Networking
Autumn 2018

i.g.batten@bham.ac.uk

Check Panopto!

- Is it running?
- Is it running?
- Seriously, is it running?

What this course is

- Some networking **theory**
- Some networking **practice**
- Some networking **business and reality**

What you will learn

- The main components of networking 2018-style, hopefully with an eye to 2028 (or at least 2019).
- Some background to explain why networking today looks like it does, much of which is rather contingent, and how people make money.
- Some theory and science to justify the technology.

The key question

- Both GCHQ and Google lament that when they ask applicants the question “you click on a link, what happens?”, people can’t really give much of an answer beyond the GUI.
- This course should get you from the click to the wire.

Who are you?

- I’m expecting people from
  - 3rd/4th years of our BSc/MSci/MEng programmes and parallel joint honours programmes
  - Advanced Computer Science and Cyber Security MSc programmes
  - Is there anyone else? ICY? Conversion MSc?

Programming Skill

- ~~I'd really like you to be able to program, well, in C (ie, Hayo's course, and perhaps be taking Operating Systems in parallel with this)~~
- This year, I am going to make the programming content optional as an extension for the good of your soul, rather than part of the assessment. It isn't a formal learning outcome.
- I would recommend you do some programming, though.

Extended Course

- There will be an additional section in the second exercise towards the end of the course for “extended” students, which will look at deeper issues.

Some Basic Assumptions

- Spoiler: Apollo 13 did get back to earth
- Spoiler: The Titanic sinks before the end of the film
- Spoiler: Ethernet lower layers and TCP/IP upper layers **have won**. There are no serious contenders on the 10–20 year horizon. Everything else is history or weird curiosity.
- So this course is unashamedly Ethernet and TCP/IP focused, as there are no other games in town.

Week 1:

- This Introduction
- Packet v Circuit Switching, layer models (DoD 4/5, ISO 7), what's in the subnet, transport, application layer.
- Network Hardware: Switches, Routers, data/control/management plane. Software defined networks.

Week 2

- Background: LAN/WAN split, Arpanet, X.25, PSS, DEC/XNS/SNA, DoD, OSI. Why OSI Failed.
- Lower Layers: FDDI, Slotted/Token Rings, **Ethernet** (in its various forms). Touch on Transmission (SDH, WDM). ATM in passing.

Week 3

- IP: addressing, routing, concepts. **Why IPv6 is needed.**
- IP: address allocation, bootp, DHCP, SLAAC.

Inversion

在过去的几年里，我在这一点上教授了TCP的复杂细节。

- In the past few years, I taught the knotty details of TCP at this point.
- Strong knowledge of TCP is important and valuable: there’s a real shortage, 35 years after the RFCs, 40 years after Cerf’s paper, of people with good understanding of its dynamics
- But I think it needs applications to motivate it, so I’m going to switch to applications at this point and come back down the stack.

Week 4

- Application: HTTP, SMTP, IMAP, POP3, SNMP, NTP
  - For all the application lectures, I will deal with deployment and server issues as well as the protocol, so you can go out and do stuff with them.
- For the good of everyone's futures, I'm going to spend some time on security issues within these protocols.

Week 5

- TCP 1: history, basic concepts. Windows, acks.
  - Previously I had a deadline here to meet Network Security, which I was teaching in the same semester. We're more relaxed this time.
- TCP 2: Detailed operation
  - This might take more than one lecture

Week 6

- TCP 3: Options (scaling, PAWS, etc, timestamps)
- TCP 4: Nagle, silly-window.
- Implementation: how it looks in the kernel.
  - I’m going to assume some familiarity with Unix/Linux kernels, but I’ll try to make it accessible if you don’t have that

Week 7

- UDP, RTSP, other transports
- NAT and its evils. IPv6 as cure. IoT. NAT security / logging / problems.

Week 8

- Application: Voice (I will try to get Chris Gallon in), 21CN, issues and politics.
- If I can't get Chris, Jim Reid on DNS security, issues and politics
    - Marshall Rose referred to politics as "Layer 8" of the 7 later model.
- Tutorial, Catchup, Exercise feedback and discussion

Week 9

- Routing inside the enterprise: Interior (RIP, OSPF, IPv6 analogues). VLANs.
- Routing outside the enterprise: Exterior (BGP, tech and politics). PPPoA/E, VLAN stacking

Week 10 and 11

- Summary and spare, more stuff.

