Setting Up DNS Server On CentOS 7

DNS Server Installation

Scenario

For the purpose of this tutorial, I will be using three nodes. One will be acting as Master DNS server, the second system will be acting as Secondary DNS, and the third will be our DNS client. Here are my three systems details.

Primary (Master) DNS Server Details:

Operating System     : CentOS 7 minimal server

Hostname             : masterdns.unixmen.local

IP Address           : 192.168.1.101/24

Secondary (Slave) DNS Server Details:

Operating System     : CentOS 7 minimal server

Hostname             : secondarydns.unixmen.local

IP Address           : 192.168.1.102/24

Client Details:

Operating System     : CentOS 6.5 Desktop

Hostname             : client.unixmen.local

IP Address           : 192.168.1.103/24

Setup Primary (Master) DNS Server

Install bind9 packages on your server.

yum install bind bind-utils -y

1. Configure DNS Server

Edit **‘/etc/named.conf’** file.

vi /etc/named.conf

Add the lines as shown in bold:

//

// 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; **192.168.1.101;**}; **### Master DNS IP ###**

#    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";

    allow-query     { localhost; **192.168.1.0/24;**}; **### IP Range ###**

    allow-transfer{ localhost; **192.168.1.102;** };   **### Slave DNS IP ###**

    /\*

     - 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;

    dnssec-lookaside auto;

    /\* Path to ISC DLV key \*/

    bindkeys-file "/etc/named.iscdlv.key";

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

    pid-file "/run/named/named.pid";

    session-keyfile "/run/named/session.key";

};

logging {

        channel default\_debug {

                file "data/named.run";

                severity dynamic;

        };

};

zone "." IN {

    type hint;

    file "named.ca";

};

**zone "unixmen.local" IN {**

**type master;**

**file "forward.unixmen";**

**allow-update { none; };**

**};**

**zone "1.168.192.in-addr.arpa" IN {**

**type master;**

**file "reverse.unixmen";**

**allow-update { none; };**

**};**

include "/etc/named.rfc1912.zones";

include "/etc/named.root.key";

2. Create Zone files

Create forward and reverse zone files which we mentioned in the **‘/etc/named.conf’** file.

2.1 Create Forward Zone

Create **forward.unixmen** file in the **‘/var/named’** directory.

vi /var/named/forward.unixmen

Add the following lines:

$TTL 86400

@   IN  SOA     masterdns.unixmen.local. root.unixmen.local. (

        2011071001  ;Serial

        3600        ;Refresh

        1800        ;Retry

        604800      ;Expire

        86400       ;Minimum TTL

)

@       IN  NS          masterdns.unixmen.local.

@       IN  NS          secondarydns.unixmen.local.

@       IN  A           192.168.1.101

@       IN  A           192.168.1.102

@       IN  A           192.168.1.103

masterdns       IN  A   192.168.1.101

secondarydns    IN  A   192.168.1.102

client          IN  A   192.168.1.103

2.2 Create Reverse Zone

Create **reverse.unixmen** file in the **‘/var/named’** directory.

vi /var/named/reverse.unixmen

Add the following lines:

$TTL 86400

@   IN  SOA     masterdns.unixmen.local. root.unixmen.local. (

        2011071001  ;Serial

        3600        ;Refresh

        1800        ;Retry

        604800      ;Expire

        86400       ;Minimum TTL

)

@       IN  NS          masterdns.unixmen.local.

@       IN  NS          secondarydns.unixmen.local.

@       IN  PTR         unixmen.local.

masterdns       IN  A   192.168.1.101

secondarydns    IN  A   192.168.1.102

client          IN  A   192.168.1.103

101     IN  PTR         masterdns.unixmen.local.

102     IN  PTR         secondarydns.unixmen.local.

103     IN  PTR         client.unixmen.local.

3. Start the DNS service

Enable and start DNS service:

systemctl enable named

systemctl start named

4. Firewall Configuration

We must allow the DNS service default port 53 through firewall.

firewall-cmd --permanent --add-port=53/tcp

firewall-cmd --permanent --add-port=53/udp

5. Restart Firewall

firewall-cmd --reload

6. Configuring Permissions, Ownership, and SELinux

Run the following commands one by one:

chgrp named -R /var/named

chown -v root:named /etc/named.conf

restorecon -rv /var/named

restorecon /etc/named.conf

7. Test DNS configuration and zone files for any syntax errors

Check DNS default configuration file:

named-checkconf /etc/named.conf

If it returns nothing, your configuration file is valid.

