Lecture 3

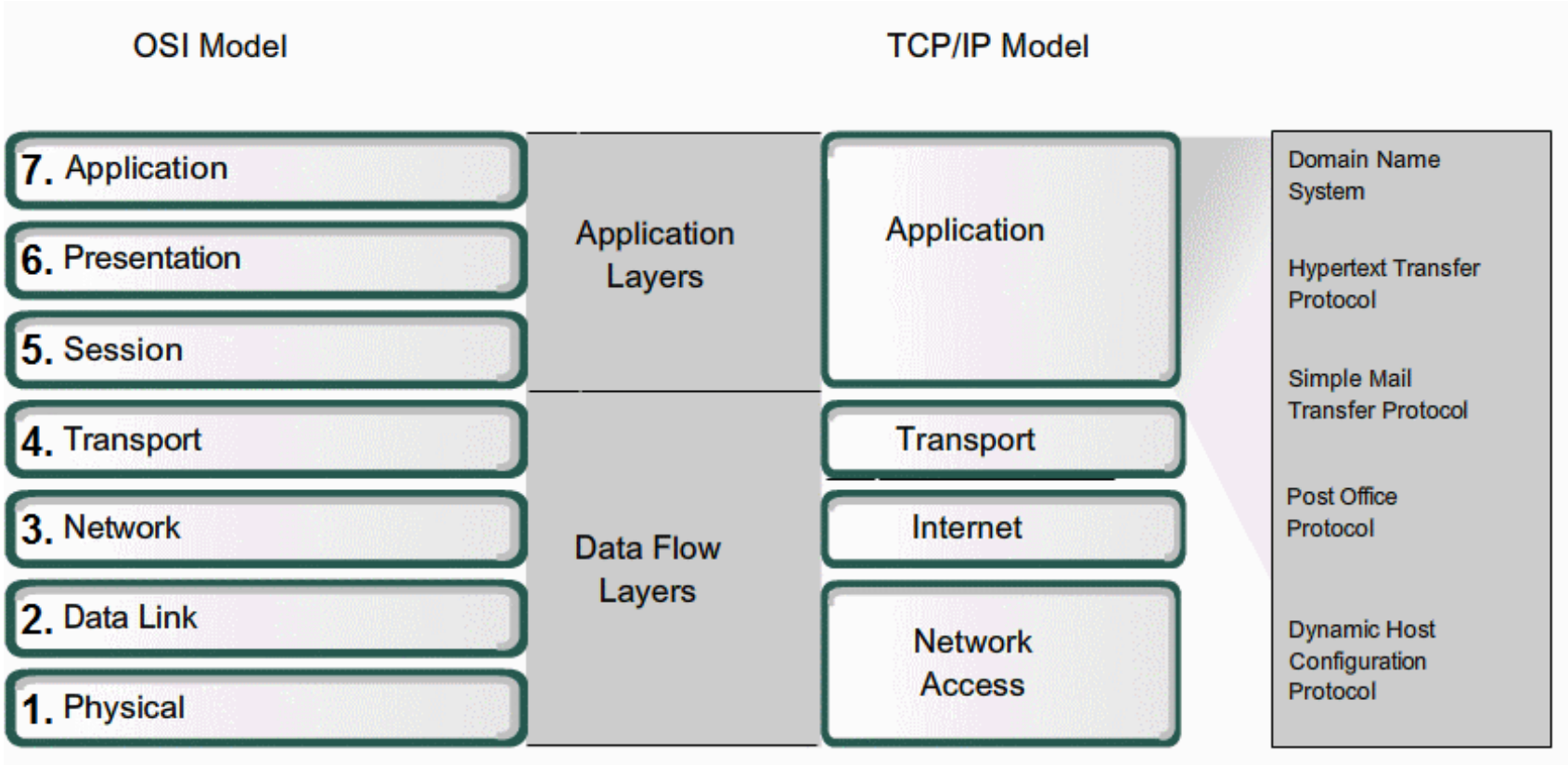
Common Internet Applications (HTTP, DNS, Email)

Topics Covered

- **Application-Layer Protocols**
- **Web Protocols**
  - Document Representation with HTML
  - Uniform Resource Locators and Hyperlinks
  - Web Document Transfer with HTTP
  - Caching in Browsers
- FTP and Telnet
- **Domain Name Systems (DNS)**
- **Electronic Mail**
  - The Simple Mail Transfer Protocol (SMTP)
  - POP3 and IMAP

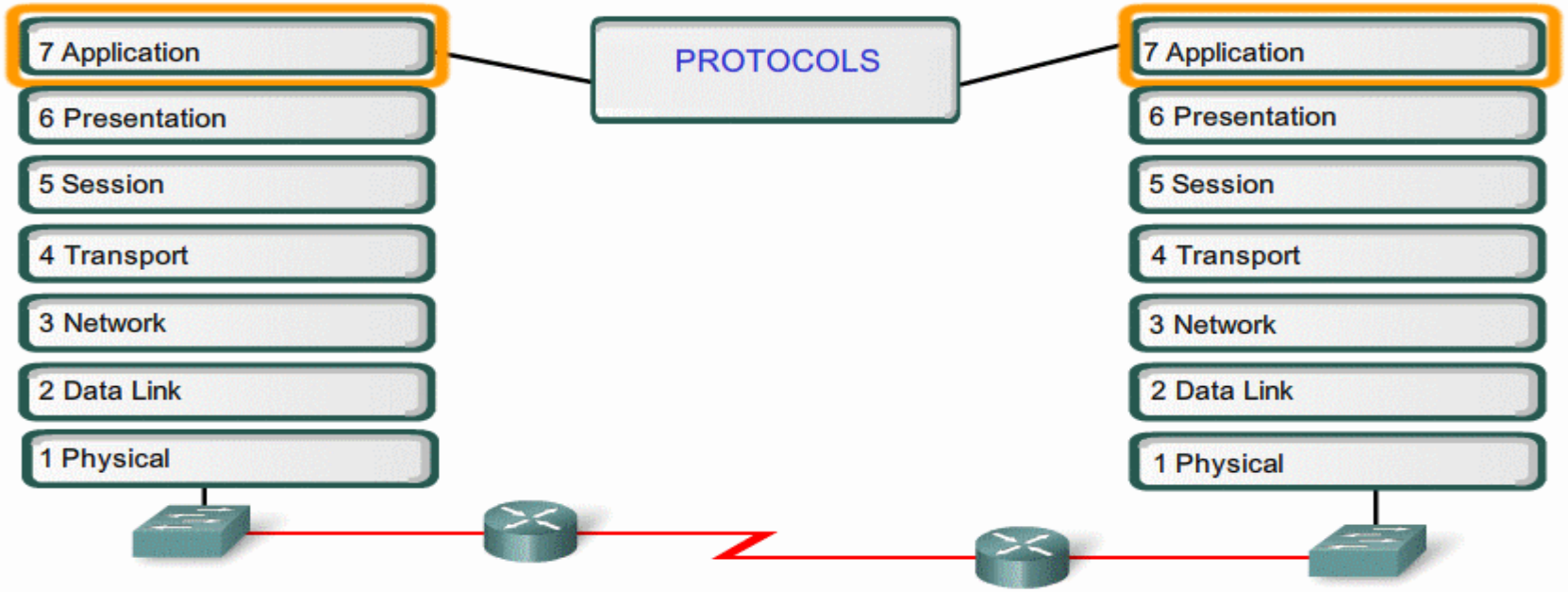
Application-Layer Protocols

- From Cisco course material
- Location on the protocol stack



Application-Layer Protocols

- From Cisco course material



Application layer protocols provide the rules for communication between applications.

Protocols:

Define processes on either end of the communication
Define the types of messages
Define the syntax of messages
Define the meaning of any informational fields
Define how messages are sent and the expected response
Define interaction with the next lower layer

Application-Layer Protocols

- Whenever a programmer creates two network applications,
  - One is the client, the other is the server
  - the programmer specifies some details called the **protocol**
  - The syntax and semantics of messages
    - Syntax– how to construct sentences (message format)
    - Semantics—how to interpret the sentence; **meaning**
  - Actions to be taken if an error arises
  - When to terminate communication
- Read more on protocol at Wikipedia
  - http://en.wikipedia.org/wiki/Communications_protocol
- There are two broad types of application-layer protocols
  - Private communication
  - Standardized service

Application-Layer Protocols

- Private communication
  - A programmer creates a pair of applications that communicate over the Internet with the intention that the pair is for **private use**
  - Interaction between the two applications is straightforward
    - code can be written without writing a formal protocol specification
- Standardized service
  - Expectation is that many programmers will create server software to offer the service or client software to access the service, in this case
    - Application protocol must be documented independent of implementation
    - The specification must be precise and unambiguous (clear)
- The size of a protocol specification depends on the complexity of the service

