# Networking Lab 8 DNS

The Domain Name System (DNS) is an essential part of the Internet that maps human readable domain names to IP addresses. The DNS client on your host connects to a DNS server, or resolver, that looks up domain names using the DNS tree structure. In this lab, we will use DNS tools to create DNS requests from our computers and examine the requests and responses using Wireshark.

## Reading

Read slides 5 - 12 in CyberAces Module 2 - Networking - Layer 7  
PDF: <https://assets.contentstack.io/v3/assets/blt36c2e63521272fdc/bltdca26f38ac8a4449/625459ec00a8bb4b7795fc1f/CyberAces_Module2-Networking-Layer7.pdf>

## DNS Tools

There are two primary DNS tools that are available on Windows and Linux systems. The older tool is nslookup. It is available on both Windows and Linux but is deprecated on Linux. The newer tool is dig, which is mainly available on Linux. Dig was created to replace nslookup, but dig is not installed by default in Windows. Nslookup and dig have many options, but we’ll just use the option to specify the type of record we want to see.

## DNS Records

We will look at a few of the DNS record types that are available (see slide 7 in the reading.)

* A, or address. Gives the IP address of a domain name
* CNAME, or Canonical Name (Alias). Sometimes the response is just an alias to another domain name
* NS, or Name Server. Gives the name and/or IP address of the DNS server that is authoritative for a domain
* TXT, or text. Used for miscellaneous information, often about spam protection
* PTR, or Pointer. Used for reverse lookup. It gives the domain name for an IP address (not always available.)

## NSLOOKUP

We will use nslookup in interactive mode. The commands, set type=ns or set type=a, tell nslookup whether we want name server (NS) or address (A) records. Also, note the “dot” at the end of the request for [www.vccs.edu](http://www.vccs.edu)., which tells nslookup that is the entire name we are looking for, or Fully Qualified Domain Name (FQDN). (If you don’t do that, your host may add its own domain name to the end if it doesn’t receive a response.) Here is an example which tells us the name and IP addresses for the vccs.edu domain. The servers 164.106.1.1 and 2.1 are the ones that other DNS servers go to when looking for addresses in the vccs.edu domain.

PS C:\Users> nslookup

Default Server: svgs-dc01.svgs.local

Address: 172.18.10.39

> set type=ns

> vccs.edu.

Server: svgs-dc01.svgs.local

Address: 172.18.10.39

Non-authoritative answer:

vccs.edu nameserver = ns2.vccs.edu

vccs.edu nameserver = ns1.vccs.edu

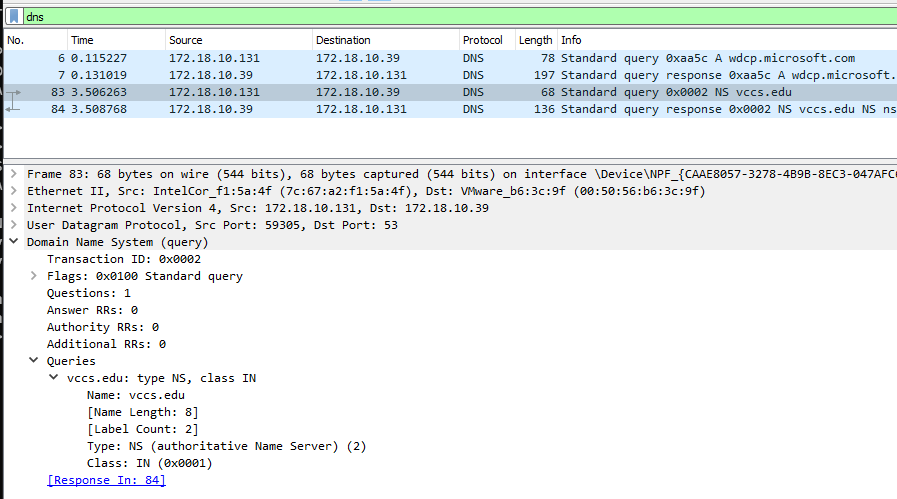
ns2.vccs.edu internet address = 164.106.2.1

