

**Activity 2.1.3 Protocols and Bandwidth**

Introduction

|  |  |
| --- | --- |
| We live in a world where computer networks are ubiquitous. In the first decades of computing, computers were not linked together. Solid media like floppy disks were the best means of transferring data between machines.  Creating a system to connect many different computers required protocols and standards, many of which were developed in the 1960s at Xerox PARC. All computers on the Internet must agree to use these standards and protocols for the communication to function. Can you think of a time when you had to perform a task in a very specific way in order for the product to be usable? |  |

Materials

* Paper and a writing utensil
* Computers with Internet access
* Firefox web browser with FireSSH installed
* AWSS credentials: username, password, hostname

Procedure

1. Greet your partner to practice professional skills. Set team norms for pair work.
2. There is a human protocol for a business introduction. First impressions in business (as elsewhere) are lasting. People do infer personality and confidence from your introduction. A business introduction includes both a verbal introduction and a handshake and is done while standing. Read the protocol, and then practice by exchanging an introduction with each member of your group. The protocol is:
   1. The verbal introduction should give your full name and a title or context. Allie might say, "Hello, my name is Allie Trusting. I'm a sophomore at Washington High School."
   2. One person starts their verbal introduction and extends their hand a relaxed distance toward the other person. The handshake lasts 3-4 seconds with 2-3 shaking motions.
   3. The handshake should be as firm as you would use to turn a door handle. The webs between thumb and index finger should meet.
3. In this activity we dive into some of the details of how the Internet works. There is a high demand for employees in **computer system design**. These jobs are approximately equal in number to jobs in applications programming. These computer specialists create and maintain networks and the computer hardware on the networks. They work with the layers of abstraction at which data are exchanged among computers in a network. They might have one of the following job titles.

* Computer Engineers and Research Scientists create new technology
* Computer Support Specialists troubleshoot users' problems
* Computer Systems Analysts analyze an institution's technology and recommend improvements
* Computer Network Architects design an institution's network
* Computer Systems Administrators install and maintain networks

Reference the Bureau of Labor Statistics (BLS) *Occupational Outlook Handbook* at http://www.bls.gov/ooh/ . Select **Computer and Information Technology** from among the occupation groups.



1. The five groups listed earlier are among the BLS information. Which of these occupation groups has the highest median salary?
2. Which of the five occupation groups needs the fewest years of education?

**Part I: Low-level protocols and the TCP/IP handshake**

1. As we dive into the miniscule details of a protocol, keep in mind our high level objectives:

* To understand how data are exchanged over the Internet.
* To understand that protocols evolve over time, though a particular version of a protocol does not change.
* To understand that protocols are compromises among competing objectives like security and speed, based upon decisions that can be political.

Think of decisions that you or another person made that were compromises among competing objectives. Describe one of these decisions that stayed in effect for some time after the decision was made.

1. An important type of protocol for computing is called a **handshake**. A handshake initiates a connection between two computers. The **Transmission Control Protocol (TCP)** and the **Internet Protocol (IP)** are used for exchanging most packets through the Internet. They are used together and are called TCP/IP. Other protocols, like **Hypertext Transfer Protocol (HTTP)** for Web data, use TCP/IP to get packets to the other computer in an end-to-end connection and to verify that all the packets arrive without errors. Sending a message through TCP/IP begins with three transmissions in a **TCP handshake**. An example of the TCP handshake is shown below.

Client: “SYN 6”

Server: “ACK 7, SYN 58”

Client: "ACK 59"

SYN and ACK stand for synchronize and acknowledge, respectively. Both the client and the server ask to SYNchronize with the computer at the other end of the connection. They each include a random number at which they will start numbering the data they send. In this example, the client picked 6 and the server picked 58; they could have been any random numbers.

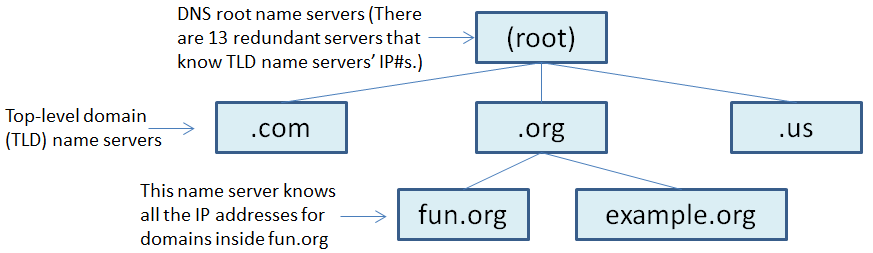
To establish a connection, both of the computers must each respond to a SYNchronize packet by sending an ACKnowledge packet, stating the sequence number of the data it is ready to receive. ACK 59 indicated the client was waiting for packet 59. We show this as just one plus the SYN 58 packet sequence number, but in the real TCP protocol, the synchronize and acknowledge numbers are counting bytes.

