COSC264
Introduction to Computer Networks and the Internet

Email, DNS

Dr. Barry Wu
Wireless Research Centre
University of Canterbury
barry.wu@canterbury.ac.nz

Review of Previous Lecture

- Web and HTTP
  - Non-persistent & persistent
    - Pipeling
  - Messages, cookies
  - Web cashing

Outline

- Electronic Mail
  - SMTP, POP3, IMAP
- DNS

Email

- Invented by Ray Tomlinson, first used in 1960s [wiki];
- One of the Internet's most important killer applications to date!
- The latest RFC is RFC5321.

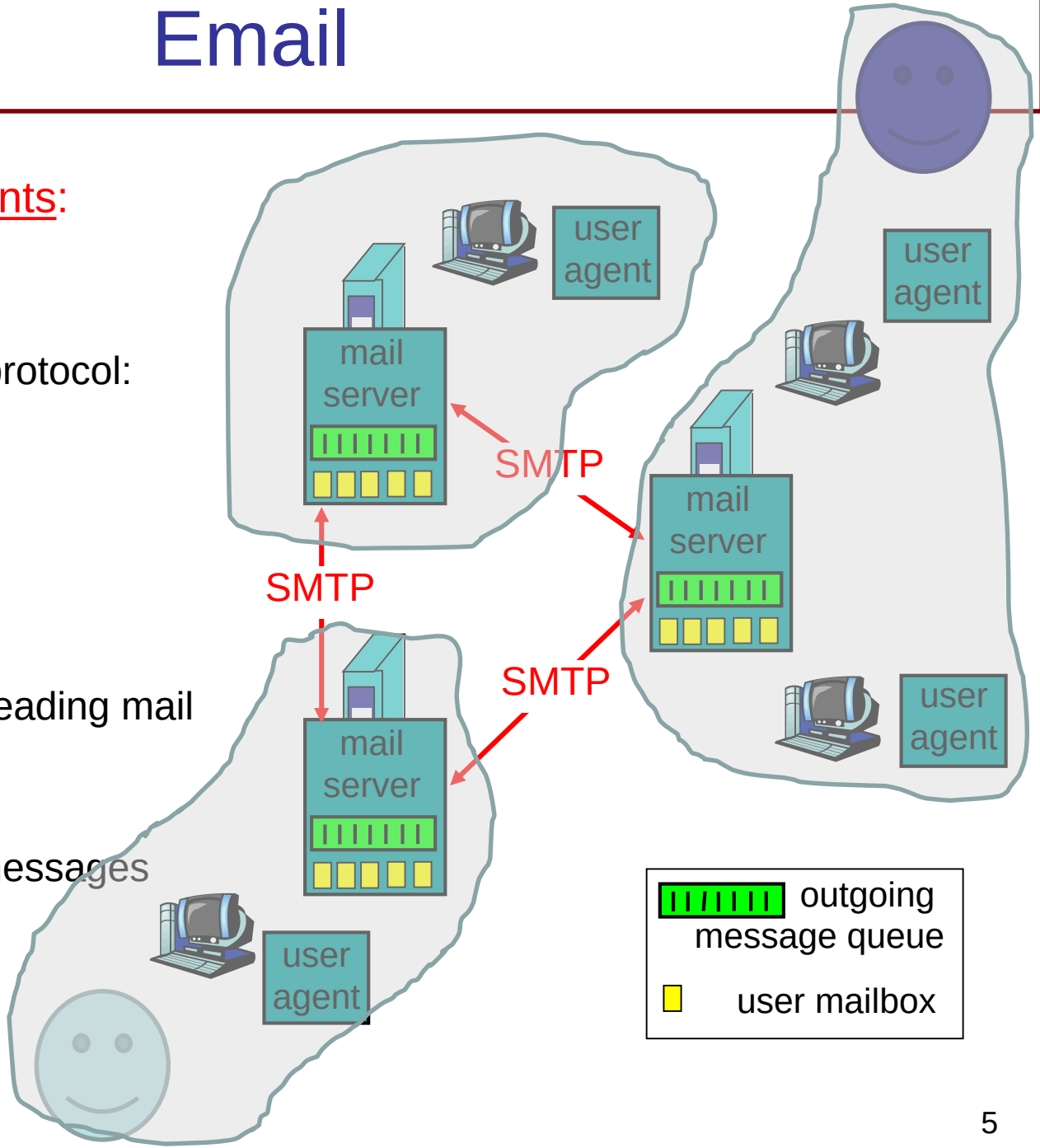
Email

Three major components:

- user agents
- mail servers
- simple mail transfer protocol: SMTP

User Agent

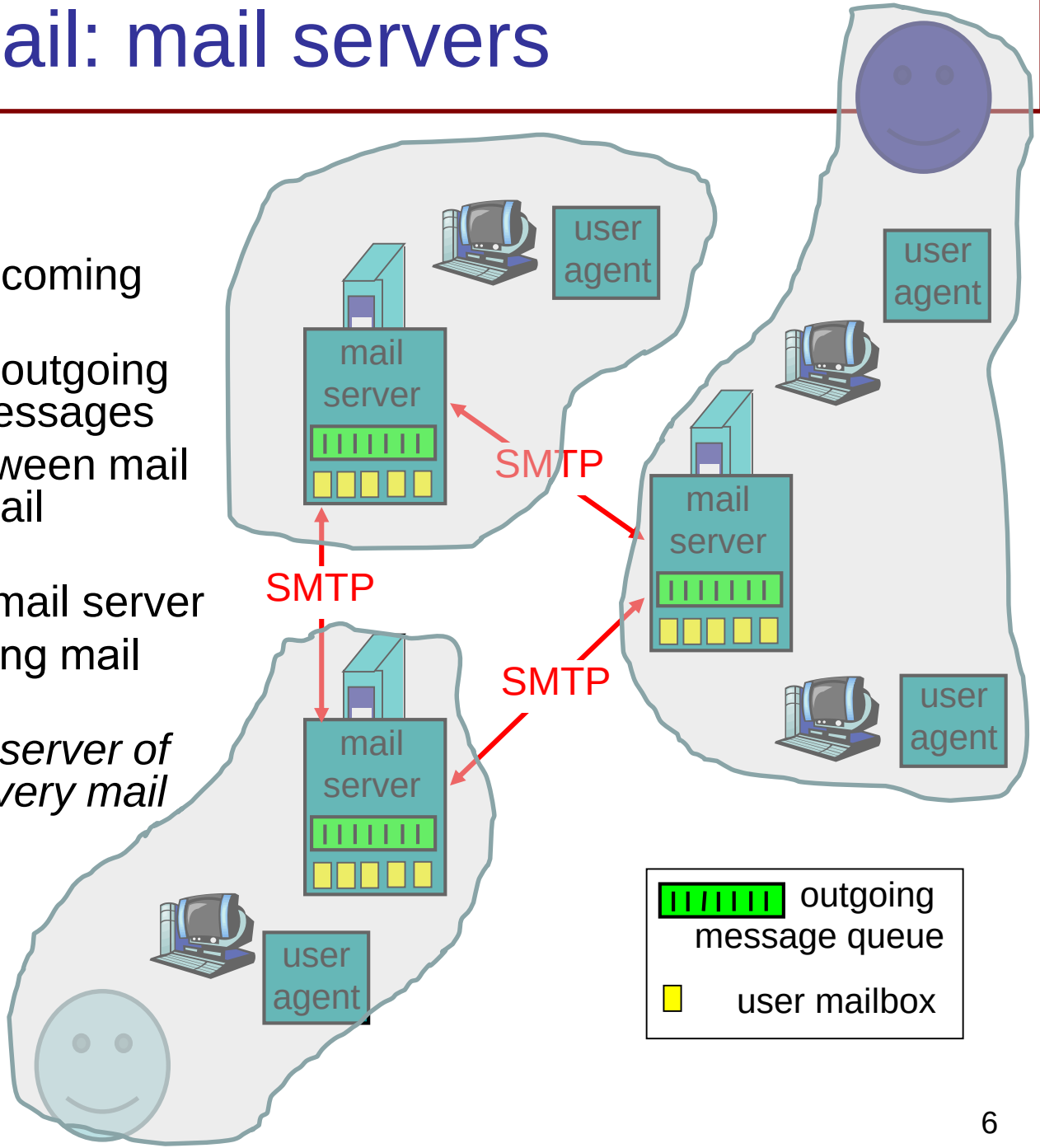
- a.k.a. “mail reader”
- composing, editing, reading mail messages
- e.g., Outlook
- outgoing, incoming messages stored on server



