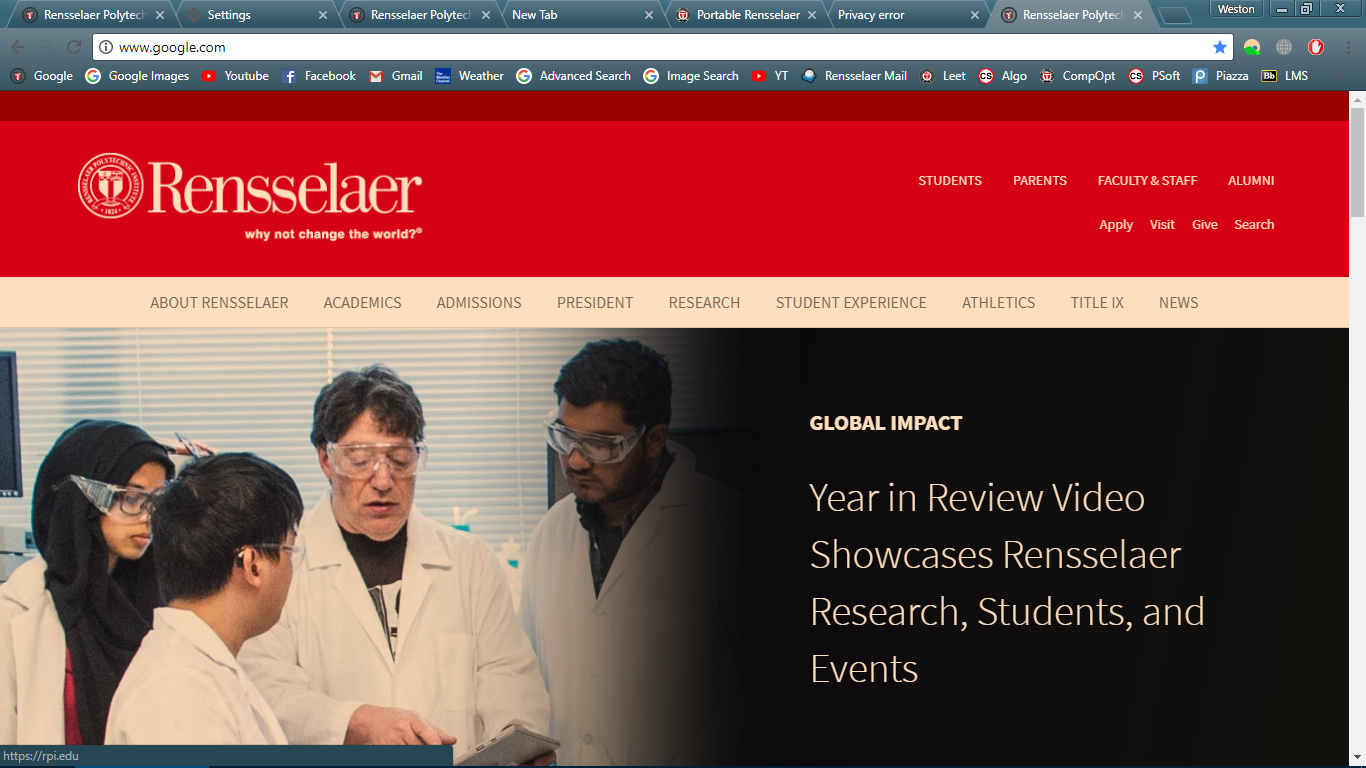
Weston Jones 1/25/18

Intro to ITWS – Section 1 Lab 1: Network & Protocols

1. **Basic Network Information**
   1. **Ipconfig /all for your LAN Ethernet Adapter and your general IP configuration:**
      1. **What is your hostname?** WesLaptop2
      2. **What is your MAC address?** 34-64-A9-CF-4C-34
      3. **Who is the vendor of your Ethernet adapter?** Realtek PCIe GBE Family Controller
      4. **What is your IPv4 address?** 192.168.116.1
      5. **What is your IPv6 address?** 2620:0:2820:a5:d71:2844:6c96:e4f1
      6. **What are the IP (v4) addresses of your DNS servers?** 128.113.28.67 and 128.113.26.77
      7. **What is the IP (v4) address of your default gateway** 128.113.54.126
   2. **Netstat -f**
      1. **What kind of transport layer protocols are in use?** I only see TCP protocols active on my machine.
      2. **What kind of application layer protocols are in use?** The command line doesn’t specify, but I’d assume HTTPS because it runs over TCP and seems to be the most common application layer protocol.
   3. **Netstat -b**
      1. **What applications are using your network ports, and what application layer protocol are they using? (only name up to three)** Chrome.exe, Dropbox.exe, and winword.exe – all are using https.
   4. **nslookup 192.0.32.10: nslookup it allows you to look up IPs or hostnames against a name server (by default, the first DNS server configured in your network settings).**
      1. **Upon running “nslookup 192.0.32.10”, what is the host name of the name server you are using** dns9.net.rpi.edu
      2. **What name is associated with 192.0.32.10? What is the significance of this name?** 192.032.10 is associated with ccnso.icann.org – It's the website for ccNSO, the country code names organization, a subdivision within ICANN responsible for managing issues concerning country level domain names. The name itself exists because letters are easier to remember than numbers, and so it makes more sense to associate names with numerical addresses.
      3. **Perform a “netstat –n” in your console (this tells netstat not to resolve names against the name server – notice that it’s quicker). Do an nslookup on a foreign IP from your netstat results. What name did you uncover? (If the DNS server can’t resolve the IP you picked, try another one.)** I tried every single IP that came up and confirmed with Professor Plotka that I was running the command correctly, but my computer kept spitting back “No existent domain” errors.
      4. **Do an “nslookup 127.0.0.1”. What name is returned? What is the significance of this name?** The name localhost is returned. This is because 127.0.0.1 is an address that always refers to the local machine – termed “localhost” in this instance.
      5. **What is the IP address of lms.rpi.edu?** 128.113.0.1
      6. **What is the IP address of www.rpi.edu? Is it IPv4 or IPv6?** 2620:0:2820:4::2128.113.0.2 is the address of ww.rpi.edu. It’s an IPv6 address.
   5. **tracert** [**www.ucla.edu**](http://www.ucla.edu)
