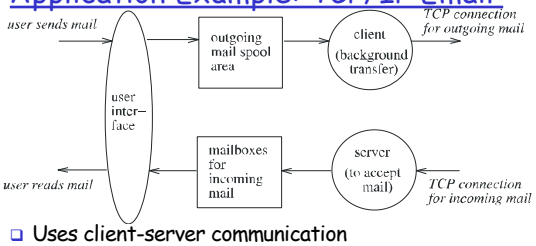
Application Example: TCP/IP Email



- Uses client-server communication
- Not interactive; transfer of e-mail messages takes place in the background (``spooling'')
- Reliable service; uses TCP (server port 25)

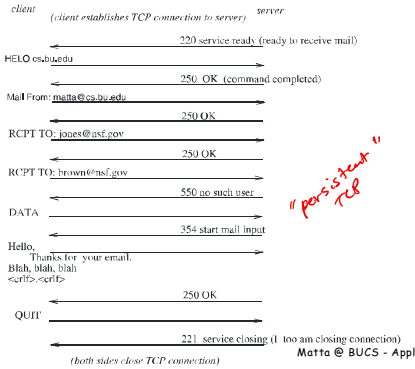
45

TCP/IP Email's Standards

- RFC 822
  - specifies the format of the e-mail message
  - header and body separated by a blank line
  - header consists of keyword-value pairs: To:, FROM:, ...
- SMTP (Simple Mail Transfer Protocol)
  - command-response protocol that defines the exchange of mail between client and server
- MIME (Multipurpose Internet Mail Extensions)
  - Encodes arbitrary data (e.g. binary image) in plain ASCII text
  - SMTP supports only ASCII messages

46

SMTP Client-server Commands



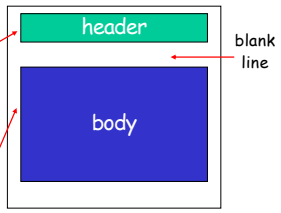
47

Mail message format

RFC 822: standard for text message format:

- header lines, e.g.,
  - To:
  - From:
  - Subject:

  different from SMTP commands!
- body
  - the "message", ASCII characters only



48

Message format: MIME extension

- MIME: Multipurpose Internet Mail Extensions, RFC 2045, 2056
- additional lines in message header declare MIME content type
- A message can have multiple parts, eg text, image, etc.

MIME version

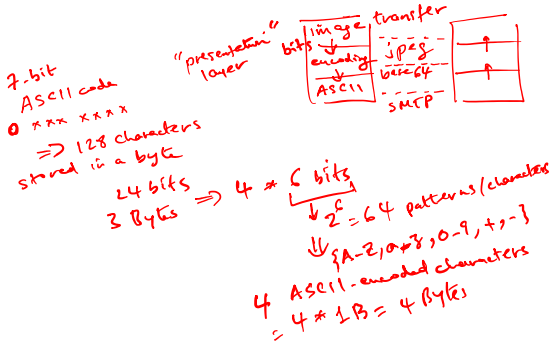
method used to encode data

multimedia data type, subtype, parameter declaration

encoded data

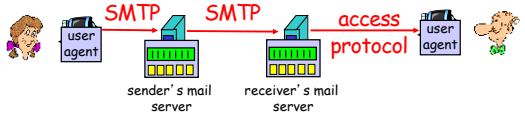
```
From: alice@crepes.fr
To: bob@hamburger.edu
Subject: Picture of yummy crepe.
MIME-Version: 1.0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

base64 encoded data .....
.........................
......base64 encoded data
```


49



50

Mail Access Protocols



- SMTP: delivery/storage to receiver's server
- Mail access protocol: retrieval from server
  - POP: Post Office Protocol [RFC 1939]
    - authorization (agent <-->server) and download
  - IMAP: Internet Mail Access Protocol [RFC 1730]
    - more features (more complex)
    - manipulation of stored msgs on server
  - HTTP: gmail, Hotmail , Yahoo! Mail, etc.

51

POP3 protocol

authorization phase

- client commands:
  - `user`: declare username
  - `pass`: password
- server responses
  - `+OK`
  - `-ERR`

transaction phase, client:

- `list`: list message numbers
- `retr`: retrieve message by number
- `dele`: delete
- `quit`

```
S: +OK POP3 server ready
C: user bob
S: +OK
C: pass hungry
S: +OK user successfully logged on
```

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

POP is a "state-less" protocol

52

POP3 (more) and IMAP

POP + state ∨ session layer ∨

More about POP3

- Previous example uses "download and delete" mode
- Bob cannot re-read e-mail if he changes client
- "Download-and-keep": copies of messages on different clients
- POP3 is stateless across sessions

IMAP

- Keep all messages in one place: the server
- Allows user to organize messages in folders
- IMAP keeps user state across sessions:
  - names of folders and mappings between message IDs and folder name

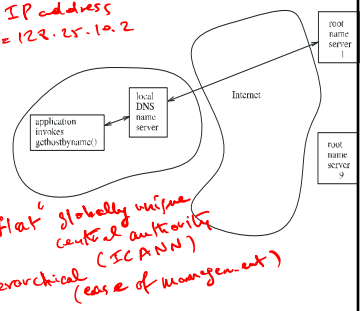
53

DNS: Internet Domain Name System

foo.cs.bu.edu → IP address = 128.25.10.2
Internet name
name resolution

1) how names are structured?
2) how to maintain this mapping info & query efficiently

