TEU00311

What is the Internet doing to me? (witidtm)

Stephen Farrell
stephen.farrell@cs.tcd.ie

https://github.com/sftcd/witidtm
https://down.dsg.cs.tcd.ie/witidtm

The Internet is not the web

- Still an important point!
- The web is (roughly) the set of computers that speak the HTTP protocol
  - HTTP == HyperText Transfer Protocol
- Email doesn’t use HTTP, but rather (mostly) the Simple Mail Transfer Protocol (SMTP) which is a couple of decades older than HTTP
- Mobile networks (3G, 4G, 5G...) mostly run over IP using a bunch of protocols you’d prefer to never have to know about
- Many “triple-play” television services run over multicast IP
- But lots of our interactions with the Internet are via the web
- We’ll look more at the web now...

Mechanics of the Web
(Plumbing:-)

So you click a link...

- What do you think happens?

So you click a link...

1) 10 minute video that’s a bit simplified and out of date, but that might be a good thing:-

    https://gizmodo.com/what-actually-happens-when-you-click-on-a-link-1665573786

2) Domain Name System (DNS) resolution
   This is a recursive protocol that uses a world-wide piece of infrastructure (the DNS) that depends on a complex name registration system (next slides)

3) Transport Connection Established (TCP)

4) Transport Layer Security (TLS or SSL) session established

    Not mentioned in video but now happens >70% of the time when HTTPS used

5) HTTP request sent

6) HTTP response received
   Probably contains HTML with links so GOTO step 1 for each one

What’s a URL?

- A web “link” is really a Uniform Resource Locator
  - Generalisation: Uniform Resource Indicator (URI) which can be a name instead of a locator
  - Name/Locator or Identifier/Locator concepts are common in networking and often boringly controversial (so we’ll skip that:-)
- URI/URL definitions are in RFC3986
  - https://tools.ietf.org/html/rfc3986
  - But that’s also been “forked” by browser makers (a fine example of boring controversy;-)
    - https://url.spec.whatwg.org/
- URIs are used as web links but also for many other things, e.g. voice over IP signalling, e.g. SIP URIs can represent phone numbers
  - sip:1-999-123-4567@voip-provider.example.net
  - https://en.wikipedia.org/wiki/SIP_URI_scheme

Parts of a URI/URL

```
  foo://example.com:8042/over/there?name=ferret#nose
  \_/   \______________/\_________/ \_________/ \__/
   |           |            |            |        |
scheme     authority       path        query   fragment
```

“Real” example:

https://down.dsg.cs.tcd.ie/teu00311/stuff#middle

- Mostly, the URL schemes you’ll see will be “https” or “http” but there are many more
- The “authority” part is essentially the DNS name of the host (with an optional port number)
- The “path” can be thought of as a directory/folder name on that host
- The “query” part provides a way to parameterise URLs sent to programs
- The “fragment” provides a way to “land” your browser at some place in a presumably long page

The Domain Name System (DNS)

- Internet Assigned Numbers Authority (IANA, https://iana.org/) keeps lists of “top” level names (e.g. .com, .ie), IP address allocations and protocol numbering registrations
  - That’s a fantastic bit of bookkeeping but no more than that, policies are set elsewhere despite what many “Internet Governance” folks might say
  - IANA is homed in ICANN which is one of the policy setting/operations entities
  - The RIRs are another, and the IETF controls the protocols that create the space in which the policies operate
- DNS has a single root, below which we have .com, .ie, etc. and below those we have example.com, tcd.ie etc. and below which you have whatever e.g. TCD might want e.g. down.dsg.cs.tcd.ie
- There are a set of (13 logical, physically hundreds of) servers in the network that serve the “root zone” and who can tell you set of IP addresses where you can find more authoritative information about .com or .ie. or any of the Top Level Domains (TLDs)
  - Entities that do the bookkeeping for a TLD are called registries

The Domain Name System (DNS)

- Each of those servers will be authoritative for some zone (e.g. all the hosts in *.tcd.ie) but can also delegate as happens in the case of cs.tcd.ie
- When you want a new name (e.g. jell.ie) you have to go to a registrar who works with the relevant registry (e.g. Tolerant Networks Limited is me-as-a-registrar for .ie) and then pay to rent that for a few years
  - Not all names are available – could be taken or a trade mark – some lawyers love this stuff!
- You need a server machine with an IP address for that to be useful
  - You often get from a hosting company or cloudy service provider like AWS or Azure or whomever
- And then your machine’s name and IP address need to be published in the DNS
- Then you can e.g. install apache (a web server) and make your web site and interact with e.g. LetsEncrypt.org to get a public key certificate so TLS will work
  - Then browsers can nicely visit your web site
  - Web crawlers and attackers of all sorts will also constantly ping your machine, all the time
- At that point you may decide to be an advertiser or not, if you do, you’ll probably start to record things about people who visit you and maybe you’ll sign up to some advertising platform to make money for you and them
- But you might also decide not to track anyone (what I do)
  - Modulo normal web logs!

