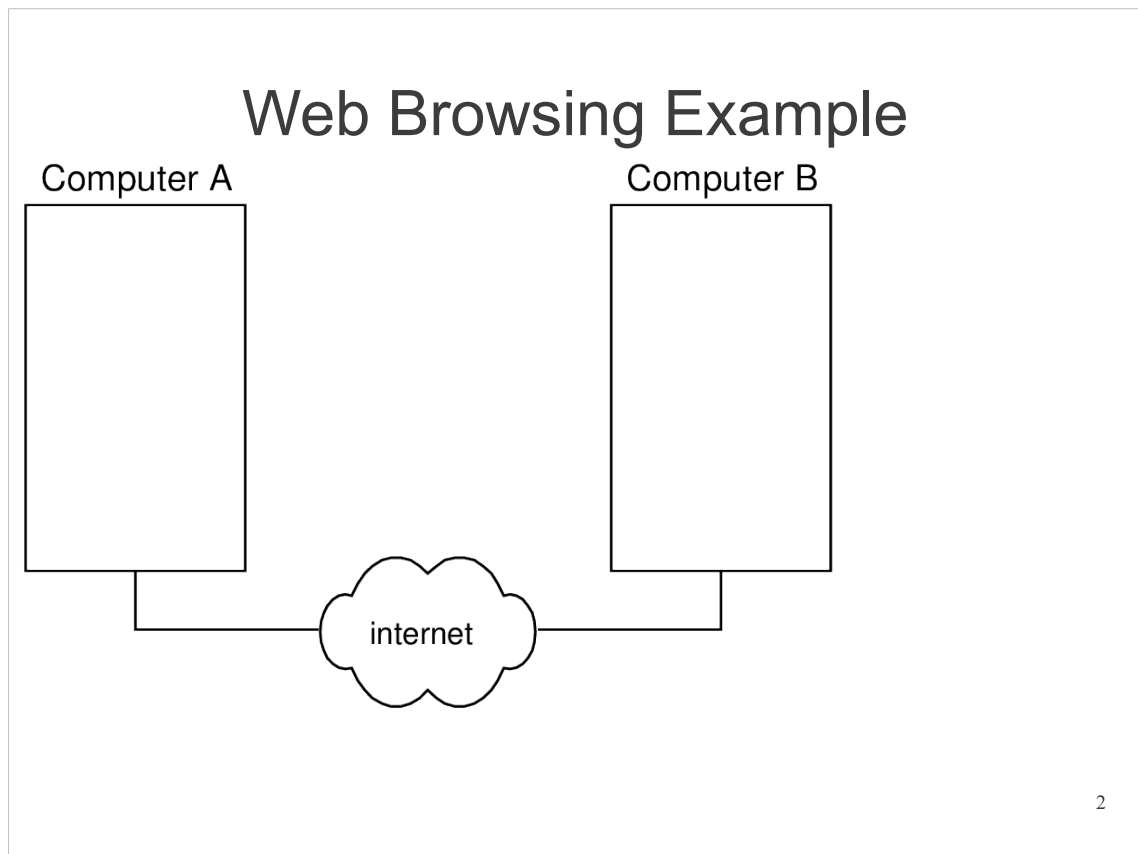