Email: mail servers

Mail Servers

- mailbox contains incoming messages for user
- message queue of outgoing (to be sent) mail messages
- SMTP protocol between mail servers to send email messages
  - client: sending mail server
  - "server": receiving mail server
  - *Both client and server of SMTP run on every mail server.*



Basic operation of SMTP

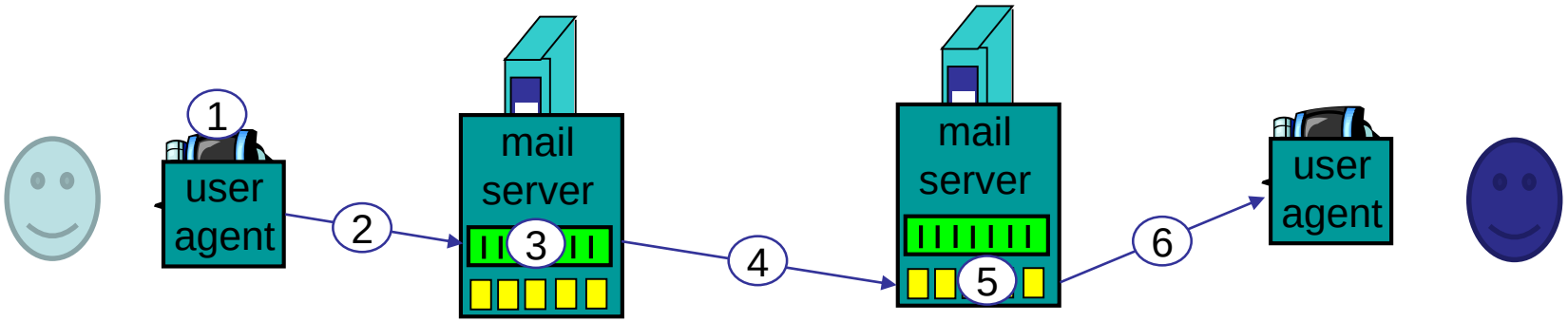
1) Alice uses UA (user agent) to compose message and "to" `bob@someschool.edu`
2) Alice's UA sends message to her mail server; message placed in message queue
3) Client side of SMTP opens TCP connection with Bob's mail server

   -> there are no intermediate servers!

4) SMTP client sends Alice's message over the TCP connection

   -> if there are more messages – they are sent via a persistent TCP connection

5) Bob's mail server places the message in Bob's mailbox
6) Bob invokes his user agent to read message



A *broken* dialogue with an SMTP server

```
duser@192.168.88.155:~/libbgpdump-1.4.99.11$ telnet exchange.canterbury.ac.nz 25
Trying 132.181.107.25...
Connected to exchange.canterbury.ac.nz.
Escape character is '^]'.
220 UCEXHUBCAS01-D.canterbury.ac.nz Microsoft ESMTP MAIL Service ready at Fri, 9
 Aug 2019 16:54:01 +1200
HELO
250 UCEXHUBCAS01-D.canterbury.ac.nz Hello [10.34.40.169]
MAIL FROM:<example@example.com>
530 5.7.1 Client was not authenticated
Connection closed by foreign host.
duser@192.168.88.155:~/libbgpdump-1.4.99.11$
```

Comparison with HTTP

- HTTP: pull; SMTP: push
- both use persistent TCP connections
- both have ASCII command/response interaction, status codes

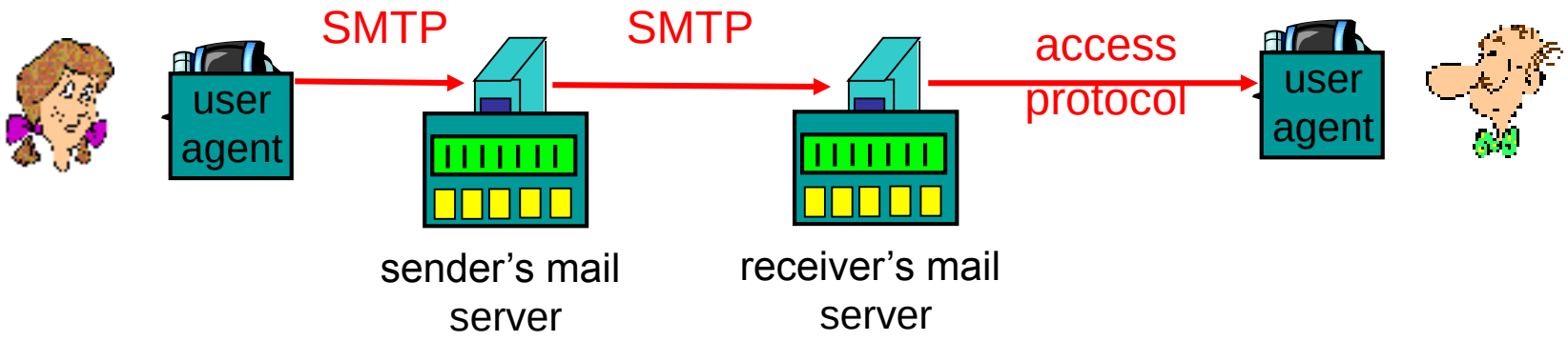
Handling documents with text and images:

- HTTP: each object encapsulated in its own response msg
- SMTP: multiple objects sent in multipart msg