Web Protocols

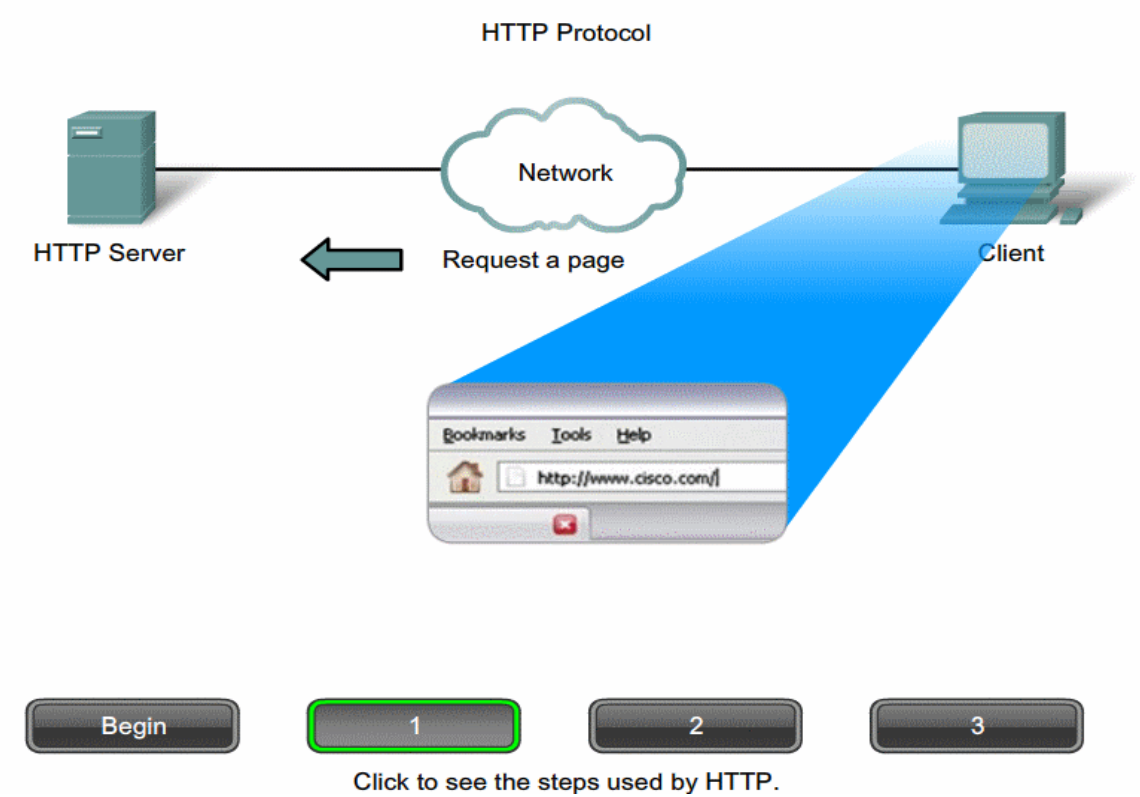
- The World Wide Web (WWW) is one of the most widely used **services** in the Internet
- Web is complex
  - many protocol standards have been devised to specify various aspects and details
- Figure 4.2 (below) illustrates major WWW standards

Standard	Purpose
HyperText Markup Language (HTML)	A representation standard used to specify the contents and layout of a web page
Uniform Resource Locator (URL)	A representation standard that specifies the format and meaning of web page identifiers
HyperText Transfer Protocol (HTTP)	A transfer protocol that specifies how a browser interacts with a web server to transfer data

Figure 4.2 Three key standards that the World Wide Web service uses.

HTTP operations

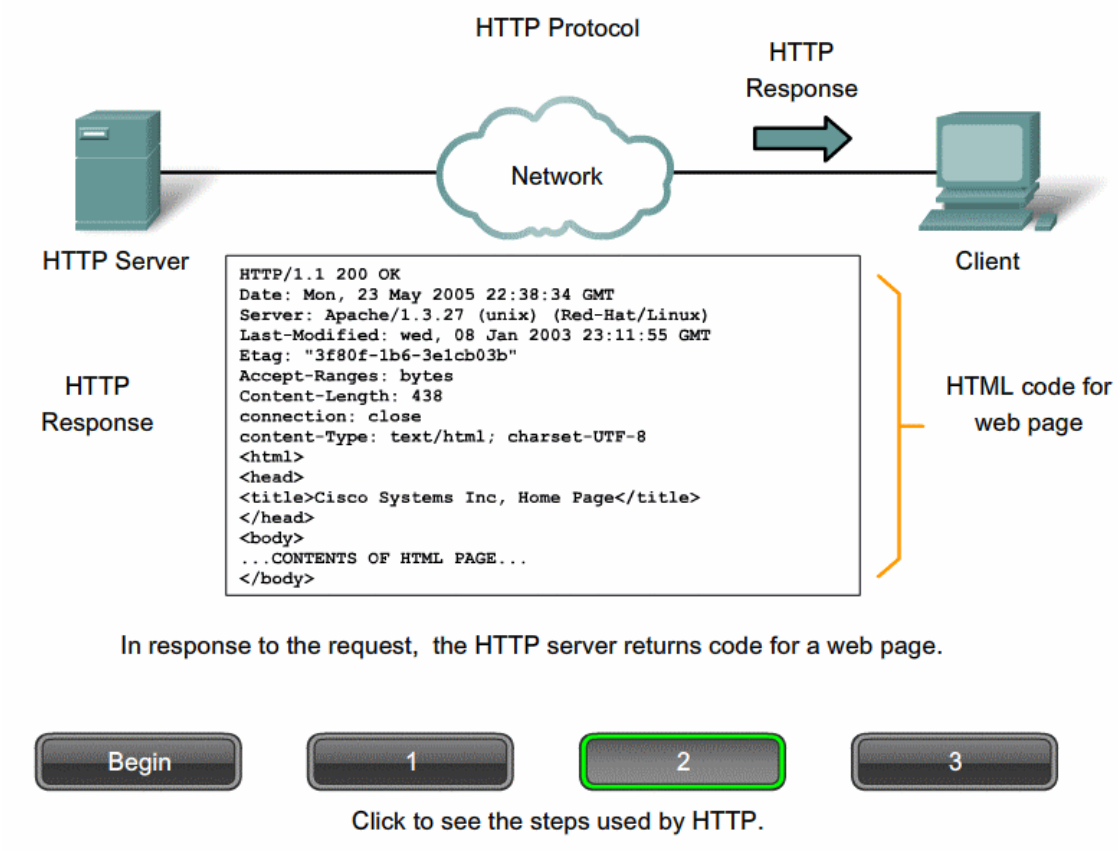
- *From Cisco course material*
- When a client wants to access Cisco home page, identified by a URL address



HTTP operations

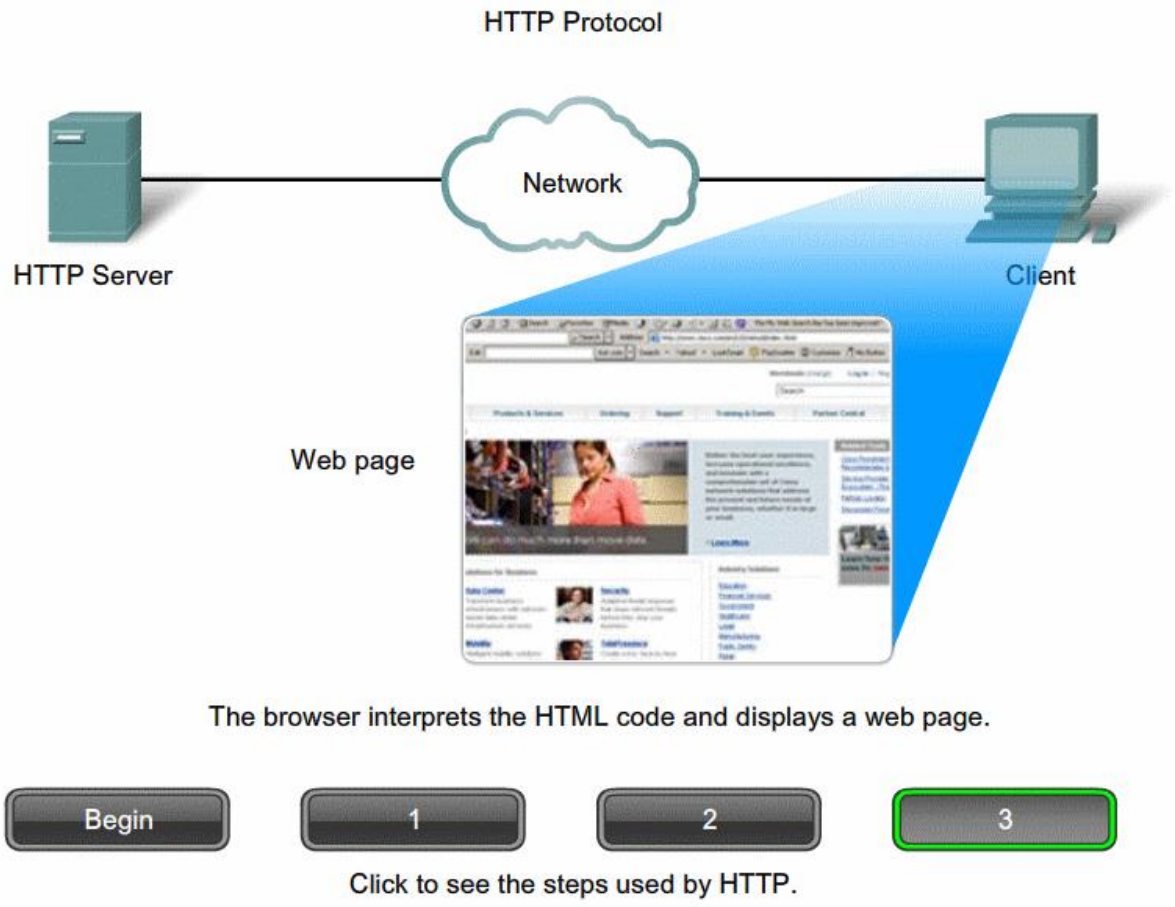
- From Cisco course material

Response from Cisco server



HTTP operations

- From Cisco course material
- Displayed by browser based on HTML code



Document Representation with HTML

```
<HTML>

  <HEAD>
      <TITLE>
            text that forms the document title
      </TITLE>
  </HEAD>

  <BODY>
       body of the document appears here
  </BODY>

</HTML>
```

Figure 4.3 The general form of an HTML

Here is an icon of a house.

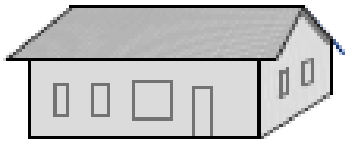


Figure 4.4 Illustration of figure alignment in HTML.