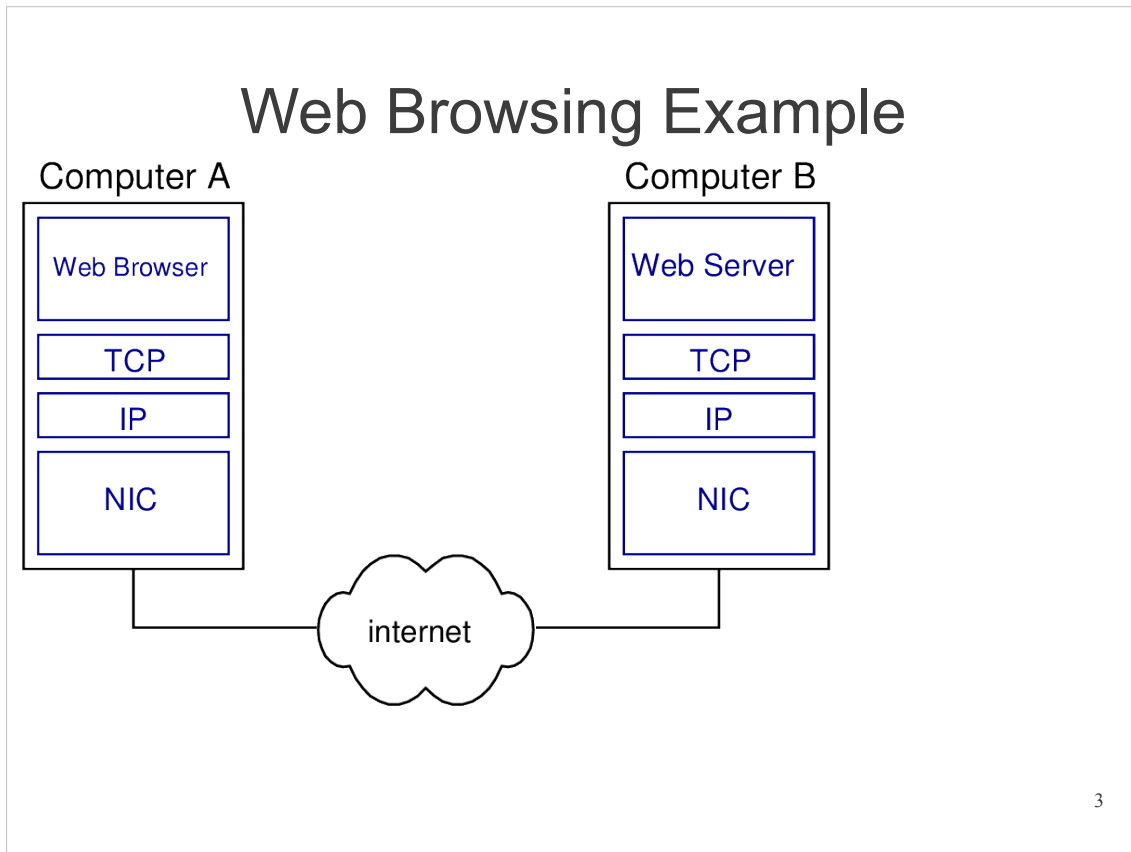


