DNS Server
Installation and Configuration

The "BIND" package in Linux refers to Berkeley Internet Name Domain, which is a widely used implementation of the Domain Name System (DNS) protocol. It provides the software necessary to run a DNS server on a Linux system.

The main component of the BIND package is the named daemon (named for "name daemon"), which is responsible for handling DNS queries.

BIND's behavior is controlled by configuration files, primarily /etc/named.conf and zone files typically located in /var/named or /etc/bind/zones (depending on the distribution and configuration). These files define the zones the server is authoritative for.

It is assumed that you have assigned a proper host name and a manual IP address to the server.

Login with a user having sudo permissions. Make sure the system is updated.

Installing DNS Server

sudo yum install bind

```
[admin@demosrv ~]$ sudo yum install bind
[sudo] password for admin:
Last metadata expiration check: 0:00:09 ago on Tuesday 23 September 2025 06:37:09 PM.
Dependencies resolved.
================================================================================
 Package                Architecture    Version                 Repository      Size
================================================================================
Installing:
 bind                   x86_64          32:9.18.33-3.el10       Internal-AppStream  333 k
Installing dependencies:
 bind-libs              x86_64          32:9.18.33-3.el10       Internal-AppStream  1.3 M
 bind-license           noarch          32:9.18.33-3.el10       Internal-AppStream   13 k
 fstrm                  x86_64          0.6.1-12.el10           Internal-AppStream   28 k
 libmaxminddb           x86_64          1.9.1-4.el10            Internal-AppStream   42 k
 libuv                  x86_64          1:1.49.2-2.el10         Internal-AppStream  258 k
 protobuf-c             x86_64          1.5.0-6.el10            Internal-BaseOS      32 k
Installing weak dependencies:
 bind-dnssec-utils      x86_64          32:9.18.33-3.el10       Internal-AppStream  150 k
 bind-utils             x86_64          32:9.18.33-3.el10       Internal-AppStream  224 k

Transaction Summary
================================================================================
Install  9 Packages

Total size: 2.3 M
Installed size: 6.4 M
Is this ok [y/N]: _
```

Type y to continue.

This will install the bind package which is the DNS server in Linux.

Initial DNS Server Configuration

The main configuration file of the DNS server is /etc/named.conf. You need to edit this file and configure some essential parameters.

sudo vi /etc/named.conf

```
10 options {
11         listen-on port 53 { 127.0.0.1; };
```

The DNS server listens on port 53. The above line makes the DNS server listen only on the 127.0.0.1 which is the localhost address. This makes server accept queries only locally. The network clients will not able to send queries to this server as the port is not open on the network address of the server. Thus change the line to following. Add the server IP address after 127.0.0.1.

```
10 options {
11         listen-on port 53 { 127.0.0.1; 192.168.100.1; };
```

```
12          listen-on-v6 port 53 { ::1; };
```

This line enables IPv6 support for the DNS server. As we are not using IPv6, comment out the line as shown below.

```
12          #listen-on-v6 port 53 { ::1; };
```

The following configuration allows DNS server to accept queries only from the localhost i.e. from the server only.

```
19          allow-query     { localhost; };
```

But we want DNS server to accept queries from any client over the network. Thus change the above line to following.

```
19          allow-query     { any; };
```

The following configuration informs the DNS server to consider the configuration from these files also.

```
57 include "/etc/named.rfc1912.zones";
58 include "/etc/named.root.key";
```

You can define the domains that you will register with this DNS server in the named.rfc1912.zones file. However it is recommended to define your own separate file for user defined zones. Thus we will add a name of the file here (line 59). Thus DNS server will consider the configuration within this file also.

```
57 include "/etc/named.rfc1912.zones";
58 include "/etc/named.root.key";
59 include "/etc/myzones.zones";
```

Keep all other things as default. Save the file.
The permissions on the DNS server configuration files are as shown below.

```
[admin@demosrv ~]$ ls -all /etc/named*
-rw-r-----. 1 root named 1761 Sep 23 19:12 /etc/named.conf
-rw-r-----. 1 root named 1034 Feb 13  2025 /etc/named.rfc1912.zones
-rw-r--r--. 1 root named  686 Feb 13  2025 /etc/named.root.key
```

We need to create a file /etc/myzones.zones and assign same permissions.
You need to give following commands for this.

sudo touch /etc/myzones.zones

sudo chgrp named /etc/myzones.zones

sudo chmod 640 /etc/myzones.zones

This finishes the initial DNS server configuration.

