COMP3310/6331 – Tute/Lab #2

At this stage students will have (also) covered:

- *LAN principles, Ethernet and Wifi*
- *At this stage it is still too early to do coding exercises .*

Activity	
1	**Introductions**: • Still looking for a 3310 class-rep – nominations are being sought, see first lecture
2	**Discussion of lectures**: • To-date: Any comments on pace, content, style? • Planning ahead: Any particular real-world network systems, technologies, challenges, etc. that you'd like to have a guest lecture on (Weeks 10-12)?
3	**Questions for reflection:** 1. How is statistical multiplexing different to TDM/FDM/SDM/etc? a. *It's just "having a go" and seeing if you can get data through, within a fixed and communally-shared bandwidth. TDM/FDM/SDM are like a circuit, they guarantee an allocation of bandwidth, but they waste bandwidth when nobody is using their allocation.* 2. Why do we use frame-flags to show the start/stop of a frame, rather than just the framelength? a. *Way too easy to get out of sync, even just a single-bit error. Everything thereafter is garbled. Having regular start/stop indicators gives everyone a chance to recover, within a frame-length.* 3. What's the difference between Collision Avoidance and Collision Detection? a. *CA is a collective behaviour, where everyone waits a random time before sending, to avoid everyone starting at the same time (very likely collision). CD is active individual behaviour, where you listen for collisions and stop, then back off.* 4. How does RTS/CTS solve the Hidden Terminal Problem? Why does it need to include the length of the message (N bytes) as part of its information? a. *Best shown over a diagram, from slide 24 (and earlier) of T3a. The CTS is seen by everyone in range of the target access point (the one that has received an RTS from somebody, and broadcasts the CTS). The CTS advises uninvolved devices from engaging in a conversation with it, so there are no collisions, and to wait until N bytes have passed before they try to send their message.* 5. What's 1000Base-T Ethernet? (how fast, what medium) a. *1000Mb/s (1Gb/s) over Twisted-Pairs (copper)* 6. Why does Ethernet send a 'heartbeat' signal ("Normal Link Pulse") when not sending data? a. *To show it is there, newly connected or just quiet. This lets the device at the other end turn on its Link light.* 7. Why do 802.3/802.11 frames start with a preamble? a. *To wake up the receiver(s) and enable clock-sync to occur* 8. Why do WiFi frames start with **fixed-rate** preambles and headers?

	a. *The signal rate varies with RF quality (noise, distance, etc.) Need to start at a known fixed rate *for each frame* and see if both ends can agree or slow down.* 9. What are the (critical) benefits of running the Spanning Tree protocol/algorithm across your Ethernet network? a. *Remove loops, to avoid broadcast storms, duplicate frames and mis-ordered frames.* 10. What are the key benefits of using VLANs across your Ethernet network? a. *Segregation of traffic for security or performance.* 11. Why is the channel spacing in the 2.4GHz band of WiFi such a problem? Why isn't it the same on the 5GHz band? a. *Channels overlap on 2.4G, so only three at best don't interfere with each other. 5GHz has much more room, so has clearly separated channels.* 12. What's the point of a WiFi **Beacon** frame? (Who sends it, and why?) a. *Access Points send it, advertising their SSIDs, channels, rates, etc. Allows clients to discover who is there just by listening.*
4	**Some hands-on activities:** Let's take a closer look at an active network, the Ethernet in the lab or your home network. If you have a laptop with wifi you possibly can also try the following analysis for wifi frames – please share the views with students who don't have laptops. • The lab machines have a program called wireshark installed, fire it up. For your laptop, go to www.wireshark.org and install the appropriate package for your system. On linux you'll have to use one of "sudo/gksudo/gksu wireshark" for necessary root permissions. You'll be using wireshark fairly often, so good to get familiar with it now. • Wireshark will listen to and record ('capture') traffic on the network from the time you hit 'start' until you 'stop'. You need to specify the interface in the capture/options. Run a capture for 20 seconds or so to get a good number. If there's not much traffic, open a web-browser, start the capture and then connect to some website. ○ *NB: WiFi seems to be a bit fiddly on some operating systems, so you may need to dig into the documentation a little further https://wiki.wireshark.org/CaptureSetup/WLAN* • The top panel displays all the traffic going by. Are you looking at frames or packets? Which frames/packets do you expect to see, and not see? ○ *You see both (Ethernet/WiFi) frames and (IP) packets. Packets are encapsulated inside the frames.* ○ *On a cable from a switch you will only see frames destined for you. You only see other people's frames on a shared medium.* • If you click on any single entry you can unpack it in the middle panel, i.e. "un-encapsulating" it (use the arrows on the left). The lowest panel shows the raw bytes. You should be able to work out the Ethernet frame structures if you look at slide 9 from T3b (Ethernet/Wifi). Note that wireshark does not show or count pre-ambles, the start-byte, or the checksum at the end.

- The source and destination addresses are (mostly) IP addresses, which the lectures will cover this week. You can drill down to just the Ethernet (and/or WiFi) layer for this tute.
- Identify the source and destination MAC addresses for various entries.
  - See if you can identify where the addresses carry flag-bits (unicast/multicast and vendor/global).
  - Can you find the address bytes in the raw data as well?
  - Which vendors does wireshark identify on your network? Which one is in your machine, which one is in the Ethernet switch (or wireless Access Point) you connect to?
  - Do you see any broadcasts in your capture? Where are they from?
- Note the length of each entry (you can sort on length). What is the longest you see, and why does it have that value? What is the shortest, and why that value? Reference the above slide.
  - *Longest I usually see is 1514, could be up to 1522: 1500 maximum Ethernet payload plus overhead bytes (6+6 MAC addresses, 4-byte tag (or a 4-byte longer max payload), 2-byte "type/length", and the 4-byte checksum, which wireshark doesn't seem to show/count.)*
  - *Shortest I usually see is 42, only for ARPs. That's the smallest payload, without any overhead, so for those frames the standard is **not** followed (no padding). Shortest should be 64, or it's a "runt" frame.*
- The Protocol column references the payload of the frames. How many different protocol types do you see there? We'll cover TCP and UDP very soon, what other protocols are on the network?

- Instead of setting a time limit you can also set a 'filter', and just collect frames or packets of a particular type. This can be set before or after a capture. Type 'ICMP' into the filter bar at the top of wireshark and start a capture. Open a terminal window and run 'ping 150.203.1.1'. That sends a handful of packets onto the network towards a particular host. Can you see each outgoing and returning packet? How many are sent/received?

- Looking ahead to this week's lectures: The operating system keeps track of Ethernet/MAC addresses in a cache, just like a switch. To see it in the terminal window (or command window on Windows) run 'arp -a'. How many hosts do you see there? The 'static' physical addresses have special meaning, you should see two broadcast addresses amongst others. The 'dynamic' addresses are learnt. Try running 'ping' to an Internet address that is listed there, but change the fourth number to something (1-254) that isn't already listed there. If it replies, do you see its MAC address being added to the cache?

- If you're not familiar with the unix command line, look at the 'man' (manual) pages for 'arp' and 'ping' - type "man arp" and "man ping" at the command prompt. For Windows users try "ping /?" and "arp /?" in the command window. These explain what these commands do and what options they have. You'll be using a lot of similar command line tools in coming weeks, so good to get familiar with both the tools and how to get more details about any of them.