IP Addressing and Routing

- How does a browser make that TCP connection to a web site after it gets the IP address?
- Regional Internet Registries (RIRs) such as RIPE allocate Autonomous System (AS) numbers and blocks of IP addresses to network operators (e.g. ISPs, enterprises, hosting companies)
- Addresses are further allocated downstream, eventually to e.g. your laptop via DHCP, or (semi-)manually to a server hosting a web site in a data centre
- ASes tell one another about who has what blocks of addresses and how to route packets to one another via the Border Gateway Protocol (BGP) – remember there are tens of thousands of ASes
- There are “private” address ranges as well as public IPv4 and IPv6 addresses. Often private IPv4 addresses (e.g. 10.0.0.1 or 192.168.1.1) are used within home networks and Network Address Translation (NAT) is used to map those to shared public IP addresses - mainly to get around the fact that all 4 billion IPv4 addresses have been allocated by IANA already
- Once your browser has a source IP address and knows the web site (destination) IP address then they can establish the TCP connection, assuming the relevant ISPs have done a good job with BGP, which mostly happens
  - The source address is needed to get the answer from the web site

HyperText Markup Language (HTML)

- Simplistically, the web pages that you download via HTTPS are HTML files
  - There’s also lots of non-HTML content: video, Javascript, images
  - Reality is nowhere near this simple… but it can be!
- HTML describes the structure of the web page
- Browser renders that
- Key concept: hypertext links

HTML for a trivial web page

https://down.dsg.cs.tcd.ie/witidtm/examples/trivial.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title>A trivial web page</title>
</head>
<!-- Background white, links blue (unvisited), navy (visited), red (active) -->
<body bgcolor="#FFFFFF" text="#000000" link="#0000FF" vlink="#000080" alink="#FF0000">
    <p>The trivial content is just a link to the
        <a href="https://jell.ie/news/">jell.ie news</a>
    </p>
</body>
</html>
```

A link on our trivial web page

https://down.dsg.cs.tcd.ie/witidtm/examples/trivial.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title>A trivial web page</title>
</head>
<!-- Background white, links blue (unvisited), navy (visited), red (active) -->
<body bgcolor="#FFFFFF" text="#000000" link="#0000FF" vlink="#000080" alink="#FF0000">
    <p>The trivial content is just a link to the
        <a href="https://jell.ie/news/">jell.ie news</a>
    </p>
</body>
</html>
```

HyperText Transfer Protocol

- Me: GOTO white/black-board
- You: draw your own diagram or take a pic:-)
- HTTP/1 is a text-based protocol and still widely used
- HTTP/2 is semantically the same but binary and with a few other efficiency improvements – go-faster-stripes basically
- h3 is in-development now, and may turn out to be HTTP/2 over QUIC
- HTTP/TLS = HTTPS – back to the drawing

TLS Sessions per “front page”

I recently (2019Q1) did some tests, loading the “front page” of these sites and recording the network traffic. The numbers below are the number of separate TLS sessions that were created when the initial page HTML is rendered, to display content and (mostly) ADs

site	N	min	max	stdev	avg
ietf.org	22	4	12	2.12	9.29
irishtimes.com	22	74	**158**	22.47	126.36
jell.ie	22	4	12	1.96	6.73
nytimes.com	22	29	**98**	16.07	81.23
rte.ie	19	38	63	6.33	50.32
tcd.ie	22	69	**102**	10.61	92.18
www.ietf.org	14	4	7	0.73	5.07

- N = count of tests done
- Min,max,stdev,avg refer to the number of TLS sessions for each test
- Automation tool used was Selenium on Ubuntu which mostly used FF, but also chrome/opera for some tests
- Browsers/selenium drivers are “out of the box” with no special config, nor plug-ins, extensions etc.

Why did the Irish Times front page have an average of 120+ TLS sessions?

Why did tcd.ie have an average > 92?

Why did the Irish Times front page have an average of 120+ TLS sessions?

Why did tcd.ie have an average > 92?

