Computer Networks and Distributed Systems

Application Layer

Dr Fidelis Perkonigg

March 15, 2018

Overview

- Application Level Protocols
- Client & Server and Peer-to-Peer Model
- Domain Name Service (DNS)
- World Wide Web (WWW) and URLs
- HTTP, SMTP, POP, IMAP

Client and Server Model

Client

- Often user-invoked application running on local machine
- Initiates connection to server
- Uses service/resource from server
- E.g. web browser, e-mail clients, chat client
- Resource use usually temporary

Server

- Often started at boot time (as a "daemon" in UNIX or "service" in Windows)
- Waits for connections
- Can handle multiple clients
- Provides controlled access to resources/services
- E.g. web server, e-mail server, chat server

Peer-to-Peer Model

- Hosts offering resources/services to one another
- A centralised server may coordinate
- E.g. internet telephony, agent systems, file sharing (BitTorrent)
- Often implemented as client and server on same host

Issues with IP Addresses

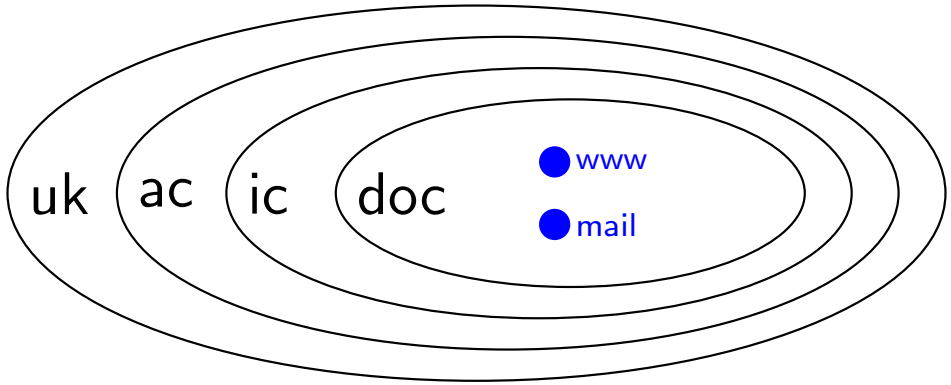
- 146.179.40.24
  - Not very memorable
  - IP addresses of a machine may change (e.g. moved to another network)
- Solution: Assign user-friendly names to IP addresses
- E.g.: www.imperial.ac.uk instead of 146.179.40.24
- Aliases possible

Mapping Names to IP Addresses

- Local file
  - Initially used in early ARPANET
  - See /etc/hosts on GNU Linux
  - Difficult to maintain
- Centralised server
  - Clients query database to perform name lookup/resolution
  - Single point of failure; hard to maintain and administer
- Distributed look-up system
  - Domain Name System (DNS)

Domain Name System (DNS)

- Internet is inter-network of autonomous networks
  - Must support independent administration of names
  - Must avoid conflicts between names
- DNS names form hierarchies (e.g. doc.ic.ac.uk):



- Requires uniqueness of complete name only
  - www.doc.ic.ac.uk different to www.ee.ic.ac.uk

Domain Names Management

uk	United Kingdom (country ID)
ac	Academic network (standard sub-domain within UK)
ic	Imperial College (assigned by UK academic net administration; owned by Imperial College)
doc	Department of Computing is assigned by College admin

- Top-level structure conveys meaning
- Local domains follow organisational structure
- Registrars of top level domains: ICANN[1] and IANA[2]
- Within each domain, naming managed independently

[1] Internet Corporation for Assigned Names and Numbers
[2] Internet Assigned Numbers Authority

www Machine in DoC is assigned by local administrator

mail Machine in DoC is assigned by local administrator

- Names may reflect what service the machine provides
  - e.g. www, mail, cate, ...
- Type of name may reflect what type of machine
  - E.g. name of birds are servers, colours are printers, towns are workstations

Top Level Domains

Domain	Specification	Example
com	Commercial	cisco.com
net	Network providers	internic.net
edu	Educational institution	mit.edu
gov	Government organisation	whitehouse.gov
mil	US Military organisation	navy.mil
org	Other organisation	linux.org
country code	Domain administered by country	de, es, fr, uk, cn, us
2nd level country domains	Sub-domains administered by organisation for that country	ac.uk, co.uk, gov.uk