## How Web Browsing and HTTP Works



Lets consider an example that shows how web browsing and HTTP work. The example will cover a simple, but very common case. There are many more details of HTTP that are not covered by this example.

We have two computers on an internet.

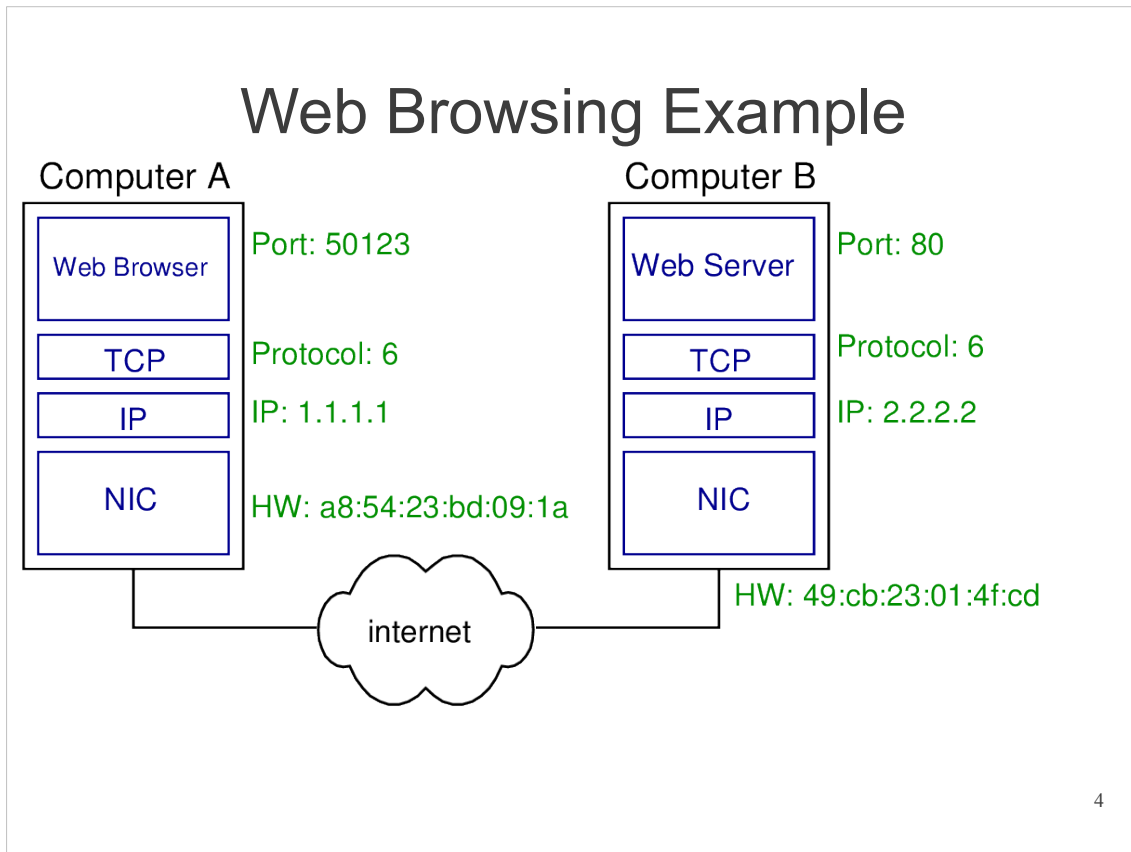


From a layering perspective, the software/hardware related to web browsing in each computer are shown on the slide.

Computer A (where the person sits) runs a web browser. They want to visit a web site on computer B, which runs a web server. A web browser and web server are software applications and considered part of the application layer.

The application protocol used to communicate between web browser and web server is HTTP (not shown). HTTP uses TCP as a transport protocol, which in turn uses IP as the network protocol. TCP and IP are implemented in the operating system.

The data link layer and physical layer protocols used on the computers may differ, especially if they are on different subnets. Lets hide the details of the protocols and say IP sends data via the Network Interface Card (NIC). That is usually hardware (on the motherboard or a separate PCI card).



This slides shows examples of addresses used.

