Computer Networks

Lecture 10

Application layer protocols

Application layer

The **application layer** is the OSI layer closest to the end user, which means both the OSI application layer and the user interact directly with the software application. This layer interacts with software applications that implement a communicating component.

Application-layer functions typically include:

- identifying communication partners,
- determining resource availability,
- synchronizing communication.

Application layer

- When identifying communication partners, the application layer determines the identity and availability of communication partners for an application with data to transmit.
- When determining resource availability, the application layer must decide whether sufficient network or the requested communication exists.
- In synchronizing communication, all communication between applications requires cooperation that is managed by the application layer

Application layer

Some examples of application-layer implementations include:

- E-mail
- Web service
- File transfer
- Network management
- Time synchronization

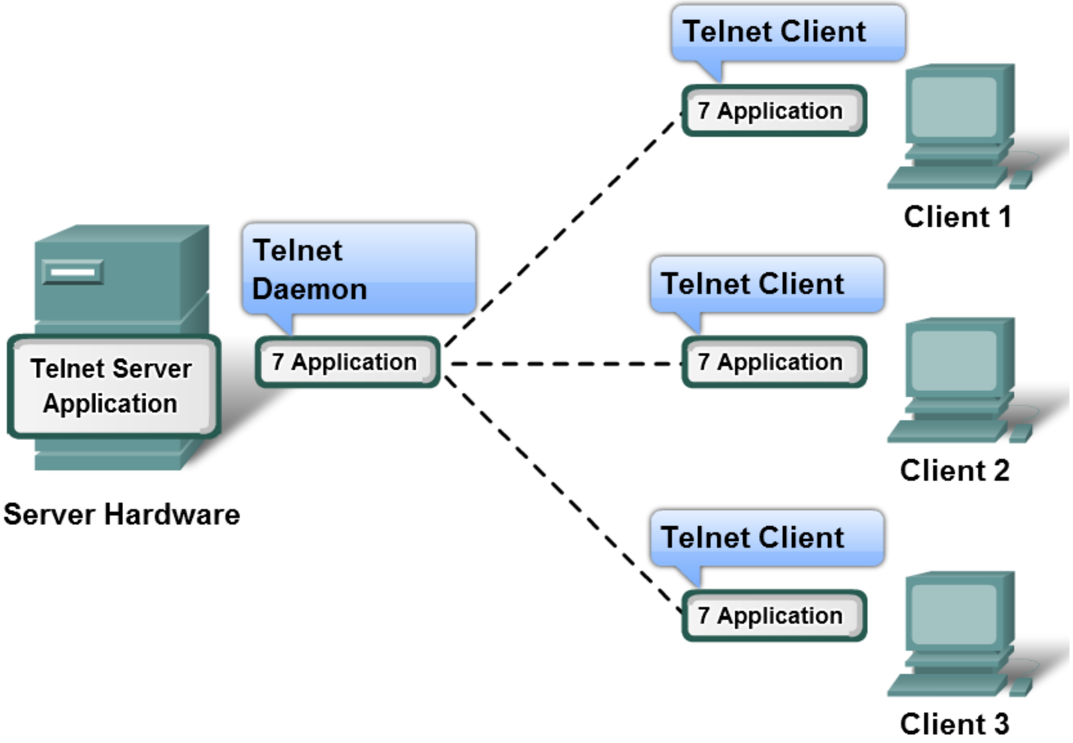
Some examples of protocols: HTTP, HTTPS, Telnet, FTP, ICMP, SMTP, POP3, DNS, SNMP, NTP, RDP ...

The protocols are generally defined by **Requests for Comments** (RFCs). The Internet Engineering Task Force maintains the RFCs as the standards for the TCP/IP suite.

Application layer

A **single application** may employ **many different** supporting Application layer services; thus what appears to the user as one request for a web page may, in fact, amount to dozens of individual requests. And for each request, multiple processes may be executed.

For example, a client may require several individual processes to formulate just one request to a server.



E- mail

E-mail service was developed more than 30 years ago. It was started to use in ARPANET project.

E-mail specification is provided in RFC 822.

