Putting it all together and the web

CS 356

Lecturer: Venkat Arun

What happens when we connect to the internet?

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	EAP	23	Request, Identity
2	0.042703	c2:a7:11:45:1b:7e	Cisco_f5:37:2f	EAPOL	18	Logoff
3	0.042708	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	EAP	29	Response, Identity
4	0.042709	10.154.247.152	3.20.1.85	TCP	54	50307 → 443 [ACK] Seq=1 Ack=1 Win=2048 Len=0
5	0.042711	10.154.247.152	174.142.116.47	TCP	54	56941 → 443 [ACK] Seq=1 Ack=1 Win=2048 Len=0
6	0.042713	10.154.247.152	128.83.185.40	DNS	69	Standard query 0x5ddf A apple.com
7	0.042714	10.154.247.152	128.83.185.40	DNS	92	Standard query 0x0c4c HTTPS mobile.events.data.microsoft.com
8	0.042716	10.154.247.152	128.83.185.40	DNS	92	Standard query 0x0680 A mobile.events.data.microsoft.com
9	0.042717	10.154.247.152	128.83.185.40	DNS	90	Standard query 0x9234 A www.msftncsi.com.edgesuite.net
10	0.042732	::	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
11	0.042734	::	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
12	0.051120	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	EAP	40	Request, MD5-Challenge EAP (EAP-MD5-CHALLENGE)
13	0.053375	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	EAP	26	Response, Legacy Nak (Response Only)
14	0.059151	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	EAP	24	Request, Protected EAP (EAP-PEAP)
15	0.061479	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	TLSv1…	179	Client Hello
16	0.079044	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	EAP	1022	Request, Protected EAP (EAP-PEAP)
17	0.079592	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	EAP	24	Response, Protected EAP (EAP-PEAP)
18	0.085223	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	EAP	1018	Request, Protected EAP (EAP-PEAP)
19	0.085547	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	EAP	24	Response, Protected EAP (EAP-PEAP)
20	0.090146	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	TLSv1…	822	Server Hello, Certificate, Server Key Exchange, Server Hello Done
21	0.206459	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	TLSv1…	154	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
22	0.212158	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	TLSv1…	75	Change Cipher Spec, Encrypted Handshake Message
23	0.213066	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	EAP	24	Response, Protected EAP (EAP-PEAP)
24	0.218058	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	TLSv1…	58	Application Data
25	0.218241	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	TLSv1…	60	Application Data
26	0.223605	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	TLSv1…	91	Application Data
27	0.223982	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	TLSv1…	114	Application Data
28	0.230115	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	TLSv1…	100	Application Data
29	0.230559	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	TLSv1…	55	Application Data
30	0.236776	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	TLSv1…	64	Application Data
31	0.237647	c2:a7:11:45:1b:7e	Cisco_9e:cb:6f	TLSv1…	64	Application Data
32	0.243278	Cisco_9e:cb:6f	c2:a7:11:45:1b:7e	EAP	22	Success
33	0.581602	::	ff02::1:ffb3:efce	ICMPv6	86	Neighbor Solicitation for fe80::1001:ae5e:d5b3:efce
34	0.581609	fe80::1001:ae5e:d5…	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
35	0.605694	c2:a7:11:45:1b:7e	Cisco_76:b5:02	ARP	42	Who has 10.154.0.1? Tell 10.154.247.152
36	0.606158	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0xb8b19c5f
37	0.610345	fe80::1001:ae5e:d5…	ff02::2	ICMPv6	62	Router Solicitation
38	0.630464	c2:a7:11:45:1b:7e	Cisco_76:b5:02	ARP	42	Who has 10.154.0.1? Tell 10.154.247.152
39	0.645761	fe80::1001:ae5e:d5…	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
40	0.647857	fe80::1001:ae5e:d5…	ff02::fb	MDNS	195	Standard query response 0x0000 PTR, cache flush CMPS-A56229.local NSEC, cache flush E.C.F.E.3.B.5.D.E.…E.A.1.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.E.F.ip6.arpa
41	0.675491	c2:a7:11:45:1b:7e	Cisco_76:b5:02	ARP	42	Who has 10.154.0.1? Tell 10.154.247.152
42	0.760295	c2:a7:11:45:1b:7e	Cisco_76:b5:02	ARP	42	Who has 10.154.0.1? Tell 10.154.247.152
43	0.927507	c2:a7:11:45:1b:7e	Cisco_76:b5:02	ARP	42	Who has 10.154.0.1? Tell 10.154.247.152
44	0.953371	fe80::1001:ae5e:d5…	ff02::fb	MDNS	361	Standard query 0x0000 PTR lb._dns-sd._udp.local, "QU" question PTR _airport._tcp.local, "QU" question PTR _rdlink._tcp.local, "QU" question PTR _uscan._tcp.…
45	1.147076	fe80::1001:ae5e:d5…	ff02::fb	MDNS	288	Standard query 0x0000 ANY CMPS-A56229._airplay._tcp.local, "QU" question ANY 5AC2722B6690@CMPS-A56229._raop._tcp.local, "QU" question ANY CMPS-A56229._compa…
46	1.251872	c2:a7:11:45:1b:7e	Cisco_76:b5:02	ARP	42	Who has 10.154.0.1? Tell 10.154.247.152
47	1.397592	fe80::1001:ae5e:…	ff02::fb	MDNS	288	Standard query 0x0000 ANY CMPS-A56229._airplay._tcp.local, "QM" question ANY 5AC2722B6690@CMPS-A56229._raop._tcp.local, "QM" question ANY CMPS-A56229._compa…