The MIME extension for Non-ASCII data

- MIME
  - Multipurpose Internet Mail Extension

SMTP and Mail access protocols

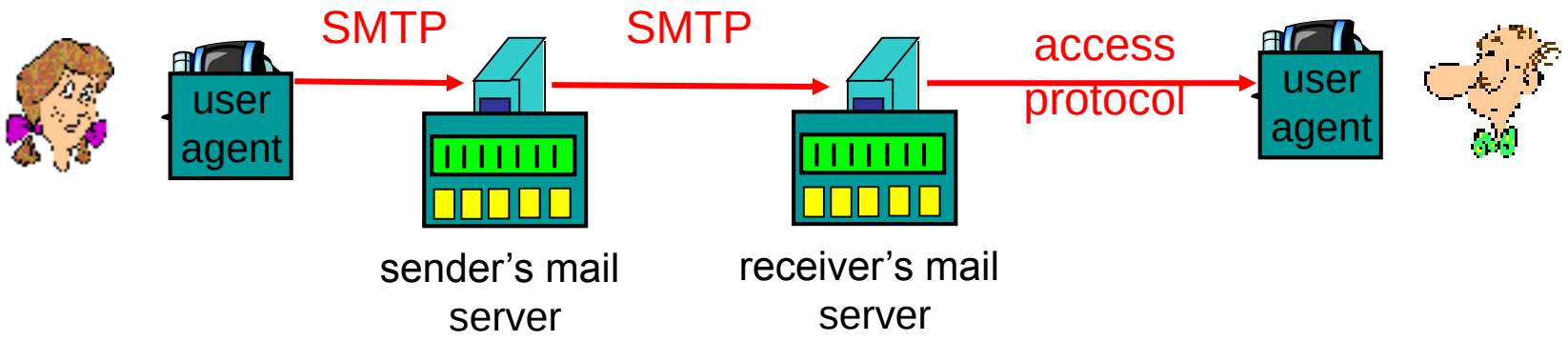


Questions:

Why does Alice needs an intermediate mail server?

Why does Bob use a different protocol?

Mail access protocols



- SMTP: delivery/storage to receiver's server
- Mail access protocol: retrieval from server
  - POP3: 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, outlook, etc.

POP3 protocol

C: telnet mailserver 110

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
```

```
duser@192.168.88.155:~/libbgpdump-1.4.99.11$ telnet pop.163.com 110
Trying 123.126.97.79...
Connected to pop3.163.idns.yeah.net.
Escape character is '^]'.
+OK Welcome to coremail Mail Pop3 Server (163coms[10774b260cc7a37d26d71b52404dcf
5cs])
user barrywuh
+OK core mail
pass [redacted]
+OK 686 message(s) [259652435 byte(s)]
```

