Lab Exercise 2: Web Proxies and DNS

AIM

Web Proxy: To understand the working of a web proxy.
DNS: To understand how DNS works by using the nslookup tool and Wireshark.

EXPERIMENT 1: Web Proxy

Tools

For this experiment, we will use the netstat utility available in Linux. It can print information about network connections, routing tables, interface statistics, etc. However, for this experiment we are only interested in its ability to print information about active network connections on our computer.

Exercise

Follow the steps described below. You will notice certain questions as you attempt the exercise. Write down the answers for your own reference. The solutions will be put up on the webpage at the end of the lab. If you have any questions or are experiencing difficulty with executing the lab please consult your lab teacher.

Step 1: Open a web browser window and open the main CSE webpage, www.cse.univdhaka.edu

Step 2: Open an xterm window and increase the size of the window to a reasonably large size.

Step 3: We will now look at the end-points (sockets) of the TCP connection established between your computer and the DU web server. In the xterm type,

netstat –t –n

If you encounter a large number of connections and your screen scrolls down, then follow instructions in Step 3b. Otherwise, continue to Step 4.

Step 3b (Optional) Employing a similar approach to the first lab experiment on FTP can filter out all connections not of interest. In the xterm window try and locate the process id (pid) of your browser process by typing,
ps –U yourloginname
Now remember the pid for the browser process and type the following at the prompt,
netstat –t –n –p | grep processid

Step 4: Note down the IP address of remote end of the connection (i.e. Non-local socket).

Step 5: Now change the URL in your browser by typing in: www.iit.univdhaka.edu. Immediately run netstat as above and note down the IP address of the remote socket of the connection.

Question 1. Is the IP address of the remote end of the TCP connection the same?

Step 6: Now change the URL in your browser by typing in: www.microsoft.com (a site external to the DU network) Note the IP address of the remote socket by running netstat as before.

Step 7: Now change the URL to: www.cnn.com. Note the IP address of the remote socket.

Question 2. Is the IP address of the remote end of the socket different from the one recorded in step 6? Is the pattern you observe with internal sites similar to the pattern with external sites? Suggest why the patterns are similar/dissimilar?

EXPERIMENT 2: DNS

As described in the lecture, the Domain Name System (DNS) translates hostnames to IP addresses, fulfilling a critical role in the Internet infrastructure. In this lab, we'll take a closer look at the client side of DNS. Recall that the client's role in the DNS is relatively simple – a client sends a query to its local DNS server, and receives a response back. As shown in the textbook, much can go on “under the covers”, invisible to the DNS clients, as the hierarchical DNS servers communicate with each other to either recursively or iteratively resolve the client’s DNS query. From the DNS client’s standpoint, however, the protocol is quite simple – a query is formulated to the local DNS server and a response is received from that server. Before beginning this lab, you’ll probably want to review DNS in the text. In particular, you may want to review the material on local DNS servers, DNS caching, DNS records and messages, and the TYPE field in the DNS record.

Tools

nslookup

In this experiment, we’ll make extensive use of the nslookup tool, which is available on our lab computers (and also most Linux/Unix and Microsoft platforms) To run nslookup in Linux/Unix, you should open a xterm window and type the nslookup command on the command line.

In it is most basic operation, nslookup tool allows the host running the tool to query any specified DNS server for a DNS record. The queried DNS server can be a root DNS server, a top-level-domain DNS server, an authoritative DNS server, or an intermediate DNS server (see the textbook for definitions of these terms). To accomplish this task, nslookup sends a DNS query to the specified DNS server, receives a DNS reply from that same DNS server, and displays the result.

Exercise

Follow the steps described below. You will notice certain questions as you attempt the exercise. Write down the answers for your own reference. The solutions will be put up on the webpage at the end of the lab. If you have any questions or are experiencing difficulty with executing the lab please consult your lab instructor.

Step 1: Open an xterm window and increase the size of the window to a reasonably large size.

Step 2: Use nslookup to find out details about the CSE web server. Type,

nslookup www.cse.univdhaka.edu

Question 1. What is the canonical name for the CSE web server? What is its IP address? Suggest a reason for having an alias for this server.

Question 2. The first line of the above input indicates a server address. What is the purpose of this server?

