Dacorum U3A

Computer Support Group

27th July 2018

Agenda

- Open forum
- Identify subjects for breakout groups and later meetings
- Web Browsers
- Tea and Coffee break (about 3.00 pm?)
- Continue Presentation
- Breakout groups looking at individual problems

Introduction

How is a web page processed?

Start by looking at a couple of January's slides

What does the server do?

What does the browser do?

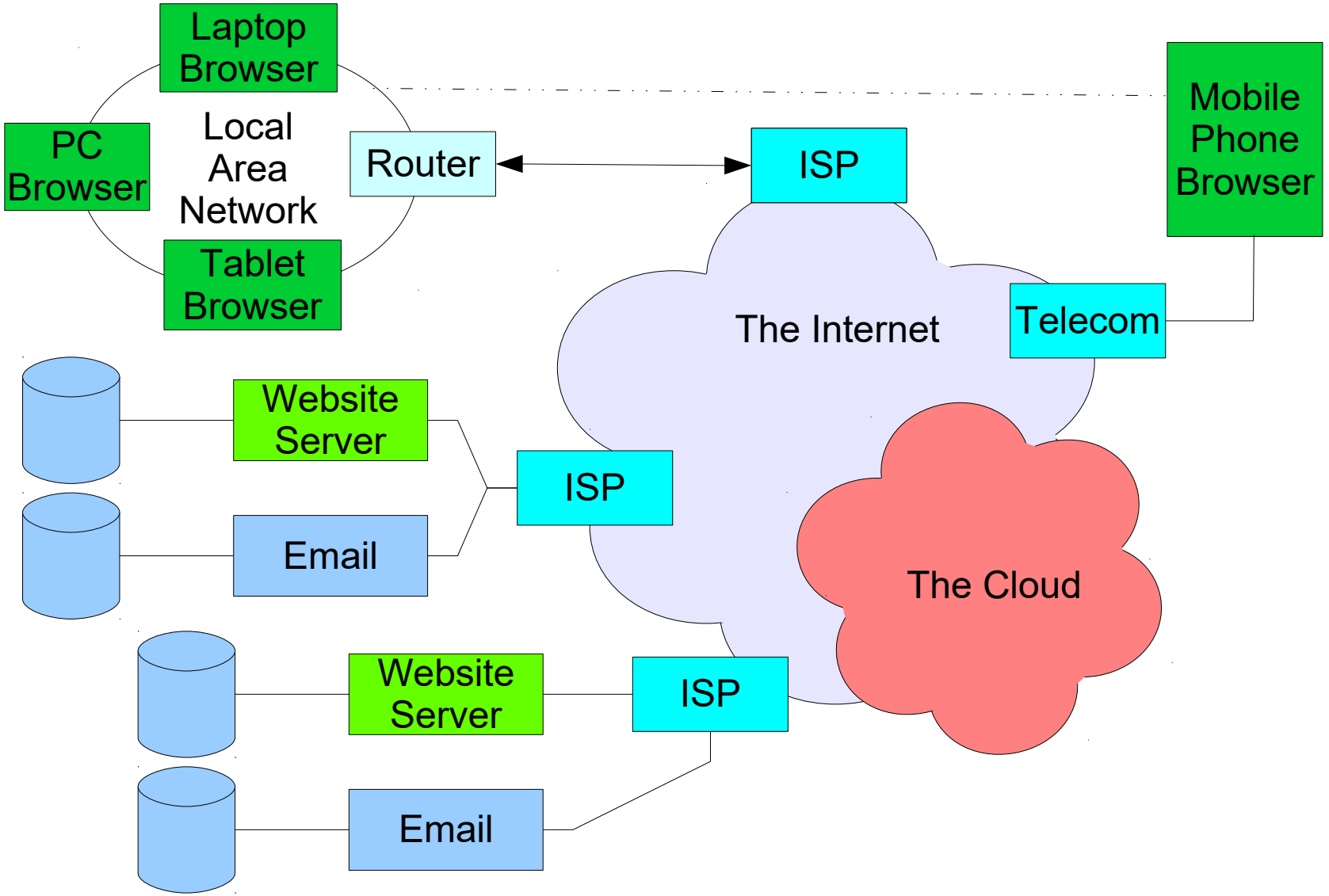
Cache and History

Cookies, Forms and Passwords

Simple HTML

Fraudulent Web pages

Network View



IP Address

Every device connected to the Internet has an unique address called the IP address.