```
retr 686
+OK 4053 octets
Received: from ucpmdf1p.canterbury.ac.nz (unknown [132.181.2.27])
        by mx26 (Coremail) with SMTP id TMCowAA3kbDGj1BdSrEhDw--.35971S3;
        Mon, 12 Aug 2019 05:59:39 +0800 (CST)
Received: from CONVERSION-DAEMON.it.canterbury.ac.nz by it.canterbury.ac.nz
 (PMDF V6.7-x02 #2267) id <0PW300R01EF8GM@it.canterbury.ac.nz> for
 barrywuh@163.com; Mon, 12 Aug 2019 09:59:34 +1200 (NZST)
Received: from UCEXHubCAS04-I.canterbury.ac.nz ([132.181.7.54])
 by it.canterbury.ac.nz (PMDF V6.7-x02 #2267)
 with ESMTPS id <0PW300CJBEF9EW@it.canterbury.ac.nz> for barrywuh@163.com; Mon,
 12 Aug 2019 09:59:33 +1200 (NZST)
Received: from UCEXMBX03-I.canterbury.ac.nz ([fe80::49ae:76ca:7b42:d8e2])
 by UCEXHubCAS04-I.canterbury.ac.nz ([fe80::a9a9:df3d:7009:e5de%13])
 with mapi id 14.03.0439.000; Mon, 12 Aug 2019 09:59:33 +1200
Date: Sun, 11 Aug 2019 21:59:33 +0000
From: Barry Wu <barry.wu@canterbury.ac.nz>
Subject: Test for email for COSC264
X-Originating-IP: [10.34.40.169]
To: "barrywuh@163.com" <barrywuh@163.com>
Message-id:
 <3ED135E755A27346BAF2B655127B8502B05ACF@UCEXMBX03-I.canterbury.ac.nz>
```

POP3 (more) and IMAP

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

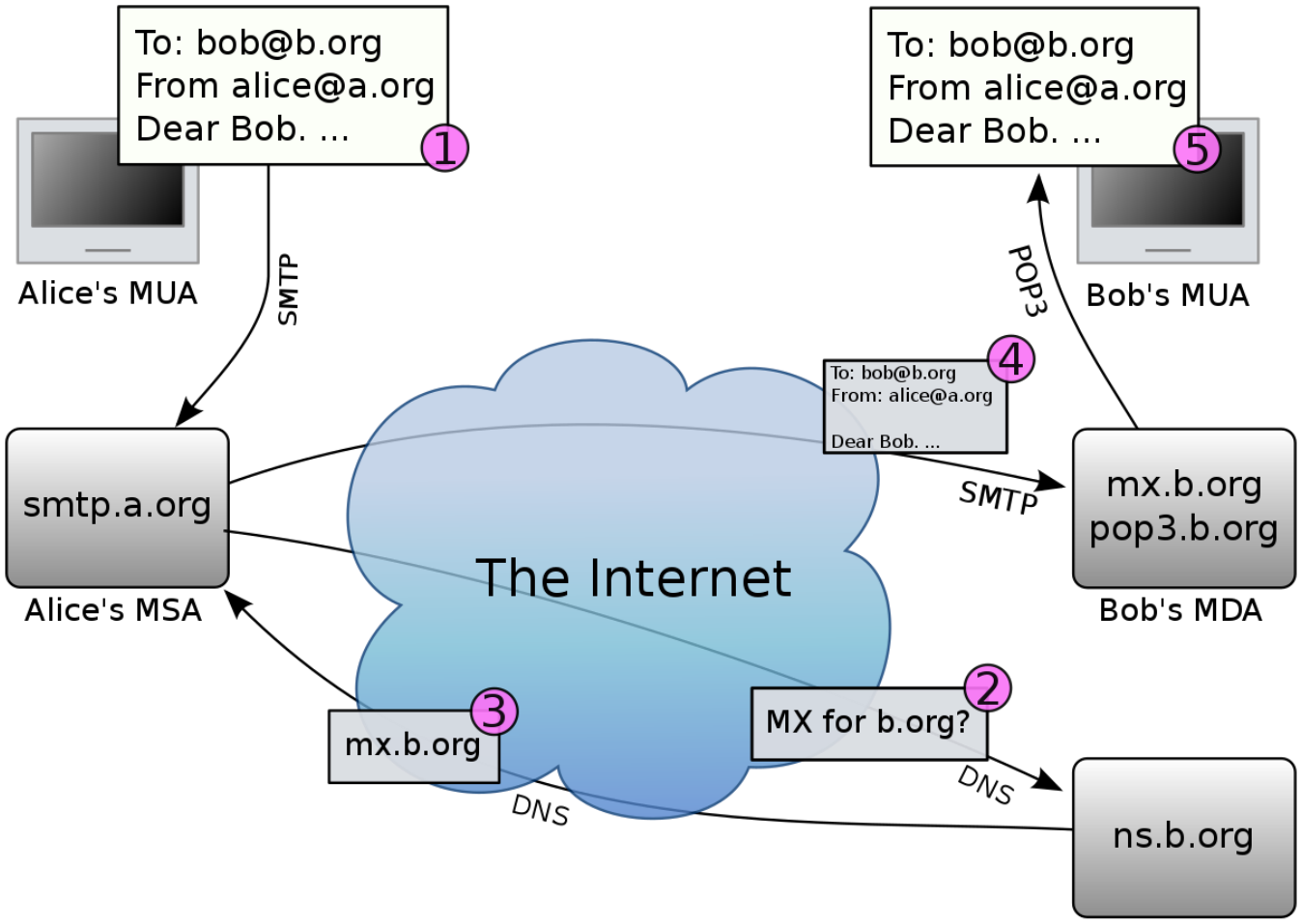
IMAP

- 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

Web-based email

- Hotmail appeared in mid 1990s;
- User communicates with its remote mailbox via HTTP.

A complete picture: DNS involved



[Wiki_email]

Outline

- Electronic Mail
  - SMTP, POP3, IMAP
- DNS

Some history

2.1. The history of domain names

The impetus for the development of the domain system was growth in the Internet:

- Host name to address mappings were maintained by the Network Information Center (NIC) in a single file (HOSTS.TXT) which was FTPed by all hosts [RFC-952, RFC-953]. The total network

bandwidth consumed in distributing a new version by this scheme is proportional to the square of the number of hosts in the network, and even when multiple levels of FTP are used, the outgoing FTP load on the NIC host is considerable. Explosive growth in the number of hosts didn't bode well for the future.

- The network population was also changing in character. The timeshared hosts that made up the original ARPANET were being replaced with local networks of workstations. Local organizations were administering their own names and addresses, but had to wait for the NIC to change HOSTS.TXT to make changes visible to the Internet at large. Organizations also wanted some local structure on the name space.