Document Representation with HTML

- HyperText Markup Language (HTML) is a representation standard that specifies the syntax of a web page
- HTML has the following general characteristics:
  - Uses a textual representation
  - Describes pages that contain multimedia
  - Follows a declarative rather than procedural paradigm
  - Provides markup specifications instead of formatting
  - Permits a hyperlink to be embedded in an arbitrary object
  - Allows a document to include metadata
    - http://en.wikipedia.org/wiki/Metadata
- HTML allows a programmer to specify a complex web page that contains graphics, audio, and video, as well as text
  - We should have used hypermedia in the name instead of hypertext

Something New ??

- HTML5 is the 5-th version of HTML
- Introduced application programming interfaces (APIs) for complex web applications
  - a candidate for cross-platform mobile applications
- WebRTC (Web Real Time Communications)
- Web Socket API
- Web of Things vs Internet of Things
  - https://webofthings.org/2016/01/23/wot-vs-iot-12/

Uniform Resource Locators and Hyperlinks

- The Web uses a syntactic form known as a Uniform Resource Locator (URL) to specify a web page
- The general form of a URL is:

protocol://computer_name:port/document_name?parameters

- where
  - protocol is the name of the protocol used to access the document
  - computer_name is the domain name of the computer on which the document resides
  - port (optional) port number at which the server is listening
  - document_name (optional) name of the document
  - ?parameters is optional for the page
  - Example:

http://www.netbook.cs.purdue.edu/toc/toc01.htm

Uniform Resource Locators and Hyperlinks

- In a typical URL, a user can omit many of the parts

www.netbook.cs.purdue.edu

- Which omits
  - the protocol (http is assumed)
  - the port (80 is assumed)
  - the document name (index.html is assumed)
  - and parameters (none are assumed)
- A URL contains the information a browser needs to retrieve a page
- Browser uses the separator characters
  - **colon**, **slash**, and **question mark**, to divide the URL into four components:
    - a protocol, a computer name, a document name, and parameters
- Browser uses the computer name and protocol port to form a connection to the server on which the page resides
- Browser uses the document name and parameters to request a page

Web Document Transfer with HTTP

- HyperText Transfer Protocol (HTTP) is the primary transfer protocol that a browser uses to interact with a web server
- HTTP protocol http://www.w3.org/Protocols/
- A browser is a web client that extracts a server name from a URL, contacts the server, gets and displays the page

- Read on HTTP at Wikipedia
  - http://en.wikipedia.org/wiki/HTTP
- HTTP can be characterized as follows:
  - Uses **textual** control messages (why textual? Not binary?)
  - Transfers binary data files (**why** file contents sent in binary?)
  - Can download or upload data
  - Incorporates caching

HTTP requests

- Once it establishes a connection
  - a browser sends an HTTP request to the server
- Figure 4.5 (below) lists the four major request types

Request	Description
GET	Requests a document; server responds by sending status information followed by a copy of the document
HEAD	Requests status information; server responds by sending status information, but does not send a copy of the document
POST	Sends data to a server; the server appends the data to a specified item (e.g., a message is appended to a list)
PUT	Sends data to a server; the server uses the data to completely replace the specified item (i.e., overwrites the previous data)

Figure 4.5 The four major HTTP request types.

HTTP GET request

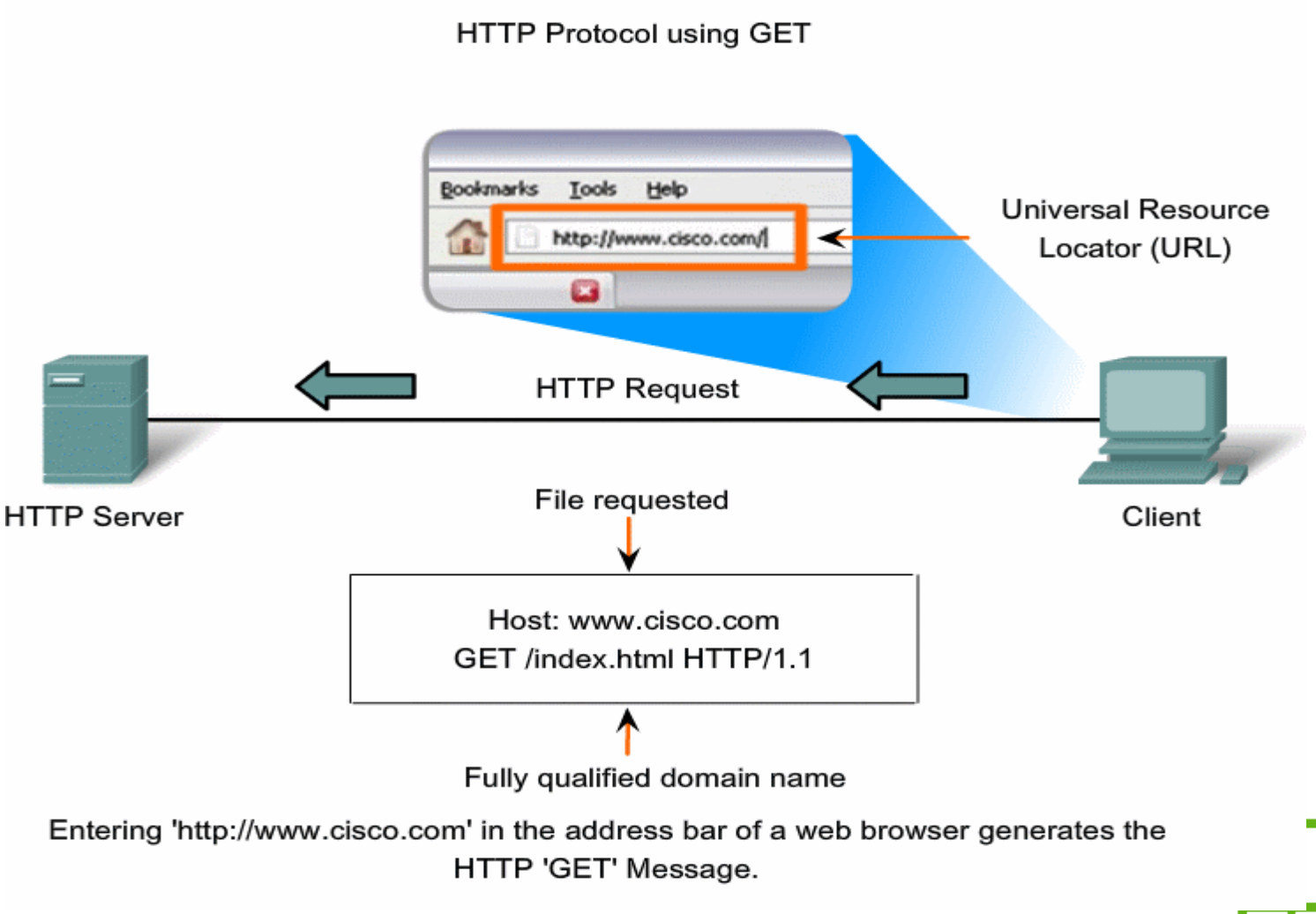
- The most common form of interaction begins with the browser requesting a page from the server
- The browser (client) sends a GET request to the HTTP server
- The server responds by sending a header, a blank line, and the requested document
- A GET request has the following form:

  GET /item http-version CRLF

  – item gives the URL for the item being requested,
  – http-version specifies a version of the protocol (HTTP/1.0 or HTTP/1.1)
  – CR LF denotes two ASCII characters
    - carriage return and linefeed, that are used to signify the end of a line of text
- Version information is important in HTTP
  – it allows the protocol to change and yet remain backward compatible
  – a browser sends version information which allows
  – a server to choose the highest version that they can both understand

HTTP GET request

- From Cisco course material



HTTP Server Responses

- The first line of a response header contains a status code
  - that tells the browser whether the server handled the request
  - If the request was incorrectly formed or the requested item was not available, the status code pinpoints the problem

Status Code	Corresponding Status String
200	OK
400	Bad Request
404	Not Found

- Additional lines of the header give further information, such as
  - Its content length (why length is required?)
  - when it was last modified (why it is needed?)
  - and the content type (what is the purpose?)