ns1.vccs.edu internet address = 164.106.1.1

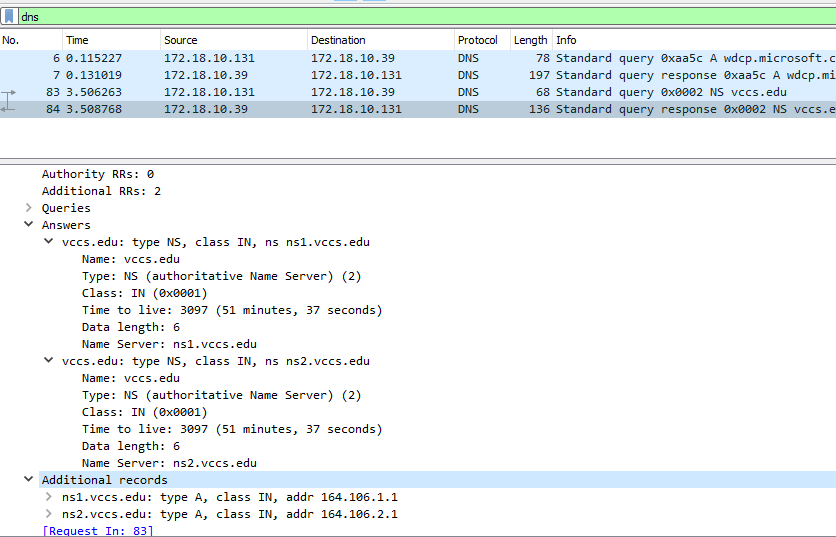
## NS record

Start a packet capture in Wireshark, and then use nslookup to issue a request for the name server records of a domain of your choosing. Don’t forget the line, set type=ns. Use a display filter dns in Wireshark to show only DNS traffic.

If you used vccs.edu, as above, Wireshark should look like this for the request generated by nslookup.

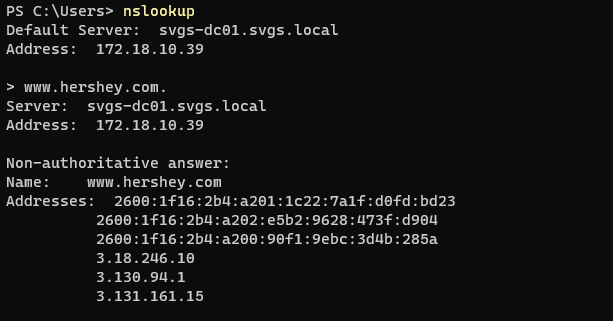


Here is the response from the DNS server. Note that the response for the NS record contained the names, but not the IP addresses of the servers. Our server kindly included additional A records that contain the IP addresses.

