

Week #4

Implementation of a Local DNS Server

DNS (Domain Name System) is the Internet's phone book; it translates hostnames to IP addresses (and vice versa). This translation is through DNS resolution, which happens behind the scene.

The objectives of this lab are to understand:

- DNS and how it works
- Install and set up a DNS server
- Functionality and operations

Lab Setup

DNS Server: 10.0.2.15

User/Client: 10.0.2.4

First Test:

Ping a computer such as www.flipkart.com. Please use Wireshark to show the DNS query triggered by your ping command and DNS response. Describe your observation. (Take a screenshot).

dns					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000000	127.0.0.1	127.0.0.53	DNS	78 Standard query 0x47b0 A www.flipkart.com
2	0.000034996	127.0.0.1	127.0.0.53	DNS	78 Standard query 0xdab2 AAAA www.flipkart.com
3	0.000251993	10.0.2.15	192.168.43.1	DNS	78 Standard query 0x204f A www.flipkart.com
4	0.000444357	10.0.2.15	192.168.43.1	DNS	78 Standard query 0x6a01 AAAA www.flipkart.com
5	0.103950680	192.168.43.1	10.0.2.15	DNS	237 Standard query response 0x204f A www.flipkart.com CNAME flipk...
6	0.104236761	127.0.0.53	127.0.0.1	DNS	108 Standard query response 0x47b0 A www.flipkart.com CNAME flipk...
7	0.116236147	192.168.43.1	10.0.2.15	DNS	155 Standard query response 0x6a01 AAAA www.flipkart.com CNAME fl...
8	0.116517229	10.0.2.15	192.168.43.1	DNS	74 Standard query 0x8a27 AAAA flipkart.com
9	0.119693522	192.168.43.1	10.0.2.15	DNS	74 Standard query response 0x8a27 AAAA flipkart.com
10	0.119850890	127.0.0.53	127.0.0.1	DNS	92 Standard query response 0xdab2 AAAA www.flipkart.com CNAME fl...
13	0.239281454	127.0.0.1	127.0.0.53	DNS	88 Standard query 0x8f51 PTR 110.78.53.163.in-addr.arpa
14	0.239523849	10.0.2.15	192.168.43.1	DNS	88 Standard query 0xbb0d PTR 110.78.53.163.in-addr.arpa
15	0.250051672	192.168.43.1	10.0.2.15	DNS	88 Standard query response 0xbb0d No such name PTR 110.78.53.163...
16	0.250329368	127.0.0.53	127.0.0.1	DNS	88 Standard query response 0x8f51 No such name PTR 110.78.53.163...

Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface any, id 0
Linux cooked capture
Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.53
User Datagram Protocol, Src Port: 35328, Dst Port: 53
Domain Name System (query)

0000	00 00 03 04 00 00 00 00 00 00 00 00 00 08 00
0010	45 00 00 3e 01 00 40 00 40 11 3b 6c 7f 00 00 01	E-> .@. @;1...
0020	7f 00 00 35 8a 00 00 35 00 2a fe 71 47 b0 01 00	...5...5 -*qG...
0030	00 01 00 00 00 00 00 00 03 77 77 08 66 6c 69 www fl1
0040	70 6b 61 72 74 03 63 6f 6d 00 00 01 00 01	pkart co m....

Part 1: Setting Up a Local DNS Server

Task 1: Configure the User Machine

On the client machine 10.0.2.4, we need to use 10.0.2.15 as the local DNS server. This is achieved by changing the resolver configuration file (**/etc/resolv.conf**) of the user machine, so the server 10.0.2.15 is added as the first nameserver entry in the file, i.e., this server will be used as the primary DNS server. Add the following entry to the **/etc/resolvconf/resolv.conf.d/head** file.

nameserver 10.0.2.15

Run the following command for the change to take effect. **sudo resolvconf -u**
The following screenshot shows how to set DNS server on the client machine.

```
itsatul@Ruby:~$ sudo nano /etc/resolvconf/resolv.conf.d/head
[sudo] password for itsatul:
itsatul@Ruby:~$ sudo resolvconf -u
itsatul@Ruby:~$ sudo cat /etc/resolvconf/resolv.conf.d/head
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
# 127.0.0.53 is the systemd-resolved stub resolver.
# run "systemd-resolve --status" to see details about the actual nameservers.

nameserver 10.0.2.15
```

Also, add 10.0.2.15 in ‘Additional DNS servers’ field in IPv4 settings of client machine.