      1. **How many hops did it take for you to reach** [www.ucla.edu](http://www.ucla.edu)**?**  17 hops.
      2. **From the tracert you just ran, locate an IPv4 address of a router along the path. Visit** [**http://www.iplocation.net/**](http://www.iplocation.net/) **and enter the IPv4 address. Were you able to determine the router’s location? If so, where is it?** All the addresses along the path are IPv6, but most of them seem to be located in New York, Ann Arbor Michigan, or Los Angeles.
      3. **Enter your own IPv4 address in the website. What did you learn?** The website either says I’m located in Troy or Hartford, Connecticut (Because RPI also has a campus there, I suppose).
2. **SSH to RCS-Linux**
   1. **netstat -e**
      1. **What transport layer protocols do you see in use?** The terminal shows DGRAM and STREAM as connection types which, from my own research, use UDP and TCP respectively.
      2. **What application layer protocols do you see in use?** SSH itself is an application layer protocol, but some of the connections running on the linux server are using HTTPS
   2. **host www.rpi.edu**
      1. **What IPv4 addresses are used by www.rpi.edu?** I just see 128.113.0.2
   3. **traceroute www.google.com**
      1. **How many hops did it take to get to Google?** Just 7 hops.
   4. **In your Windows Command Prompt console use netstat (as in section 1.b) to find your current SSH connection. Copy the netstat line showing the SSH connection to here.** TCP [2620:0:2820:1:c59f:3595:9856:6816]:52188 rcs-linux7.rpi.edu:ssh ESTABLISHED
3. **Overriding DNS**
   1. **Use jEdit, notepad or some other text editor to edit your hosts file: C:\Windows\System32\drivers\etc\hosts (/etc/hosts on linux/unix). Using the examples in the comments at the top of the hosts file as a guide, give www.rpi.edu a new host name – you can use any IP address of www.rpi.edu for this exercise (you need only one IP).**
   2. **Visit the new hostname in your browser and capture a screenshot of the browser window. (CTRL-SHIFT-ALT PrntScrn)**
   3. **Paste the screenshot here.**



1. **Take a look at some packets**
   1. **Run WireShark and select “Capture Options, start a capture with detailed options”.**
      1. **Set the Capture Filter to “tcp port 80” (or “tcp port http”)**
      2. **Set “Stop Capture after 4 packets”**
      3. **Click “start”**
   2. **In your browser, visit** [**http://m.rpi.edu/i/**](http://m.rpi.edu/i/)
   3. **Looking back at the WireShark output, look for the TCP three-way handshake.** 
      1. **How are the SYN and ACK flags represented in the TCP header? (Feel free to investigate** [**http://en.wikipedia.org/wiki/Transmission\_Control\_Protocol**](http://en.wikipedia.org/wiki/Transmission_Control_Protocol) **as well.)** The SYN flag is represented as 002 in hexadecimal and the ACK flag is represented as 010 in hexadecimal.
   4. **Look at the HTTP Request header. Copy your us+er-agent here (right-click to copy)**. User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/63.0.3239.132 Safari/537.36\r\n