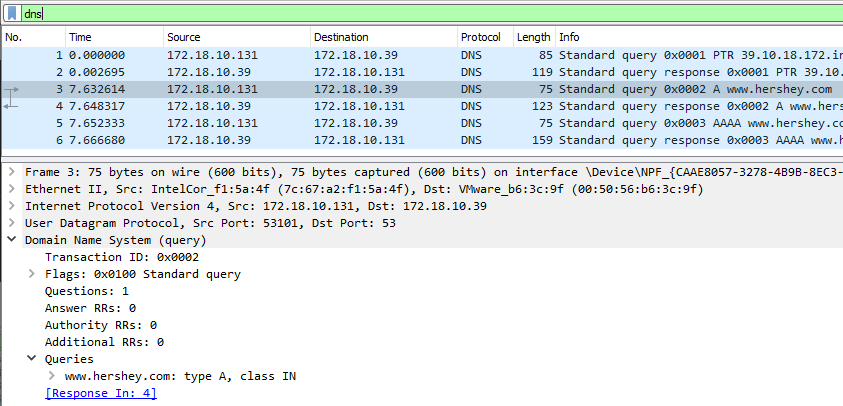


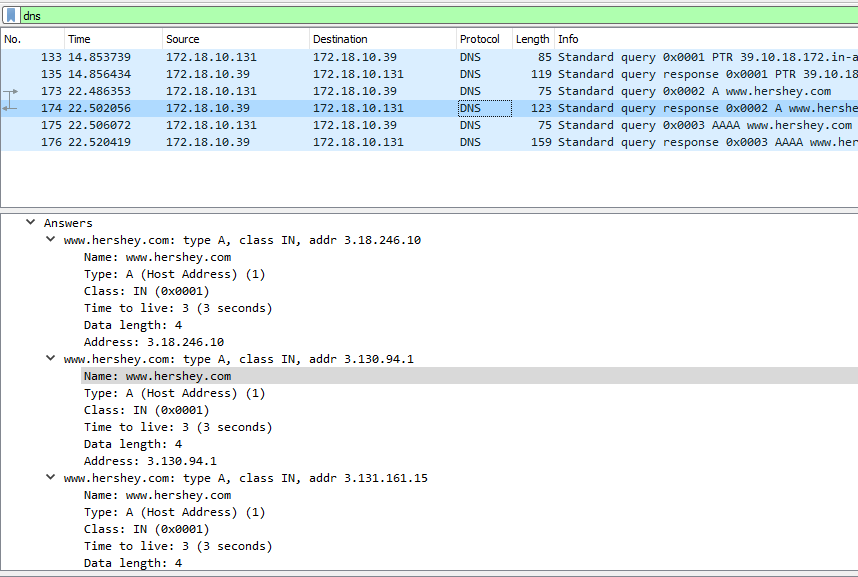
## A record

Often, your computer’s DNS requests are for the A records of the web servers you browse to. Since most places host their web servers with a cloud provider, the records can be complicated. Here is a simple one. Start a packet capture and then use nslookup to generate a request for the A record for [www.hershey.com](http://www.hershey.com). Note: nslookup asks for A records by default. Since we previously set nslookup to request NS records, you will need to either: 1) enter set type=A, or 2) stop and restart nslookup. Here, I have chosen to restart nslookup.



The workstation sent a request to the DNS server for the A record for www.hershey.com.



The response is simple and returns three A records with three IP addresses. This site is supported by three servers for redundancy and load balancing. Some DNS servers will rotate which answer is first, second, etc., as one form of load balancing.  


Also note that in the top pane, the last request and response is for an AAAA record, or IPv6 address. Windows asks for this even when it does not have an IPv6 connection to the Internet. It can talk to other computers on the local subnet with IPv6, however.

## Hand In (1)

Many sites that are hosted by cloud providers have a CNAME (alias) record that gives you the domain name of the site using the provider’s own domain. For example, a site may give a CNAME that points to a long address at AWS, Microsoft Azure, CloudFront, etc.

Use Wireshark to record nslookup requests for [www.vccs.edu](http://www.vccs.edu), [www.brcc.edu](http://www.brcc.edu), and [www.svgs.k12.va.us](http://www.svgs.k12.va.us). These use CNAME records to point to the provider domains. What are the cloud providers for those three sites?

## MX Record

When you send an email to [janestudent@svgs.k12.va.us](mailto:janestudent@svgs.k12.va.us) your mail server needs to find the address for the svgs.k12.va.us mail server, and it does it with DNS. The MX record gives the name and priority of the mail server. The priority is used when the site uses multiple mail servers for redundancy or load sharing; the backup server will have a higher priority number (I don’t see this in use much.) The MX record returns the name of the mail server, but not the IP address. (The IP addresses shown in this screenshot are NS name server records, not mail server.)  
A computer screen with white text

Description automatically generated

Strictly speaking, the DNS server only had to give us the name svgs-k12-va-us.mail.protection.outlook.com, but it helpfully included additional records with the IP addresses as well

Here are the packet captures of the DNS requests and replies.

A screenshot of a computer

Description automatically generated

The request and response for the MX record are in frames 124 and 126. Frames 122 and 123 are interesting, in that my laptop requested the MX record for svgs.k12.va.us.svgs.local. This happened because I did not put a ‘.’ (period) at the end of the request to tell the laptop this was a complete, or Fully Qualified Domain Name (FQDN).  


The laptop assumed (wrongly) that I might be wanting to talk to something on my local network and had left off the laptop’s domain. It appended the domain of the laptop, svgs.local, to the request. The server had no idea what svgs.k12.va.us.svgs.local is, and answered “No such name” in frame 123. The laptop reissued the request without the svgs.local in frame 1248 and the DNS server responded in frame 126.

In the MX response below, the DNS server only gave us the name of the mail server, but it kindly gave the IP addresses as additional records. If it had not done that, we would have needed to request the A record for svgs-k12-va-us.mail.protection.outlook.com. Also note that the name ends in outlook.com. This mail server is hosted by Microsoft in the cloud.

A screenshot of a computer program

Description automatically generated

## Hand In (2)

Pick an email address (besides @svgs.k12.va.us) and use nslookup and Wireshark to find the IP address of the mail server.