HTTP server response header

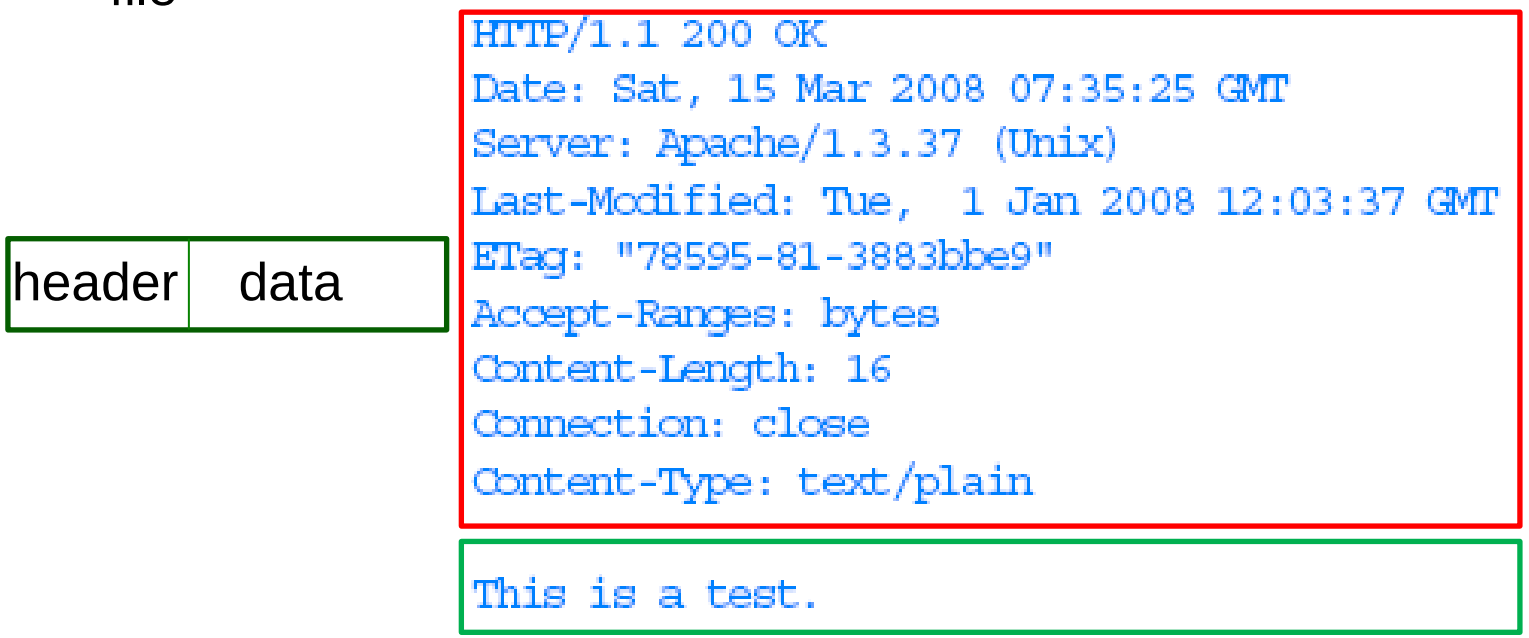
- Figure 4.6 shows the general format of lines in a basic response header

```
HTTP/1.0 status_code status_string CRLF
Server: server_identification CRLF
Last-Modified: date_document_was_changed CRLF
Content-Length: datasize CRLF
Content-Type: document_type CRLF
CRLF
```

Figure 4.6 General format of lines in a basic response header.

HTTP server response example

- Figure 4.8 shows sample output from an Apache web server
- The item being requested is a text file containing 16 characters
  - (i.e., the text This is a test. plus a NEWLINE character)
- Although the GET request specifies HTTP version 1.0, the server runs version 1.1
- The server returns 9 lines of header, a blank line, and the contents of the file



Caching in Browsers

- Caching provides an important optimization for web access
  - because users tend to visit the same web sites repeatedly
- Much of the content at a given site consists of large images
  - Graphics Image Format (GIF)
  - Joint Picture Encoding Group (JPEG)
- Such images often contain backgrounds or banners
  - they do not change frequently
- A browser can reduce download times significantly
  - by saving a copy of each image in a cache on the user's disk and using the cached copy
- What happens if the document on the web server changes after a browser stores a copy in its cache?
  - How can a browser tell whether its cached copy is stale?

Caching in Browsers

- Whenever a browser obtains a new document from a web server
- A browser saves the Last-Modified date information along with the cached copy
  - A browser makes a HEAD request to the server and compares the Last-Modified date of the server's copy to the Last-Modified date in the cached
  - If the cached version is stale, the browser downloads the new version
- Algorithm 4.1 summarizes caching, but omits several minor details
  - For example, HTTP allows a web site to include a No-cache header that specifies a given item should not be cached
- Browsers do not cache small items
  - because the time to download the item with a GET request is approximately the same as the time to make a HEAD request and keeping many small items in a cache can increase cache lookup times

Caching in Browsers

Algorithm 4.1

Given:

A URL for an item on a web page

Obtain:

A copy of the page

Method:

```
if (item is not in the local cache) {
        Issue GET request and place a copy in the cache;
} else {
    Issue HEAD request to the server;
    if (cached item is up-to-date) {
        use cached item;
    } else {
        Issue GET request and place a copy in the cache;
    }
}
```

Algorithm 4.1 Caching in a browser used to reduce download times.

State Information and Cookies

- How to make web server to remember client info?
  - Using IP addresses? (consider dynamic IP, NAT)

- Cookie
  - A message given to a web browser by the server, and saved in a text file.
  - The message is sent back to the server each time the browser requests a page from the server.
  - Browser can be configured on cookies
  - Short-term cookie: only live in a session
  - Long-term cookie: saved in the disk

WireShark and HTTP Protocol

- WireShark
  - Download and install WireShark yourself
  - Download Wireshark labs zip files
  - Get started with WireShark

- Analyse trace http-ethereal-trace-2 using filter http
- We are going to use it a lot in the course

File Transfer Protocol (FTP)

- 3 traditional Internet
  - FTP,
  - email, and
  - remote login (telnet)
- FTP
  - FTP server and ftp client allow the file uploading/downloading across the Internet
  - Separate connections for control messages and data transfer
  - It is not secure (plain text in communications)
  - Standalone FTP client application
  - Web browser also implements FTP protocol
    - Try ftp://ftp.sunet.se
- Ref: http://en.wikipedia.org/wiki/File_Transfer_Protocol
- https://**filezilla**-project.org/

A typical FTP session

- Figure 4.10

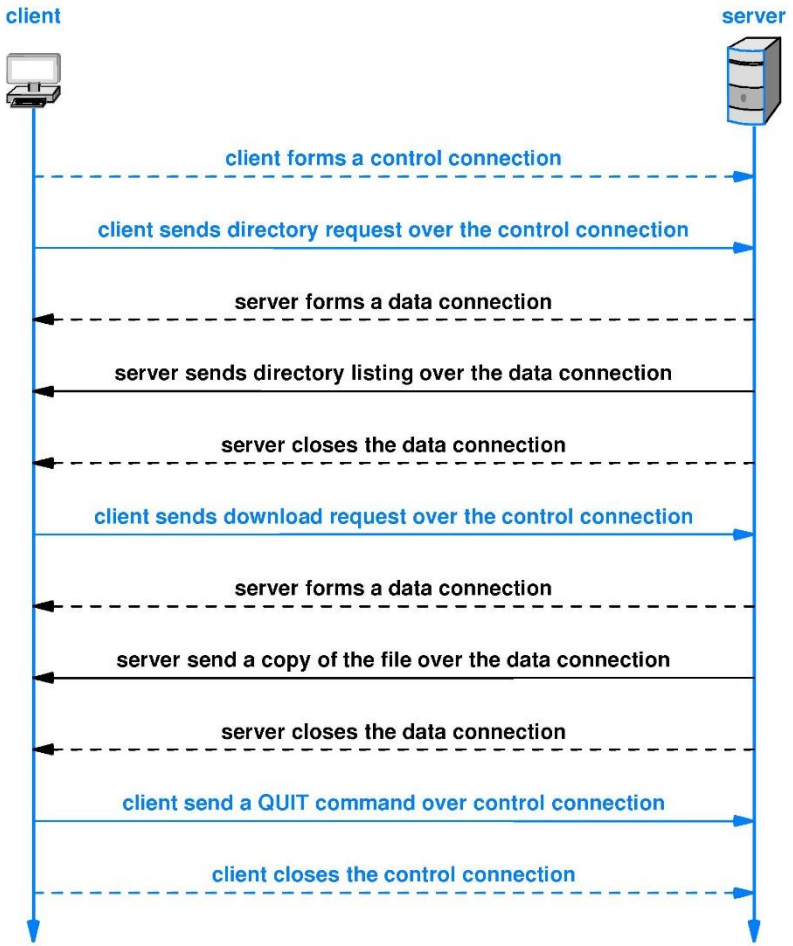
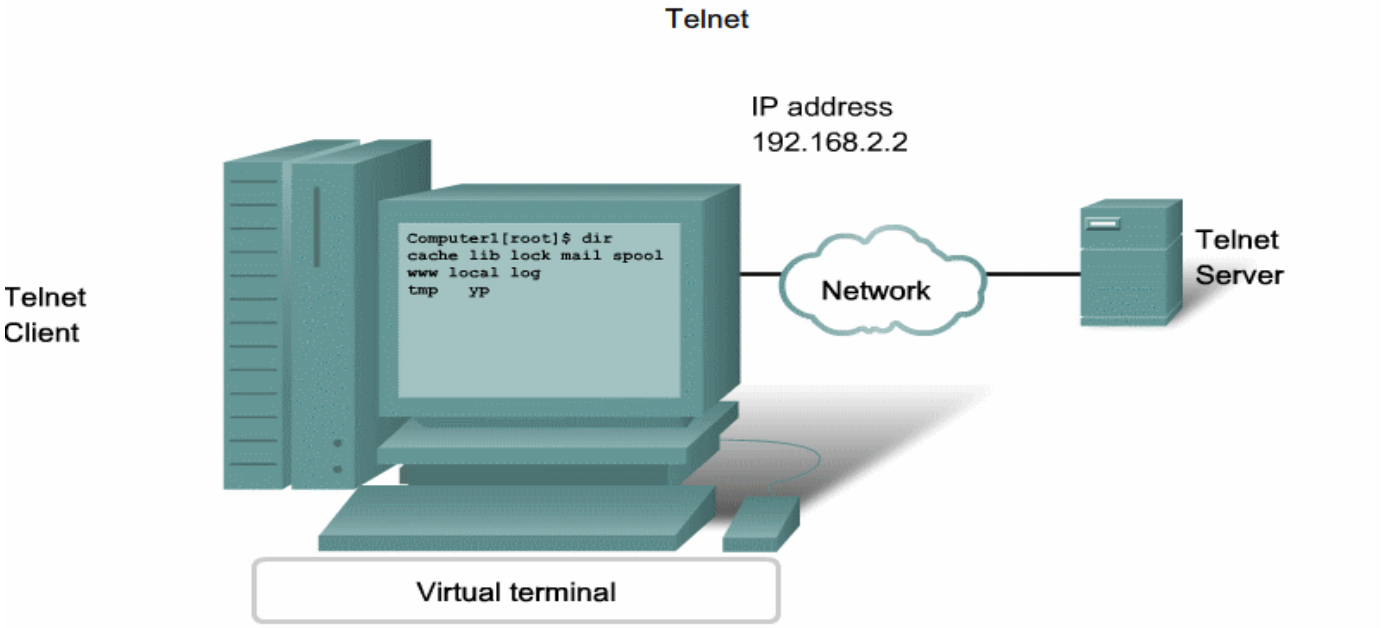


Figure 4.10 Illustration of FTP connections during a typical session.