The screenshot shows the NetworkManager configuration window for a wired connection. The 'IPv4' tab is selected. Under 'IPv4 Method', 'Automatic (DHCP)' is chosen. The 'DNS' section is expanded, showing 'Automatic' as the method and '10.0.2.15' entered in the text field. The 'Routes' section is also expanded, showing 'Automatic' as the method. A checkbox at the bottom is labeled 'Use this connection only for resources on its network'.

Second Test:

Ping a computer such as www.flipkart.com. Please use Wireshark to show the DNS query triggered by your ping command and DNS response. Describe your observation. (Take a screenshot).

```
itsatul@Ruby:~$ ping www.flipkart.com
PING flipkart.com (163.53.78.110) 56(84) bytes of data.
64 bytes from 163.53.78.110 (163.53.78.110): icmp_seq=1 ttl=53 time=109 ms
64 bytes from 163.53.78.110 (163.53.78.110): icmp_seq=3 ttl=53 time=101 ms
64 bytes from 163.53.78.110 (163.53.78.110): icmp_seq=4 ttl=53 time=99.1 ms
^C
--- flipkart.com ping statistics ---
4 packets transmitted, 3 received, 25% packet loss, time 3009ms
rtt min/avg/max/mdev = 99.052/103.078/109.234/4.420 ms
itsatul@Ruby:~$
```

dns						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.4	10.0.2.15	DNS	78	Standard query 0x3a00 A www.flipkart.com
2	0.000025259	10.0.2.4	10.0.2.15	DNS	78	Standard query 0x3b0f AAAA www.flipkart.com
3	0.000330146	10.0.2.15	10.0.2.4	ICMP	106	Destination unreachable (Port unreachable)
4	0.000330327	10.0.2.15	10.0.2.4	ICMP	106	Destination unreachable (Port unreachable)
5	0.000397197	127.0.0.1	127.0.0.53	DNS	78	Standard query 0x3a00 A www.flipkart.com
6	0.000418839	127.0.0.1	127.0.0.53	DNS	78	Standard query 0x3b0f AAAA www.flipkart.com
7	0.000670324	10.0.2.4	192.168.43.1	DNS	78	Standard query 0x45fe A www.flipkart.com
8	0.000781060	10.0.2.4	192.168.43.1	DNS	78	Standard query 0xf71e AAAA www.flipkart.com
9	0.230978808	192.168.43.1	10.0.2.4	DNS	108	Standard query response 0x45fe A www.flipkart.com CNAME flipk...
10	0.231250192	127.0.0.53	127.0.0.1	DNS	108	Standard query response 0x3a00 A www.flipkart.com CNAME flipk...
11	0.268590218	192.168.43.1	10.0.2.4	DNS	155	Standard query response 0xf71e AAAA www.flipkart.com CNAME fl...
12	0.268915488	10.0.2.4	192.168.43.1	DNS	74	Standard query 0xece6 AAAA flipkart.com
13	0.283964192	192.168.43.1	10.0.2.4	DNS	74	Standard query response 0xece6 AAAA flipkart.com
14	0.284160675	127.0.0.53	127.0.0.1	DNS	92	Standard query response 0x3b0f AAAA www.flipkart.com CNAME fl...
17	0.393849245	10.0.2.4	10.0.2.15	DNS	88	Standard query 0x6757 PTR 110.78.53.163.in-addr.arpa
18	0.394116565	10.0.2.15	10.0.2.4	ICMP	116	Destination unreachable (Port unreachable)
19	0.394170429	127.0.0.1	127.0.0.53	DNS	88	Standard query 0x6757 PTR 110.78.53.163.in-addr.arpa
20	0.394413924	10.0.2.4	192.168.43.1	DNS	88	Standard query 0xee12 PTR 110.78.53.163.in-addr.arpa
21	0.663585773	192.168.43.1	10.0.2.4	DNS	176	Standard query response 0xee12 No such name PTR 110.78.53.163...
22	0.663906735	127.0.0.53	127.0.0.1	DNS	88	Standard query response 0x6757 No such name PTR 110.78.53.163...

Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface any, id 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.0.2.4, Dst: 10.0.2.15
User Datagram Protocol, Src Port: 33567, Dst Port: 53
Domain Name System (query)

0000	00 04 00 01 00 06 08 00	27 27 eb 59 00 00 08 06Y..
0010	45 00 00 3e dc 5b 40 00	40 11 46 41 0a 00 02 04	E->[0: @ FA...
0020	0a 00 02 0f 83 1f 00 35	00 2a 18 4e 3a 00 01 005..N:...
0030	00 01 00 00 00 00 00 00	03 77 77 77 08 66 6c 69www.fli
0040	70 6b 61 72 74 03 63 6f	6d 00 00 01 00 01	pkart-co m.....

Task 2: Set Up a Local DNS Server

Note: If bind9 server is not already installed, install using the command

\$ sudo apt-get update \$ sudo apt-get install bind9

Step 1: Configure the BIND9 Server.

BIND9 gets its configuration from a file called `/etc/bind/named.conf`. This file is the primary configuration file, and it usually contains several “include” entries. One of the included files is called `/etc/bind/named.conf.options`. This is where we typically set up the configuration options. Let us first set up an option related to DNS cache by adding a dump-file entry to the options block. The above option specifies where the cache content should be dumped to if BIND is asked to dump its cache.

```
GNU nano 4.8 /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    dump-file "/var/cache/bind/dump.db";
```

The above option specifies where the cache content should be dumped to if BIND is asked to dump its cache. If this option is not specified, BIND dumps the cache to a default file called `/var/cache/bind/named_dump.db`.

Step 2: Start DNS server

We start the DNS server using the command:

\$ sudo service bind9 restart

The two commands shown below are related to DNS cache. The first command dumps the content of the cache to the file specified above, and the second command clears the cache.

```
itsatul@Pearl:~$ sudo service bind9 restart
itsatul@Pearl:~$ sudo rndc dumpdb -cache
itsatul@Pearl:~$ sudo rndc flush
itsatul@Pearl:~$
```

Step 3: Use the DNS server

Task 3: Host a Zone in the Local DNS server.

Assume that we own a domain, we will be responsible for providing the definitive answer regarding this domain. We will use our local DNS server as the authoritative nameserver for the domain. In this lab, we will set up an authoritative server for the **pastel.com** domain. This domain name is reserved for use in documentation, and is not owned by anybody, so it is safe to use it.

Step 1: Create Zones

We had two zone entries in the DNS server by adding the following contents to **/etc/bind/named.conf** as shown in the below screenshot. The first zone is for forward lookup (from hostname to IP), and the second zone is for reverse lookup (from IP to hostname).

```
itsatul@Pearl:~$ sudo nano /etc/bind/named.conf
[sudo] password for itsatul:
itsatul@Pearl:~$ sudo cat /etc/bind/named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";

zone "pastel.com" {
    type master;
    file "/etc/bind/pastel.com.db";
};

zone "10.0.2.in-addr.arpa" {
    type master;
    file "/etc/bind/10.0.2.db";
};
itsatul@Pearl:~$
```

Step 2: Setup the forward lookup zone file

We create **pastel.com.db** zone file with the following contents in the **/etc/bind/** directory where the actual DNS resolution is stored.



```
Open  pastel.com.db
      /etc/bind
1 $TTL 3D
2 @      IN      SOA ns.pastel.com. admin.pastel.com. (
3         2008111001
4         8H
5         2H
6         4W
7         1D)
8
9 @      IN      NS      ns.pastel.com.
10 @     IN      MX      10 mail.pastel.com.
11
12 www   IN      A       10.0.2.101
13 mail  IN      A       10.0.2.102
14 ns    IN      A       10.0.2.10
15 *.pastel.com. IN A 10.0.2.100
```

The symbol '@' is a special notation representing the origin specified in **named.conf** (the string after "zone"). Therefore, '@' here stands for **pastel.com**. This zone file contains 7 resource records (RRs), including a SOA (Start Of Authority) RR, a NS (Name Server) RR, a MX (Mail eXchanger) RR, and 4 A (host Address) RRs.