- The applications on the Internet were getting more sophisticated and creating a need for general purpose name service.

The result was several ideas about name spaces and their management [IEN-116, RFC-799, RFC-819, RFC-830]. The proposals varied, but a common thread was the idea of a hierarchical name space, with the hierarchy roughly corresponding to organizational structure, and names using "." as the character to mark the boundary between hierarchy levels. A design using a distributed database and generalized resources was described in [RFC-882, RFC-883]. Based on experience with several implementations, the system evolved into the scheme described in this memo.

DNS: Domain Name System

People: many identifiers:

- SSN, name, passport #

Internet hosts, routers:

- IP address (32 bit) - used for addressing datagrams
- "name", e.g., ww.yahoo.com - used by humans

Q: map between IP addresses and name ?

Domain Name System:

- *distributed database* implemented in hierarchy of many *name servers*
- *application-layer protocol* (udp on port 53)

  enables host, routers, name servers to communicate to *resolve* names (address/name translation)
  - note: core Internet function, implemented as application-layer protocol
  - complexity at network's "edge"

DNS

DNS services

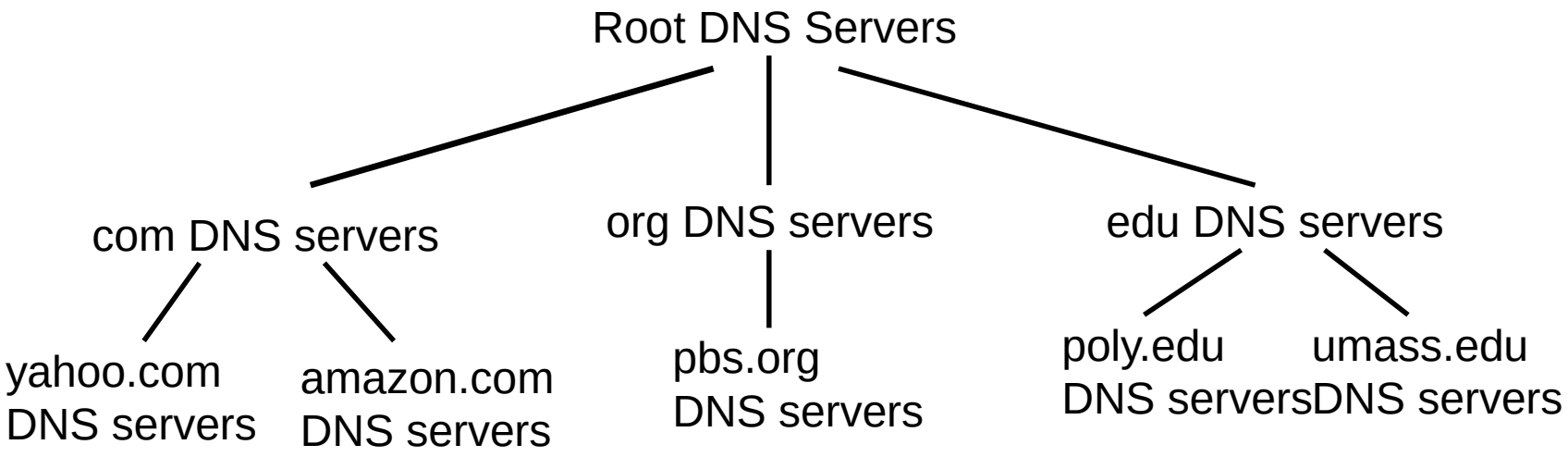
- Hostname to IP address translation
  - E.g., www.northwestern.edu
- Host aliasing
  - Canonical and alias names
  - E.g., dell.com www.dell.com
- Mail server aliasing
  - E.g., bob@hotmail.com
- Load distribution
  - Replicated Web servers: set of IP addresses for one canonical name
  - E.g., cnn.com

Why not centralize DNS?

- single point of failure
- traffic volume
- distant centralized database
- maintenance

doesn't *scale!*

Distributed, Hierarchical Database

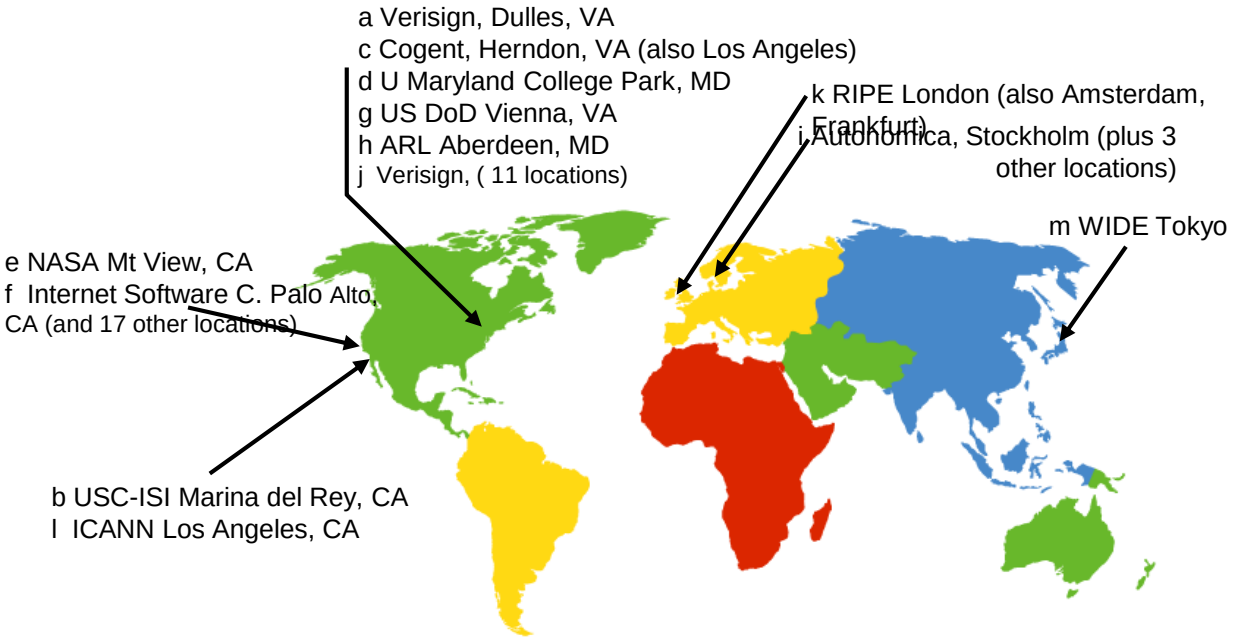


Client wants IP for www.amazon.com; 1st approx:

- Client queries a root server to find com DNS server
- Client queries com DNS server to get amazon.com DNS server
- Client queries amazon.com DNS server to get IP address for www.amazon.com

DNS: Root name servers

- contacted by local name server that can not resolve name
- root name server:
  - contacts TLD server if name mapping not known