Telnet or Remote Login

- Common tool for networking
- From Cisco course material



Telnet provides a way to use a computer, connected via the network, to access a network device as if the keyboard and monitor were directly connected to the device.

SSH and sftp

- Secure shell (SSH)
  - A network protocol for secure communication
  - Replacement of telnet or rlogin
  - http://en.wikipedia.org/wiki/Secure_Shell
  - OpenSSH
- SFTP
  - Secure FTP
  - Windows client -- WinSCP
- SSH server and client
  - Linux: ssh client/server installation
  - Windows: puTTY http://www.chiark.greenend.org.uk/~sgtatham/putty/

Domain Name System (DNS)

- Important piece of Internet infrastructure
- Runs at the application layer
- DNS provides a service that maps human-readable symbolic names to computer addresses
- Provide name resolution services

  domain name ⟷ IP address

- Whenever an application needs to translate a name
  - the application becomes a client of the naming system
  - the client sends a request message to a name server
  - server finds the corresponding address and sends a reply message
    - if it cannot answer a request, a name server temporarily becomes the client of another name server, until a server is found that can answer the request
- **Name resolution**
  - Given a name, find its IP address

Domain Name System (DNS)

- Syntactically, each name consists of a sequence of alpha-numeric segments separated by periods
  - For example, a computer in the Computer Science Department at Purdue University has the domain name:

    mordred.cs.purdue.edu
  - A computer at Cisco, Inc. has the domain name:

    anakin.cisco.com
- Domain names are hierarchical, with the most significant part of the name on the right
  - The left-most segment of a name (mordred and anakin in the examples) is the name of an individual computer
  - Other segments in a domain name identify the group that owns it
    - For example, the segment purdue gives the name of a university, and cisco gives the name of a company

Domain Name System (DNS)

- DNS does not specify the number of segments in a name
- DNS does specify values for the most significant segment, which is called a top-level domain (TLD)
  - Controlled by the Internet Corporation for Assigned Names and Numbers (ICANN)
  - ICANN designates one or more domain registrars to administer a given top-level domain and approve specific names
- Some TLDs are generic, meaning they are generally available
  - Other TLDs are restricted to specific groups or government agencies
- Figure 4.16 lists example top-level DNS domains
- An organization applies for a name under one of the existing top-level domains
  - most US corporations choose to register under the com domain
- DNS allows organizations to use a geographic registration
  - For example, the Corporation For National Research Initiatives registered the domain:

    cnri.reston.va.us

Figure 4.16
Example top-level domains and the group to which each is assigned

Domain Name	Assigned To
aero	Air transport industry
arpa	Infrastructure domain
asia	For or about Asia
biz	Businesses
com	Commercial organizations
coop	Cooperative associations
edu	Educational institutions
gov	United States Government
info	Information
int	International treaty organizations
jobs	Human resource managers
mil	United States military
mobi	Mobile content providers
museum	Museums
name	Individuals
net	Major network support centers
org	Non-commercial organizations
pro	Credentialed professionals
travel	Travel and tourism
country code	A sovereign nation

About top domain .nu

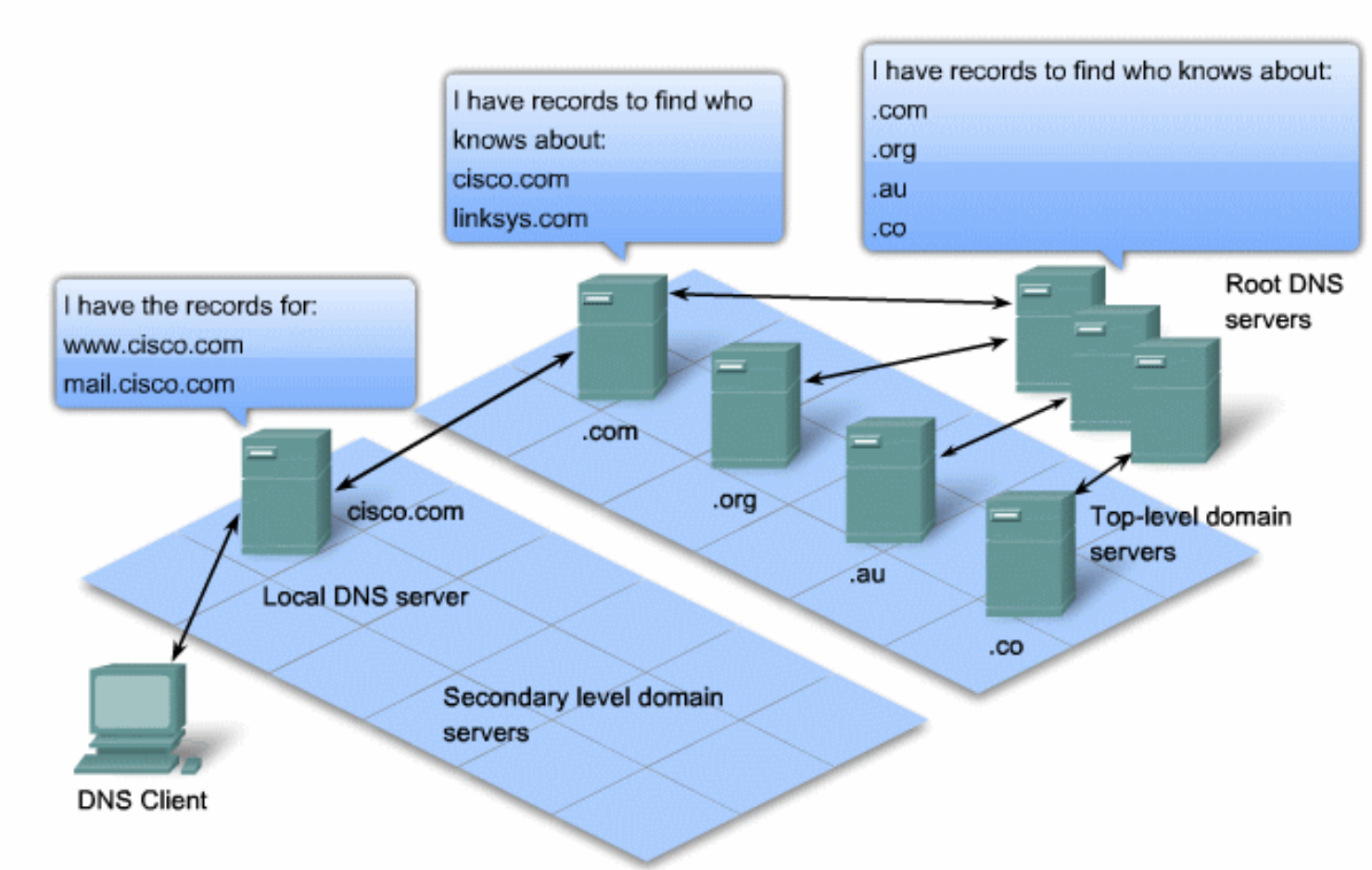
- .nu is a top level domain for the island state of Niue
  http://en.wikipedia.org/wiki/.nu

- Since in Sweden and some other countries, “nu” means “now”, lot of companies/organizations register their domain names under this top domain:

- Examples
  - www.studera.nu
- Using http://www.iplocation.net/index.php find more about this IP address?
- How it is possible ?

A hierarchy of DNS servers

- From Cisco course material



A hierarchy of DNS servers contains the resource records that match names with addresses.

Domain Names That Begin with www

- Many organizations assign domain names that reflect the service a computer provides
  - For example, a computer that runs a server for FTP might be named:

    ftp.foobar.com
  - Similarly, a computer that runs a web server might be named:

    www.foobar.com
- Such names are mnemonic, but are not required
- The use of *www* to name computers that run a web server is merely a convention
  - an arbitrary computer can run a web server, even if the computer's domain name does not contain *www*
  - a computer that has a domain name beginning with *www* is not required to run a web server

The DNS Hierarchy and Server Model

- Each organization is free to choose the details of its servers
  - For example, a small organization that only has a few computers can contract with an ISP to run a DNS server.
- An organization that runs its own server can choose to place all names for the organization in a single physical server, or it can choose to divide its names among multiple servers
  - For example, Figure 4.17 illustrates how the hypothetical Foobar Corporation might choose to structure servers if the corporation had a candy division and a soap division