Step 3: We will now query for the NS record for the "cse.univdhaka.edu" domain. Type, *nslookup –type=NS cse.univdhaka.edu*

This causes nslookup to send a query for a type-NS record to the default local DNS server. In words, the query is saying, "please send me the host names of the authoritative DNS for "cse.univdhaka.edu". (When the –type option is not used, nslookup uses the default, which is to query for type A records, as we did in Step 2).

Question 3. What are the DNS name servers for the "cse.univdhaka.edu" domain? Find out their IP addresses?

Step 4: We will do the same query for "www.mit.edu" domain. Observe the reply accordingly.

Experiment 3: Tracing DNS with Wireshark Tools

For this experiment, we will use the Wireshark packet analyser that we used extensively in the previous lab. Before you begin capture some packets and save as dns-ethereal-trace-2 file for the DNS lab. Or you can do the experiment with the provided trace file.

Exercise

Follow the steps described below. You will notice certain questions as you attempt the exercise. Write down the answers for your own reference.

Step 1: Open an xterm and run Wireshark.

Step 2: Load the trace file dns-ethereal-trace-2 by using the File pull down menu, choosing Open and selecting the appropriate trace file. This file captures the sequence of messages exchanged between a host and its default DNS server while using the nslookup utility for obtaining the canonical name (type A record) of www.mit.edu. Now filter out all non-DNS packets by typing “dns” (without quotes) in the filter filed. Also click the right arrow for DNS in the packet-header detail window. Now focus on the last two DNS messages and answer the following questions:

1.What transport layer protocol is being used by the DNS messages? Are they sent over UDP or TCP?
2.What is the source and destination port for the DNS query message and the corresponding response?
3.To what IP address is the DNS query message sent? Is this the same as the default local DNS server?
4.How many “questions” are contained in the DNS query message? What “Type” of DNS queries are they? Does the query message also contain any “answers”?
5.Examine the DNS response message. Provide details of the contents of the “Answers”, “Authority” and “Additional Information” fields. What can you infer from these?
6.Consider the subsequent TCP SYN packet sent by your host. Does the destination IP address of the SYN packet correspond to any of the IP addresses provided in the DNS response message?
7. This web page contains images. Before retrieving each image, does your host issue new DNS queries?

Step 3: Again capture packets and save as dns-ethereal-trace-3. Load the trace file dns-ethereal-trace-3 by using the File pull down menu, choosing Open and selecting the appropriate trace file. This file captures the sequence of messages exchanged between a host and its default DNS server while using the nslookup utility for obtaining the name servers (type NS record) of the “mit.edu”. Now filter out all non-DNS packets by typing “dns” (without quotes) in the filter field. Also click the right arrow for DNS in the packet-header detail window. Now focus on the last two DNS messages and answer the following questions:

1.To what IP address is the DNS query message sent? Is this the same as your default local DNS server?
2.Examine the DNS query message? What “Type” of DNS query is it? Does the query message also contain any “answers”?
3.Examine the DNS response message. Provide details of the contents of the “Answers”, “Authority” and “Additional Information” fields. What can you infer from these?
4.Provide a screenshot.

Step 4: You can confirm the above records by using nslookup yourself. Type the following two commands and observe the output and compare it to your answers for Steps 2 and 3,

nslookup www.mit.edu

nslookup -type=NS mit.edu

5.To what IP address is the DNS query message sent? Is this the same as your default local DNS server?
6.Examine the DNS query message? What "Type" of DNS query is it? Does the query message also contain any "answers"?
7.Examine the DNS response message. *What MIT nameservers does the response message provide? Does this response message also provide the IP addresses of the MIT namesers?*
8. *Provide a screenshot.*

Experiment 3: DNS Server configuration

Steps to be done:

1.Check bind package is installed, if not use yum to install bind

2.To enable DNS service check /etc/host.conf file

3.Configure /etc/named.conf file to point to your DNS tables

4.Configure DNS database files

5.Restart DNS server.

6.Advanced configuration.