With a desktop browser... you can see what’s happening: “shift-ctrl-I”

(but we’re getting a bit ahead with that)

Scaling and Content Delivery Networks

Big web sites aren’t trivial

- Caveat: I don’t run a big web site! So I don’t really know this stuff in full detail.
- There’s a web called “high scalability” that publishes “war stories” for people who manage large systems
- One article gives a nice perspective from a sys admin point-of-view – probably too much detail for us, but good as a description of how a “middle-sized” web site has developed in the back-end over the last decade
  - https://www.betabrand.com/ “Best. Pants. Ever.” (sigh;-)
  - https://boxunix.com/2018/12/10/from-bare-metal-to-kubernetes/
  - If you do read that, I’d say the take-away is that there’s enough complexity in even a middle-sized merchant web site that many things can go wrong on their end, including some that could affect you (e.g. data leaks, which we’ll look at later)
- Another article talks about how Netflix do stuff...

“Netflix: What Happens When You Press Play?”

- We’ll look at some quotes from this Dec. 2017 article about Netflix (author: Todd Hoff)
  - https://highscalability.com/blog/2017/12/11/netflix-what-happens-when-you-press-play.html

  Some 2017 Netflix statistics:
- Netflix has more than 110 million subscribers.
- Netflix operates in more than 200 countries.
- Netflix has nearly $3 billion in revenue per quarter.
- Netflix adds more than 5 million new subscribers per quarter.
- Netflix plays more than 1 billion hours of video each week. As a comparison, YouTube streams 1 billion hours of video every day while Facebook streams 110 million hours of video every day.
- Netflix played 250 million hours of video on a single day in 2017.
- Netflix accounts for over 37% of peak internet traffic in the United States.
- Netflix plans to spend $7 billion on new content in 2018.

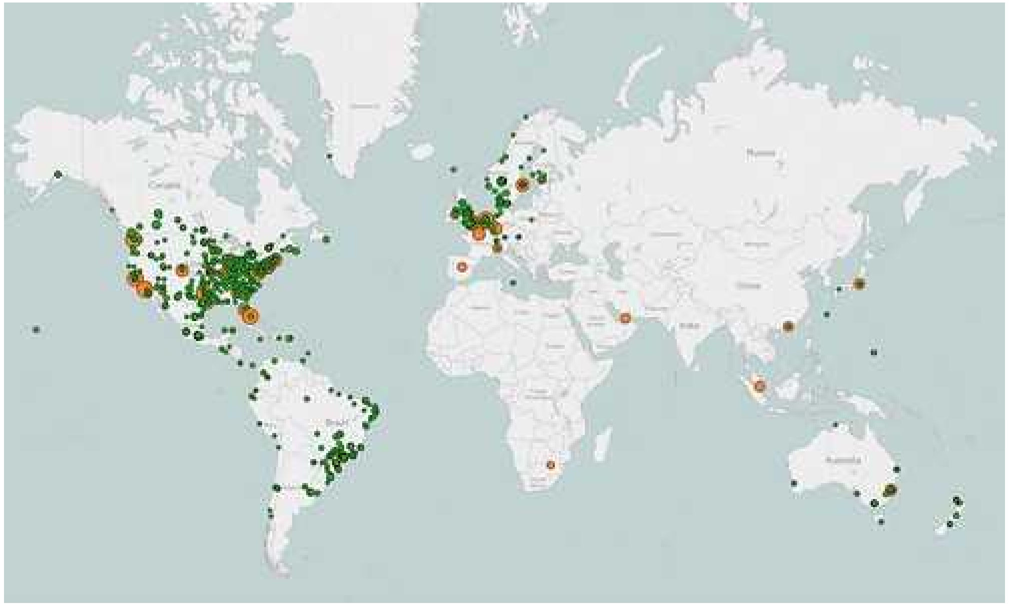
The quotes...