Check Forward zone:

named-checkzone unixmen.local /var/named/forward.unixmen

Sample output:

zone unixmen.local/IN: loaded serial 2011071001

OK

Check reverse zone:

named-checkzone unixmen.local /var/named/reverse.unixmen

Sample Output:

zone unixmen.local/IN: loaded serial 2011071001

OK

Add the DNS Server details in your network interface config file.

vi /etc/sysconfig/network-scripts/ifcfg-enp0s3

TYPE="Ethernet"

BOOTPROTO="none"

DEFROUTE="yes"

IPV4\_FAILURE\_FATAL="no"

IPV6INIT="yes"

IPV6\_AUTOCONF="yes"

IPV6\_DEFROUTE="yes"

IPV6\_FAILURE\_FATAL="no"

NAME="enp0s3"

UUID="5d0428b3-6af2-4f6b-9fe3-4250cd839efa"

ONBOOT="yes"

HWADDR="08:00:27:19:68:73"

IPADDR0="192.168.1.101"

PREFIX0="24"

GATEWAY0="192.168.1.1"

**DNS="192.168.1.101"**

IPV6\_PEERDNS="yes"

IPV6\_PEERROUTES="yes"

Edit file **/etc/resolv.conf**,

vi /etc/resolv.conf

Add the name server ip address:

nameserver      192.168.1.101

Save and close the file.

Restart network service:

systemctl restart network

8. Test DNS Server

dig masterdns.unixmen.local

Sample Output:

; <<>> DiG 9.9.4-RedHat-9.9.4-14.el7 <<>> masterdns.unixmen.local

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25179

;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; OPT PSEUDOSECTION:

; EDNS: version: 0, flags:; udp: 4096

;; QUESTION SECTION:

;masterdns.unixmen.local.    IN    A

;; ANSWER SECTION:

masterdns.unixmen.local. 86400    IN    A    192.168.1.101

;; AUTHORITY SECTION:

unixmen.local.        86400    IN    NS    secondarydns.unixmen.local.

unixmen.local.        86400    IN    NS    masterdns.unixmen.local.

;; ADDITIONAL SECTION:

secondarydns.unixmen.local. 86400 IN    A    192.168.1.102

;; Query time: 0 msec

;; SERVER: 192.168.1.101#53(192.168.1.101)

;; WHEN: Wed Aug 20 16:20:46 IST 2014

;; MSG SIZE  rcvd: 125

nslookup unixmen.local

Sample Output:

Server:        192.168.1.101

Address:    192.168.1.101#53

Name:    unixmen.local

Address: 192.168.1.103

Name:    unixmen.local

Address: 192.168.1.101

Name:    unixmen.local

Address: 192.168.1.102

Now the Primary DNS server is ready to use.

It is time to configure our Secondary DNS server.

Setup Secondary(Slave) DNS Server

Install bind packages using the following command:

yum install bind bind-utils -y

1. Configure Slave DNS Server

Edit file **‘/etc/named.conf’**:

vi /etc/named.conf

Make the changes as shown in bold.

//

// 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; **192.168.1.102;** };

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";

allow-query     { localhost; **192.168.1.0/24;** };

.

.

.

.

zone "." IN {

type hint;

file "named.ca";

};

**zone "unixmen.local" IN {**

**type slave;**

**file "slaves/unixmen.fwd";**

**masters { 192.168.1.101; };**

**};**

**zone "1.168.192.in-addr.arpa" IN {**

**type slave;**

**file "slaves/unixmen.rev";**

**masters { 192.168.1.101; };**

**};**

include "/etc/named.rfc1912.zones";

include "/etc/named.root.key";

2. Start the DNS Service

systemctl enable named

systemctl start named

Now the forward and reverse zones are automatically replicated from Master DNS server to ‘/var/named/slaves/’ in Secondary DNS server.

ls /var/named/slaves/

Sample Output:

unixmen.fwd  unixmen.rev

3. Add the DNS Server details

Add the DNS Server details in your network interface config file.

vi /etc/sysconfig/network-scripts/ifcfg-enp0s3

TYPE="Ethernet"