1. Host asks a name server to resolve domain name
2. Host or name server might have to ask other name servers
3. Name server answers host

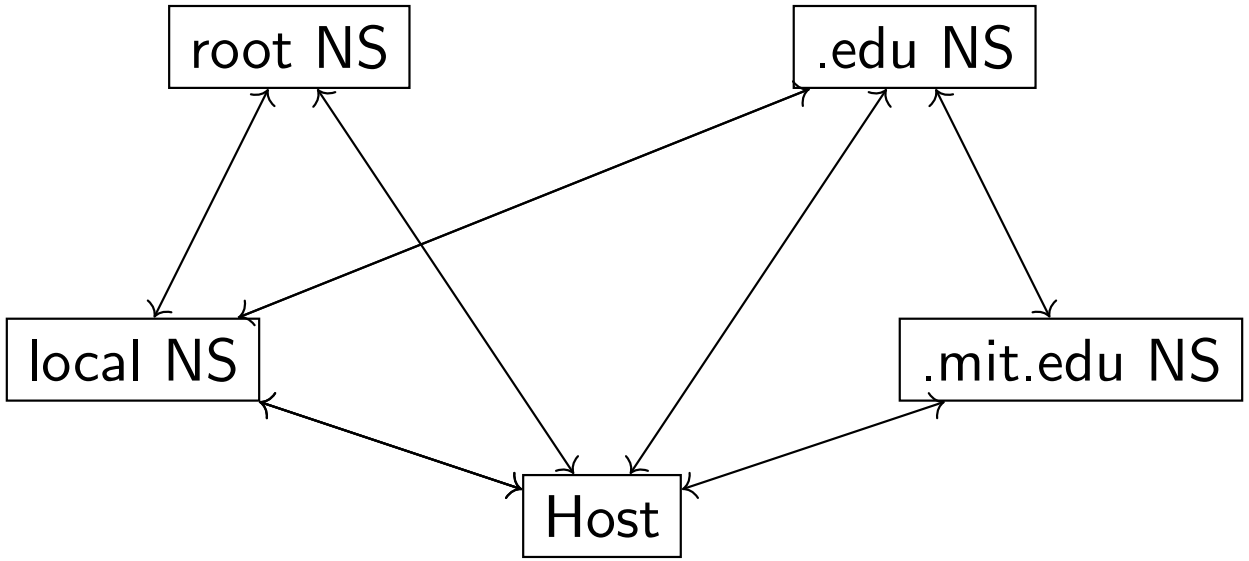
- Host knows (local) name server
  - Configured statically or dynamically using DHCP
  - Name server known by its IP address
  - Usually at least 2 local name servers to avoid single point of failure

Types of DNS Records

A	Name to IP address mapping
MX	Name of mail server for domain
NS	Name of name server for domain
CNAME	Aliase names
PTR	IP address to name mapping (for reverse lookups)
RRSIG	DNSSEC Signature

DNS lookup

Example: www.eecs.mit.edu



- 13 root name servers[3] (labeled A - M) know location of top-level servers
- Iterative vs recursive query
- Name servers do not have to support recursive lookups
- DNS uses UDP
- Local NS knows about you

[3] http://www.root-servers.org

- A lot of messages just to figure out where to connect to
  - DNS can be a bottleneck for some applications (e.g. the Web)
  - Critical point of failure
- Servers cache responses as they are often needed again
- Cached answers are non-authoritative[4] and may be wrong, out-of-date
  - DNS entries give TTL based on volatility
  - Caches expire records after their TTL has expired
  - Stable names can safely be cached for much longer
- Troubleshooting DNS: clients such as *dig* and *nslookup*

[4] Authoritative answers are returned from a name server managing that domain

World Wide Web (WWW)

- Developed by Tim Berners-Lee (CERN in 1989)
- Core application that uses the Internet
- Supports transfer and display of documents
- Documents include multimedia content and hyperlinks
- Client/server based
  - Clients (browser): Firefox, Chrome, Chromium, Explorer, Opera, vimb, lynx, ...
  - Web servers: Apache, IIS[5], nginx, ...