- “Netflix collects a lot of information. Netflix knows what everyone has watched when they watched it and where they were when they watched. Netflix knows which videos members have looked at but decided not to watch. Netflix knows how many times each video has been watched…and a lot more.”
- “When browsing around looking for something to watch on Netflix, have you noticed there’s always an image displayed for each video? That’s called the header image. The header image is meant to intrigue you, to draw you into selecting a video. The idea is the more compelling the header image, the more likely you are to watch a video. And the more videos you watch, the less likely you are to unsubscribe from Netflix.”
- “Everyone used to see the same header image. Here’s how it worked. Members were shown at a random one picture from a group of options, like the pictures in the above Stranger Things collage. Netflix counted every time the video was watched, recording which picture was displayed when the video was selected.For our Stranger Things example, let’s say when the group picture in the center was shown, Stranger Things was watched 1,000 times. For all the other pictures, it was watched only once each. Since the group picture was the best at getting members to watch, Netflix would make it the header image for Stranger Things forever. ”
- “That’s why Netflix now personalizes all the images they show you. Netflix tries to select the artwork highlighting the most relevant aspect of a video to you. How do they do that? Remember, Netflix records and counts everything you do on their site. They know which kind of movies you like best, which actors you like the most, and so on. Let’s say one of your recommendations is the movie Good Will Hunting. Netflix must choose a header image to show you. The goal is to show an image that lets you know about a movie you’ll probably be interested in. Which image should Netflix show you?”

Content Deliery Network

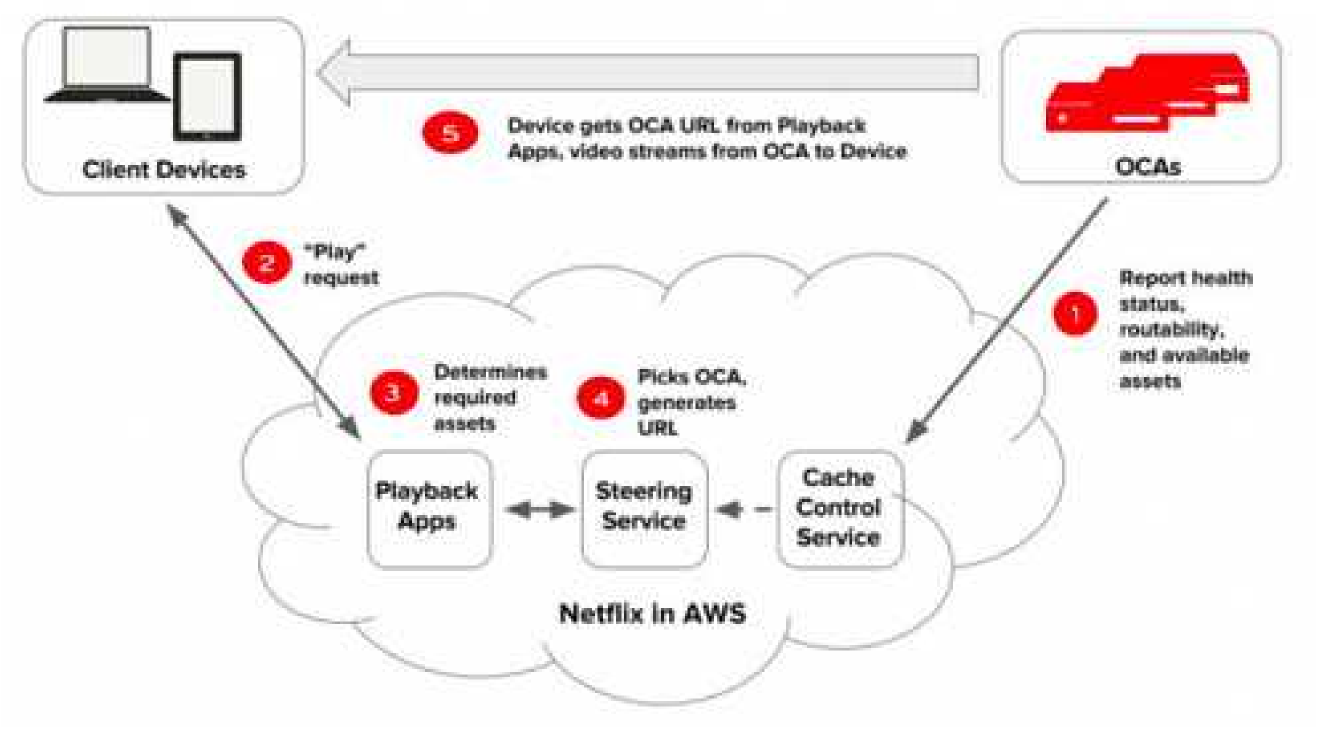
- Netflix use “hundreds of thousands” of Amazon EC2 instances for their application logic – stuff that happens before video is streaming
  - Anyone know what “Amazon EC2” is? If not, you wanna?
- For video, Netflix built their own Content Delivery Network (CDN) – most web sites use 3rd party CDNs like Akamai, Cloudflare etc.
  - CDNs represent a kind of Internet centralisation that you may not have known about?
- Basic idea is to put large data files near where the customer is so data (video) gets there faster, and to be more reliable when failures happen
- So where are Neflix CDN points-of-presence (PoPs)?