1. Create a playwright-style script for a TCP handshake, similar to the one above, just using different numbers. Write it down.
2. Using different numbers yet again, initiate a two-way connection with another student in your group aloud.
3. The TCP/IP protocol lets the two computers keep track of what data the other computer has received. What might be a disadvantage of agreeing that both ends will confirm receipt like this?

**Part II: Domain Name Service**

1. The **Domain name service (DNS)** protocol lets a user get the IP address of a computer at a particular domain name. The DNS protocol is a weak point in the Internet from a security perspective. DNSSEC is the protocol presumed as the replacement, but adoption has been slow. Today the **Internet Corporation for Assigned Names and Numbers (ICANN)**, a non-profit non-governmental organization, controls domain names and IP numbers.

DNS servers are organized in a tree of delegated authority. The root points to several **top-level domain** servers like .com and .org. Each name server is responsible for knowing the IP numbers assigned to names in its domain, but it can delegate that responsibility to servers lower in the tree. All these servers are **authoritative DNS name servers**. They are the final authority on which IP numbers go with which domain names.



1. There is a computer with domain name www.casos.cs.cmu.edu. There is a name server you could ask to get the IP address for this computer. What domain is this name server in charge of?
2. Name servers delegate autonomous authority to servers below them. For example, the .org name server doesn't keep track of IP addresses for all computers using domain names within .org. Why is this necessary?
3. The DNS uses a second set of computers called **recursive DNS name servers**. These are the ones your computer actually talks to. The recursive name servers respond to your requests by getting IP numbers from the authoritative name servers.

One example is diagrammed below. In this example, your computer wants to initiate a connection with the web server www.example.org, but it doesn't know the web server's IP address.

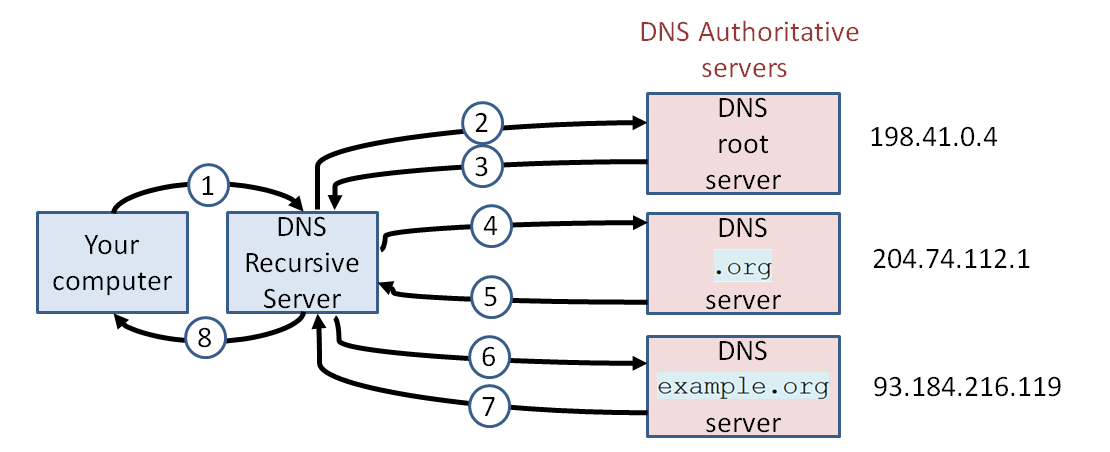
Your computer will use TCP/IP to ask your recursive name server for www.example.org's IP address. If your domain name server doesn't know, it will ask the root name server.

The root name server will respond with the IP address of the .org name server(s).

Then your recursive name server will ask the .org name server. The .org name server will respond with the IP address of the example.org name server.

When asked, the example.org name server will respond with the IP address for www.example.org.

