

Contents

1	Introduction (0:00–3:00)	2
2	The Internet (3:00–100:00)	2
2.1	DNS (3:00–5:00, 12:00–20:00)	2
2.2	TCP and UDP (5:00–12:00, 20:00–24:00)	2
2.3	Ports (24:00–38:00)	3
2.4	E-mail (38:00–80:00)	4
2.5	HTTP (80:00–100:00)	6

1 Introduction (0:00–3:00)

- Exam 1 is next week during normal lecture hours. You'll find resources to help you prepare for the exam, which will be comprehensive, on the course website. After lecture today, there will also be a review section.
- Assignments are graded on a \checkmark -, \checkmark , \checkmark + basis whereas exams are graded on a percentile basis. We'll soon make a tool available via web or e-mail that will allow you to verify that you've submitted your assignments on time.

2 The Internet (3:00–100:00)

2.1 DNS (3:00–5:00, 12:00–20:00)

- Recall from last week that when you type in a URL into your browser's address bar, one or more requests are made to discover the IP address that corresponds to that URL. Once that address has been found, the request is directed along a series of routers until it reaches the server identified by that IP address. The server then locates the content that is being requested and sends it back to your computer where it is interpreted and rendered by your browser.
- As we mentioned last week, some ISPs, including Comcast, have hijacked this DNS lookup process for their own profit. In the event that you mistype a URL, you will not be redirected to a blank error page, but rather to a page full of advertisements sponsored by Comcast. You can, however, opt out of this "service" by clicking through a few links. And if Comcast can hijack your malformed URL requests, then so might any adversary who is positioned so as to intercept your packets!

2.2 TCP and UDP (5:00–12:00, 20:00–24:00)

- Since the inception of the internet, protocols have been in place which help ensure that requests and responses reach their destinations even if connection problems arise. TCP is one such protocol.
- Messages are not transmitted in whole across the internet, but rather in multiple parts called packets. If we think of a message as a letter, we can imagine that we will tear up this letter into multiple pieces and place each piece in a separate envelope with the same address. What should each of these packets contain? First, a return address so that the response can be properly directed. Second, the order of the packet. When we send these packets out, we have no guarantee that they'll pass along the same routes. As a result, we have no guarantee that they'll arrive at their destination in the order that they were transmitted. Thus, we need to pass along the order of the packets so that the recipient can properly assemble the original message.

- Let's imagine each student in this classroom is an entity on the internet. If we want to transmit a message to Mark in the back row, we can distribute our packets to multiple different students on the front row and trust them to send those packets along to Mark, albeit via different routes. Unfortunately, we can't always trust that these students or routers will *succeed* in passing our packets along. Occasionally, they might drop or lose our packets. In this event, Mark, our destination, will notify us that he didn't receive one of the packets and we will recreate and resend it. Finally, Mark receives all the packets and reconstructs the original message, which reads "Woohoo: message received!"
- There exist internet applications, however, that have no interest in this retransmission aspect of TCP. Video streaming applications, for example, simply forge ahead if packets are lost. If a user is receiving a video stream and misses some of its packets, he may experience a small skip or some fuzziness in playback. If he is forced to wait for these lost packets to be retransmitted, however, playback will pause entirely until those packets have been received. This alternative protocol which does not support retransmission is called UDP.

2.3 Ports (24:00–38:00)

- In order to support multiple different applications, servers have individually numbered ports assigned to each. HTTP web requests are generally handled on port 80, e-mail on port 25, and HTTPS web requests (via SSL) on port 443. We can, in fact, specify a port number in our browser's address bar. <http://www.cnn.com:80/> will take us to CNN's homepage as usual.
- If you'd like to run a web server off your home computer, you'll probably have to choose a port other than 80 since most ISPs will block incoming traffic on that port. Generally it's against their Terms of Service (TOS) to run a web server through them, but if you do it on a port other than 80 and there's not much traffic to it, you'll reduce your chances of getting caught.
- Question: there's not really an advantage to typing :80 at the end of a URL, because the web browser by default assumes that you want to make a request on port 80.
- Question: how does China block Facebook? Because they have a router positioned between all users and the rest of the internet, they can simply drop all packets that are destined for Facebook's IP addresses. To get around this, as we mentioned, you could make use of a VPN, although China could, in turn, block the ports that VPN applications use.
- By default, your home router allows all outgoing traffic but blocks all incoming traffic that wasn't initiated internally. If you want to be able