Netflix PoPs (2017)



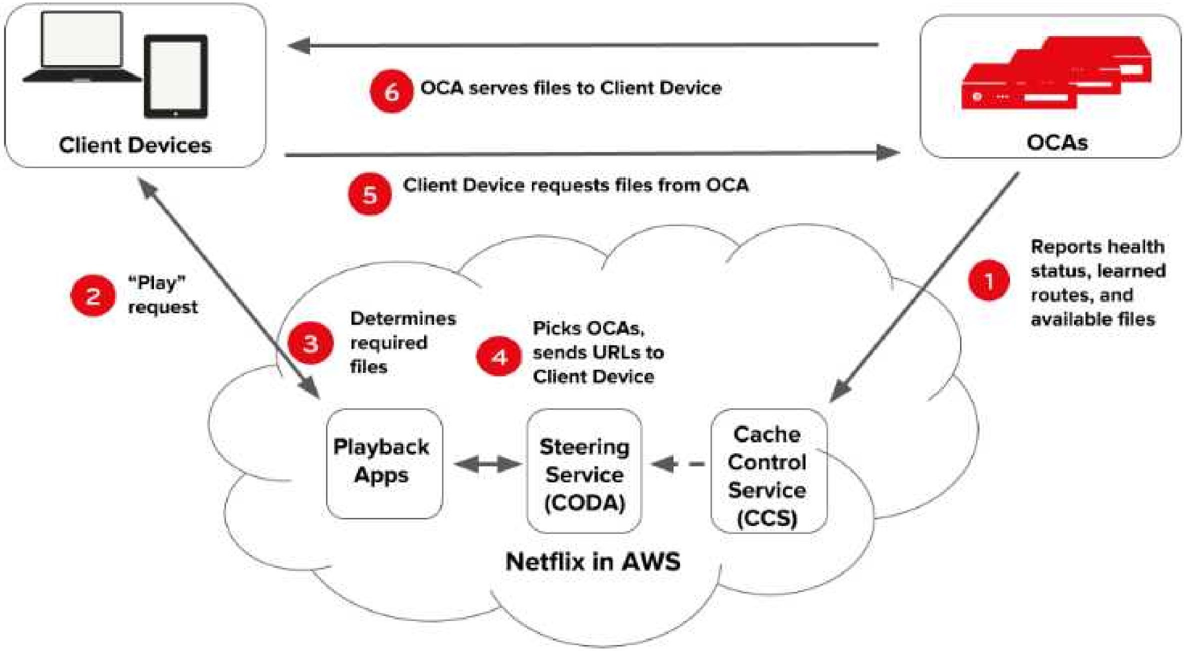
- PoPs are mostly hosted by ISPs or IXPs
- Because Netflix send lots of data (37% of the internet traffic in the US) and otherwise the ISPs would have to pay interconnect charges to the ISPs between them and the content
- That’s done with Neflix-designed hardware physically located in the ISP’s network – called an Open Connect Appliacnce (OCA)

So what happens when you press Play?


Client Devices
5
Device gets OCA URL from Playback Apps, video streams from OCA to Device
OCAs
2
"Play" request
1
Report health status, routability, and available assets
3
Determines required assets
4
Picks OCA, generates URL
Playback Apps
Steering Service
Cache Control Service
Netflix in AWS