Your recursive name server will finally respond to you and **cache** the answer in case the same question comes up again soon. The information is cleared from cache in 1-3 days when it expires. This flurry of messages makes the system scalable. The messages are vulnerable to imposters, but making it less vulnerable with DNSSEC requires bigger messages and longer processing time.



* 1. The six computers involved are listed below A –F. Using the diagram above, record. Record who says each of messages 1-8 below, and to whom.

A. Your computer:

B. Your name server:

C. root name server:

D. .org name server:

E. example.org name server:

F. www.example.org web server:

1. from\_\_to\_\_: What is the IP address of www.example.org ?

2. from\_\_to\_\_: What is the IP address of www.example.org ?

3. from\_\_to\_\_: Ask 204.74.112.1 .

4. from\_\_to\_\_: What is the IP address of www.example.org ?

5. from\_\_to\_\_: Ask 93.184.216.119 .

6. from\_\_to\_\_: What is the IP address of www.example.org ?

7. from\_\_to\_\_: The IP# of www.example.org is 93.184.216.119 .

8. from\_\_to\_\_: The IP# of www.example.org is 93.184.216.119 .

* 1. Record people's names to play parts A – F above. As directed by your teacher, act this out for the class, following the script as above.

**Part III: Observing the protocols from a server**

1. In a later activity, you will publish a web page on the server. Our purpose now is to explore a computer's connection to the Internet. You will use several commands on a server to explore protocols used by the Internet:

* ifconfig and ipconfig to check your own IP number
* dig and nslookup to get an IP number for a domain name
* ping to test latency
* traceroute and tracepath to see routing of IP packets

Open a **remote shell** on the cse-ssh.pltw.org server from which you can interact with the DNS as follows. A shell is a command line interface to an operating system. A remote shell is a command line interface to the operating system of one computer, provided to a user on another computer.

1. Launch Firefox.
2. Launch the FireSSH add-on from within Firefox, as shown below on the left or on the right, depending on whether the Firefox menu toolbar is visible. This will open a new tab, labeled by FireSSH's purple sea urchin. This will also open an Account Manager dialog box.

**Secure Shell (SSH)** was developed in 1995 at Helsinki University of Technology as a replacement for the Telnet protocol, which was deemed insecure.

|  |  |  |
| --- | --- | --- |
| Select **Firefox** > **Web** **Developer** > **FireSSH** | OR | Select **Tools** > **Web Developer** > **FireSSH** |

1. Set FireSSH not to remember your credentials and then provide your credentials as follows.
2. FireSSH might be configured to remember your credentials. Although it would be a violation of local schools' Acceptable User Policies and a gross violation of professional ethics, another user would be able to log in as you. Configure FireSSH not to remember your credentials. Do this by selecting **Tools** > **Options**... . In the Options dialog box that appears, make sure that **Remember passwords** is unchecked.

|  |  |
| --- | --- |
|  |  |

1. Create a profile in FireSSH as shown below. The profile includes log-in credentials for a user account on a web server. Your teacher will provide these. Select **Create an account** from the dropdown menu. Enter the host, login, and password. Select **Connect**.

|  |  |
| --- | --- |
|  | Host:  Login Username:  Password: (don't write down passwords on paper that could be seen) |

1. The remote server is running a Linux operating system. The command line prompt shows your username. In the brackets is your current directory; ~ is an abbreviation for your home directory.

example@pav-www-0102 [~]#

**Part IIIA: The NIC gets you onto the Internet**

1. A computer is connected to the Internet through a **network interface card (NIC)**. A NIC is set up using the program ifconfig (in a UNIX-like operating system) or ipconfig (in a Windows operating system). We haven't permitted you to reconfigure the NIC, but you can view information about the configuration and status of the NIC. A **System Administrator** might run this program as part of their job duties, which encompass a wide variety of responsibilities, including managing user accounts, configuring hardware, automating processes with scripts, backing up data, and maintaining a secure network.

From within FireSSH, use /sbin/ifconfig to run the ifconfig program, located in the  /sbin  directory, as shown below. Your input is shown in red.

example@pav-www-0102 [~]# /sbin/ifconfig

eth0 Link encap:Ethernet HWaddr 0E:57:A7:79:2C:1D

inet addr:10.0.132.231 Bcast:10.0.132.255.0 Mask:255.255.255.0

inet6 addr: fe80::c57:a7ff:fe79:2c1d/64 Scope:Link

The output includes much more text than shown here! The output has two **records**, each one showing IP numbers assigned to a NIC. Each record has several **fields**, including the inet addr field which indicates the IPv4 address. The NIC will only tell the computer about IP packets if they are addressed to an IP number matching the inet addr field of one of these records. What is the IPv4 address assigned to the eth0 NIC?