## TXT Record

The TXT record can hold anything the DNS administrator wants to put in it. Usually, I have seen it used for:

1. Spam prevention info
2. Info for cloud mail providers
3. Capture The Flag Contests (CTFs)

The mail and spam information is fairly advanced stuff, so I would normally have skipped TXT records. However, since TXT records are sometimes used in CTF questions you may see them for DNS questions.

## Hand In (3)

Is there a TXT record for the domain that holds your email address? (For example, the domain for [me@yahoo.com](mailto:me@yahoo.com) is yahoo.com.) If so, what is it?

Note: If you get no answers when you use set type=txt in nslookup, it may be that your Internet Service Provider is blocking it. If so, change your DNS server to Google, server 8.8.8.8, and it should work.

## PTR Record

Pointer records are “reverse lookup” records. For those, the request is an IP address, and the response is a domain name. Most IP addresses do not have PTR records. DNS admins usually create PTR records for mail servers and other major servers so that people can verify that the IP address belongs to the server.

## Hand In (4)

Do a reverse lookup for the IP address 8.8.8.8 as follows:

PS C:\Users> nslookup

Default Server: Comtrend.Home

Address: 172.16.0.1

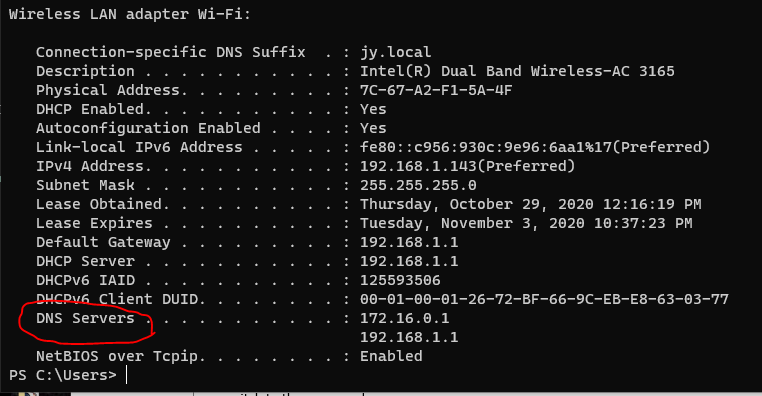
> set type=ptr

> 8.8.8.8

What does DNS say about the 8.8.8.8 address?

## Your DNS server

When you start nslookup it tells you the IP address of the DNS server it is using; when you look at DNS queries in Wireshark you can see the IP address of the DNS server (hopefully these are the same.) You can also see the address of the DNS server your computer is using from the command line.

You have used the command, ipconfig, to learn the IP address of your computer. You can also use it to learn the IP address of your DNS server if you add the /all switch to the command.  
  
<snip>  


## Hand In (5)

What is the IP address of the DNS server used by your computer? How did your computer learn that IP address? (Hint: Take a look at Networking Lab 4)

# DIG (optional)

Note: dig is only available in Linux, so use your Linux VM.

Here is an example of using dig to query the DNS server 164.106.1.1 for the name server record of [www.vccs.edu](http://www.vccs.edu). The command can be simpler, of course; if you just want to query your default DNS server for [www.vccs.edu](http://www.vccs.edu), you would just enter dig www.vccs.edu.

The format for dig is shown below. “@[IP address]” is the address of the DNS server you want to use for the query. “Type” is the record type you want the server to return. We’ll be using Name Server (NS) and Address (A) records. Slide 7 in the CyberAces module shows you the other possibilities. In the example for nslookup, we queried the server 164.106.1.1 to find the IP address of [www.vccs.edu](http://www.vccs.edu). Here’s what it looks like in dig.

[john@localhost ~]$ dig @164.106.1.1 www.vccs.edu NS

; <<>> DiG 9.8.2rc1-RedHat-9.8.2-0.37.rc1.el6\_7.4 <<>> @164.106.1.1 www.vccs.edu NS

; (1 server found)

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12370

;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:

;www.vccs.edu. IN NS

;; AUTHORITY SECTION:

vccs.edu. 3600 IN SOA ns1.vccs.edu. alyon.vccs.edu. 458660325 1200 180 1209600 3600

;; Query time: 13 msec

;; SERVER: 164.106.1.1#53(164.106.1.1)

;; WHEN: Mon Nov 9 15:20:12 2015

;; MSG SIZE rcvd: 76