Apps

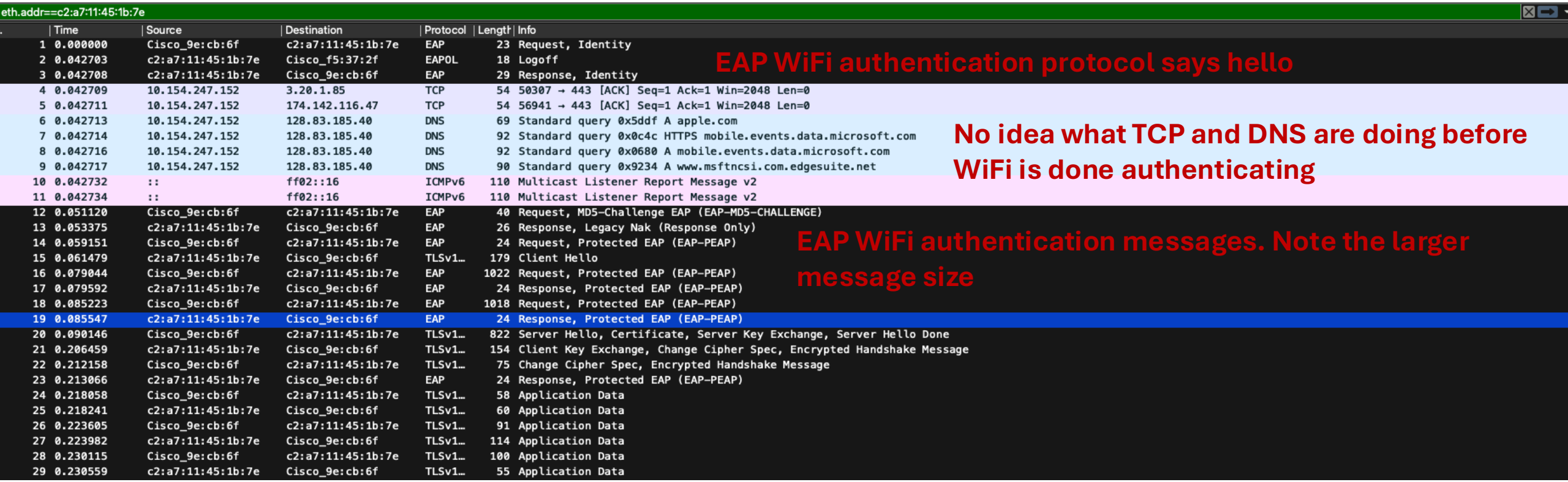
Logging into WiFi

Other machines making DNS requests. We have not yet received an IP address, so the NIC listens to all packets

My machine eventually gets an IP address of 10.155.11.97, so this is not intended for me

Note: most of this lecture will be a live demo. I have included a few details for your convenience. To review material, watch the lecture video

I filter to only include messages to my ethernet address



After connection, DNS goes whirr….