    - TLD server contacts authoritative name server if name mapping not known
  - gets mapping
  - returns mapping to local name server



13 root name servers worldwide
each server is actually a cluster of replicated servers

TLD and Authoritative Servers

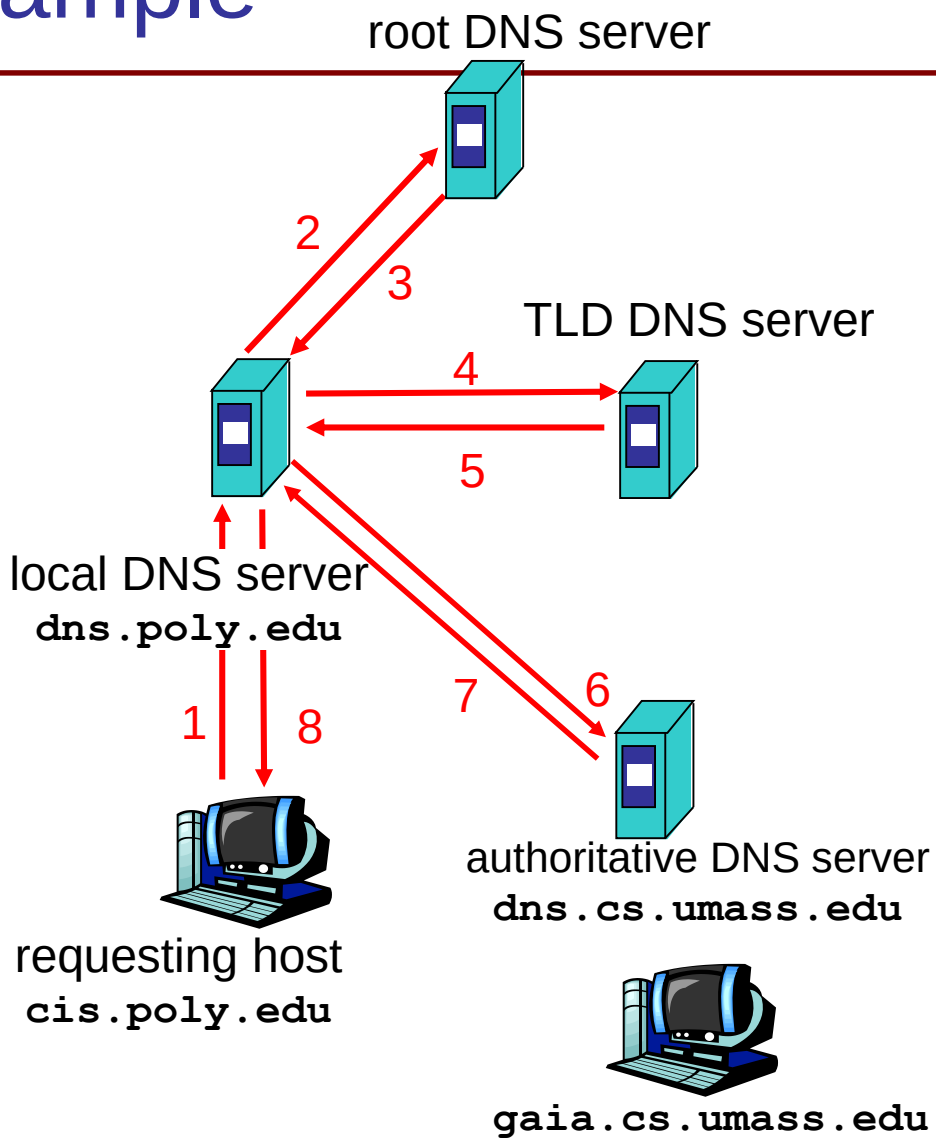
- Top-level domain (TLD) servers: responsible for com, org, net, edu, etc, and all top-level country domains uk, fr, ca, jp.
  - "Network solutions" maintains servers for com TLD
  - "Educause" for edu TLD
- Authoritative DNS servers: organization's DNS servers, providing authoritative hostname to IP mappings for organization's servers (e.g., Web and mail).
  - Can be maintained by organization or service provider

Local Name Server

- Does not strictly belong to hierarchy
- Each ISP (residential ISP, company, university) has one.
  - Also called “default name server”
    - When you connect to an ISP, you have to type the address of the default DNS server
- When a host makes a DNS query, query is sent to its local DNS server
  - Acts as a proxy, forwards query into hierarchy.

Example

- Host at cis.poly.edu wants IP address for gaia.cs.umass.edu



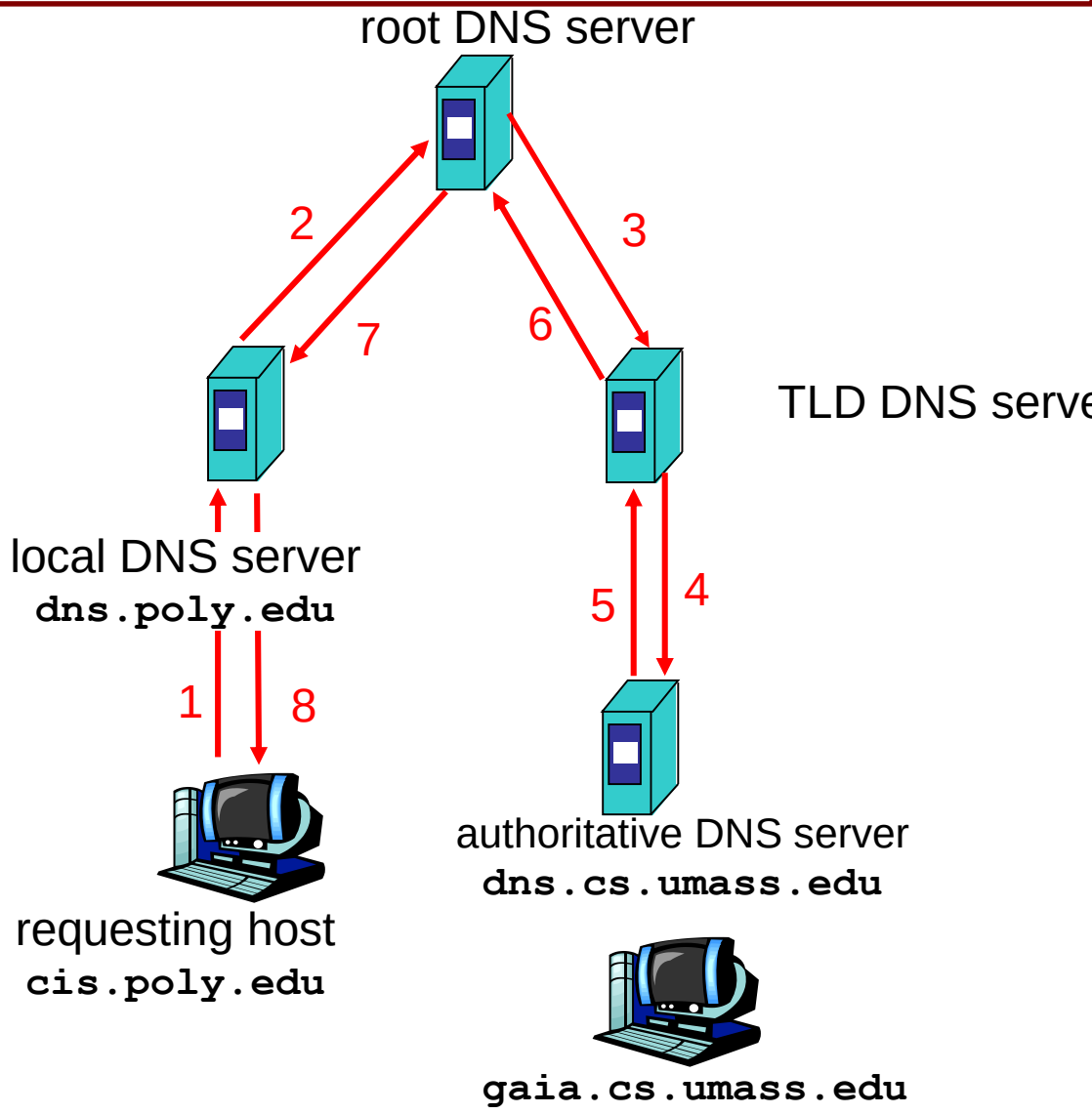
Recursive queries

recursive query:

- puts burden of name resolution on contacted name server

iterated query:

- contacted server replies with name of server to contact
  - Used in practice
- "I don't know this name, but ask this server"



DNS: caching and updating records

- once (any) name server learns mapping, it *caches* mapping
  - the DNS server can provide the desired IP address even if it is not authoritative for that hostname
- cache entries timeout (disappear) after some time
  - because hosts and mapping between host names and IP addresses are by no means permanent
- TLD servers typically cached in local name servers
  - Thus root name servers not often visited

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 IP address of authoritative name server for this domain
- Type=CNAME
  - **name** is alias name for some "cannonical" (the real) name `www.ibm.com` is really `servereast.backup2.ibm.com`
  - **value** is cannonical name
- Type=MX
  - **value** is name of mailserver associated with **name**

Try www.canterbuy.ac.nz

33	78.934584	10.34.40.169	132.181.2.225	DNS	80	Standard query 0x4401 A www.canterbury.ac.nz