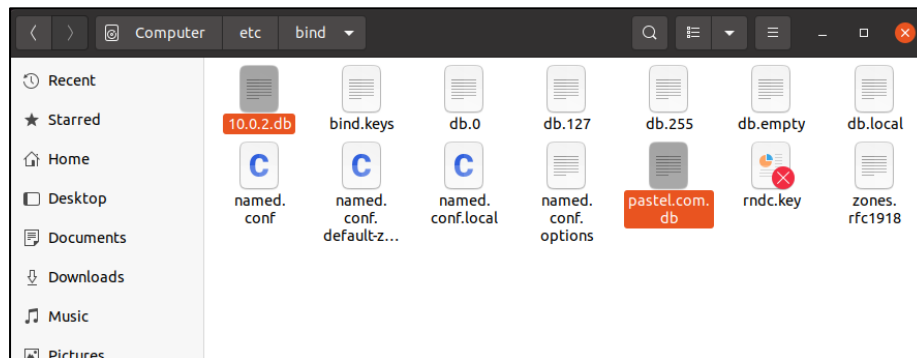
Step 3: Setup the reverse lookup zone file

We create a reverse DNS lookup file called **10.0.2.db** for the example.net domain to support DNS reverse lookup, i.e., from IP address to hostname in the **/etc/bind/** directory with the following contents.



```
1 $TTL 3D
2 @      IN      SOA  ns.pastel.com. admin.pastel.com. (
3          2008111001
4          8H
5          2H
6          4W
7          1D)
8 @      IN      NS   ns.pastel.com.
9
10 101    IN      PTR  www.pastel.com.
11 102    IN      PTR  mail.pastel.com.
12 10     IN      PTR  ns.pastel.com.
```

Step 4: Copy the above files into **/etc/bind** location.



Task 4: Restart the BIND server and test

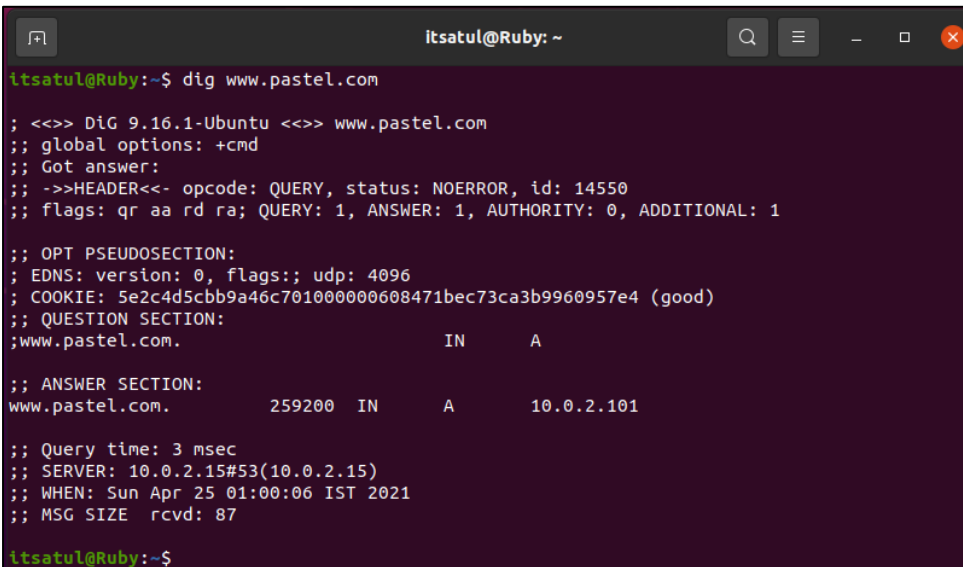
Step 1: When all the changes are made, remember to restart the BIND server. Now we will restart the DNS server using the following command:

\$ sudo service bind9 restart

```
itsatul@Pearl:~$ sudo service bind9 restart
itsatul@Pearl:~$
```

Step 2: Now, go back to the client machine and ask the local DNS server for the IP address of www.pastel.com using the dig command.