I added a filter to only include packets with my IP address

ip.addr==10.155.11.97sc

No.	Time	Source	Destination	Protocol	Length	Info
80	6.455730	10.155.0.1	10.155.11.97	DHCP	342	DHCP Offer - Transaction ID 0xb8b19c60
82	7.474542	10.155.0.1	10.155.11.97	DHCP	342	DHCP ACK - Transaction ID 0xb8b19c60
93	9.157328	10.155.11.97	224.0.0.251	MDNS	228	Standard query response 0x0000 PTR, cache flush CMPS-A56229.local PTR, cache flush CMPS-A56229.local NSEC, cache flush E.C.F.E.3.B.5.D.E.5.E.A.1.0.0.1.0.0.0…
98	9.385733	10.155.11.97	224.0.0.251	MDNS	367	Standard query 0x0000 PTR lb._dns-sd._udp.local, "QU" question PTR _airport._tcp.local, "QU" question PTR _rdlink._tcp.local, "QU" question PTR _uscan._tcp.…
102	9.481653	10.155.11.97	128.83.185.40	DNS	99	Standard query 0xd478 PTR lb._dns-sd._udp.0.0.155.10.in-addr.arpa
103	9.481685	10.155.11.97	128.83.185.40	DNS	93	Standard query 0xc24f PTR lb._dns-sd._udp.public.utexas.edu
104	9.481692	10.155.11.97	128.83.185.40	DNS	78	Standard query 0x3649 SVCB _dns.resolver.arpa
105	9.487295	128.83.185.40	10.155.11.97	DNS	198	Standard query response 0xd478 No such name PTR lb._dns-sd._udp.0.0.155.10.in-addr.arpa SOA chisos.ots.utexas.edu
106	9.487298	128.83.185.40	10.155.11.97	DNS	158	Standard query response 0x3649 No such name SVCB _dns.resolver.arpa SOA a.root-servers.net
107	9.487301	128.83.185.40	10.155.11.97	DNS	178	Standard query response 0xc24f No such name PTR lb._dns-sd._udp.public.utexas.edu SOA chisos.ots.utexas.edu
108	9.544675	10.155.11.97	128.83.185.40	DNS	69	Standard query 0x127c A apple.com
109	9.554029	10.155.11.97	128.83.185.40	DNS	73	Standard query 0xa297 A www.apple.com
110	9.559662	10.155.11.97	128.83.185.40	DNS	83	Standard query 0x05c2 A www.msftconnecttest.com
111	9.565001	10.155.11.97	128.83.185.40	DNS	77	Standard query 0xd330 A www.microsoft.com
112	9.581270	10.155.11.97	128.83.185.40	DNS	75	Standard query 0x175b A www.spotify.com
113	9.583237	10.155.11.97	128.83.185.40	DNS	75	Standard query 0x7ec8 A www.outlook.com
114	9.585459	10.155.11.97	128.83.185.40	DNS	84	Standard query 0x1bf3 A detectportal.firefox.com
115	9.586078	10.155.11.97	128.83.185.40	DNS	82	Standard query 0x0018 A api.apple-cloudkit.com
116	9.586739	10.155.11.97	128.83.185.40	DNS	74	Standard query 0x88df A www.google.com
117	9.588463	10.155.11.97	128.83.185.40	DNS	71	Standard query 0x0588 A example.org
118	9.595624	10.155.11.97	31.13.70.50	TCP	78	57026 → 443 [SYN, ECE, CWR] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1688987110 TSecr=0 SACK_PERM
119	9.595839	10.155.11.97	31.13.71.50	TCP	78	57027 → 80 [SYN, ECE, CWR] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=3552385995 TSecr=0 SACK_PERM
120	9.603632	10.155.11.97	128.83.185.40	DNS	84	Standard query 0x2759 A skydrive.wns.windows.com
121	9.603937	10.155.11.97	128.83.185.40	DNS	73	Standard query 0x079f A ipv4only.arpa
122	9.604242	10.155.11.97	128.83.185.40	DNS	84	Standard query 0x3c7c AAAA skydrive.wns.windows.com
