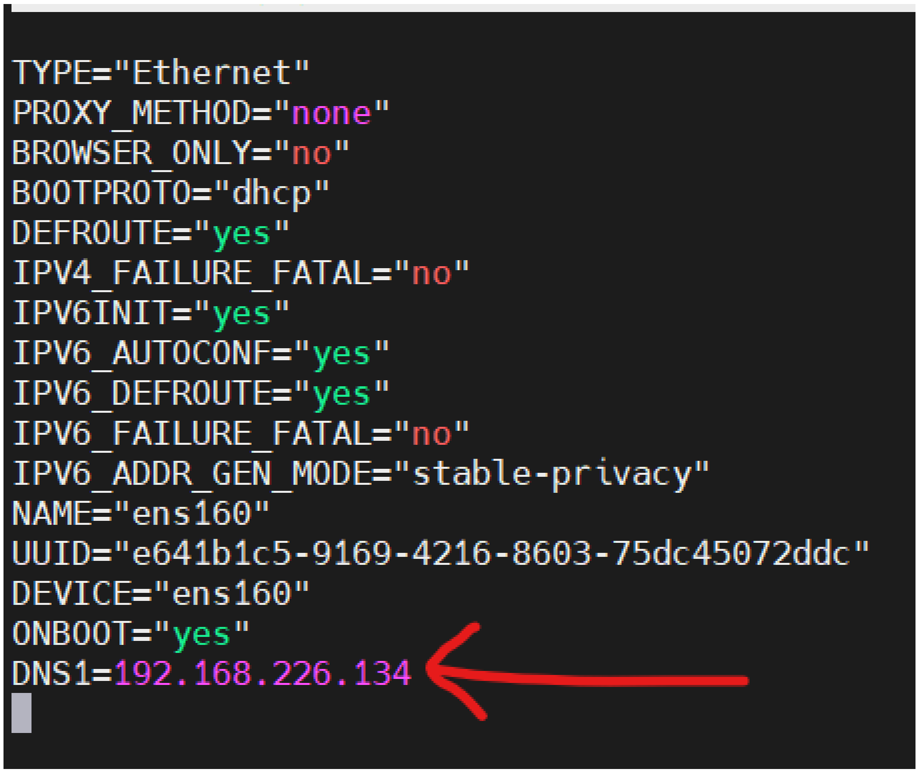
# firewall-cmd --list-ports
# firewall-cmd --permanent --add-port=53/udp
# firewall-cmd --reload
# firewall-cmd --list-ports



```
[root@zimbra ~]# firewall-cmd --list-ports

[root@zimbra ~]# firewall-cmd --permanent --add-port=53/tcp
success
[root@zimbra ~]# firewall-cmd --permanent --add-port=53/udp
success
[root@zimbra ~]# firewall-cmd --reload
success
[root@zimbra ~]# firewall-cmd --list-ports
53/tcp 53/udp
[root@zimbra ~]#
```

