Föreläsning 9 – del 2
Distribuerade applikationer

Sven Karlsson
svenka@it.kth.se
08-752 1375

FTP

- Used to transfer large files
- The nodes communicate with text strings much like SMTP and HTTP
- The server holds all files like a web server
- The client connects to the server
- Each user must log on to the server, i.e., he/she must type a login name and a password
- Files can be transfered as ASCII files or binary files. You should normally only use the binary mode as files might otherwise be corrupted.
  - ASCII mode will convert between different encodings

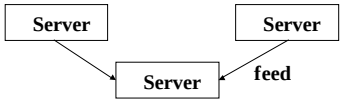
FTP session

Connect to host S, port L,
  220 Service ready <CRLF>
USER Doe<CRLF>
  331 User name ok, need password<CRLF>
PASS mumble<CRLF>
  230 User logged in<CRLF
RETR test.pl1<CRLF>
  150 File status okay; about to open data connection<CRLF
  226 Closing data connection, file transfer successful<CRLF>
QUIT <CRLF>

NNTP

- Is the protocol behind news which is used for discussion groups. Web based chat rooms has become popular lately but news is much more powerful.
- Again, NNTP is based on ASCII strings and is a client-server protocol just like SMTP.
- NNTP can also be used between servers but *news articles* are commonly sent between servers using some other protocol and using special connections, so called *feeds*.

NNTP servers



- Servers are organized in network where articles are fed from server to server. Each server gets fed from normally at least two servers. Here NNTP might not be used. Duplicate articles are removed.
- Clients use NNTP to communicate with servers. The users subscribe to *newsgroups*, i.e., discussions.

NNTP example

(requests connection on TCP port 119)

200 wombatvax news server ready - posting ok<CRLF>

LIST<CRLF>

215<CRLF>

net.wombats 00543 00501 y<CRLF>

net.unix-wizards 10125 10011 y<CRLF>

.<CRLF>

GROUP net.unix-wizards<CRLF>

211 104 10011 10125 net.unix-wizards group selected<CRLF>

NNTP example II

STAT 10110<CRLF>

223 10110 <23445@sdcsvax.ARPA> article retrieved - statistics only<CRLF>

HEAD<CRLF>

221 10110 <23445@sdcsvax.ARPA> article retrieved - head follows (text of the header appears here)<CRLF>

. <CRLF>

NNTP example II

BODY

222 10110 <23445@sdcsvax.ARPA> article retrieved - body follows (body text here) <CRLF>

.<CRLF>

QUIT<CRLF>

205 goodbye.

Telnet

- Back in the old days, i.e., ten years ago. People used so called terminals to connect to mainframes.
- Most terminals had screens and keyboards and allowed you to run applications just like you do in ”xterm” windows under X. However, without any fancy graphics.
- When you went to a terminal you had to state your user name and type a password. Just like you do on most UNIX systems and also some other operating systems.
- When you used a terminal you used a *terminal session*.

Telnet II

- Telnet allows you to open a terminal session to a possibly remote system.
- Telnet opens a plain TCP connection to the remote system and each character you write will be sent to that remote system.
- Each character written to the screen will be sent to you.
- You and the remote system belive you are sitting in front of a terminal directly connected to the remote system.

Opening a telnet session

- While opening the TCP connection the telnet application and the remote computer might send information to each other as to what type of terminal is used.
- The telnet application can often emulate a few different types of terminals.
- No specific ”telnet” protocol is used after the connection is setup. The telnet application behaves as a terminal!
- Both parties can end a telnet session.

Telnet session

The Telnet application open a TCP connection

It sends information as to which terminal it is going to be

The remote computer can also send settings

(After this point anything can happen! However, you usually have to log on to the remote system with a login name and a password.)

Rlogin

- Rlogin is similar to telnet in that the remote system belive you are on a terminal. However, you never need to log on to the remote system with passwords. The remote system trust the computer/system you are using.
- Rlogin sets up a TCP connection and tell the remote system who you are.
- The remote system might choose to not accept you!
- Rlogin is a major security hazard and you are normally not allowed or even able to use it.