Assessment

- Quiz released 22 Oct, due 27 Oct, just to check progress
- Exercise set 29 Oct, due 9 Oct
- Exercise set 27 Nov, due 7 Dec
- Each worth 10% of total marks, so don’t sweat them too much.

Office Hours

- Tuesdays 11–12, Fridays 10-11 in Room 132
- I.G.Batten@bham.ac.uk
  - Or Canvas discussions
- https://igb.batten.eu.org/
- Canvas/Panopto will (I hope) contain full recordings

Books

- “Distributed Systems: Concepts and Designs” by Coulouris, Dollimore, Kindberg and Blair is also used for Distributed Systems and covers a lot (was used in the past when the two courses were combined)
- TCP/IP Illustrated, Volume 1 by W. Richard Stevens is the essential book on TCP/IP
- TCP/IP Illustrated, Volume 2 is **not** necessary (it documents kernel implementations) but is a fascinating read (really)
- I will also expect you to read RFCs as we go along.

Things I've left out

- There is hopefully some spare time in which we can fit in some of the following.
  - Wireless networking (802.11[abgn])
  - Network management (SNMP in more detail)
  - Network design issues

The basic problem

- To get us going, let's look at the most basic problem of modern networking and see if we can think of some solutions. I'll return to this as something like a running example.

A problem

- An author has written a novel, and wants to send a copy to their US publisher 3000 miles away.
- They want to be certain the publisher receives a complete, unaltered copy.
- But the only communications mechanism they have is airmail letters, as we have been transported back to 1985.

PARK ROW BOOKS

This is not wildly unrealistic

- As we will see, all modern computer networking functions by dividing data into small (relative to the volumes we are moving) chunks and sending them as separate messages. An airmail letter is ~1000 words, ~4Kbytes, which is the right order of magnitude.
- There is a finite limit on how fast data can cross a network, set by (as an absolute floor on delay) the speed of light - 186 000 miles per second means that a 6000 mile round trip is a minimum of 32ms.

A basic approach

- Here is one slow, basic approach.
    1. The author sends a page of the novel
    2. The publisher copies it and sends back the copy.
    3. If the author hears nothing for twice the usual delay, they a letter was lost and re-send the most recent page.
    4. If the author receives back a copy which is faulty, they re-send.
    5. If the author receives back a copy which matches, they send the next page, or if they have sent the last page, they stop.
- There are some details to tidy up about page numbers, marking the last page, what “copy” means, etc, but the general idea stands.
- There are significant security holes in this in the face of a capable attacker, but let’s leave that for Network Security next semester.
- For our Semester 1 purposes we assume the post office to be unreliable, but not hostile.

This is ridiculous as an analogue example

- 384 pages by (at a minimum) four day turnaround = ~ five years.
- In ~~my lifetime~~ the dim and distant past, people played Chess, Diplomacy and other board games by post, taking years.
- There is also a genre called “epistolary novels” told in the form of exchanges of letters. However…

It’s bad electronically too

- Assume:
  - each page is 4KByte,
  - and we can send a page and receive back the reply (let’s say “acknowledgement”) in 32ms (ie, no delays other than the speed of light).
- It therefore takes 12.28s to send 1.5MB, which is 125KB per second, or 1Mbps.
  - (384 messages * 0.032 seconds) = 12.288 seconds
  - (384*4000 B)/12.288s = **125000 bytes per second**

So for Friday…

- Stick to the analogue example: sending a novel using airmail envelopes
  - How might you reduce the time taken to send the novel?
  - What fresh problems might your solution cause?
  - What conditions might you relax to get more performance?
- Assume that you **cannot** make the envelopes bigger or otherwise improve the performance of the post office.
  - This is again realistic: the maximum packet size of a network is not a fundamental limit like the speed of light, but the ~1500 byte limit of Ethernet circa 1979 is still very much with us.