123	9.605275	10.155.11.97	128.83.185.40	DNS	92	Standard query 0xe98c A 1-courier.sandbox.push.apple.com
124	9.605615	10.155.11.97	128.83.185.40	DNS	84	Standard query 0x197f A 1-courier.push.apple.com
125	9.608811	10.155.11.97	128.83.185.40	DNS	88	Standard query 0xd070 A profile.accounts.firefox.com
126	9.620699	10.155.11.97	128.83.185.40	DNS	85	Standard query 0xa697 A push.services.mozilla.com
127	9.621002	10.155.11.97	128.83.185.40	DNS	103	Standard query 0xb25c A sync-1-us-west1-g.sync.services.mozilla.com
128	9.624176	10.155.11.97	128.83.185.40	DNS	81	Standard query 0x6757 HTTPS outlook.office365.com
129	9.624487	10.155.11.97	128.83.185.40	DNS	81	Standard query 0x63f9 A outlook.office365.com
130	9.637269	10.155.11.97	128.83.185.40	DNS	69	Standard query 0xfbbf A slack.com
131	9.637952	10.155.11.97	128.83.185.40	DNS	69	Standard query 0x9a43 HTTPS slack.com
132	9.638038	10.155.11.97	128.83.185.40	DNS	69	Standard query 0x2a5e A sentry.io
133	9.646444	31.13.70.50	10.155.11.97	TCP	74	443 → 57026 [SYN, ACK, ECE] Seq=0 Ack=1 Win=65535 Len=0 MSS=1250 SACK_PERM TSval=800502752 TSecr=1688987110 WS=256
134	9.646586	10.155.11.97	31.13.70.50	TCP	66	57026 → 443 [ACK] Seq=1 Ack=1 Win=131264 Len=0 TSval=1688987161 TSecr=800502752
135	9.648152	31.13.71.50	10.155.11.97	TCP	74	80 → 57027 [SYN, ACK, ECE] Seq=0 Ack=1 Win=65535 Len=0 MSS=1250 SACK_PERM TSval=1798905966 TSecr=3552385995 WS=256
136	9.648265	10.155.11.97	31.13.71.50	TCP	66	57027 → 80 [ACK] Seq=1 Ack=1 Win=131264 Len=0 TSval=3552386048 TSecr=1798905966
137	9.654420	128.83.185.40	10.155.11.97	DNS	198	Standard query response 0x1bf3 A detectportal.firefox.com CNAME detectportal.prod.mozaws.net CNAME prod.detectportal.prod.cloudops.mozgcp.net A 34.107.221.82
138	9.654421	128.83.185.40	10.155.11.97	DNS	85	Standard query response 0x127c A apple.com A 17.253.144.10
139	9.654423	128.83.185.40	10.155.11.97	DNS	272	Standard query response 0x7ec8 A www.outlook.com CNAME outlook.office365.com CNAME ooc-g2.tm-4.office.com CNAME outlook.ms-acdc.office.com CNAME SAT-efz.ms-…
140	9.654423	128.83.185.40	10.155.11.97	DNS	337	Standard query response 0xe98c A 1-courier.sandbox.push.apple.com CNAME 1.courier-sandbox-push-apple.com.akadns.net CNAME us-sandbox-courier-4.push-apple.co…
141	9.654425	128.83.185.40	10.155.11.97	DNS	176	Standard query response 0x3c7c AAAA skydrive.wns.windows.com CNAME client.wns.windows.com CNAME wns.notify.trafficmanager.net AAAA 2603:1030:40c:e::
142	9.654426	128.83.185.40	10.155.11.97	DNS	209	Standard query response 0xa297 A www.apple.com CNAME www-apple-com.v.aaplimg.com CNAME www.apple.com.edgekey.net CNAME e6858.dsce9.akamaiedge.net A 23.201.1…
143	9.654427	128.83.185.40	10.155.11.97	DNS	233	Standard query response 0x05c2 A www.msftconnecttest.com CNAME ncsi-geo.trafficmanager.net CNAME www.msftncsi.com.edgesuite.net CNAME a1961.g2.akamai.net A …
144	9.654427	128.83.185.40	10.155.11.97	DNS	135	Standard query response 0x175b A www.spotify.com CNAME edge-web.dual-gslb.spotify.com A 35.186.224.24