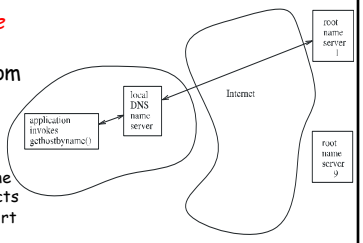
names: "flat" globally unique centrally authority (ICANN)
hierarchical (ease of management)



54

DNS: Internet Domain Name System

- a *distributed database* used by TCP/IP applications to map from hostnames to IP addresses
- *name servers*
  - user-level library routine **gethostbyname()** contacts local name server via port 53
  - name server returns IP address of requested hostname



55

Why geeks say Ted Cruz is wrong about ICANN

Sen. Ted Cruz is pushing hard for Congress to prevent Obama from relinquishing US authority over the Internet body, citing concerns that it will give China and Russia greater flexibility to censor free speech.

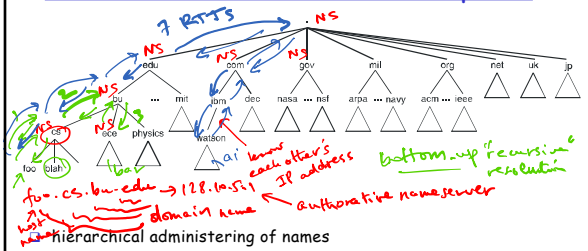
By Laurent Belsie, Staff writer | SEPTEMBER 24, 2016

Save for later

Darron Cummings/AP/File | View Caption

56

The Internet Domain Name Space



- hierarchical administering of names
  - e.g. foo.cs.bu.edu
  - leftmost name is typically a host name (has an IP address)
  - next leftmost name is organization administering that host (e.g., cs)
  - next leftmost name is organization administering all of subnames to the left (e.g., bu administers cs, ece, physics domains)

57

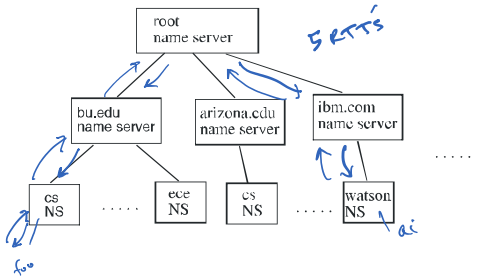
The Internet Domain Name Space (cont'd)

- rightmost (highest) domain is organization, structure, country

domain	*usage*	*example*
com	business	watson.ibm.com
edu	educational	cs.bu.edu
gov	US non-military gov't	nasa.gov
mil	US military	arpa.mil
org	non-profit organization	acm.org
jp	Japan	osaka-u.ac.jp

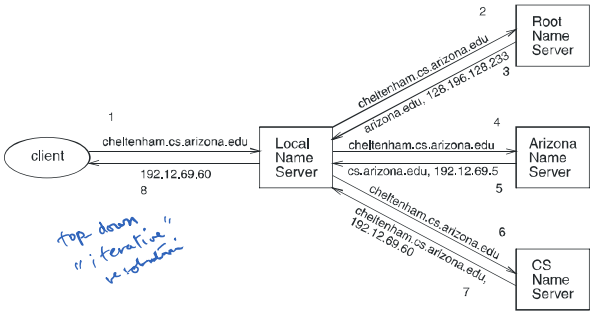
58

DNS: resolving names



59

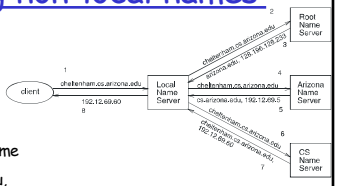
DNS: resolving non-local names



60

DNS: resolving non-local names

- no single name server has complete information
- if local name server can't resolve address, contacts root name server:
  - currently, 13 root name servers world-wide
  - each has addresses of name servers for all level-two name servers (e.g., bu.edu, ibm.com)
  - contacted root server returns IP address of name server which should be contacted next
  - contacted level-two name server may itself return a pointer to another name server



- name resolution for the local name server is an iterative process of following name server pointers
- DNS protocol specifies packet formats for exchanges with DNS servers

61

DNS Caching

- Once (any) name server learns mapping, it *caches* mapping
  - cache entries timeout (disappear) after some time
  - marked as "non-authoritative" mapping with address of authoritative server

62

DNS Records

DNS: distributed db storing resource records (RR)

RR format: **(name, value, type, ttl)**

- Type=A
  - **name** is hostname
  - **value** is IP address
- Type=NS
  - **name** is domain (e.g. foo.com)
  - **value** is name of authoritative server for this domain
- Type=CNAME
  - **name** is alias name for some "canonical" (the real) name `www.ibm.com` is really `servereast.backup2.ibm.com`

  Or `ibm.com` is really `www.ibm.com`
  - **value** is canonical name
- Type=MX
  - **value** is name of mail server associated with **name**