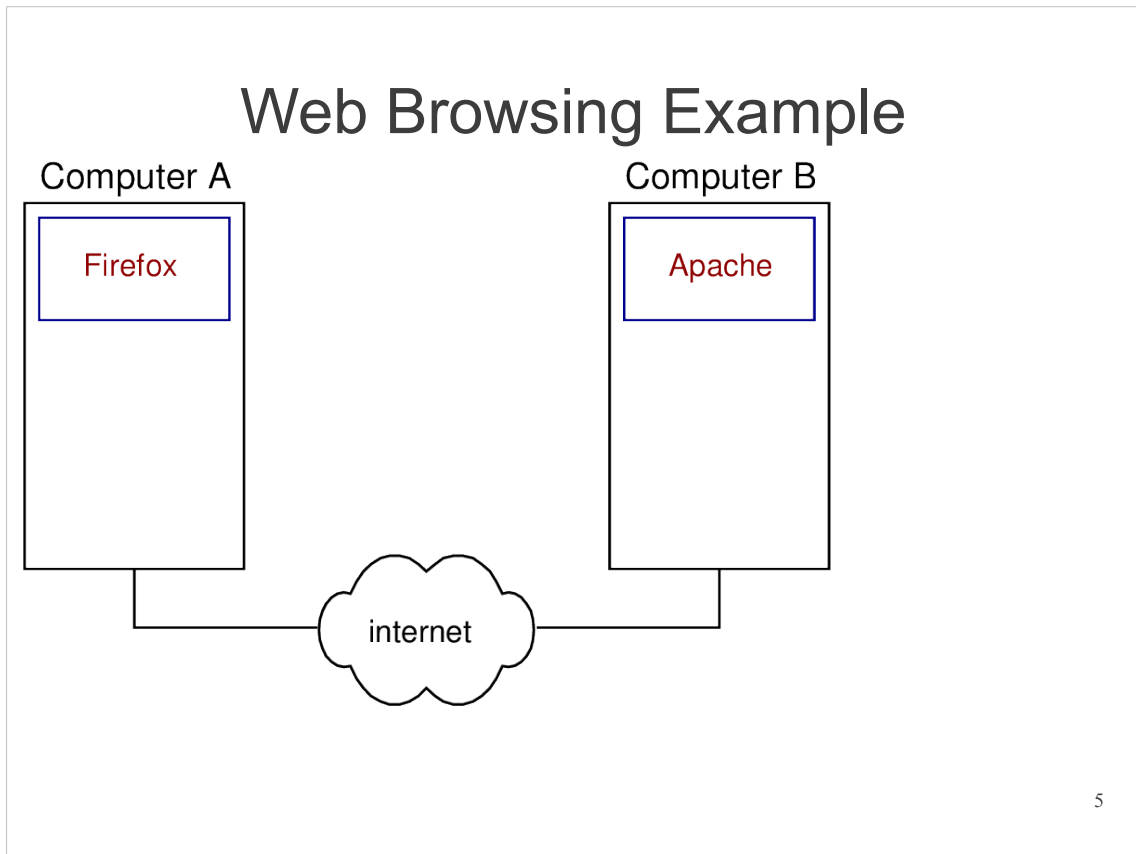
NICs are identified by hardware addresses (48-bits, shown as 12-digit hexadecimal). They are used for LAN communication, and are not so relevant for this example.

Computer interfaces on an internet are identified by IP addresses. Simply, consider the address of computer A to be 1.1.1.1 and that of B to be 2.2.2.2.

The transport protocol used by the application is identified by a protocol number. TCP has protocol number 6. (UDP is 17, ICMP is 1; look in /etc/protocols to see the list). Note that both sides use the same transport protocol.

Applications (or more precisely sockets used by applications; a single application, like web browser, may use multiple sockets and ports) are identified by port numbers. When an application protocol wants to communicate via TCP/IP, it uses a unique port number on that computer. Application servers commonly use default or well-known port numbers (see /etc/services). For example, most web servers use port 80. Application clients usually have port numbers assigned by the operating system (they are dynamic and not known in advance). In this example, the web browser uses port 50123.