Sample Rlogin session

<null> ***(Note: no CRLF!)***
client-user-name<null>
server-user-name<null>
terminal-type/speed<null>

For example:
<null>
bostic<null>
kbostic<null>
vt100/9600<null> *(after this anything goes)*

DNS - RFC 1034

- You don't want to remember IP addresses to surf the net. You want to use symbolic names such as www.aftonbladet.se!
- DNS is a service that translates symbolic names into IP addresses

Note that the previously release lecture notes have a typing error. They state RFC 1084 instead of 1034

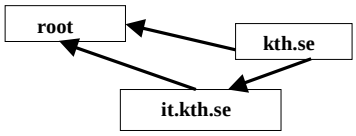
DNS

- DNS is hierarchial and uses, once again, an ASCII string based client-server protocol.
- Each organisation usually have at least one DNS server and the symbolic name can give clues:
  - Normally each ”.” corresponds to a organisation, a *domain*.
  - it.kth.se has its own dns server
  - kth.se of course also has its own dns server
  - One single set of DNS servers can manage multiple domains.
- The DNS servers are linked to each other
  - kth.se knows about it.kth.se but it.kth.se does not necesserily know about kth.se

Network of DNS servers

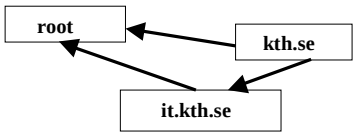
- Each organisation can have at least two DNS servers at each level which in turn can know any number of DNS servers
- When you need to know the IP number of a symbolic name you ask a DNS server which in turn could ask another DNS server.
- DNS servers remember the most recent queries but otherwise only knows about its own domains.

Network of DNS servers II



- Root DNS servers are special. They are DNS servers that know about .com .edu etc. domains.
  - Each DNS server should have a list of root servers
  - The hierarchial system is used for performance reasons

Network of DNS servers II



- kth.se knows about it.kth.se. it.kth.se should know at least one root node
  - If kth.se gets a query for sardine.it.kth.se it will return it.kth.se not sardine.it.kth.se
  - it.kth.se will ask the root server if it gets a query it doesn't know anything about

DNS queries

- There are two types of queries
  - Recursive
    - This is the normal query type. If a DNS server has to ask some other DNS server then it is a recursive query. If, for instance, it.kth.se gets a query for www.aftonbladet.se it will have to ask a root server for se. se for aftonbladet.se and then aftonbladet.se for www.aftonbladet.se. The query is a recursive query
  - Non-recursive
    - A DNS server can immediately answer a non-recursive or iterative query. If it.kth.se was asked for sardine.it.kth.se, it would be a non-recursive query
- You can also ask the DNS for the symbolic name of an IP address

DNS II

- A DNS server should know about several root servers as some of them might not work
  - Each node using IP should also have a long list of DNS servers so that at least one could be working
- However, there are no guarantees
  - Not long ago the DNS in large portions of the US crashed
  - Microsoft's DNS was cracked

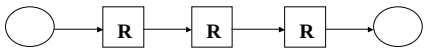
Problems with DNS

- Fragile
  - You cannot know for sure that any DNS server you know will actually be able to process your queries. They might not be operational.
- Stale IP addresses
  - Once inserted into the system an IP address can be *cached* or stored at numerous servers. The DNS protocols used between servers should update the IP addresses if they ever change but this do not always work!

Traceroute

- Traceroute is an application that tries to find out what route a packet might take on the Internet.
- Traceroute sends several datagrams to the destination with different number of hops in the TTL field of the IP header.
- The routers that drops the packets sends back a ICMP packet saying that they dropped the packet.

Traceroute II



- A ICMP datagram is sent when the TTL becomes 0.
- This can be used to find out the identity of each router and the round-trip-time to each of the routers.

Summary

- ISDN
- ATM
- ADSL
- ”Cable modems”
- Distributed applications

In the textbook: Parts here and there and ch. 11, 19 Not everything said in this lecture is in the book

Next time: Security and related topics. Ch 18