The DNS Hierarchy and Server Model

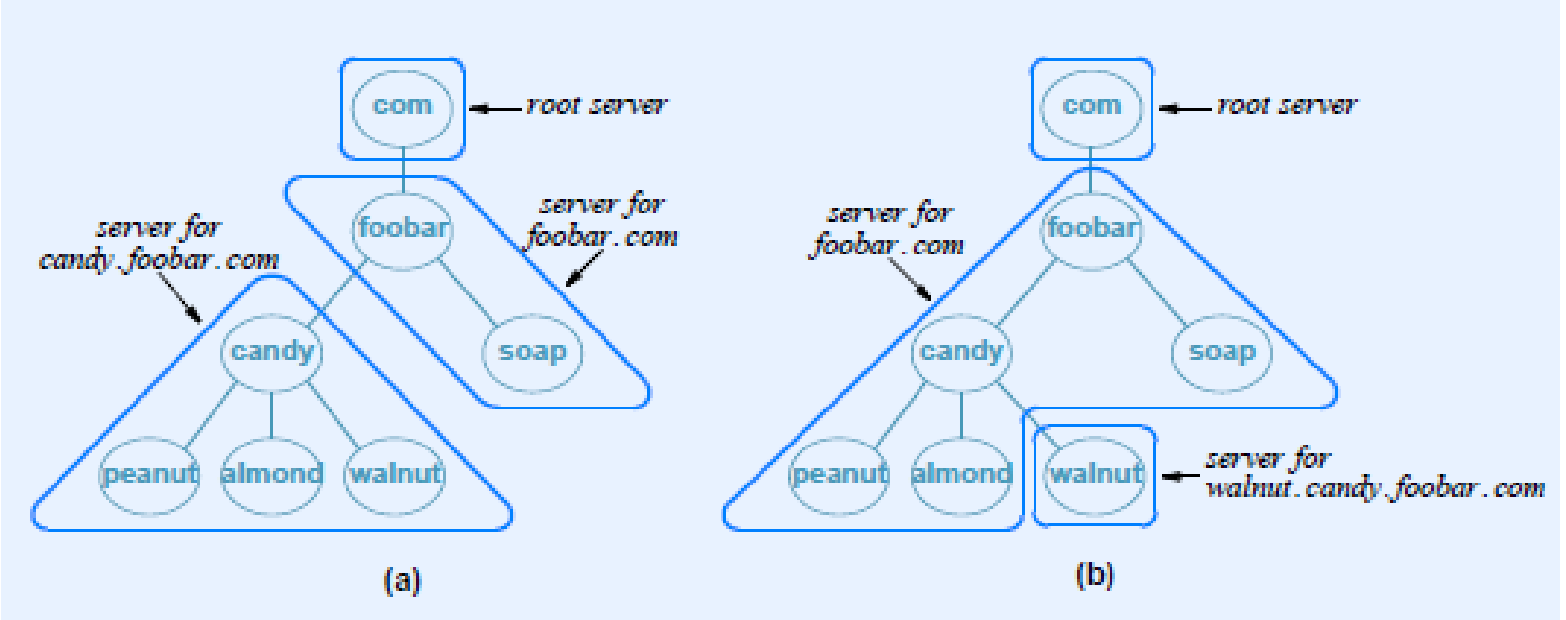


Figure 4.17 A hypothetical DNS hierarchy and two possible assignments of names to servers.

The DNS Hierarchy and Server Model

- DNS is designed to allow each organization to assign names to computers or to change those names without informing a central authority
  - To achieve autonomy, each organization is permitted to operate DNS servers for its part of the hierarchy
    - Purdue University operates a server for names ending in *purdue.edu*
    - IBM Corporation operates a server for names ending in *ibm.com*
- Each DNS server contains information that links the server to other domain name servers up and down the hierarchy
  - a given server can be replicated, such that multiple physical copies of the server exist
- Replication is useful for heavily used servers, such as root servers that provide information about top-level domains
  - administrators must guarantee that all copies are coordinated so they provide exactly the same information

Name Resolution

- Given name and type, find the IP address
- Reverse resolution
  - –Given IP address, find name
- Resolution methods
  - –Recursive resolution
  - –Iterative resolution

Recursive resolution

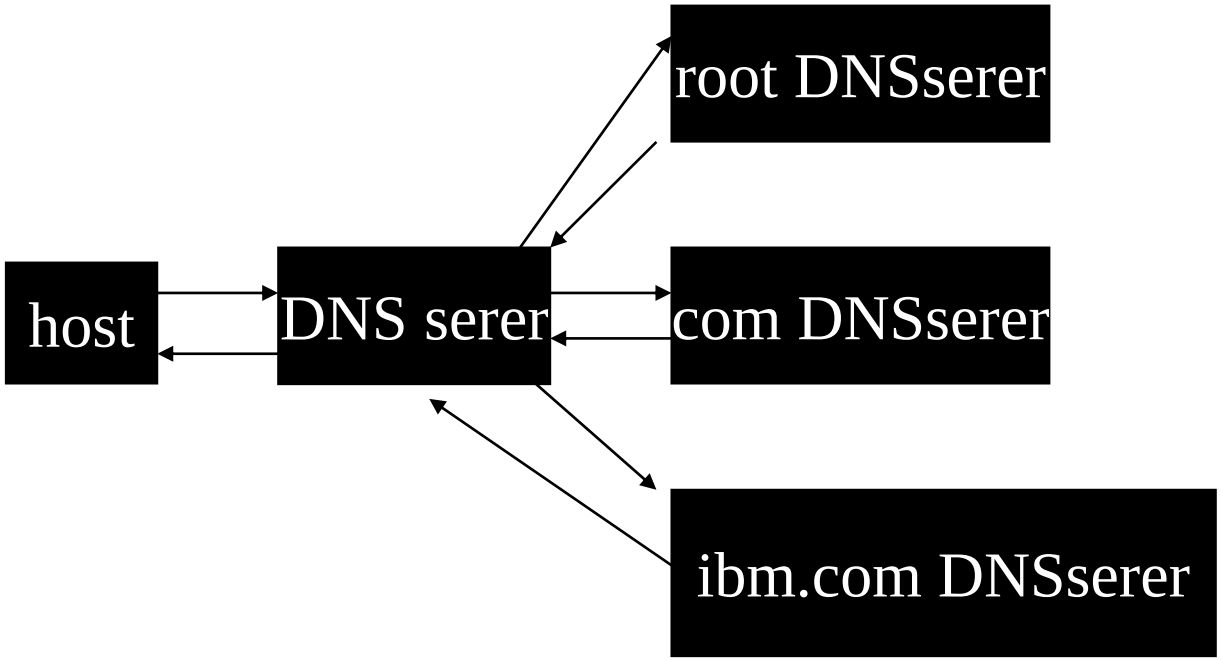
- If the name server does not have the requested information, it goes and finds it somewhere, then reports back.
- Example: how to resolve www.ibm.com

Iterative resolution

- An Interative request will respond with the answer(resolved) or the name of a DNS that may be able to answer the question.
- Iterative resolution is usually used between Domain Name Servers.
- Recursive resolution is usually used by the host

Name resolution

- Example: how to resolve www.ibm.com



Caching in DNS Servers

- The locality of reference principle that forms the basis for caching applies to the Domain Name System in two ways:
  - Spatial: A user tends to look up the names of local computers more often than the names of remote computers
  - Temporal: A user tends to look up the same set of domain names repeatedly
- DNS exploits spatial locality
  - a name resolver contacts a local server first
- To exploit temporal locality
  - a DNS server caches all lookups
- Algorithm 4.4 summarizes the process

Caching in DNS Servers

- According to the algorithm, when a request arrives for a name outside the set for which the server is an authority
  - further client-server interaction results
- The server temporarily becomes a client of another name server
- When the other server returns an answer
  - the original server caches the answer and sends a copy of the answer back to the resolver from which the request arrived
- In addition to knowing the address of all servers down the hierarchy
  - each DNS server must know the address of a root server
- How long should items be cached?
  - if an item is cached too long, the item will become stale
  - DNS specify a cache timeout for each item

Algorithm 4.4

Given:

A request message from a DNS name resolver

Provide:

A response message that contains the address

Method:

```
Extract the name, N, from the request
if ( server is an authority for N ) {
    Form and send a response to the requester;
else if ( answer for N is in the cache ) {
    Form and send a response to the requester;
else {   /* Need to look up an answer */
    if ( authority server for N is known ) {
        Send request to authority server;
    } else {
        Send request to root server;
    }
    Receive response and place in cache;
    Form and send a response to the requester;
}
```

Algorithm 4.4 Steps a DNS server takes to resolve a name.

Types of DNS Entries

- Each entry in a DNS database consists of three items:
  - a domain name
  - a record type
    - The record type specifies how the value is to be interpreted
  - a value
- A query sent to a DNS server specifies both a domain name and a type
  - the server only returns a binding that matches the type of the query
- The principal type maps a domain name to an IP address
  - DNS classifies such bindings as type A
    - type A lookup is used by applications such as FTP, ping, or a browser
  - DNS supports several other types, including type MX
    - that specifies a Mail eXchanger
    - when it looks up the name in an email address, SMTP uses type MX

Types of DNS Entries

- Each entry in a DNS server has a type
- When a resolver looks up a name
  - the resolver specifies the type that is desired
  - the DNS server returns only entries that match the specified type

- It is possible for the corporation to divide the workload between separate computers
  - by mapping type A lookups to one computer and type MX lookups to another
    - Example hkr.se
  - By mapping domain name to different IP addresses at different times
    - Example: www.google.com

Aliases and CNAME Resource Records

- The DNS offers a CNAME
  - it is analogous to a symbolic link in a file system; the entry provides an alias for another DNS entry
  - aliases can be useful; suppose Foobar Corporation has two computers, named as:

    hobbes.foobar.com and *calvin.foobar.com*
- Suppose that foobar decides to run a web server on computer hobbes, and wants to follow the convention of using the name www
- Organization could choose to rename computer hobbes
- A much easier solution exists:
  - the organization can create a CNAME entry for *www.foobar.com* that points to hobbes
- Whenever a resolver sends a request for *www.foobar.com*, the server returns the address of computer hobbes

Aliases and CNAME Resource Records

- The use of aliases is especially convenient
  - it permits an organization to change the computer used for a particular service without changing the names or addresses:
    - For example, Foobar Corporation can move its web service from hobbes → calvin
    - changing the CNAME record in the DNS server, the two computers retain their original names and IP addresses
- The use of aliases also allows an organization to associate multiple aliases with a single computer
  - Thus, Foobar Corporation can run an FTP server and a web server on the same computer, and can create CNAME records:

    www.foobar.com

    ftp.foobar.com

Abbreviations and the DNS

- DNS does not incorporate abbreviations - a server only responds to a full name
- Most resolvers can be configured with a set of suffixes that allow a user to abbreviate names
  - For example, each resolver at Foobar Corporation might be programmed to look up a name twice:
    - once with no change and once with the suffix foobar.com appended
- If a user enters a full domain name
  - the local server will return the address, and processing will proceed
- If a user enters an abbreviated name
  - it will first try to resolve the name
  - and will receive an error because no such name exists
  - then it will try appending a suffix and looking up the resulting name