34	78.936312	132.181.2.225	10.34.40.169	DNS	172	Standard query response 0x4401 A www.canterbury.ac.nz A…
35	79.021055	10.34.40.169	132.181.2.225	DNS	77	Standard query 0x461e A ocsp.digicert.com
36	79.022492	132.181.2.225	10.34.40.169	DNS	373	Standard query response 0x461e A ocsp.digicert.com CNAM…
37	79.651294	10.34.40.169	132.181.2.225	DNS	83	Standard query 0x07a3 A static.canterbury.ac.nz
38	79.654045	132.181.2.225	10.34.40.169	DNS	175	Standard query response 0x07a3 A static.canterbury.ac.n…

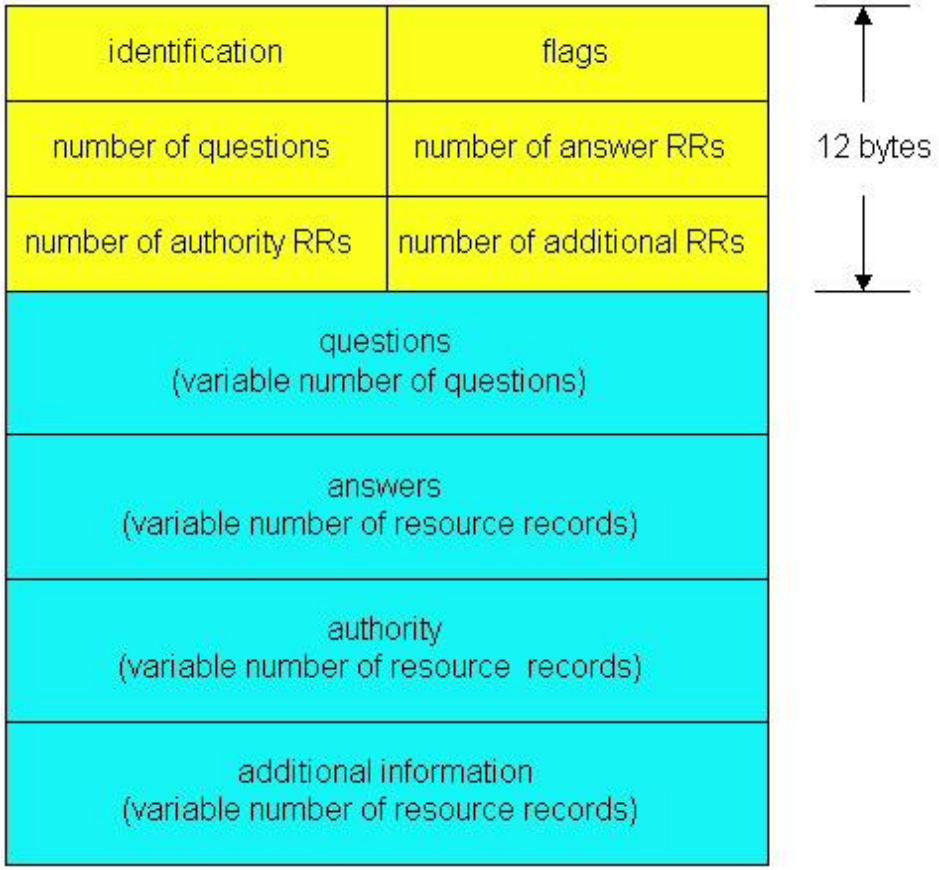
- Queries
  - www.canterbury.ac.nz: type A, class IN
    - Name: www.canterbury.ac.nz
    - [Name Length: 20]
    - [Label Count: 4]
    - Type: A (Host Address) (1)
    - Class: IN (0x0001)
- Answers
  - www.canterbury.ac.nz: type A, class IN, addr 132.181.106.9
    - Name: www.canterbury.ac.nz
    - Type: A (Host Address) (1)
    - Class: IN (0x0001)
    - Time to live: 300
    - Data length: 4
    - Address: 132.181.106.9
- Authoritative nameservers
  - canterbury.ac.nz: type NS, class IN, ns intdns2.canterbury.ac.nz
    - Name: canterbury.ac.nz
    - Type: NS (authoritative Name Server) (2)
    - Class: IN (0x0001)
    - Time to live: 300
    - Data length: 10
    - Name Server: intdns2.canterbury.ac.nz
  - canterbury.ac.nz: type NS, class IN, ns intdns1.canterbury.ac.nz
    - Name: canterbury.ac.nz
    - Type: NS (authoritative Name Server) (2)
    - Class: IN (0x0001)

DNS protocol, messages

DNS protocol : *query* and *reply* messages, both with same *message format*

msg header

- identification: 16 bit # for query, reply to query uses same #
- flags:
  - query or reply
  - recursion desired
  - recursion available
  - reply is authoritative



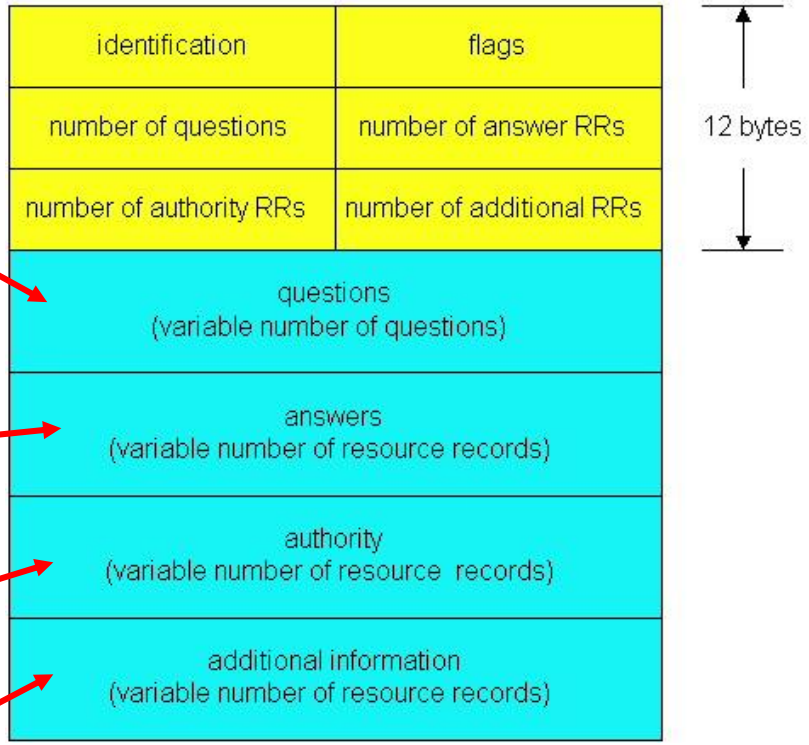
DNS protocol, messages

Name, type fields for a query
(Name, Type)
e.g., (ibm.com, CNAME)

RRs in reponse to query
(Type, Value, TTL)
(CNAME, serv.bckup.ibm.com,5)

records for authoritative servers

additional “helpful” info that may be used
e.g., (serv.bckup.ibm.com, 254.24.54.42,A)

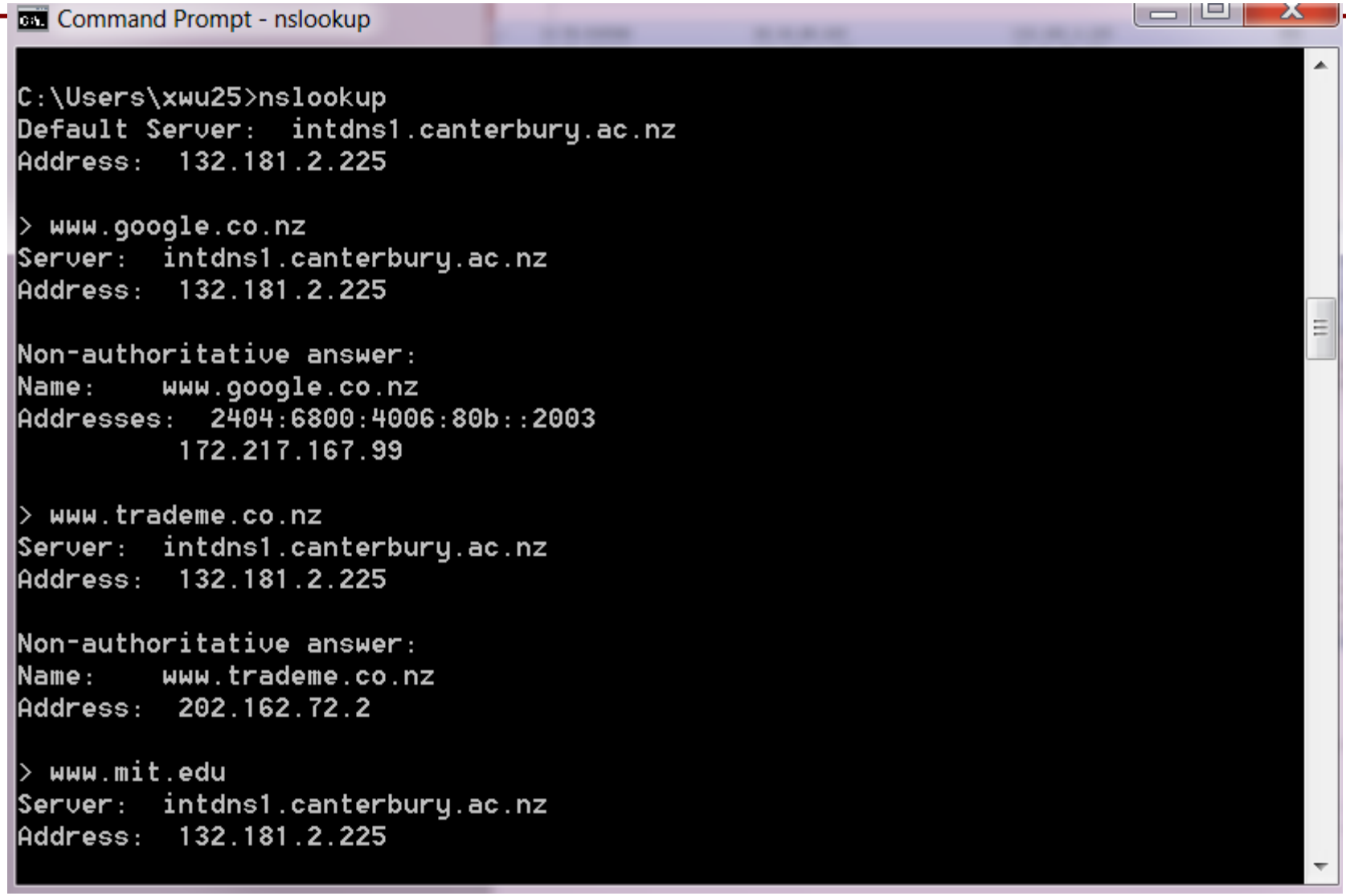


33	78.934584	10.34.40.169	132.181.2.225	DNS
34	78.936312	132.181.2.225	10.34.40.169	DNS
35	79.021055	10.34.40.169	132.181.2.225	DNS
36	79.022492	132.181.2.225	10.34.40.169	DNS
37	79.651294	10.34.40.169	132.181.2.225	DNS
38	79.654045	132.181.2.225	10.34.40.169	DNS
39	79.656486	10.34.40.169	132.181.2.225	DNS