BOOTPROTO="none"

DEFROUTE="yes"

IPV4\_FAILURE\_FATAL="no"

IPV6INIT="yes"

IPV6\_AUTOCONF="yes"

IPV6\_DEFROUTE="yes"

IPV6\_FAILURE\_FATAL="no"

NAME="enp0s3"

UUID="5d0428b3-6af2-4f6b-9fe3-4250cd839efa"

ONBOOT="yes"

HWADDR="08:00:27:19:68:73"

IPADDR0="192.168.1.102"

PREFIX0="24"

GATEWAY0="192.168.1.1"

**DNS1="192.168.1.101"**

**DNS2="192.168.1.102"**

IPV6\_PEERDNS="yes"

IPV6\_PEERROUTES="yes"

Edit file **/etc/resolv.conf**,

vi /etc/resolv.conf

Add the name server ip address:

nameserver      192.168.1.101

nameserver      192.168.1.102

Save and close the file.

Restart network service:

systemctl restart network

4. Firewall Configuration

We must allow the DNS service default port 53 through firewall.

firewall-cmd --permanent --add-port=53/tcp

5. Restart Firewall

firewall-cmd --reload

6. Configuring Permissions, Ownership, and SELinux

chgrp named -R /var/named

chown -v root:named /etc/named.conf

restorecon -rv /var/named

restorecon /etc/named.conf

7. Test DNS Server

dig masterdns.unixmen.local

Sample Output:

; <<>> DiG 9.9.4-RedHat-9.9.4-14.el7 <<>> masterdns.unixmen.local

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18204

;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; OPT PSEUDOSECTION:

; EDNS: version: 0, flags:; udp: 4096

;; QUESTION SECTION:

;masterdns.unixmen.local.    IN    A

;; ANSWER SECTION:

masterdns.unixmen.local. 86400    IN    A    192.168.1.101

;; AUTHORITY SECTION:

unixmen.local.        86400    IN    NS    masterdns.unixmen.local.

unixmen.local.        86400    IN    NS    secondarydns.unixmen.local.

;; ADDITIONAL SECTION:

secondarydns.unixmen.local. 86400 IN    A    192.168.1.102

;; Query time: 0 msec

;; SERVER: 192.168.1.102#53(192.168.1.102)

;; WHEN: Wed Aug 20 17:04:30 IST 2014

;; MSG SIZE  rcvd: 125

dig secondarydns.unixmen.local

Sample Output:

; <<>> DiG 9.9.4-RedHat-9.9.4-14.el7 <<>> secondarydns.unixmen.local

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60819

;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; OPT PSEUDOSECTION:

; EDNS: version: 0, flags:; udp: 4096

;; QUESTION SECTION:

;secondarydns.unixmen.local.    IN    A

;; ANSWER SECTION:

secondarydns.unixmen.local. 86400 IN    A    192.168.1.102

;; AUTHORITY SECTION:

unixmen.local.        86400    IN    NS    masterdns.unixmen.local.

unixmen.local.        86400    IN    NS    secondarydns.unixmen.local.

;; ADDITIONAL SECTION:

masterdns.unixmen.local. 86400    IN    A    192.168.1.101

;; Query time: 0 msec

;; SERVER: 192.168.1.102#53(192.168.1.102)

;; WHEN: Wed Aug 20 17:05:50 IST 2014

;; MSG SIZE  rcvd: 125

nslookup unixmen.local

Sample Output:

Server:        192.168.1.102

Address:    192.168.1.102#53

Name:    unixmen.local

Address: 192.168.1.101

Name:    unixmen.local

Address: 192.168.1.103

Name:    unixmen.local

Address: 192.168.1.102

Client Side Configuration

Add the DNS server details in **‘/etc/resolv.conf’** file in all client systems

vi /etc/resolv.conf

# Generated by NetworkManager

search unixmen.local

nameserver 192.168.1.101

nameserver 192.168.1.102

Restart network service or reboot the system.

Test DNS Server

Now, you can test the DNS server using any one of the following commands:

dig masterdns.unixmen.local

dig secondarydns.unixmen.local

dig client.unixmen.local

nslookup unixmen.local

That’s all about now. The primary and secondary DNS servers are ready to use.

If you want to setup DNS server on Ubuntu systems, check the following link.