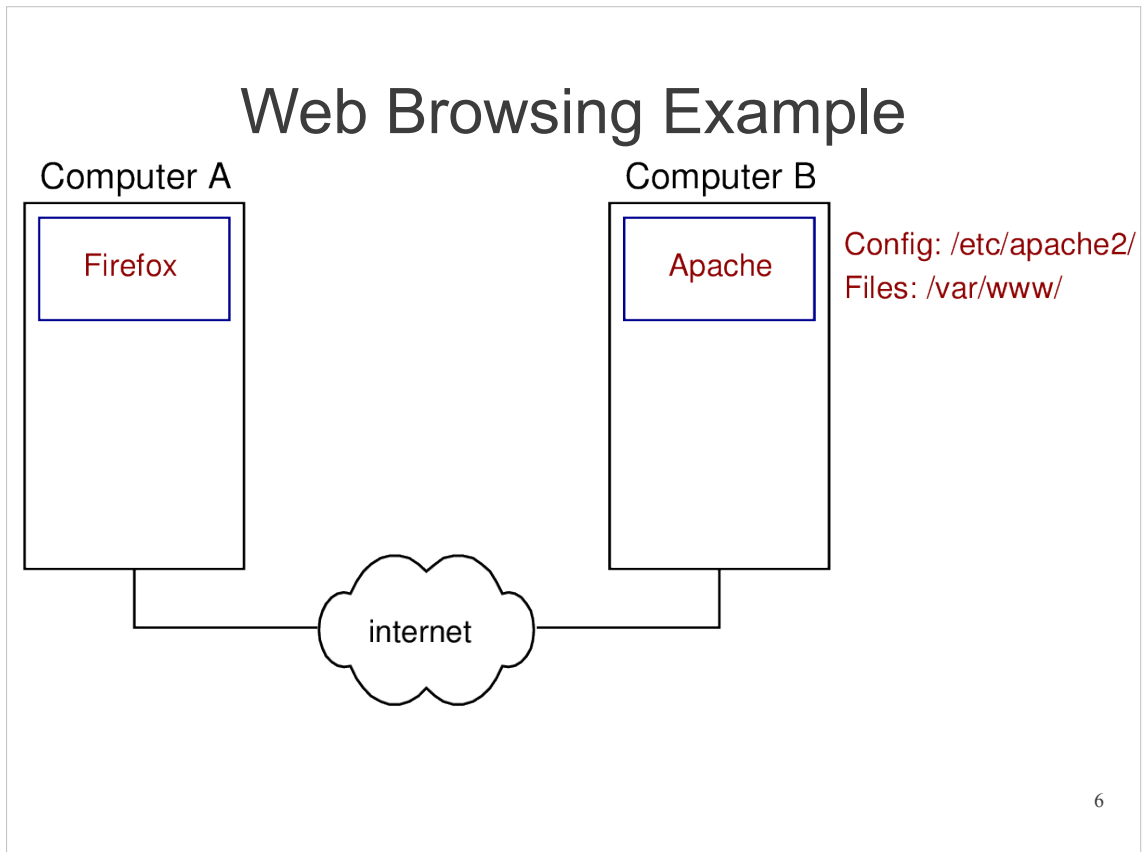
For internet communications we can talk about a unique connection between a client and server. Each connection is identified by 5 addresses: source port, destination port, protocol number, source IP and destination IP.



This shows examples of web browser and web server applications.

Firefox is a common web browser. Others include Internet Explorer, Chrome, Opera, Safari and many mobile phone based browsers. Other, command-line based browsers, are lynx and wget.