[5] Internet Information Server

Web Standards

- HyperText Transfer Protocol (HTTP)
  - Used by browsers to get resources from servers
  - World Wide Web Consortium (W3C) and IETF
- HyperText Mark-up Language (HTML)
  - Describes structure of web pages
  - Supports text with images, formatting, and hyperlinks
  - Cascading Style Sheets (CSS) and JavaScript
- Uniform Resource Identifier (URI)
  - A string of characters to identify a resource
- Uniform Resource Locator (URL)
  - Is a URI
  - Used to identify resources on servers, their location, and how to obtain them

Uniform Resource Locators (URLs)

`scheme://user:password@host:port/path?query#fragment`

scheme	Protocols such as http(s), ftp, news, mail, file, ...
host	IP address or domain name with optional port number and access credentials
path	Path to resource on the host which may resemble a file system path
query	An optional sequence of key-value pairs
fragment	An optional identifier to a secondary resource such as a section heading

```
http://www.imperial.ac.uk/computing/
http://www.imperial.ac.uk:80/computing/
https://en.wikipedia.org/wiki/Uniform_Resource_Locator#Syntax
https://duckduckgo.com/?q=uniform+resource+locators
```

Hypertext Transfer Protocol (HTTP)

- HTTP server usually listens on port 80
- Uses TCP
- Stateless, transaction-oriented protocol
  1. Client opens connection to server
  2. Request sent from client to server
  3. Server responds
  4. Connection closed

- Request format
  - Request line (method, identifier, version)
  - Header (additional info)
  - Body (data)
- Example:

```
GET /~fp910/computer_networks.html HTTP/1
Host: www.doc.ic.ac.uk
User-agent: Mozilla/5.0
Accept-Language: en-GB
```

Request line
Zero or more header lines
Blank line
Object body (possibly empty)

/~fp910/external/computer_networks.html

HTTP Methods

- **GET**: Client requests resource from server
  - No permanent action on server is implied
  - Most common method
- **POST**: Append/send data to named resource
  - Used to submit client data from web forms
- **HEAD**: Requests only header of web page
  - Useful when deciding if changed and to test validity of links
- Others: PUT, DELETE, LINK, UNLINK

- Reply format
  - Status line (version, code, optional message)
  - Header (additional info)
  - Body (data, MIME compatible)
- Example:

HTTP/1.1 200 OK	Status line
Date: Fri, 19 Aug 2017 13:53:49 GMT Server: Apache Last-Modified: Fri, 19 Aug 2017 10:51:38 GMT Content-Length: 154 Content-Type: text/html	Zero or more header lines
	Blank line
	Object body (possibly empty)

1xx	Informational
2xx	Successful operation: OK (200), Accepted (202), No response (204)
3xx	Redirection: Moved permanently (301), Not modified (304)
4xx	Client error: Bad request (e.g. syntax error) (400), Unauthorised (401), Payment required (402), Not found (404)
5xx	Server error: Service unavailable (503), HTTP version unsupported (505)

- HTTP protocol is stateless
  - Server does not keep track of client requests, which simplifies server design
  - But websites would like to keep track of customers
- HTTP/1.0 uses non-persistent connections
  - New TCP connection opened for every request
  - Page with 20 images will open 21 TCP connections
  - Adds server load, setup costs, and delays
  - TCP congestion control inefficient for short transfers

- Websites want to identify users (e.g. logins, user preferences, recording user behaviour)
- Persistent information through cookies
- Data sent by web server and stored by web browser
- Name/value pair with expiry time
- Set-Cookie header in HTTP response
- Allows server application to maintain state between HTTP requests
- IP addresses not practical identifiers for users due to NAT and IP sharing

Persistent Connections
HTTP/1.1

- Persistent connections
  - Client opens TCP connection
  - HTTP requests pipelined through this connection
  - Multiple requests with one TCP overhead for connection establishment
- Backwards compatible and now in widespread use

Electronic Mail

Movement of structured text messages between systems

- Features
  - Asynchronous communication
  - One-to-many communication
  - Multi-media content as attachments
- Limitations
  - Messages can be accidentally lost or intentionally blocked
  - Basic system has no authentication and confidentiality
    - Messages can be modified or forged
    - The message can be read by others
    - Can be overcome by using secure connections to mail servers and public key encryption