to access your home computer from outside your home network, you need to allow certain incoming traffic. For example, Remote Desktop Protocol (RDP) on PCs and screen-sharing on Macs, which allow you to control a computer remotely, require external connections to your home network. RDP operates on port 3389, so if you'd like to be able to control your home PC remotely,¹ you need to configure your router to allow incoming traffic on port 3389. Furthermore, you need to instruct your router to forward traffic on this port to the IP address of the home PC you wish to control. This configuration is called *port forwarding*. The IP address of your home PC will also need to be static so that this port forwarding won't break if your home PC is suddenly assigned a new IP address on the network.

2.4 E-mail (38:00–80:00)

- Whereas the internet is actually a physical entity encompassing thousands of interconnected servers and routers, the world wide web (WWW), e-mail, and other internet services are virtual entities which exist on top of the internet's framework.
- E-mail addresses, which are case-insensitive, are generally of the form `username@domain.tld`, where TLD stands for top-level domain. Other e-mail addresses might look like so: `username@subdomain.domain.tld`. Subdomains help system administrators organize servers and departments within larger institutions (e.g. universities) and also allow for a greater number of unique e-mail addresses. Unfortunately, some websites don't recognize e-mail addresses that contain subdomains. Bank of America, for example, rejected David's e-mail address, `malan@post.harvard.edu`, as invalid when he tried to fill out one of their forms. A very annoying programming mistake on their part!
- What does it mean to BCC someone? BCC stands for *blind carbon copy*. If you BCC someone on an e-mail, he will receive a copy of the e-mail, but the other recipients of the e-mail will be unaware that he has. This is sometimes a privacy measure to prevent recipients from knowing who else is in on the conversation and sometimes a courtesy measure to prevent recipients' replies from being sent to the people who were BCC'ed. If, however, someone who has been BCC'ed happens to reply to all, then the original recipients will suddenly know that others were in on the original message. Beware!
- What are some of the indicators of spam? First, if you don't know the sender. Second, if there are attachments or clickable links that seem suspicious. Clickable links that have been shortened are particularly dangerous because you may not even recognize the domain that they lead to. Be very wary of clicking on any links or downloading attachments you don't

¹RDP is only supported by certain versions of Windows.

recognize! Third, typos and improper grammar might indicate that it wasn't actually a human who composed the message. Fourth, any request for money or login credentials is almost certainly fraudulent!

- Another spammer trick is to create subdomains on an illegitimate domain which mimic a legitimate domain. For example, you might be tempted to click on a link that begins with `fas.harvard.edu` simply because you failed to read the rest of the URL. Simply using HTML, spammers can also make a link appear to lead to a legitimate domain when in fact it leads to an illegitimate one. Some browsers and e-mail clients allow you to hover over clickable links to see where they actually lead before you click them. If you do click these false links, you may be led to a website which exactly mimics a legitimate site like Bank of America. This is very easy to do since the HTML source code which makes Bank of America's homepage appear as it does can be viewed in every major browser.
- Much like web requests, e-mail messages are bounced from router to router as they travel from origin to destination. In the case of e-mails, the IP addresses of these routers are recorded in the e-mail's headers, which you can view in your mail client with a little bit of poking around. In Gmail, if you click the down arrow in the top right of the message and select Show Original, you can work your way from bottom to top of the lines that begin with "Received: by" to see the routers the message traveled through. If you use a stand-alone mail client, the first of these headers will generally reveal the IP address of your home network. If you're really paranoid, this might be a cause for concern, but honestly it's not too risky. Still, using a web mail client like Gmail will remedy this situation as the first header will reveal the IP address of the Google server you're connected to, not that of your home network.
- Interestingly, this ability to view the IP address of the e-mail's origin (at least in the case that the sender was using a stand-alone mail client) can be used against those who lie about their location. If you send an e-mail from a stand-alone client and state that you're in the Bahamas when you're actually on Harvard's campus, we'll know that you're not missing class because you're out of town, but rather because you're lazy!
- How do spammers get your e-mail address? Chances are you didn't sign up for their mailing list. Instead, they probably guessed your username just by random chance. E-mail usernames are generally short in length, so spammers might send out e-mails to every combination of 5, 6, 7, 8 characters at Gmail. Even if only a tiny fraction of these e-mails are actually delivered, the spammers have succeeded. Spammers also crawl the web searching for e-mail addresses, so if yours is posted on a public-facing web site, it might be harvested this way. If you receive a piece of spam with an unsubscribe link at the bottom, you probably shouldn't click it. Although this seems counterintuitive, consider that if you click it,