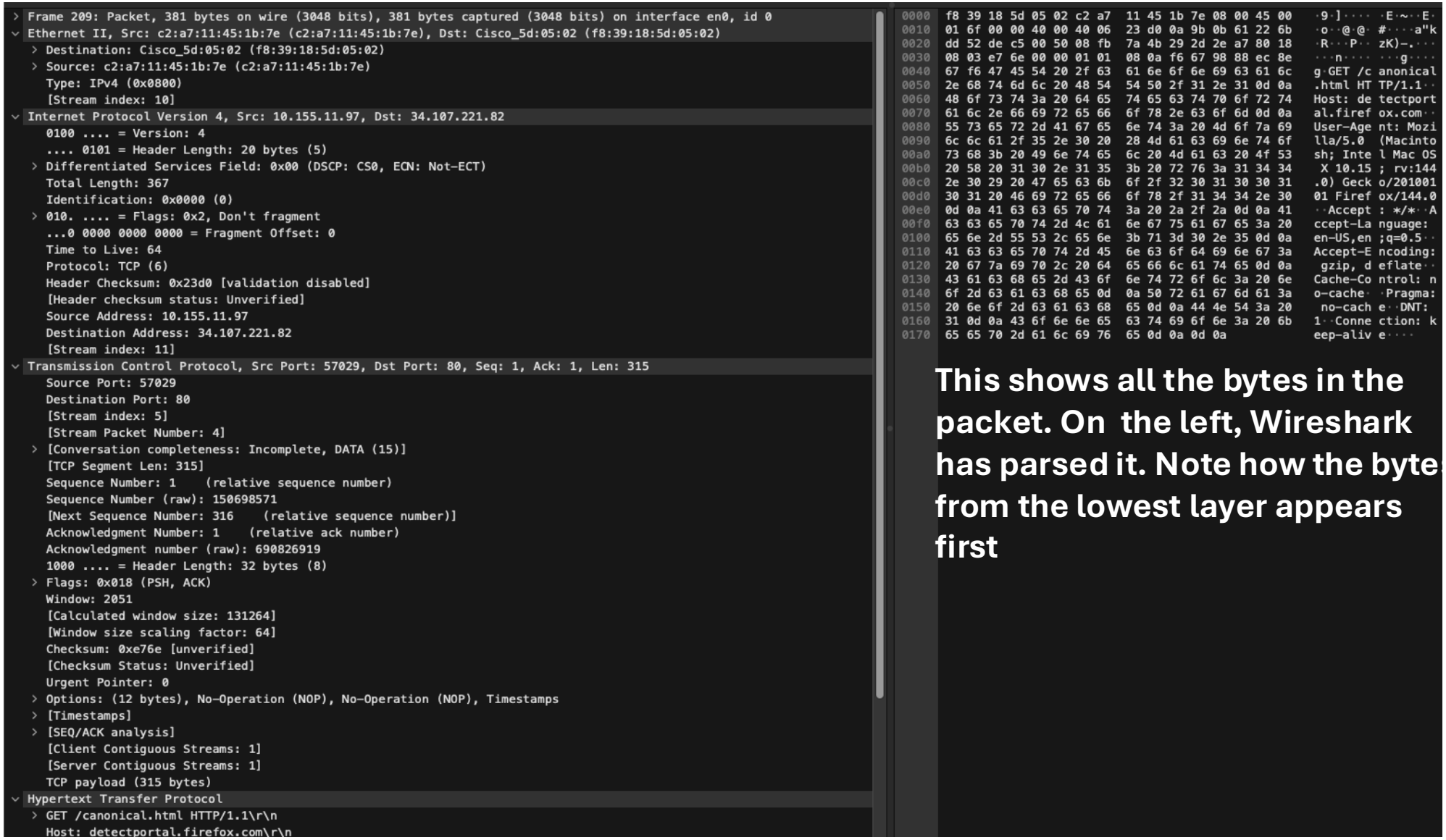
Apps

DHCP gives me my IP address

Every program in my machine starts making DNS requests

A TCP connection is established. I did a reverse DNS lookup with "nslookup 31.13.70.50" to find out this was Whatsapp

Example of an HTTP packet



This shows all the bytes in the packet. On the left, Wireshark has parsed it. Note how the bytes from the lowest layer appears first

What happens when we load a web page

Live demo in class

We will look at HTTP headers, responses etc.