**Part IIIB: DNS turns domain names into IP addresses**

1. On Windows and UNIX-like machine, the nslookup command will ask your recursive name server to find out what IP address is assigned to a domain name. Look up pltw.org as shown below.

example@pav-www-0102 [~]# nslookup pltw.org

Server: 10.0.134.17

Non-authoritative answer:

Name: pltw.org

Address: 190.93.241.249

Note that the results indicate a non-authoritative answer. This means that the information was from the cache of the recursive name server 10.0.134.17 rather than freshly obtained from the name server for .org. Do you get the same response if you look up the IP addresses of subdomains of pltw.org?

1. Use nslookup to find the IP address for the web server of your favorite web page. Simply type  nslookup  followed by the URL of your favorite website. Record your information here.

example@pav-www-0102 [~]# nslookup www.coolsite.com

URL of favorite website: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

IP address of favorite web server:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. To see the work of the authoritative DNS servers, use the dig program with the +trace option as shown below.

example@pav-www-0102 [~]# dig www.example.org +trace

www.example.org. 4697 IN A 93.184.216.119

There is much more output from the dig program (domain information groper) than shown here. First, the recursive DNS server reports the domain name of one or more root servers. The recursive server asks one of the root servers for the IP address of www.example.org. That root DNS server responds with the domain names (and IP addresses, though not shown in the output) for the .org name servers. The NS records refer to other name servers, while the A record shown above finally gives the address. How many separate machines are serving DNS for the top-level .org domain?

**Part IIIC: Latency and bandwidth**

1. The ping command is another commonly used tool to see behind the curtain in a computer network, on both Windows and UNIX-like systems. The command uses a simple protocol: one host sends a packet asking for an echo, and the target host responds. The ping command repeats the process for many iterations. It reports how long the round trips took and whether any packets were dropped. Use Ctrl-c to **kill** the  ping  **process**. A process is a program or distinct thread of a program. Killing a process tells the processor to stop executing it. Execute the command with your own choice of domain name in place of www.example.org. The last line of output is shown below.

example@pav-www-0102 [~]# ping www.example.org

rtt min/avg/max/mdev = 17.385/17.439/17.585/0.108 ms

This output shows that the packets traveled round trip from pltwcs.org to www.example.org in 17.439 ms, on average. A signal can travel one million feet of copper wire in 1 ms! What is the fastest response you can find?

1. Ping tells you about the **latency** between two computers’ devices. Latency is the delay in a connection. Latency is different than **bandwidth**, the rate at which data can be pushed into or out of a device or through a connection. Bandwidth is limited by the narrowest "pipe" along the way: the processor, the network card, cables, switching hubs, etc. Bandwidth and latency can be observed through a browser. In a new tab, navigate to http://www.speedtest.net . Once the page loads, select a target location on the detailed map to start the test. Sample results are shown below.



* 1. Record the target location you selected, along with your results. Include units.

Target location: \_\_\_columbia\_\_ Ping: \_\_\_\_\_\_\_\_

Download speed: \_\_1.86Mbps\_\_\_\_\_ Upload speed: \_\_\_\_\_14.73Mbps\_\_\_

1. The example download speed above is 5.28 million bits per second. Shown below is work calculating how long it would take to retrieve a 3 Gb file.

3 Gb = 3 billion bytes = 24 billion bits

|  |  |  |  |
| --- | --- | --- | --- |
| 24 billion bits | • | 1 second | = 4545 seconds |
|  |  | 5.28 million bits |  |

Given the download speed you measured, calculate how long it would take to download a 450 kb file Show your work following the example. 24193 sec