Adding a Forward Lookup Zone

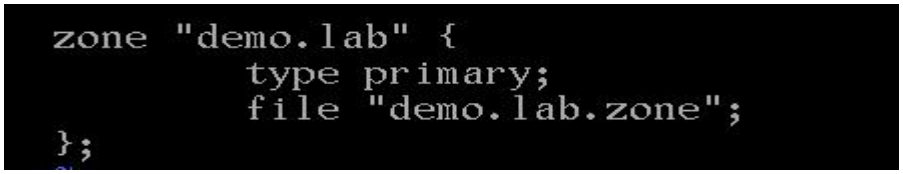
Now we will add the first zone to the DNS. The zone is nothing but a domain. You need to declare the zone (domain) in /etc/myzones.zones file and then create a zone file for this domain which will contain the various records for the domain.

sudo vi /etc/myzones.zones

Type following lines.

```
zone "demo.lab" {
        type primary;
        file "demo.lab.zone";
};
```

This will be as shown below.



Save the file.

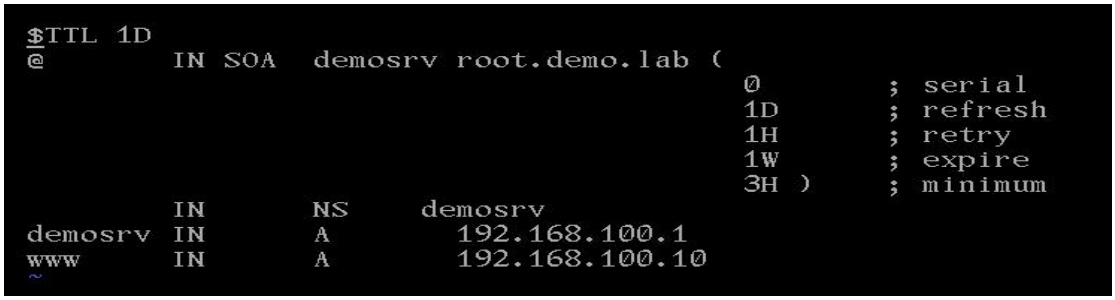
Now you need to create the zone file in /var/named directory by the name specified in the above configuration.
We need to type a lot of information in the file. Thus instead of typing everything we will use an existing file and then modify it as per our requirement.

sudo cp /var/named/named.localhost /var/named/demo.lab.zone

Now edit the new file.

sudo vi /var/named/demo.lab.zone

Modify it as shown below.



demosrv is the hostname of the server. Make sure you provide the name that you specified.
Root.demo.lab is the email ID of the administrator of this domain.

$TTL, defined in RFC 2308, is followed by a number to be used as the default TTL (time-to-live).
SOA (Start of Authority) Record: This is the first and most important record, declaring the DNS zone and specifying the authoritative name server for that zone. It also includes the email address of the administrator.
NS (Name Server) Record: These records list the DNS servers responsible for managing the zone, helping to distribute the load and ensure reliability.
Serial — the zone serial number, incremented when the zone file is modified, so the slave and secondary name servers know when the zone has been changed and should be reloaded.

Refresh — This is the number of seconds between update requests from secondary and slave name servers.

Retry — This is the number of seconds the secondary or slave will wait before retrying when the last attempt has failed.

Expire — This is the number of seconds a master or slave will wait before considering the data stale if it cannot reach the primary name server.

Minimum — Previously used to determine the minimum TTL, this is used for negative caching. This is the default TTL if the domain does not specify a TTL.

Save the file.

Now assign the permissions on the file to the named group.

sudo chgrp named /var/named/demo.lab.zone

```
[admin@demosrv ~]$ sudo chgrp named /var/named/demo.lab.zone
```

Now start the service.

sudo systemctl start named

In case if the service fails to start, it means there is some syntax error in one of the configuration file. To check give following command

sudo journalctl -xeu named

Read the output carefully, it will tell you the name of the file and the location in that file where there is an error.