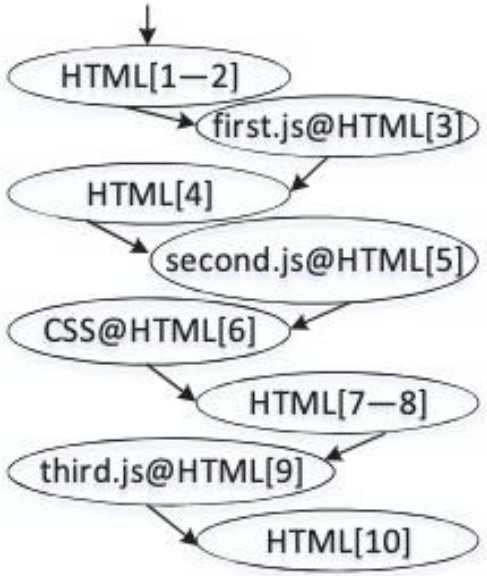
We will also see how web servers are “stateless”

The structure of a web page

- Web pages have three main types of files:
  - HTML – The first file that is loaded. Specifies the initial structure of the “Document Object Model (DOM)” that tells the browser the layout of the web page. The DOM may be modified by scripts loaded afterwards
  - JavaScript – A programming language run by the browser. Can execute arbitrary code, modify the DOM, send/receive other HTTP(S) requests, request other objects, …
  - CSS – Tells the browser how to style the page (very useful for developing web pages, not important from a networking pov)

Dependencies in a web page

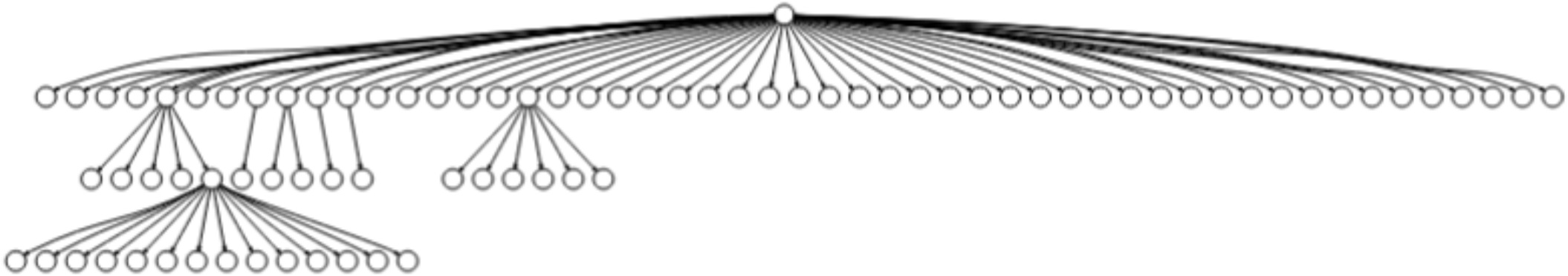
```
1  <h1>Text</h1>
2  <p>Text</p>
3  <script src="first.js"/>
       <!--Reads <p> tag-->
4  <b>Text</b>
5  <script src="second.js"/>
       <!--Accesses no DOM nodes-->
       <!--or JS state from first.js----->
6  <link rel="stylesheet" href="...">
       <!--CSS-->
7  <span>Text</span>
8  <span>Text</span>
9  <script src="third.js"/>
       <!--Writes <b> tag-->
10 <span>Text</span>
```



Example of an HTML file (left) first loaded when you open a web page. It instructs the browser to load other objects (i.e. javascript and CSS files). They need to be loaded in the order shown on the right to avoid violating dependencies

Images taken from "Polaris: Faster Page Loads Using Fine-grained Dependency Tracking" by Netravali et al.

Modern web pages are complex



Dependency graph from weather.com (taken from the same paper). A web page has many objects. Some have even deeper dependency trees

Questions to ponder

- How would you design the WiFi authentication protocol? What considerations might you use?
- Why are DNS packets the first to show up when we connect to WiFi?
- Would you restrict what javascript can do in any way?
- What are the benefits of a stateless design for web servers? What does stateless even mean in this context? Clearly websites maintain state about us. Think about the consequences this has when a TCP connection breaks, or we move between different IP addresses