To:, Cc:, Sender:, Reply-to:	E-mail addresses
From:	Who sent the message
Received:	List of mail servers that processed mail
Subject:	Short summary of message
Message-Id:	Unique ID for message, which helps to detect duplicate messages, responses

- E-mail address: <user name>@<domain name>
- Mail server for domain found through DNS MX records

E-mail Subsystems

- Mail User Agents (MUA)
  - E-mail client for composition, display, filing, ...
  - E.g. Thunderbird, Outlook, mutt, K9, ...
- Message Transfer Agents (MTA)
  - Mail server that handles message transfers
  - Holds (spools) mail if destination is unreachable
  - E.g. sendmail, postfix, exim, Exchange, ...
- Mail Delivery Agent (MDA)
  - Stores messages for the recipient
  - Many MTA come with an MDA
  - E.g. procmail, fetchmail
- Example: MX records and typical e-mail body

Simple Mail Transfer Protocol (SMTP)

- Protocol[6] used by MTAs and MUAs to deliver mail
- Two parties (client and server) communicate using TCP and port 25
- Three basic steps:
  1. Start transaction (HELO, MAIL FROM)
  2. Exchange data (RCPT TO, DATA, .)
  3. Complete transaction (QUIT)

[6] RFC 821 and 5321

```
# telnet mx1.cc.ic.ac.uk 25
Trying 155.198.5.151...
Connected to mx1.cc.ic.ac.uk.
Escape character is '^]'.
220 mx1.cc.ic.ac.uk ESMTP Exim 4.86 Wed, 27 Jul 2016 13:59:14 +0100
HELO imperial.ac.uk
250 mx1.cc.ic.ac.uk Hello dromio.doc.ic.ac.uk [146.169.24.15]
MAIL FROM: just.testing2050@gmail.com
250 OK
RCPT TO: fp910@imperial.ac.uk
250 Accepted
DATA
354 Enter message, ending with "." on a line by itself
From: someone@sb.com
To: Fidelis Perkonigg <f.perkonigg10@imperial.ac.uk>
Subject: Course Test
body text
.
250 OK id=1bSORK-0008T4-R9
QUIT
221 mx1.cc.ic.ac.uk closing connection
Connection closed by foreign host.
```

Received Mail
Example

Multipurpose Internet Mail Extensions (MIME)

- The standard message format has some serious limitations
  - Only text
  - 7-bit (text) content
  - Essentially good exclusively for the English language
- Any content that does not fit the 7-bit (ASCII) character set must be encoded (e.g. base64)
- The MIME specification[7] defines useful extensions including new content-types:
  - text/plain - this is a normal ASCII message
  - text/html - this is an HTML-formatted message
  - image/jpeg - this message contains (only) an image file
  - multipart/mixed - this message consists of multiple parts

[7] RFC 204{5,6,7,8,9}

Mail Body
Example

Post Office Protocol (POP)

- SMTP designed for permanently available hosts
  - E.g. internet mail servers
  - SMTP delivers mail to mailbox at mail provider (often ISP)
  - Not usually used to deliver mail to users desktop
- Post Office Protocol allows user access to mailbox
  - POP client (MUA) connects to mailbox (POP) server
  - Client connects to server (port 110)
  - POP authenticates user (with password)
  - Mailbox downloaded for processing
- POP only supports downloads of Inbox and usually deletes retrieved mail from server

Internet Message Access Protocol (IMAP)

- Designed to leave e-mails on mail server
- Designed to handle multiple folders
- Useful when using multiple MUAs or machines
- Other features
  - Partial downloads of e-mails/attachments
  - Offline mode when unconnected
  - Server-side searches
- IMAP is more complex and adds server load

POP vs. IMAP

Feature	POP	IMAP
Where is e-mail stored	Users PC	Server
Where is e-mail read	Off-line	On-line
Connect time required	Little	Much
User of server resources	Minimal	Extensive
Multiple mailboxes	No	Yes
Who backs up mailboxes	User	ISP
Partial message downloads	No	Yes
Are disk quotas an issue?	No	Could be in time
Simple to implement	Yes	No

Further Information

A lot of information and tutorials can be found on the World Wide Web Consortium (W3C) web page[8]

[8] `www.w3.org`