Dig stands for (Domain Information Groper) is a network administration command-line tool for querying DNS name servers. It is useful for verifying and troubleshooting DNS problems and also to perform DNS lookups and displays the answers that are returned from the name server that were queried. dig is part of the BIND domain name server software suite.



```
itsatul@Ruby: ~
itsatul@Ruby:~$ dig www.pastel.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.pastel.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 14550
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 5e2c4d5cbb9a46c701000000608471bec73ca3b9960957e4 (good)
;; QUESTION SECTION:
;www.pastel.com.                IN      A

;; ANSWER SECTION:
www.pastel.com.                259200  IN      A      10.0.2.101

;; Query time: 3 msec
;; SERVER: 10.0.2.15#53(10.0.2.15)
;; WHEN: Sun Apr 25 01:00:06 IST 2021
;; MSG SIZE rcvd: 87

itsatul@Ruby:~$
```

We can see that the ANSWER SECTION contains the DNS mapping. We can see that the IP address of www.pastel.com is now 10.0.2.101, which is what we have setup in the DNS server.

Step 3: Observe the results in Wireshark capture.

Apply a display filter ... <Ctrl-/>					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000000	127.0.0.1	127.0.0.1	UDP	45 44265 → 44265 Len=1
2	0.000044482	::1	::1	UDP	65 47372 → 47372 Len=1
3	0.000092972	10.0.2.4	10.0.2.15	DNS	99 Standard query 0x7869 A www.pastel.com OPT
4	0.000573866	10.0.2.15	10.0.2.4	DNS	131 Standard query response 0x7869 A www.pastel.com A 10.0.2.101 OPT

▶ Frame 4: 131 bytes on wire (1048 bits), 131 bytes captured (1048 bits) on interface any, id 0

▶ Linux cooked capture

▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.4

▶ User Datagram Protocol, Src Port: 53, Dst Port: 46125

▶ Domain Name System (response)

```
▶ Frame 4: 131 bytes on wire (1048 bits), 131 bytes captured (1048 bits) on interface any, id 0
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.4
▶ User Datagram Protocol, Src Port: 53, Dst Port: 46125
▼ Domain Name System (response)
  Transaction ID: 0x7869
  ▶ Flags: 0x8580 Standard query response, No error
  Questions: 1
  Answer RRs: 1
  Authority RRs: 0
  Additional RRs: 1
  ▶ Queries
  ▼ Answers
    ▼ www.pastel.com: type A, class IN, addr 10.0.2.101
      Name: www.pastel.com
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 259200 (3 days)
      Data length: 4
      Address: 10.0.2.101
    ▼ Additional records
      ▼ <Root>: type OPT
        Name: <Root>
        Type: OPT (41)
        UDP payload size: 4096
        Higher bits in extended RCODE: 0x00
        EDNS0 version: 0
        ▼ Z: 0x0000
          0... .. = DO bit: Cannot handle DNSSEC security RRs
          .000 0000 0000 0000 = Reserved: 0x0000
          Data length: 28
        ▼ Option: COOKIE
          Option Code: COOKIE (10)
          Option Length: 24
          Option Data: 5024af451f8912d80100000060847254867c4a60cae75527
          Client Cookie: 5024af451f8912d8
          Server Cookie: 0100000060847254867c4a60cae75527
      [Request In: 3]
      [Time: 0.000480894 seconds]
```

To load and clear DNS cache, use the below commands.

```
itsatul@Pearl:~$ sudo rndc dumpdb -cache
itsatul@Pearl:~$ sudo rndc flush
itsatul@Pearl:~$
```

Observation Notebook Requirements:

For 'ping www.flipkart.com', answer the following questions

- 1) Locate the DNS query and response messages. Are then sent over UDP or TCP?
 - The DNS query and response messages are visible in the screenshots.They are sent over UDP.
- 2) What is the destination port for the DNS query message? What is the source port of DNS response message?
 - The destination and source ports of the DNS query and response messages are the same. The source port of DNS response message is 53.
- 3) To what IP address is the DNS query message sent? Use ipconfig to determine the IP address of your local DNS server. Are these two IP addresses the same?
 - The DNS query message is sent to the IP Address 10.0.2.15 which is the same for local DNS Server.
- 4) Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?
 - The DNS Query is of type A since it requests for an authoritative record. The answer section is empty since it does not have any answer.