Host Files

- Text file lists host names and IP addresses
  - /etc/hosts on UNIX/Linux
  - Hosts c:\windows\system32\drivers\etc
- Alias
  - Nickname for a node's host name
- static

# IP Address	host name	aliases
132.55.78.109	bingo.games.com	bingo
132.55.78.110	parcheesi.games.com	parcheesi
132.55.78.111	checkers.games.com	checkers
132.55.78.112	darts.games.com	darts

An example of a host file

nslookup utility program

- Nslookup command in both unix and ms windows
  - Type help for how to use the command
- Given an email address like
  - Eric.chen@hkr.se
- How to locate the mail server ?
- Use command nslookup to look it up (MX type)
- Hkr.se, mx type ➔ hkr-se.mail.protection.outlook.com
- More on using nslookup, read http://support.microsoft.com/kb/200525

ipconfig program (MS Windows)

- Find usage of ipconfig by command
  ipconfig /?

  ipconfig /displaydns ➔ Displays DNS resolver cache

  ipconfig /flushdns ➔ clear DNS resolver cache

- **The Syntax and Options for Using the Ipconfig Diagnostic Utility for Network Connections**

Use WireShark to Analyze DNS Operations

- Follow the instruction given in Wireshark Lab: DNS to learn more on
  - Command nslookup
  - How DNS works

- About DNS
  - DNS uses UDP protocol as the transport service
  - DNS server port number is 53
- Reference

  DNS Message Header and Question Section Format
  http://www.tcpipguide.com/free/t_DNSMessageHeaderandQuestionSectionFormat.htm

Exercise on DNS

- Name resolution for www.studera.nu
  - what is this web site?
  - what is the top domain of this domain name?
  - is it the web server located in Sweden?
  - how to get its IP address?

Exercise on DNS

- Name resolution for www.studera.nu
  - what is this web site?
  - what is the top domain of this domain name?
  - is it the web server located in Sweden?
  - how to get its IP address?
    - using nslookup on local computer or https://network-tools.com/nslook/
  - Server location https://www.iplocation.net/

Dynamic DNS

- Provide a way to automatically update the mapping between a domain name to a changing IP address
- Read more at http://en.wikipedia.org/wiki/Dynamic_DNS

Electronic Mail

- One of the most widely used Internet applications
- Figure 4.11 illustrates the architecture of electronic email
- Email software is divided into two conceptually pieces:
  - An email interface application
    - A mechanism for a user to compose and edit outgoing messages as well as read and process incoming email
  - A mail transfer program
    - acts as a client to send a message to the mail server on the destination computer; the mail server accepts incoming messages and deposits each in the appropriate user's mailbox
- Algorithm 4.3 lists the steps taken to send an email

Electronic Mail

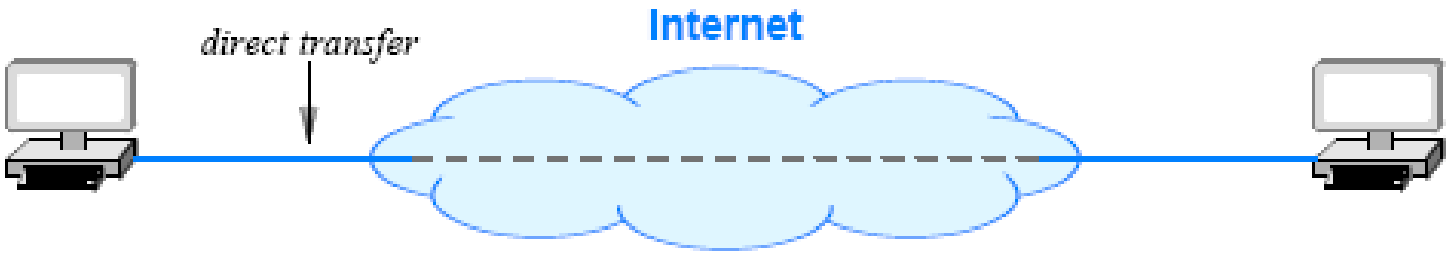


Figure 4.11 The original email configuration with direct transfer from a sender's computer directly to a recipient's computer.

Algorithm 4.3

Given:

Email communication from one user to another.

Provide:

Transmission of a message to the intended recipient.

Method:

User invokes interface application and generates an email message for user *x@destination.com*;

User's email interface program queues message for transfer;

Mail transfer program on user's computer examines the outgoing mail queue, and finds message;

Mail transfer program opens connection to *destination.com*;

Mail transfer program uses SMTP to transfer the message;

Mail transfer program closes connection;

Mail server on *destination.com* receives message and places a copy in user x's mailbox;

User x on *destination.com* runs mail interface program, which displays the user's mailbox, including the new message;

Algorithm 4.3 Steps taken to send email in the original paradigm.

Electronic Mail

- The specifications used for Internet email can be divided into three broad categories as Figure 4.12 lists

Type	Description
Transfer	A protocol used to move a copy of an email message from one computer to another
Access	A protocol that allows a user to access their mailbox and to view or send email messages
Representation	A protocol that specifies the format of an email message when stored on disk

Figure 4.12 The three types of protocols used with email.

Electronic Mail

- Mailbox
  - –Destination point for mails
  - –Can be storage or program
  - –Given unique address

E-mail Address

- Text string
- Specifies mail destination
- General form

  mailbox@computer

- *computer*
  - –Domain name of computer
  - –Actually type MX
- *mailbox*
  - –Destination on the computer

The Simple Mail Transfer Protocol (SMTP)

- The Simple Mail Transfer Protocol (SMTP) is the standard protocol that a mail transfer program uses
- SMTP can be characterized as:
  - Follows a stream paradigm (TCP)
  - Uses **textual** control messages
  - Only transfers **text** messages (ASCII) – 7 bits/char
  - Terminates message with CRLF.CRLF
  - Allows a sender to specify recipients' names and check each name
  - Sends one copy of a given message
- SMTP has a restriction to send only textual content
  - MIME standard that allows email to include attachments such as graphic images or binary files
- SMTP can send a single message to multiple recipients
  - The protocol allows a client to list users and then send a single copy of a message for all users on the list

```
Server:   220 somewhere.com Simple Mail Transfer Service Ready

Client:   HELO example.edu
Server::  250 OK

Client:   MAIL FROM:<John_Q_Smith@example.edu>
Server:   250 OK

Client:   RCPT TO:<Mathew_Doe@somewhere.com>
Server:   550 No such user here

Client:   RCPT TO:<Paul_Jones@somewhere.com>
Server:   250 OK

Client:   DATA
Server:   354 Start mail input; end with <CR><LF>.<CR><LF>
Client:   ...sends body of mail message, which can contain
Client:   ...arbitrarily many lines of text
Client:   <CR><LF>.<CR><LF>
Server:   250 OK

Client:   QUIT
Server:   221 somewhere.com closing transmission channel
```

Figure 4.13 An example SMTP session.

ISPs, Mail Servers, and Mail Access

- ISPs began offering email services
  - An ISP runs an email server and provides a mailbox for each user
    - each ISP provides interface that allows a user to access their mailbox
- Figure 4.14 illustrates the arrangement
- Email access follows one of two forms:
  - A special-purpose email interface application
  - A web browser that accesses an email web page

ISPs, Mail Servers, and Mail Access

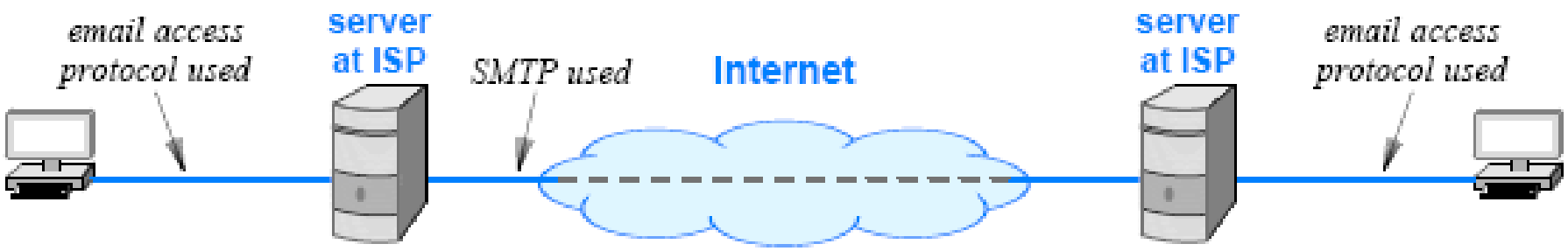


Figure 4.14 An email configuration where an ISP runs an email server and provides a user access to a mailbox.

