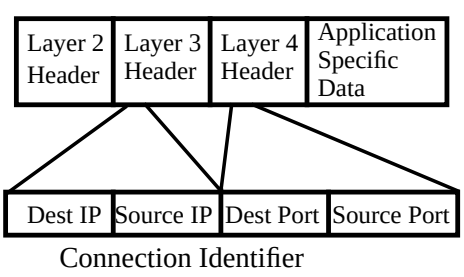
## *"Some Application Layer Stuff"*

- Layer 3 gets packets from here to there (given a four byte IP address)
- Layer 4 makes a reliable and well behaved (wrt network load) connection using timeouts and retransmission
- What's left?
  - Applications must find the "port" of their peer
  - Look up IP address based on a name
  - Infinite variety of application specific kludges

## *Ports*

- TCP and UDP each contain a 2 byte source port and a 2 byte destination port in each packet




- Each connection is independent
- Up to $2^{32}$ independent connections between a pair of computers

## *Port Field is Overloaded*

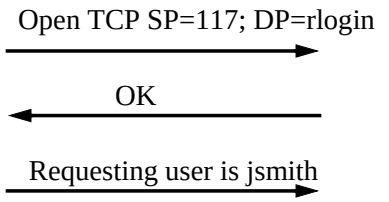
- In practice, can't have $2^{32}$ connections between a pair of computers
- "Well Known Ports" used to find applications without negotiations
- "Privileged Ports" indicate that the corresponding application is running with elevated privileges on its computer
- In many implementations, ports assigned to applications for all IP addresses and all paired ports

## *Well Known Ports*

- Telnet port is #23
- If I open a connection to any system and specify a destination port of 23, it means I want to open a telnet connection
- Implies a single computer can only open $2^{16}$ simultaneous telnet sessions to another computer
- Telnet daemon "listens" on port 23
  - O/S forwards all connect requests on port 23 to that daemon
  - That process "owns" all connections from any port and any IP address to port 23
  - Typically starts a subprocess to handle each active connection (IP address/source port combination)

## Privileged Ports

- Only privileged processes are allowed to open connections from low numbered source ports
- A time sharing system could open connections on behalf of its users and assert user identities as part of the application protocol




- Server is configured with source IP addresses to trust, and what port numbers are low enough to be trusted at that IP address

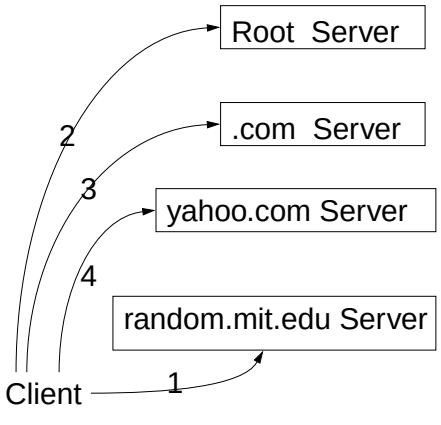
## Domain Name System (DNS)

- Distributed database for looking up 4 byte IP address from names like “www.yahoo.com”
- Control is delegated. If you control yahoo.com, you can define properties for any name of the form *.yahoo.com, *.*.yahoo.com, etc.
- You can in turn delegate control of some sub-domain to someone else

## Tying DNS together

- Every client must know the IP address of some DNS server (sometimes learned through dhcp)
  - Preferably more than one for failover
- Every server must know all information about things in its own part of the namespace (its **zone**)
- A server is either the Primary Master for its zone, or it is a secondary master and knows the IP address of the Primary Master
- Every server must know the IP addresses of servers controlling child subdomains
- Every server must know the IP address of some DNS server for the root
  - Preferably more than one for failover

## Tying DNS together




- Any DNS server can either answer your query or point you to a closer server

## DNS Query Types

- Iterative: if you don't have the data tell me about someone closer
- Recursive: if you don't have the data, look it up for me

## Caching

- After recursive lookup, or any other time, DNS server can cache information from other servers
- All data has a "time to live" **(TTL)** controlling how stale the cache may be
- Client can set "authoritative bit" in query asking server to bypass cache

## DNS Administration

- Initially, fixed small set of **Top Level Domains:**

  - .com, .edu, .gov, .mil, .org, .net, .arpa

- Later added two letter country codes as top level domains

  - .us, .au, .tv, .to, .tm, ...

