The Application Layer

CS158a

Chris Pollett

May 9, 2007.

Outline

- DNS
- E-mail
- More on HTTP

The Domain Name System (DNS)

- To refer to a process on the internet we need to give an IP address and a port.
- These numbers on not very convenient for people to use, ASCII names would be better.
- An Arpanet scheme to solve this problem was to have a file hosts.txt in which host names in ASCII together with IP addresses were listed. Then if a connection was attempted to the ASCII name, a look up of the IP address was done.
- As the internet grew it became impractical for each machine to maintain a complete list of all hosts.
- Instead a hierarchal system called the Domain Name System (DNS) was developed to solve this problem.
- To map a host name onto an IP address an application layer program calls a library procedure known as a **resolver** (for example, InetAddress’ method getHostAddress() in Java).
- The resolver sends a UDP packets to a local DNS server (Port 53), which then looks up the name and address and returns the IP address to the resolver.

More on DNS

- Domain Names are organized into a hierarchy.
  - One has **top level domains** which include com, edu, gov, int, mil, net, org, biz, info, name, and pro and one has country names such as ca.
  - **Second level domain** have the form: some string, period, followed by a top level name. For example, my-company.com
  - Then one has **third level domains** such as my-dept.my-company.com
  - In principle, you can keep going but for most US companies we only consider three levels. For foreign ones you might have 4:www.amazon.co.uk
- Names are case insensitive
- Each domain controls how it allocates names under it. For example, co.uk is a .com in the UK.

Still More on DNS

- Every domain can have a set of **resource records** associated with it.
- The most common resource record is just its IP address, but many other resource records also exist.
- When a resolver gives a domain name to a DNS server what it gets back are the resource records associated with the name.
- A resource records has the format:

  Domain_name Time_to_live Class Type Value
- Thi iis the domain name to which the record applies, how many seconds the record is good for, for internet information the class will always be IN, and the type says what kind of records it is: SOA -(start of authority) parameters for this zone, A - IP address, MX - mail exchange, does host accept email, NS- name server of the domain, CNAME -- domain name, PTR -- alias for an IP address, HINFO -- host CPU and OS, TXT -- anything the register wanted to add

Name Servers

- The DNS name space is divided into non-overlapping **zones**.
- Each zone contains part of the name tree and has name servers holding information about that zone.
- A zone will typically have a primary name server which gets information about that zone on disk, as well as one or more secondary servers.
- Zone boundaries are up the given zones administrators.
- When a resolver has a query it asks a local name server.
- If no information is available locally, the name server sends the query to the top level name server for the domain requested, which can then query its sub-zone until the name is resolved.

E-mail

- E-mail is distributed using a collection of application layer protocols:
  - SMTP (for sending e-mail)
  - POP3, IMAP for reading e-mail
- A user might get e-mail from a mail server through a client application (either a web interface, or a program like Outlook, Thunderbird, Mail, etc). This would use POP or IMAP.
- Messages are sent to mail servers using SMTP. Here what happens is a user’s client will communicate with the user’s mail server which in turn will use SMTP to communicate to the receiver’s mail server.

SMTP

- Stands for Simple Mail Transport Protocol .
  - A sample session with an SMTP server might look like:

```
S: 220 myname.edu
C: HELO client.domain
S: 250 Hello client.domain, pleased to meet you
C: MAIL FROM: bob@client.domain
S: 250 bob@client.domain … Sender ok
C: RCPT TO: sally@myname.edu
S: 250 sally@client.domain … Receiver ok
C: DATA
S:354 Enter mail, end with “.” on a line by itself
C: How about lunch?
C:.
S: 250 Message accepted for delievery
C: QUIT
S: 221 myname.edu closing connection
```

This protocol uses port 25.

Format of Mail Messages

- E-mail messages consist of a header followed by a blank line followed by the contents of the message.
- A E-mail header consists of a sequence of lines the form *something: value* .
- A header must have a From: and To: line.
- An example might be

  From: bob@client.domain

  To: sally@client.domain

  Subject: lunch?

  How about lunch?
- For non-ASCII data one can specify MIME header -- Multipurpose Internet Mail Extension header, such as Content-Type: and Content-Transfer-Encoding:
- The receiver server will often append to the header Received: lines saying when the letter got to the server and whether it was forwarded through any other servers.

POP3

- POP (Post Office Protocol) is a very simple protocol for a mail client to communicate with a mail server to find retrieve the mail for a particular user.
- A POP server listens to port 110.
- A client connecting to such a server might do:

```
telnet mailserver 110
+OK POP3 server ready
user bob
+OK
pass secret
+OK user successfully logged on
```

- The client can then issue commands such as: list - which returns what messages are available and how big they are; retr n -- retrieves message n, dele n -- deletes message n, and quit.

IMAP

- IMAP (Internet Mail Access Protocol) is a more sophisticated protocol which has the same role as POP3.
- Beyond the functionality of POP, it supports marking messages as read, multiple folders, moving messages between folders.

HTTP

- HTTP stand for hypertext transfer protocol
- As we previously demo'd in class to connect to a web server using HTTP:

```
C: telnet www.cs.sjsu.edu 80
S: Trying 130.65.86.46...
     Connected to www.cs.sjsu.edu.
     Escape character is '^]'.
C: GET /index.html HTTP/1.1
   Host: www.cs.sjsu.edu
     [blank line]
S: HTTP/1.1 200 OK
Date: Thu, 10 May 2007 18:37:34 GMT
Server: Apache/2.2.2 (Fedora)
Last-Modified: Mon, 07 May 2007 22:38:22 GMT
ETag: "f5678-25e4-f558d380"
Accept-Ranges: bytes
Content-Length: 9700
Connection: close
Content-Type: text/html
[blank line]
document
```

HTTP supports other commands such as POST, HEAD, etc

This is an example header line which has

The format *field name: value*

In general, can have several of these

The first line back says a status code for the

Response 200 means the request was

Successful. This is followed by the

response headers, a blank line and the

Returned web document.