```
▷ Frame 33: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface 0
▷ Ethernet II, Src: IntelCor_b6:fe:63 (80:19:34:b6:fe:63), Dst: JuniperN_ef:61:00 (2c:21:31:ef:61:00)
▷ Internet Protocol Version 4, Src: 10.34.40.169, Dst: 132.181.2.225
▷ User Datagram Protocol, Src Port: 59064, Dst Port: 53
◢ Domain Name System (query)
    [Response In: 34]
    Transaction ID: 0x4401
  ◢ Flags: 0x0100 Standard query
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ◢ Queries
    ◢ www.canterbury.ac.nz: type A, class IN
        Name: www.canterbury.ac.nz
        [Name Length: 20]
        [Label Count: 4]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
```

33	78.934584	10.34.40.169	132.181.2.225
34	78.936312	132.181.2.225	10.34.40.169
35	79.021055	10.34.40.169	132.181.2.225
36	79.022492	132.181.2.225	10.34.40.169
37	79.651294	10.34.40.169	132.181.2.225
38	79.654045	132.181.2.225	10.34.40.169
39	79.656486	10.34.40.169	132.181.2.225

```
▷ Ethernet II, Src: JuniperN_ef:61:00 (2c:21:31:ef:61:00), Dst: IntelCor_b6:fe:63 (80:
▷ Internet Protocol Version 4, Src: 132.181.2.225, Dst: 10.34.40.169
▷ User Datagram Protocol, Src Port: 53, Dst Port: 59064
◢ Domain Name System (response)
    [Request In: 33]
    [Time: 0.001728000 seconds]
    Transaction ID: 0x4401
  ◢ Flags: 0x8580 Standard query response, No error
      1... .... .... .... = Response: Message is a response
      .000 0... .... .... = Opcode: Standard query (0)
      .... .1.. .... .... = Authoritative: Server is an authority for domain
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... 1... .... = Recursion available: Server can do recursive queries
      .... .... .0.. .... = Z: reserved (0)
      .... .... ..0. .... = Answer authenticated: Answer/authority portion was not au
      .... .... ...0 .... = Non-authenticated data: Unacceptable
      .... .... .... 0000 = Reply code: No error (0)
    Questions: 1
    Answer RRs: 1
    Authority RRs: 2
    Additional RRs: 2
  ◢ Queries
    ▷ www.canterbury.ac.nz: type A, class IN
  ▷ Answers
  ▷ Authoritative nameservers
  ▷ Additional records
```


Command Prompt - nslookup
C:\Users\xwu25>nslookup
Default Server: intdns1.canterbury.ac.nz
Address: 132.181.2.225
> www.google.co.nz
Server: intdns1.canterbury.ac.nz
Address: 132.181.2.225
Non-authoritative answer:
Name: www.google.co.nz
Addresses: 2404:6800:4006:80b::2003
          172.217.167.99
> www.trademe.co.nz
Server: intdns1.canterbury.ac.nz
Address: 132.181.2.225
Non-authoritative answer:
Name: www.trademe.co.nz
Address: 202.162.72.2
> www.mit.edu
Server: intdns1.canterbury.ac.nz
Address: 132.181.2.225

Inserting records into DNS

- Example: just created startup “Network Utopia”
- Register name networkuptopia.com at a registrar (e.g., “Network Solutions”)
  - Need to provide registrar with names and IP addresses of your authoritative name server (primary and secondary)
  - Registrar inserts two RRs into the com TLD server:

```
(networkutopia.com, dns1.networkutopia.com, NS)
(dns1.networkutopia.com, 212.212.212.1, A)
```

- Put in authoritative server Type A record for www.networkuptopia.com and Type MX record for networkutopia.com
- How do people get the IP address of your Web site?

Outline

- Electronic Mail
  - SMTP, POP3, IMAP
- DNS

References

- [KR3] James F. Kurose, Keith W. Ross, *Computer networking: a top-down approach featuring the Internet*, 3rd edition.
- [PD5] Larry L. Peterson, Bruce S. Davie, *Computer networks: a systems approach*, 5th edition
- [TW5] Andrew S. Tanenbaum, David J. Wetherall, *Computer network*, 5th edition
- [LHBi]Y-D. Lin, R-H. Hwang, F. Baker, *Computer network: an open source approach*, International edition

Acknowledgements

- All slides are developed based on slides from the following two sources:
  - Dr DongSeong Kim's slides for COSC264, University of Canterbury;
  - Prof Aleksandar Kuzmanovic's lecture notes for CS340,Northwestern University, https://users.cs.northwestern.edu/~akuzma/classes/CS340-w05/lecture_notes.htm