Need Knowledge on

- DNS
- named.conf details
- zone files
- DNS records

Sample Example

1. Check if bind package is installed using the following command
   - *dnf list installed | grep bind*
   - output:
   - *bind*
   - *bind-chroot*
   - *bind-libs*
   - *bind-utils*

   if above packages are not shown install using the following command
   - *dnf install bind bind-chroot bind-libs bind-utils*
2. Enable DNS query sequence check /etc/host.conf,
   - should contains *order hosts,bind*

3. Configure /etc/named.conf file to point to your DNS tables

Sample

```
options {
          // DNS tables are located in the /var/named directory
          directory "/var/named";

          // Forward any unresolved requests to our ISP's name server
          // (this is an example IP address only -- do not use!)
          forwarders {
                    123.12.40.17;
          };

          /*
           * If there is a firewall between you and nameservers you want
           * to talk to, you might need to uncomment the query-source
           * directive below.  Previous versions of BIND always asked
           * questions using port 53, but BIND 8.1 uses an unprivileged
           * port by default.
           */
          // query-source address * port 53;
};

// Enable caching and load root server info
zone "named.root" {
          type hint;
          file "";
};

// All our DNS information is stored in /var/named/mydomain_name.db
// (eg. if mydomain.name = foobar.com then use foobar_com.db)
zone "mydomain.name" {
          type master;
          file "mydomain_name.db";
          allow-transfer { 123.12.41.40; };
};

// Reverse lookups for 123.12.41.*, .42.*, .43.*, .44.* class C's
// (these are example Class C's only -- do not use!)
zone "12.123.IN-ADDR.ARPA" {
          type master;
          file "123_12.rev";
          allow-transfer { 123.12.41.40; };
};

// Reverse lookups for 126.27.18.*, .19.*, .20.* class C's
// (these are example Class C's only -- do not use!)
zone "27.126.IN-ADDR.ARPA" {
          type master;
          file "126_27.rev";
          allow-transfer { 123.12.41.40; };
};
```

4. Configure DNS database
    - Root zone
    - local zone
    - Master(and slave[optional]) Zone file
    - Reverse zone(master and slave[optional])
5. Root Zone

Sample(already there)

```
; <<>> DiG 9.5.0b2 <<>> +bufsize=1200 +norec NS . @a.root-servers.net
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34420
;; flags: qr aa; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 20

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;. IN NS

;; ANSWER SECTION:
. 518400 IN NS M.ROOT-SERVERS.NET.
. 518400 IN NS A.ROOT-SERVERS.NET.
. 518400 IN NS B.ROOT-SERVERS.NET.
. 518400 IN NS C.ROOT-SERVERS.NET.
. 518400 IN NS D.ROOT-SERVERS.NET.
. 518400 IN NS E.ROOT-SERVERS.NET.
. 518400 IN NS F.ROOT-SERVERS.NET.
. 518400 IN NS G.ROOT-SERVERS.NET.
. 518400 IN NS H.ROOT-SERVERS.NET.
. 518400 IN NS I.ROOT-SERVERS.NET.
. 518400 IN NS J.ROOT-SERVERS.NET.
. 518400 IN NS K.ROOT-SERVERS.NET.
. 518400 IN NS L.ROOT-SERVERS.NET.

;; ADDITIONAL SECTION:
A.ROOT-SERVERS.NET. 3600000 IN A 198.41.0.4
A.ROOT-SERVERS.NET. 3600000 IN AAAA 2001:503:ba3e::2:30
B.ROOT-SERVERS.NET. 3600000 IN A 192.228.79.201
C.ROOT-SERVERS.NET. 3600000 IN A 192.33.4.12
D.ROOT-SERVERS.NET. 3600000 IN A 128.8.10.90
E.ROOT-SERVERS.NET. 3600000 IN A 192.203.230.10
F.ROOT-SERVERS.NET. 3600000 IN A 192.5.5.241
F.ROOT-SERVERS.NET. 3600000 IN AAAA 2001:500:2f::f
G.ROOT-SERVERS.NET. 3600000 IN A 192.112.36.4
H.ROOT-SERVERS.NET. 3600000 IN A 128.63.2.53
H.ROOT-SERVERS.NET. 3600000 IN AAAA 2001:500:1::803f:235
I.ROOT-SERVERS.NET. 3600000 IN A 192.36.148.17
J.ROOT-SERVERS.NET. 3600000 IN A 192.58.128.30
J.ROOT-SERVERS.NET. 3600000 IN AAAA 2001:503:c27::2:30
K.ROOT-SERVERS.NET. 3600000 IN A 193.0.14.129
K.ROOT-SERVERS.NET. 3600000 IN AAAA 2001:7fd::1
L.ROOT-SERVERS.NET. 3600000 IN A 199.7.83.42
M.ROOT-SERVERS.NET. 3600000 IN A 202.12.27.33
M.ROOT-SERVERS.NET. 3600000 IN AAAA 2001:dc3::35

;; Query time: 147 msec
;; SERVER: 198.41.0.4#53(198.41.0.4)
;; WHEN: Mon Feb 18 13:29:18 2008
;; MSG SIZE rcvd: 615
```