If service starts successfully, verify using following command.

sudo systemctl status named

The output should look like as below.

```
[admin@demosrv ~]$ sudo systemctl status named
[sudo] password for admin:
● named.service - Berkeley Internet Name Domain (DNS)
     Loaded: loaded (/usr/lib/systemd/system/named.service; disabled; preset: disabled)
     Active: active (running) since Wed 2025-09-24 07:47:59 IST; 19min ago
 Invocation: 831b910788774924b6652ed84d2714a4
    Process: 1686 ExecStartPre=/bin/bash -c if [ ! "$DISABLE_ZONE_CHECKING" == "yes" ]; then /usr/bin/nam>
    Process: 1689 ExecStart=/usr/sbin/named -u named -c ${NAMEDCONF} $OPTIONS (code=exited, status=0/SUCC>
   Main PID: 1690 (named)
      Tasks: 8 (limit: 22951)
     Memory: 6.2M (peak: 8M)
        CPU: 115ms
     CGroup: /system.slice/named.service
             └─1690 /usr/sbin/named -u named -c /etc/named.conf
```

Type q to exit from the output.

Open port TCP 53 and UDP 53 in the firewall

sudo firewall-cmd --add-port={53/tcp,53/udp}

sudo firewall-cmd --add-port={53/tcp,53/udp} --permanent

Verify if the DNS server is working.

Configure DNS server to use local DNS service.

sudo vi /etc/resolv.conf

And add/change the entry to look as below.

```
# Generated by NetworkManager
nameserver 192.168.100.1
```

Save the file.

Now give following command to verify that the DNS server is working properly.

nslookup www.demo.lab

You should get following output.

```
[admin@demosrv ~]$ nslookup www.demo.lab
Server:         192.168.100.1
Address:        192.168.100.1#53

Name:   www.demo.lab
Address: 192.168.100.10
```

It means the DNS server configuration for the domain demo.lab was successful.

Similarly you can add other domain entries also.

Creating a Reverse Lookup Zone

A reverse lookup zone is a Domain Name System (DNS) zone that maps an IP address to its corresponding Fully Qualified Domain Name (FQDN), which is the opposite of a typical forward lookup zone that maps a domain name to an IP address. These zones use Pointer (PTR) records to perform these lookups and are essential for network troubleshooting, security verification, and the proper functioning of some network services like email

You will declare all the zones (domains) in the /etc/myzones.zones file.

Here you will create a reverse lookup zone for the network 192.168.100.0/24 network. This is the same network in which you have configured this DNS server.

sudo vi /etc/myzones.zones

Add following lines after the earlier configuration, the file will look as below.

```
zone "demo.lab" {
        type primary;
        file "demo.lab.zone";
};

zone "100.168.192.in-addr.arpa" {
        type  primary;
        file  "rev-100.zone";
};
```

While declaring the reverse lookup zone, you need to reverse the network address.

Save the file.

Now create the rev-100.zone file in /var/named directory.

sudo cp /var/named/demo.lab.zone /var/named/rev-100.zone

Change group to named so the service gets the permission to read this file.

Sudo chgrp named /var/named/rev-100.zone

```
[admin@demosrv ~]$ sudo cp /var/named/demo.lab.zone /var/named/rev-100.zone
[sudo] password for admin:
[admin@demosrv ~]$ sudo chgrp named  /var/named/rev-100.zone
[admin@demosrv ~]$
```

Now edit the rev-100.zone file and modify it to as shown below.

```
$TTL 1D
@       IN SOA  demosrv root.demo.lab (
                                        0       ; serial
                                        1D      ; refresh
                                        1H      ; retry
                                        1W      ; expire
                                        3H )    ; minimum
        IN      NS     demosrv
demosrv IN      A        192.168.100.1
1       IN      PTR      demosrv.demo.lab.
10      IN      PTR      www.demo.lab.
15      IN      PTR      exam.demo.lab._
~
```

Make sure you add .(dot) at the end of the name in the PTR records.

These records are randomly added to test the server.

Save the file.

Restart DNS service.

sudo systemctl restart named

If you have not made any typing mistake then the service will restart successfully.

Verify the service status using

sudo systemctl status named

The service should be running.

If the service is running then check if the reverse lookup zone is working correctly.

nslookup 192.168.100.10

You should get the following output.

```
[admin@demosrv ~]$ nslookup 192.168.100.10
10.100.168.192.in-addr.arpa     name = www.demo.lab.

[admin@demosrv ~]$ _
```

This is how you have successfully configured the reverse lookup zone in the DNS server.