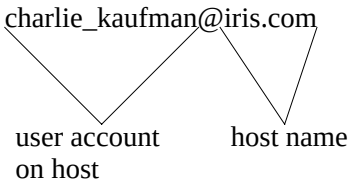
- Initially loosely administered on first come first served basis
- Subject of ongoing political, legal, and economic squabbling

## What information is stored in DNS?

- Primarily IP addresses associated with names (A records)
- Information for tying the system together (e.g. names and IP addresses of other servers)
- With **dnssec**, domain keys and digital signatures
- Application specific extra information

  - Most important is MX records

## Email addressing




- This model was too restrictive:
  - A single host not fast enough or reliable enough to handle all the mail for a large company
  - Hide the user's mail server name
  - Might want mail to go to different machine than the one named in the email address
- Solution: MX record for iris.com has list of machines that accept mail so addressed

## What is "www." for?

- Web faced same scaling problems as mail
- Could have created WX records containing names of web servers for a domain
- Instead adopted convention of using a separate name with prepended www.
- Usage is fading... most sites today accessible with or without the www.

## How does the web work?

- Terminology:
  - Internet: the communications network supporting many applications, including email, remote login, file sharing, etc.
  - World Wide Web: an application of the Internet where information is linked together with URLs (Universal Resource Locators) and displayed using a browser
  - HTTP: HyperText Transport Protocol. A protocol using TCP/IP that browsers use to communicate with web servers
  - HTML: HyperText Markup Language. A very flexible syntax for carrying information including fonts, colors, layout, embedded pictures, and "hotspot" links to other URLs
  - XML: eXtended Markup Language. An even more flexible successor to HTML

## What's in a URL?

- http://www.microsoft.com/security
- "http:" says use http protocol. Alternatives include ftp:
- "www.microsoft.com" is a domain name whose address is looked up in DNS for opening a connection
- "security" is information passed to the server to specify what information is desired
  - initially, this was usually a file name
  - today, it is typically a complex formula with sub-parts that are interpreted by the server
  - e.g. a command to run and parameters to supply to the command

## Elaborate URLs

- http://video.lycos.com/myvideocenter/viewvmail.asp?vm=578650&e=57Uk/Z0JeuvHc&r=1
- “http://” implies port to connect to and protocol to speak
- “video.lycos.com” is looked up to find the IP address
- “myvideocenter/viewmail.asp” might be the name of a program to run
- “vm=578650&e=57Uk/Z0JeuvHc&r=1” is likely a set of parameters passed to the program
- HTML often contains generated URLs that allow the server to maintain context

## Confusing URLs

- general syntax is:
  - <protocol>://<username>@<host>:<port>/<server-specific-parameters>
- What does this mean??
  - http://www.cnn.com&story=breaking_news@18.69.0.44/evarady/www/top_story.htm>

## Web Statelessness

- HTTP is “stateless”
  - Protocol has no concept of a “session”
  - TCP connection often broken and reformed between clicks
  - Designed for navigating static web pages linked together with URLs
  - Applications that want a concept of a session have to build it from dynamically generated URLs, referrals, and cookies
- Referral: if you call up a page by clicking on a URL, the server is told the URL of the page containing the reference
- Cookies: when a page is returned, it can include non-displayed information that will be returned when any other page is requested from the same server

## Constructing a Web Session

- On any request to web site, server looks for a cookie named “login_cookie=” in the request
- If none present, return a page containing a login screen (requesting user name and password)
- When fields filled in and “login” button pushed, generates a request to server containing that info
- Server returns “login_cookie=Er5sleQ” along with the login succeeded page
- Server knows all subsequent requests are associated with this session
- Logout button causes server to set “login_cookie=<null>”

# *Cookies and Privacy*

- With persistent cookies, a site can know when the same user returns on another day
- Cookies only go to the site they came from to protect privacy and security
  - Different sites can’t corrolate actions of a single user
  - Hostile sites can’t get session identifiers and impersonate user back to original server
- But... advertising embedded in web pages allows the advertiser to corrolate actions based on referrals and advertiser cookies
- If you give your name, email address, or other identification to any site, it can be associated with any other sites you visit