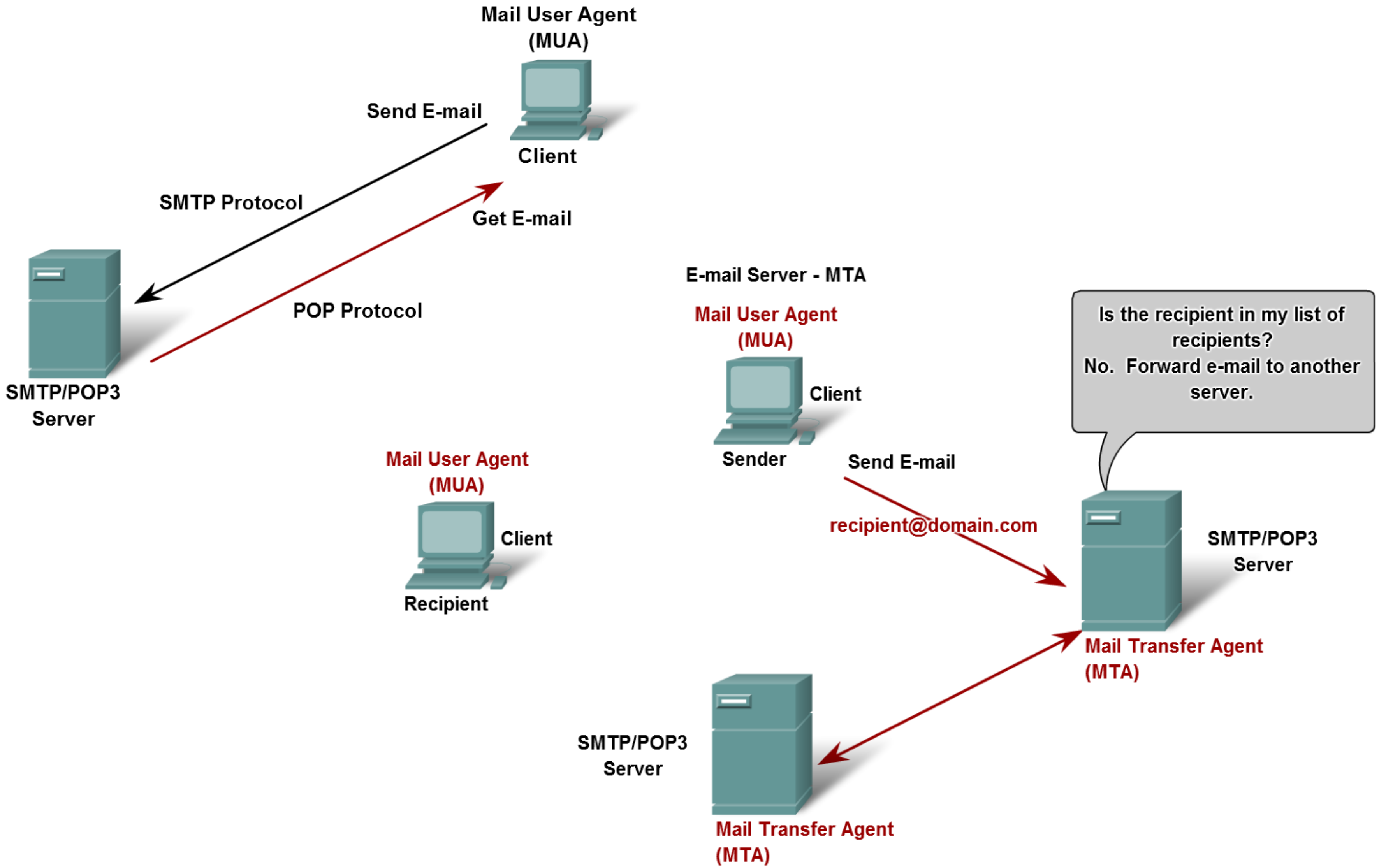
E-mail system consists of three parts:

1. Mail Transfer Agent (MTA)
2. Mail Delivery Agent (MDA)
3. Mail User Agent (MUA)

The Mail Transfer Agent (MTA) process is used to forward e-mail. The MTA receives messages from the MUA or from another MTA and based on the message header, it determines how a message has to be forwarded to reach its destination.

If the mail is addressed to a user whose mailbox is on the local server, the mail is passed to the MDA. If the mail is for a user not on the local server, the MTA routes the e-mail to the MTA on the appropriate server.

E-mail


Mail User Agent
(MUA)
Client
Send E-mail
SMTP Protocol
Get E-mail
POP Protocol
SMTP/POP3
Server
E-mail Server - MTA
Mail User Agent
(MUA)
Client
Sender
Is the recipient in my list of recipients?
No. Forward e-mail to another server.
Send E-mail
recipient@domain.com
SMTP/POP3
Server
Mail User Agent
(MUA)
Client
Recipient
Mail Transfer Agent
(MTA)
SMTP/POP3
Server
Mail Transfer Agent
(MTA)

E-mail

E-mail header (RFC 822)

Name of Fields	Meaning
To:	Receiver
Cc:	Carbon copy
From:	Sender
Bcc:	Blind carbon copy
Sender:	Sender’s e-mail address
Received:	Forward Path
Return-Path:	Return Path

MIME

MIME (Multi-Purpose Internet Mail Extensions) is an extension of the original Internet e-mail protocol that lets people use the protocol to exchange different kinds of data files on the Internet: audio, video, images, application programs, and other kinds, as well as the ASCII text handled in the original protocol, the Simple Mail Transfer Protocol (SMTP).

RFC 1341; RFC 1521 specification.

Internet Mail Extensions (**MIME**) extends the format of email to support:

- Text in character sets other than ASCII
- Non-text attachments
- Message bodies with multiple parts
- Header information in non-ASCII character sets

MIME

MIME header

Name of the Field	Meaning
MIME-Version:	MIME version
Content-Description:	Description
Content-Id:	Unique ID
Content-Transfer-Encoding:	Coding type (7 bits, 8 bits, base64)
Content-Type:	Type (text/plain; multipart/mixed)

Content-Type: *Text/richtext, Image, Audio, Video, Application, Postscript, Message, Multipart*

SMTP

SMTP (Simple Mail Transfer Protocol) sends e-mail to receiver.