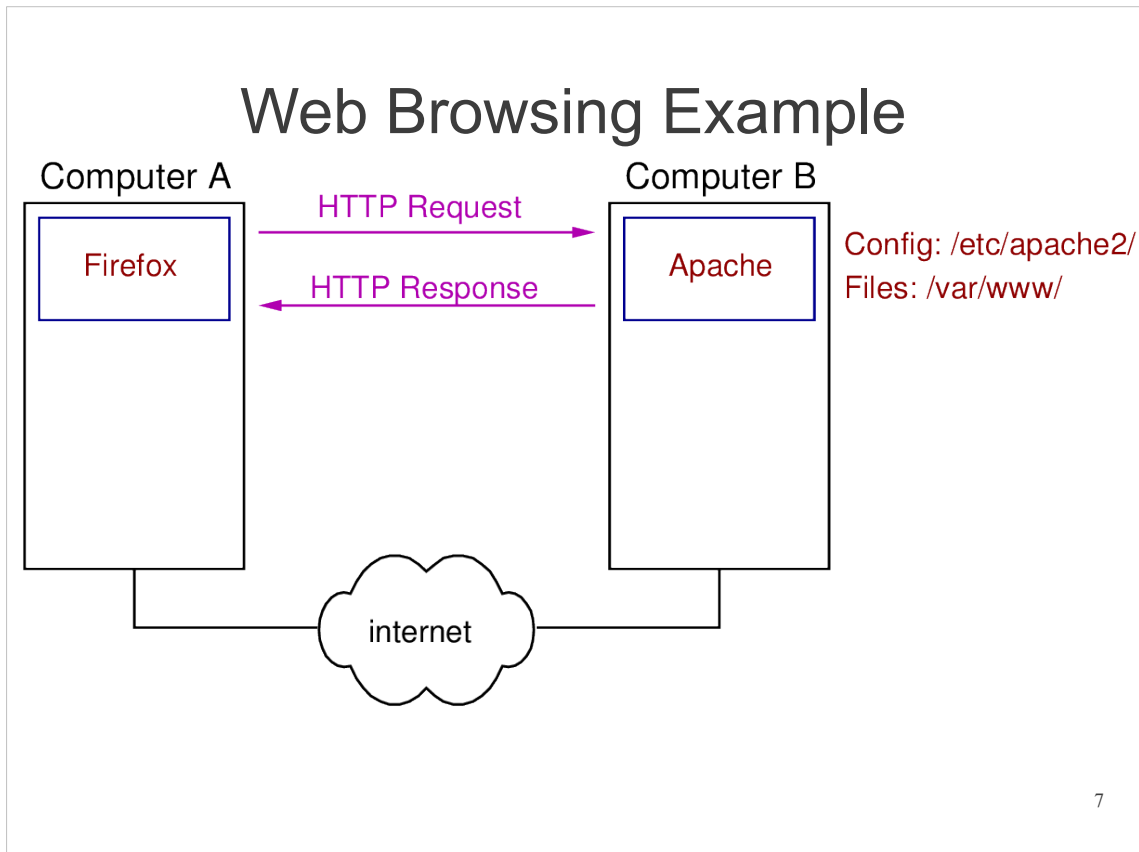
Apache is a common web server. Others include IIS, nginx, lighttpd, ...



A web server hosts files in some directory, and as web browsers requests those files, the server retrieves them from disk and sends them back to the browser.

Apache web server is configured by a set of text files. They are usually in the directory `/etc/apache2/`. They set parameter values for the web server.

The files that Apache web server hosts (and makes available to web browsers) are usually stored in the directory `/var/www/`. This is the web server root. When a request to the web server arrives with URL path `/index.html`, then it refers to the file `/var/www/index.html` on the computer filesystem. URL paths are relative to filesystem path `/var/www/`.



HTTP is the application protocol used for web browsers and web servers to communicate. The basic operation is:

- A user of the web browser types in a destination URL or clicks on a link for some destination URL, e.g. `http://www.example.com/index.html`
- The URL indicates the domain name of the destination computer running the web server; DNS maps the domain name to an IP address, e.g. `2.2.2.2`.
- The web browser sends a HTTP Request message to the IP address of the web server. The request contains the path of the URL, e.g. `/index.html`
- When the web server receives the HTTP Request, it checks if the requested file is available on the filesystem, e.g. file `/var/www/index.html`
- If the file exists, the file is read and sent back to the web browser in a HTTP Response message. The response also includes information about the status of the request (e.g. was it successful).
- When the web browser receives the HTTP Response, it parses the HTML in the received file, and displays the content on the screen.

There are many details not mentioned above like:

- HTTP messages include header fields, indicating useful information to the other side
- If the requested file doesn't exist, or is not accessible, the web server still sends a response, but with different status information, e.g. `404 Not Found`.
- Multiple, separate requests may be automatically sent by the web browser, e.g. to request images and style sheets included in web pages.

# Summary

- Web server listens on port 80
  - `/etc/services` lists the port numbers
- Web browser OS chooses own port
- HTTP: Request/Response protocol
  - Request types: GET, POST, ...
  - Response codes: 200, 301, 304, 401, 404, ...
  - TCP is protocol, number 6 (see `/etc/protocols`)
- Apache is common web server
  - Configuration in text files in `/etc/apache2/`
  - Main config file: `/etc/apache2/sites-available/default`
  - Web site is in `/var/www/`

8

See other handouts, lecture notes and websites to see further details of HTTP, Apache configuration, port numbers etc.