**Part IIID: IP packets travel multiple paths**

1. Every packet sent through the Internet is passed from one computer to the next until it arrives at its destination. The tracepath utility asks computers along the way to report back to the sender, so that the sender can see the route packets are taking. Track the route of packets from the pltw.org server back to the computer in front of you as follows.
   1. Use whatsmyip.org in a new browser tab to identify the IP address of your computer. (This might be your school's router's IP address; ipconfig would find your computer’s IP address, but it might be only internal to the school network. We'll ignore these complications.) Record the IP address.
   2. To find the paths from the pltw.org server to the computer at which you are sitting, execute the tracepath command on the server, followed by your computer’s IP address. Packets contain a maximum number of hops they can be sent, known as the time-to-live (TTL). When an Internet host forwards a packet, it **decrements** the TTL. If the TTL reaches 0, the packet is dropped and an error message is sent back to the sender's IP. The tracepath program sends IP packets that will only survive 1 hop, or 2 hops, or 3 hops, and so on, so it can receive error messages from the machines along the way to the destination, revealing their IP addresses. Some machines won't send back an error message if they get a packet that has run out of hops, resulting in "no reply."

Record the IP addresses reported by tracepath at each hop. Use your IP address instead of 195.2.6.32 as shown below. An example of hop #5 is shown below. Ctrl-c will stop the tracepath program early if desired.

example@pav-www-0102 [~]# tracepath 195.2.6.32

5: PR01.LAX03.google.com (206.223.123.21) 17.827ms

* 1. In a new browser tab, use the service at iplocation.net to determine the geographical location of the IP addresses through which the packets hopped. Sketch the path on the map below, labeling locations along the way. Note that location data may be missing or conflicting. Map the locations that seem to agree with other data. 1 Seattle, Washington.

2. Seattle, Washington.



**Part IV: Infrastructure and Organizations**

1. **Ethernet** has been a critical technology in the development of computer networks. Originally developed by Xerox PARC in the 1970s, Ethernet is a protocol for sending bits over cables. A sequence of Ethernet bits is called a frame, and an IP packet can be split across several frames. The cables are usually copper wires twisted in pairs, with bits represented as voltage. The twisted pairs of wires in one cable can carry up to 100 megabits per second. A newer technology has glass fibers that use light to represent the bits. Known as Gigabit Ethernet, the Ethernet standard for glass fiber can carry 1000 megabits per second. Bringing glass fiber optics directly to the home is a new service, with vendors competing to connect entire cities in 2014.

How much faster is Gigabit Ethernet than the bandwidth you observed in Step 16?

* 1. Create one group with the entire classroom and repeat the procedure above. What changes? Has collisions in large groups.
  2. What, if any, do the limitations of this protocol seem to be? Has collisions in large groups.

1. TCP, IP, and Ethernet were developed by researchers at Xerox PARC with U.S. military funding. Now the **Internet Engineering Task Force (IETF)** is responsible for maintaining the TCP/IP suite of protocols. In 2013 the IETF began a discussion that may lead to a more secure Internet. Some protocols are being considered to help protect the online activities of citizens from being monitored. This initiative was sparked when Eric Snowden leaked classified documentation about National Security Administration (NSA) efforts to monitor U.S. citizens. The Department of Defense (DoD), the NSA, Xerox PARC, and IETF are examples of entities that have influenced the development of computing and the Internet.
   1. Use the Internet to find a web page or article about international issues regarding the Internet and governmental programs for monitoring civilian communication. <http://www.washingtonpost.com/posteverything/wp/2014/07/14/this-is-why-the-government-should-never-control-the-internet/>
   2. Skim the article to determine what it is about.
   3. Summarize what you learned from the article. The article is about ISP’s and how the government FCC ends and its affect on certain internet services.
2. For each of the following listed entities, use the Internet to determine their responsibilities to the Internet:
3. IETF – Internet Engineering Task Force – the goal of the IETF is to make the internet work faster and at a higher quality.
4. W3C – World Wide Web Consortium-
5. ICANN
6. IETF
7. W3C
8. ICANN

**Conclusion**

1. Web pages are transmitted in packets using the HTTP protocol. DNS protocols are used by your machine and its domain name server to obtain the IP address of the web server when you request a web page. TCP is a protocol for breaking the information into packets and for reassembling packets into segments. Match each protocol with its purpose. Match the protocols on the left with the purposes on the right.

|  |  |
| --- | --- |
| * IP * TCP * DNS * HTTP | 1. How to request Web content 2. How to address and route packets 3. How to sequence packets and verify accuracy 4. How to get the IP address for retailer.com |

1. Why are protocols necessary?
2. Why do protocols get outdated?
3. How do governmental agencies, corporations, and non-governmental organizations affect the development and functioning of the Internet?