Step 9. Give/Edit the entry of your local_dns_server in network interface file

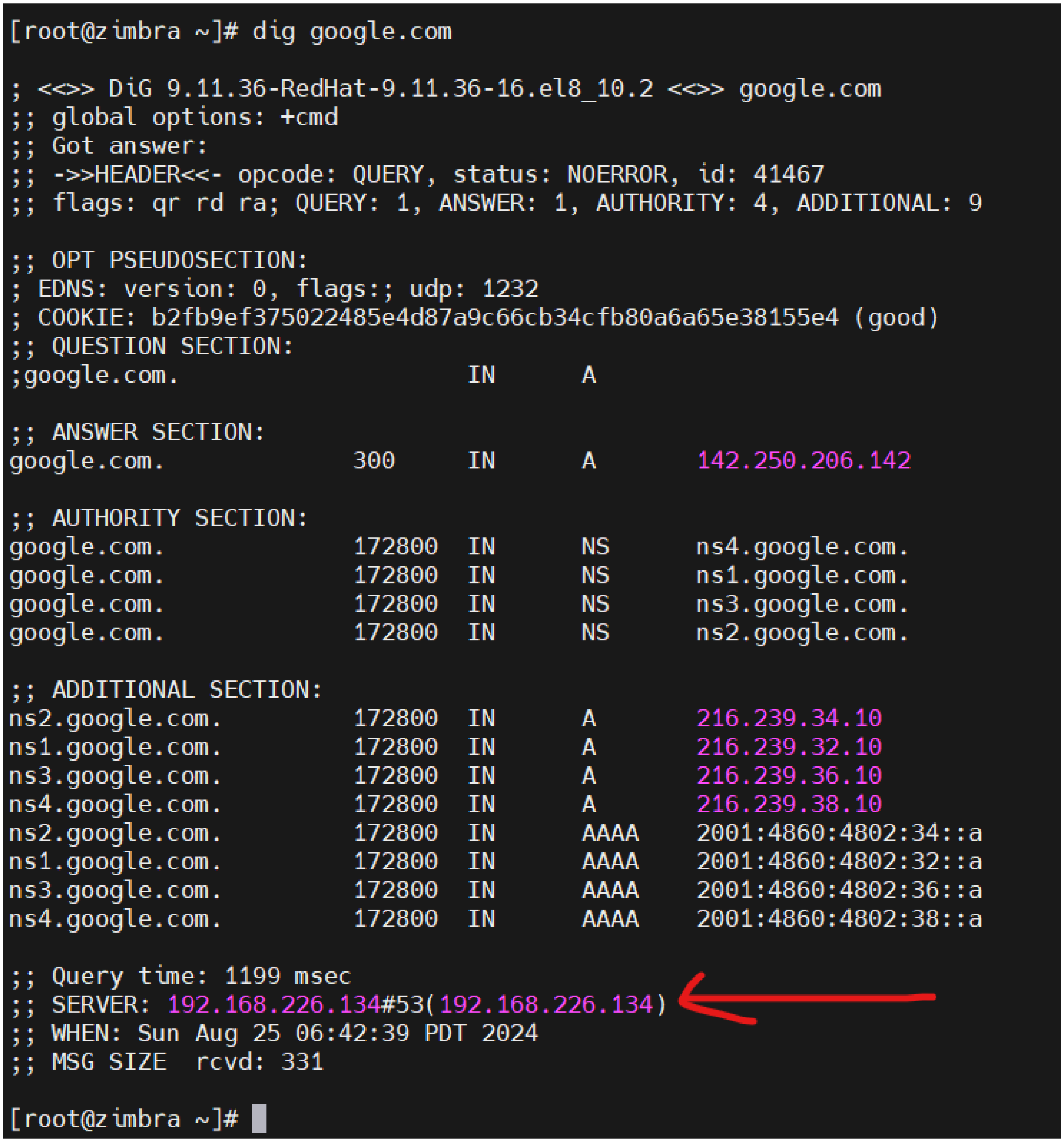
```
# nano //etc/sysconfig/network-scripts/ifcfg-end160
```



```
TYPE="Ethernet"
PROXY_METHOD="none"
BROWSER_ONLY="no"
BOOTPROTO="dhcp"
DEFROUTE="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_FAILURE_FATAL="no"
IPV6_ADDR_GEN_MODE="stable-privacy"
NAME="ens160"
UUID="e641b1c5-9169-4216-8603-75dc45072ddc"
DEVICE="ens160"
ONBOOT="yes"
DNS1=192.168.226.134
```

Step 10. Now restart the service and quarry to google

```
#systemctl restart named.service
# dig google.com
```



```
[root@zimbra ~]# dig google.com

; <<>> DiG 9.11.36-RedHat-9.11.36-16.el8_10.2 <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 41467
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 9

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: b2fb9ef375022485e4d87a9c66cb34cfb80a6a65e38155e4 (good)
;; QUESTION SECTION:
;google.com.                    IN      A

;; ANSWER SECTION:
google.com.             300     IN      A       142.250.206.142

;; AUTHORITY SECTION:
google.com.             172800  IN      NS      ns4.google.com.
google.com.             172800  IN      NS      ns1.google.com.
google.com.             172800  IN      NS      ns3.google.com.
google.com.             172800  IN      NS      ns2.google.com.

;; ADDITIONAL SECTION:
ns2.google.com.         172800  IN      A       216.239.34.10
ns1.google.com.         172800  IN      A       216.239.32.10
ns3.google.com.         172800  IN      A       216.239.36.10
ns4.google.com.         172800  IN      A       216.239.38.10
ns2.google.com.         172800  IN      AAAA    2001:4860:4802:34::a
ns1.google.com.         172800  IN      AAAA    2001:4860:4802:32::a
ns3.google.com.         172800  IN      AAAA    2001:4860:4802:36::a
ns4.google.com.         172800  IN      AAAA    2001:4860:4802:38::a

;; Query time: 1199 msec
;; SERVER: 192.168.226.134#53(192.168.226.134)
;; WHEN: Sun Aug 25 06:42:39 PDT 2024
;; MSG SIZE  rcvd: 331

[root@zimbra ~]#
```

:- Now it replying from your server in 1199 milli second

THANK YOU