It is used to route messages to the device

Old ip addresses of the form n.n.n.n (n=0-255)

Allows 'only' 4 billion devices

New IP addresses of 128 bits of the form nnnnnnnnnnnnnnnn (n=a hexadecimal digit).

Allows 'adequate' number of devices

Both forms co-exist for the foreseeable future

How does the server know what to do

All Web Page request URL start with ‘HTTP’ or ‘HTTPS’.

This is automatically added by the browser and is recognised by the server as a web page.

Eg: u3adacorum.co.uk is actually sent as http://u3adacorum.co.uk and interpreted by the server as http://www.u3adacorum.co.uk. The www is added because there is no alternative, this is not always true.

What does the server now do?

The server looks at the file extension, by convention, files containing ‘normal’ web page commands have an extension of htm, html or xhtml but this is not an absolute requirement.

However some web page generation packages (there are any number of them – e.g. page builder, DreamWeaver, Workpress…) have their own extensions. php (a very common method of generating html) usually uses .php as the file extension, PERL uses .pl.

The server then returns the html (xhtml) to the browser through the internet.

Is that all the server does?

After the browser has decoded the html it may find that its needs more data (css, javascript, image, video etc.).

If this data is not in the browser's cache the server will send it.

If it is in the browser's cache it will send the file's date and the server will send the file, if it is newer.

In unusual circumstances the browser might ask for some more processing and the return of more data.

What the server doesn't do

The server retains NO information about any previous web pages. If one page relies on information about a previous page, this must be placed in a 'cookie' or sent in the uri.

(This is not 100% true as extensions like php can keep session data and the server can write and read temporary files on database.)

There is no way to generate cookies directly by html, only by javascript. Not all users allow cookies to be used.

What does the browser do?

For EACH Web page: initiates and records the request (the server cannot do this)

Receives and decodes the response

Send any relevant Cookies

Searches the cache for the presence and date of any ‘loaded’ data

Requests any (more up to date) ‘loaded’ data. May be from another Website

Formats and displays the Web page

What does the Browser do 2

The browsers does not necessarily ask for ALL the files it requires.

Eg pictures (photographs), audio and video files. These can be displayed as they arrive or requested later by the web page. This allows the page to display asap. without having to wait for (possibly large) files which may not even be required.

In fact video can start playing BEFORE the whole has arrived completely

Cache

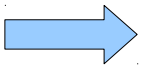
The cache holds most of the data sent from the server and the browser uses it in preference to requesting the server to send it again. However, a check is made by the server that it is not a newer version.

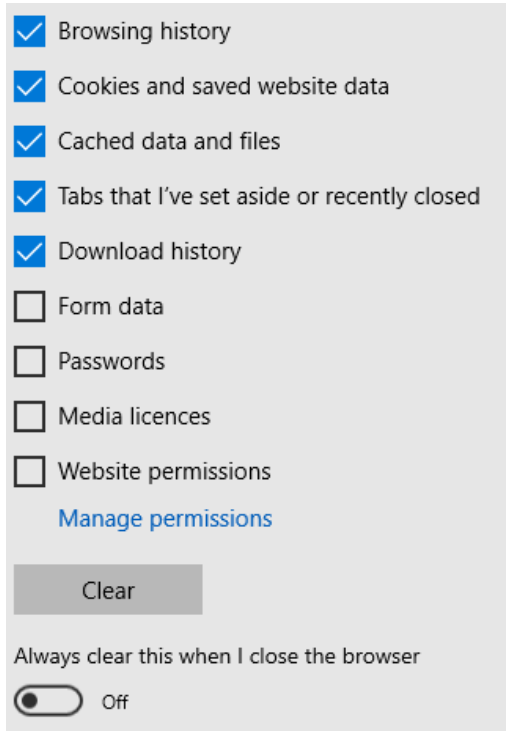
It is possible to force the server to send the data again by:

- Using Cntl-F5 (for many browsers)
- Clearing the cache (use ‘Help’ or Internet search as each browser is different)

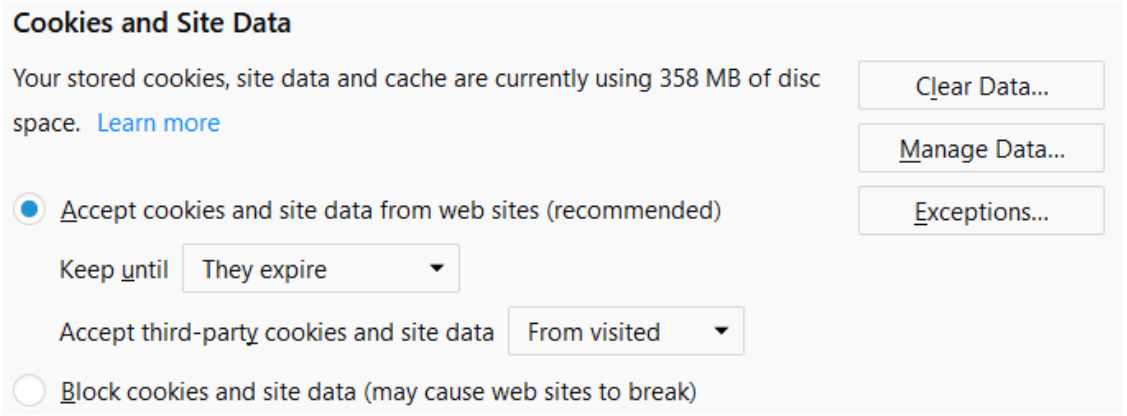
Cache 2

The cache is organised in different ways by different browsers. E.g. Edge calls it Browsing history





Firefox has separate controls for cookies and site data. Form data and passwords are separate



History

The browser keeps a history of all the web pages that have been visited.

This allows the browser to display a list of pages visited previously which partially match what you are typing in.

It is possible to display this list but again each browser is different.

I delete a lot of the entries to minimise the list and enable it to show relevant entries more quickly

Cookies

Cookies (retained data) are vital for some purposes (Banking userid, shopping basket etc). ‘Session cookies’ will be deleted when you close the browser.

You probably would prefer not to Cookies that track you, to control what advertising you see.

Browsers have built in facilities for controlling which cookies are kept but these have limited value.

Some Browsers have ‘add ons’ that allow YOU to say which cookies you want to keep. On Firefox, I use Cookie Broker whch allows me to ‘whitelist’ a site’s cookies and delete the rest.

Forms

A Web Page can include a Form which allows data to be entered which is sent to the server when a submit button is pressed.

The browser will retain this data. When the form is displayed again, it will search through previous entries into each field and prefill the field if a partial match is found.

This is useful for entering user ids etc.

Passwords

A field on a form can be defined as a password field. This will replace each character entered as an ‘*’.

Most browsers can (as an option) retain the password, encrypted under a master key. Again each browser is different

Simple HTML

HyperText Markup Language

- An intenational standard for formatting Web Pages. Latest version is HTML 5 which is not supported by older browsers.
- Entirely readable text.
- Ignores any ‘white space’ - putting extra spaces or carriage returns has NO effect on the results
- Is tolerant of errors

XHTML

eXtended HyperText Markup Language

Actually uses the international standard XML (eXtended Markup Language)

Almost identical to HTML but is not tolerant of errors in the formatting

Phishing

You receive an email asking you to sign into your account for some reason. To be helpful, a link is provided either in the email or in an attached document. The link looks legitimate (but the displayed text doesn't need to be anything to do with the underlying link). Even if you look at link itself it might look legitimate (e.g. barclays.xyz.com/signon.html).

If you click on the link, you see the page that you expect to see. In fact, it IS the page – almost.

Phishing 2

There are many ways for the criminal to forge the page, one of the easiest is:

- Display the page
- Save it (the browser can do this)
- Modify what happens what one of the controls is activated
- Save the modified page as barclays.xyz.com/signon.html

It is hard (even for me) to see what has been done!