what you're really telling the spammer is that a real, live person was the recipient of that message he thought never got delivered. As a reward, he may just send you more spam. Of course, in the case that the message you received is the result of your buying something online or subscribing to a newsletter, the unsubscribe link may actually be genuine.

- Tragically, approximately 97% of all e-mail messages are spam. If you're looking for a better spam filter, Gmail's is one of the best. They benefit both intelligent programmers and a large body of users in which to observe trends.
- Question: why not charge for e-mails being sent? Technologically, it would be near impossible to plug all the holes. Any number of e-mail servers would arise that circumvented this policy. Not to mention that it would certainly hurt many legitimate e-mail users as well.
- Frighteningly, misunderstanding of the internet extends even to those at the highest levels who make decisions to regulate it. Check out [this video](#) which documents the comments of Senator Ted Stevens as well as John Hodgman's humorous discussion of net neutrality.

2.5 HTTP (80:00–100:00)

- So far, we've discussed in detail the process of DNS lookup as well as the transmission of packets via TCP/IP. Till now, however, we've waved our hands at the protocol responsible for requesting content from a web server on port 80. This protocol is *Hypertext Transfer Protocol* (HTTP).
- To begin examining HTTP, we'll use a Firefox add-on named Live HTTP Headers. When we navigate to `cnn.com` with this add-on open, we can see the actual information that is exchanged between our browser and CNN's servers. This information is in the form of headers much like the headers we described earlier in the context of e-mail. The first of these headers reads `GET / HTTP/1.1`. The first slash tells the web server that we want the content at the very root of the web directory. The second header reads `Host: cnn.com`. We won't discuss this thoroughly here, but suffice it to say that a single web server can host multiple domains through *name-based virtual hosting* because the content of this `Host` header tells the web server which domain is being requested.
- The next header begins with `User-Agent` and tells the web server the user's browser, operating system and version, language and country. Because this information is freely transmitted with every web request, sites are able to generate very accurate reports on the popularity of browsers and operating systems.
- Skipping a few lines down we see the `Cookie` header. When you first access a website, that website may request that a small piece of information

be saved on your computer. Thereafter, whenever you access the same website (at least until you clear your cookies), this piece of information will be sent to the web server in order to identify you.

- The second block of text in Live HTTP Headers are the headers that correspond to the web response. The first header, which reads `HTTP/1.1 301 Moved Permanently` tells us that CNN doesn't actually live at `cnn.com`. Rather, it lives at `http://www.cnn.com`, as indicated by the `Location` header a few lines down. When this header is received, your browser automatically redirects you to the correct location. In years past, almost all websites began with `www`. These days, it's a matter of branding. A website will choose to be known either with the `www` or not. There are even some websites that still fail to redirect those users who access the site without a `www` in front. Rookie programming mistake!
- After we've been redirected and a second request has been made for the correct domain name, we get a second response that actually contains the web content we were asking for. Realize that all web responses come with a three-digit code, e.g. 301 or 200, indicating the status of the request. 301 means the website has moved whereas 200 means everything's okay. Error codes include 403, 404, 500, and others.
- After the last of the response headers, the actual content of the website is sent in the form of HTML. HTML is actually just plaintext, but plaintext which can be parsed and interpreted by a web browser to create a lively presentation of content.
- Incidentally, know that `/` is a slash and `\` is a backslash. So if you hear anyone say "h-t-t-p colon backslash backslash," they're just wrong.
- Internet protocols can be thought of as different layers. At the bottom, ethernet represents the physical connection to the internet. On top of that is IP, which identifies computers on the internet. TCP sits on top of IP and allows for reliable transmission of information, as we've already seen. Above TCP is HTTP, which is responsible for the content you actually care about.