ISPs, Mail Servers, and Mail Access

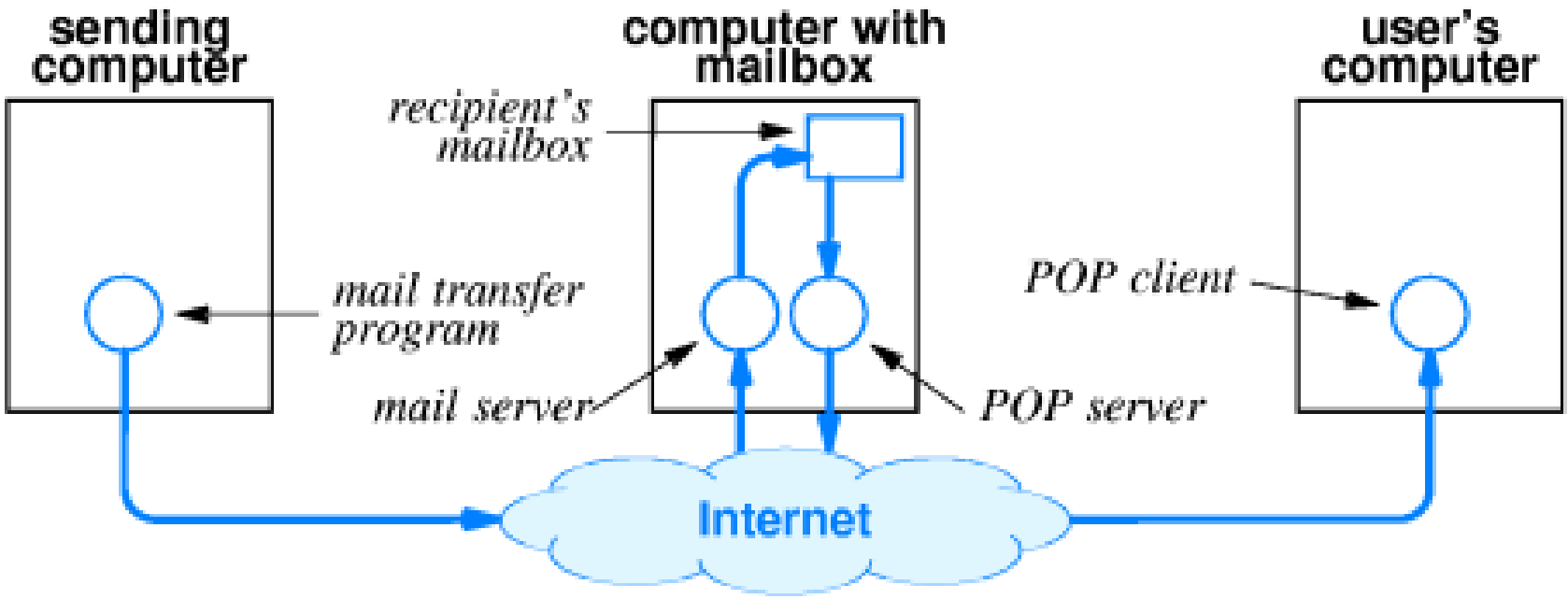
- The web browser approach is straightforward:
  - an ISP provides a special web page that displays messages from a user's mailbox
- The chief advantage of using a web page for email
  - ability to read email from any computer
  - a user does not need to run a special mail interface application
- Using a special mail application can download an entire mailbox onto a local computer, such as a laptop
  - when connected to the Internet, a user can run an email program that downloads an entire mailbox onto the laptop
  - the user can then process email when the laptop is disconnected from the Internet (e.g., while on an airplane).
  - once Internet connectivity is regained, it communicates with the server at the ISP to upload email the user has created and download any new email

Mail Access Protocols (POP, IMAP)

- A variety of mechanisms available for email access
  - Some ISPs provide free email access software to their subscribers
  - In addition, two standard email access protocols have been created
- Two protocols differ in many details
  - In particular, each provides its own authentication mechanism that a user follows to identify themselves

Acronym	Expansion
POP3	Post Office Protocol version 3
IMAP	Internet Mail Access Protocol

Illustration of POP



- Current version named POP3

Email Representation Standards (RFC2822, MIME)

- Two important email representation standards exist:
  - RFC (Request For Comments) 2822 Mail Message Format
  - Multi-purpose Internet Mail Extensions (MIME)
- RFC 2822 Mail Message Format:
  - takes its name from the IETF standards document RFC 2822
  - a mail message is represented as a text file and consists of
    - a header section
    - a blank line
    - and a body
  - Header lines each have the form:

    Keyword: information

    - where the set of keywords is defined to include From:, To:, Subject:, Cc:

MIME

Although Internet e-mail only transfers text, MIME can be used to transport binary data by encoding it in printed characters. A MIME mail message includes additional information that a receiving application uses to decode the message.

Email Representation Standards (RFC2822, MIME)

- Multi-purpose Internet Mail Extensions (MIME)
  - The MIME standard extends the functionality of email to allow the transfer of non-text data in a message
  - MIME specifies how a binary file can be encoded into printable characters, included in a message, and decoded by the receiver
  - The Base64 encoding standard is most popular, but MIME does not restrict encoding to a specific form
    - MIME permits a sender/receiver to choose a convenient encoding
    - the sender includes additional lines in the header to specify encoding used
  - MIME allows a sender to divide a message into several parts and to specify an encoding for each part independently
    - a user can send a plain text message and attach a graphic image, a spreadsheet, and an audio clip, each with their own encoding

MIME Encoding

- *Data transmitted as printable characters to avoid translation and format errors.*
- *base64 encoding ($2^6 = 64$)*
  - *splits three bytes (24 bits) into four 6 bit units.*
  - *The 6 bit units are sent as upper and lower case letters, numbers, "+" and "/".*

  http://www.motobit.com/util/base64-decoder-encoder.asp
- *quoted- printable encoding*
  - *7- bit ASCII with characters above 127 sent as an equal sign followed by two hex digits.*

  http://www.motobit.com/util/quoted-printable-encoder.asp

Email Representation Standards (RFC2822, MIME)

- MIME adds two lines to an email header
  - one to declare that MIME has been used to create the message
  - and another to specify how MIME information is included in the body
  - For example, the header lines:

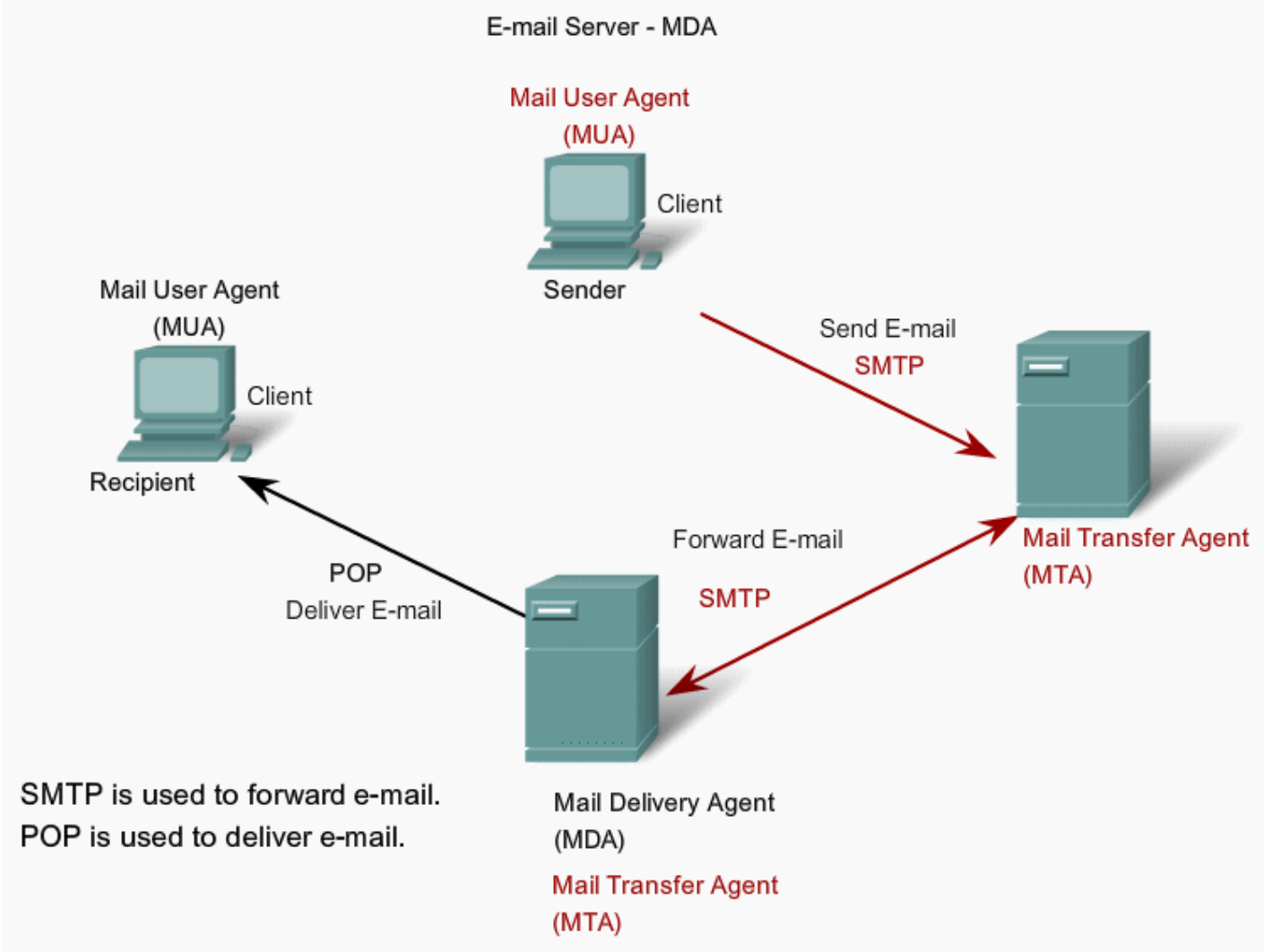
    MIME-Version: 1.0

    Content-Type: Multipart/Mixed; Boundary=Mime_separator
  - Mime_separator will appear in the message body before each part
- When MIME is used to send a standard text message

  Content-Type: text/plain
- MIME is backward compatible with email systems that do not understand the MIME standard or encoding
  - such systems have no way of extracting non-text attachments
  - they treat the body as a single block of text

Email delivery and protocols

- From Cisco course material



Review questions

- Check the review questions for Part 1

- How to view the header in a mail message? How to read the header to find the sender computer IP?
  - ref: http://whatismyipaddress.com/trace-email
- Read more on nslookup and ipconfig, and be able to use these tools