6. Local Zone file

Sample(already there)

```
$TTL 86400
@ IN SOA localhost. root.localhost. (
1997022700 ; Serial
28800 ; Refresh
14400 ; Retry
3600000 ; Expire
86400 ) ; Minimum
IN NS localhost.
1 IN PTR localhost.
```

7. Reverse local zone

Sample(already there)

```
$TTL 86400
@ IN SOA @ root (
42 ; serial (d. adams)
3H ; refresh
15M ; retry
1W ; expiry
1D ) ; minimum

IN NS @
IN A 127.0.0.1
IN AAAA ::1
```

8. Zone file

Sample

```
; This is the Start of Authority (SOA) record.  Contains contact
; & other information about the name server.  The serial number
; must be changed whenever the file is updated (to inform secondary
; servers that zone information has changed).
     @ IN SOA mydomain.name.  postmaster.mydomain.name. (
          19990811        ; Serial number
          3600            ; 1 hour refresh
          300             ; 5 minutes retry
          172800          ; 2 days expiry
          43200 )         ; 12 hours minimum

; List the name servers in use.  Unresolved (entries in other zones)
; will go to our ISP's name server isp.domain.name.com
          IN NS           mydomain.name.
          IN NS           isp.domain.name.com.

; This is the mail-exchanger.  You can list more than one (if
; applicable), with the integer field indicating priority (lowest
; being a higher priority)
```

```
          IN MX           mail.mydomain.name.

; Provides optional information on the machine type & operating system
; used for the server
          IN HINFO        Pentium/350     LINUX

; A list of machine names & addresses
     spock.mydomain.name.    IN A    123.12.41.40   ; OpenVMS Alpha
     mail.mydomain.name.     IN A    123.12.41.41   ; Linux (main server)
     kirk.mydomain.name.     IN A    123.12.41.42   ; Windows NT (blech!)

; Including any in our other class C's
     twixel.mydomain.name.   IN A    126.27.18.161  ; Linux test machine
     foxone.mydomain.name.   IN A    126.27.18.162  ; Linux devel. kernel

; Alias (canonical) names
     gopher      IN CNAME        mail.mydomain.name.
     ftp         IN CNAME        mail.mydomain.name.
     www         IN CNAME        mail.mydomain.name.
```

9. Reverse Zone

Sample

```
; This is the Start of Authority record.  Same as in forward lookup table.
     @ IN SOA mydomain.name.  postmaster.mydomain.name. (
          19990811        ; Serial number
          3600            ; 1 hour refresh
          300             ; 5 minutes retry
          172800          ; 2 days expiry
          43200 )         ; 12 hours minimum

; Name servers listed as in forward lookup table
          IN NS           mail.mydomain.name.
          IN NS           isp.domain.name.com.

; A list of machine names & addresses, in reverse.  We are mapping
; more than one class C here, so we need to list the class B portion
; as well.
     40.41       IN PTR    spock.mydomain.name.
     41.41       IN PTR    mail.mydomain.name.
     42.41       IN PTR    kirk.mydomain.name.

; As you can see, we can map our other class C's as long as they are
; under the 123.12.* class B addresses
     24.42       IN PTR    tsingtao.mydomain.name.
     250.42      IN PTR    redstripe.mydomain.name.
     24.43       IN PTR    kirin.mydomain.name.
     66.44       IN PTR    sapporo.mydomain.name.

; No alias (canonical) names should be listed in the reverse lookup
; file (for obvious reasons).
```

What to do?

1. Make a plan for your DNS and zone
2. Your plan must contains at-least one mail server and one slave DNS server
3. Design and configure your zone files

4. Write on named.conf options and zone configuration and location
5. Write your understanding on DNS zone files such as root, local, primary, slave --zone and reverse zone. What are purposes of these zone files.
6. Write you understanding on DNS RR(resource record) such as A, AAAA, MX, CNAME, NS, SOA, PTR in details.
7. Tests using dig tool; You must include test questions and test results – for resource records listed in 6 and your full zone database
8. Include everything in your report and submit.

END OF LAB