Open Connect Appliance - Global

More up to date pic



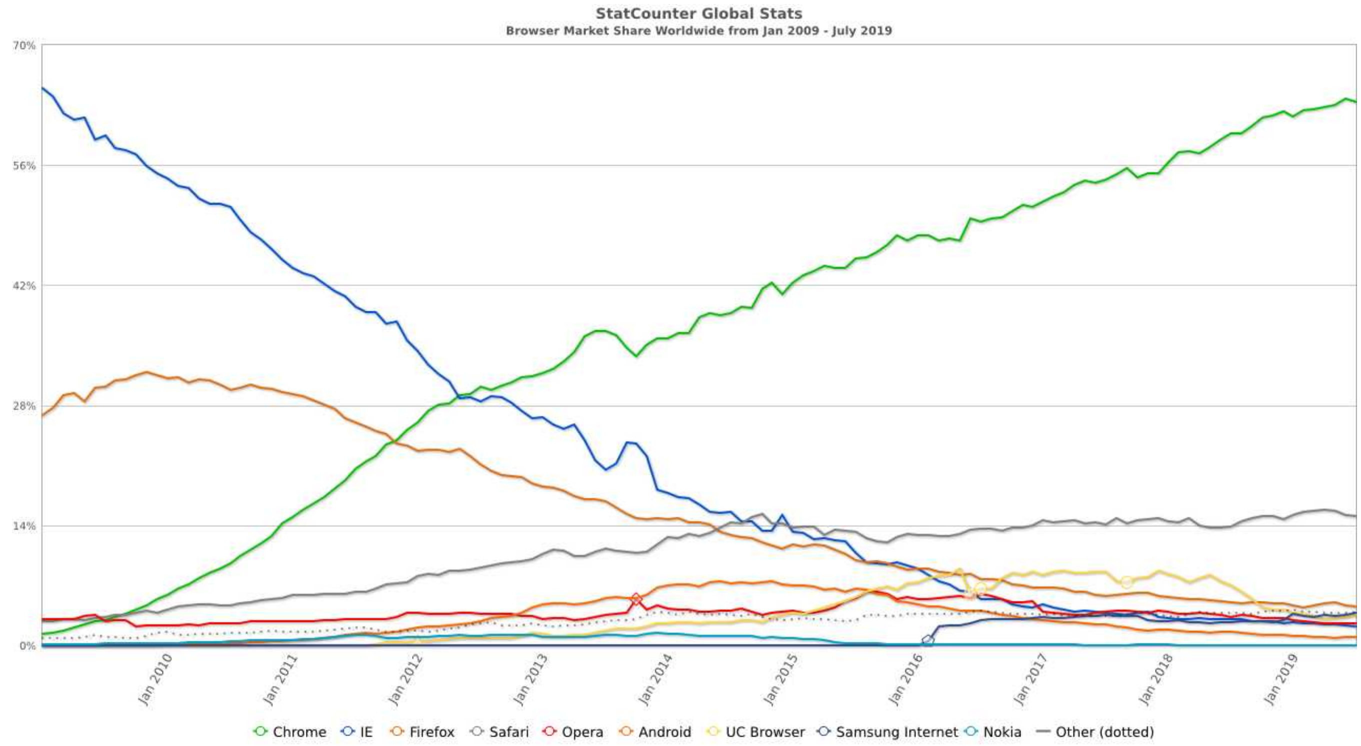
Client Devices
OCAs
6 OCA serves files to Client Device
5 Client Device requests files from OCA
1 Reports health status, learned routes, and available files
2 "Play" request
3 Determines required files
4 Picks OCAs, sends URLs to Client Device
Playback Apps
Steering Service (CODA)
Cache Control Service (CCS)
Netflix in AWS

I have no idea if the differences in these diagrams are significant:-)

https://openconnect.netflix.com/Open-Connect-Overview.pdf

Browser Hygiene

Browser worldwide market share



2009-2019

Chrome
Safari
Firefox
Opera
Android
UC browser
...
Brave
Vivaldi

https://gs.statcounter.com/browser-market-share#monthly-200901-201907

Overall browser landscape

- Browser defaults are chosen by browser implementers (Google, Mozilla, Apple,...)
  - Allow Javascript and cookies, do telemetry, try get you to login, keep state ...
- Historically, browser-makers seemed to care most about market share
  - Performance and rendering were their main concerns as they lose market share if they’re slower or sites don’t render sites (well)
- They started getting significantly better at security a while back (2013+)
- Some browser-makers are starting to get a bit better at privacy
- IMO they don’t behave as if they think you should be the one in control

My browser setup

- Default browser: FF “nightly” + NoScript/Ghostery & disallowing cookies, with some white-listed sites with search via DuckDuckGo (“!g” works too, if needed:-)
  - This is the only browser that saves logins, but not for sensitive things (we’ll consider passwords later)
  - Some sites don’t work with the above; mostly: screw ‘em
- Tor Browser: If searching for anything sensitive (e.g. medical info)
- If-need-be: chromium/incognito with no write-access to disk and so that it shoots it’s own brain out on exit (at least I hope so;-)
  - Use that e.g. for airline/hotel bookings
- If-all-else-fails: Brave or Vivaldi
- On phone: sailfish browser with no JS/no cookies and 2ndary open-kimono browsers if-need-be (Webcat/Web pirate)
- Recommend you figure out some browser-hygiene you consider ok and follow that
  - Requires some self-discipline!
  - Be willing to help others do the same!

Your browser setup?

- What browser(s) do you use? What extensions/plugins?
  - (your stuff here)

- We’ll return to cookies etc shortly