SMTP by default uses TCP port 25. SMTP connections secured by SSL, known as SMTPS, default to port 465

SMTP is specified in RFC 821 …RFC 5321.

Algorithm

Email is submitted by a mail client (MUA, mail user agent) to a mail server (MSA) using SMTP. The boundary MTA has to locate the target host. It uses the DNS to look up the mail exchanger record (MX record) for the recipient's domain. The returned MX record contains the name of the target host. The MTA next connects to the exchange server as an SMTP client. Once the MX target accepts the incoming message, it hands it to a mail delivery agent (MDA) for local mail delivery. Once delivered to the local mail server, the mail is stored for batch retrieval by authenticated mail clients (MUAs).

SMTP

SMTP is a **connection-oriented**, **text-based** protocol in which a mail sender communicates with a mail receiver by issuing command strings and supplying necessary data over a reliable ordered data stream channel, typically TCP connection.

An SMTP session consists of commands originated by an SMTP client (the initiating agent, sender, or transmitter) and corresponding responses from the SMTP server (the listening agent, or receiver) so that the session is opened, and session parameters are exchanged.

A session may include zero or more SMTP transactions.

SMTP

An SMTP transaction consists of three command/reply sequences:

1. **MAIL command** is used to establish the return address (Return-Path, reverse-path, bounce address).
2. **RCPT command** is used to establish a recipient of this message. This command can be issued multiple times, one for each recipient. These addresses are also part of the envelope.
3. **DATA** is uded to signal the beginning of the message text. It consists of a message header and a message body separated by an empty line. DATA is actually a group of commands, and the server replies twice:
    - once to the DATA command proper, to acknowledge that it is ready to receive the text, and
    - the second time after the end-of-data sequence, to either accept or reject the entire message

POP3

POP3 (Post Office Protocol v.3, 110 port) is used by local e-mail clients to retrieve e-mail from a remote server over a TCP/IP connection.

POP has been developed through several versions, with version 3 (POP3) being the current standard. Usually is used together with SMTP.

POP3 specifications are provided in RFC 1225, 1939.

POP3 has commands to login to e-mail server and to logoff, retrieve e-mails, delete e-mails.

POP3 supports encrypting:

- **TLS** (transport layer security)
- **SSL** (secure socket layer), 995 port

POP3 dialog

POP3 dialog status:

1. **Authorization**
   Client makes authorization procedure.

2. **Transaction**
   Client receives information about e-mail box status, receives or delete e-mail.

3. **Renewal**
   Server deletes e-mails and finishes session.

IMAP

Internet Message Access Protocol (IMAP) is a protocol for e-mail retrieval and storage developed in 1986 at Stanford University as an alternative to POP.

IMAP, unlike POP, specifically allows multiple clients simultaneously connected to the same mailbox, and through flags stored on the server, different clients accessing the same mailbox at the same or different times can detect state changes made by other clients.

The current version, IMAP version 4 is defined by RFC 3501.

An IMAP server typically listens on well-known port 143. IMAP over SSL (IMAPS) is assigned well-known port number 993.

IMAP

IMAP supports both on-line and off-line modes of operation. E-mail clients using IMAP generally leave messages on the server until the user explicitly deletes them. This and other characteristics of IMAP operation allow multiple clients to manage the same mailbox.

Most e-mail clients support IMAP in addition to POP3 to retrieve messages. IMAP offers access to the mail storage. Clients may store local copies of the messages, but these are considered to be a temporary cache.

IMAP

Incoming e-mail messages are sent to an e-mail server that stores messages in the recipient's e-mail box. The user retrieves the messages with an e-mail client that uses one of a number of e-mail retrieval protocols.

Some clients and servers preferentially use vendor-specific, proprietary protocols, but most support SMTP for sending e-mail and POP and IMAP for retrieving e-mail, allowing interoperability with other servers and clients.

For example, Microsoft's Outlook client uses MAPI, a Microsoft proprietary protocol to communicate with a Microsoft Exchange Server.

HTTP

HTTP (Hypertext Transfer Protocol, 80 port) – The Hypertext Transfer Protocol (HTTP), one of the protocols in the TCP/IP suite, was originally developed to publish and retrieve HTML pages and is now used for distributed, collaborative information systems. HTTP is used across the WWW for data transfer and is one of the most used application protocols.

HTTP specifies a **request/response protocol**.

When a client, typically a web browser, sends a request message to a server, the HTTP protocol defines the message types the client uses to request the web page and also the message types the server uses to respond.

HTTP

HTTP specifies a request/response protocol. When a client, typically a web browser, sends a request message to a server, the HTTP protocol defines the message types the client uses to request the web page and also the message types the server uses to respond. The three common message types are GET, POST, and PUT.

GET is a client request for data. A web browser sends the GET message to request pages from a web server. As shown in the figure, once the server receives the GET request, it responds with a status line, such as HTTP/1.1 200 OK, and a message of its own, the body of which may be the requested file, an error message, or some other information.

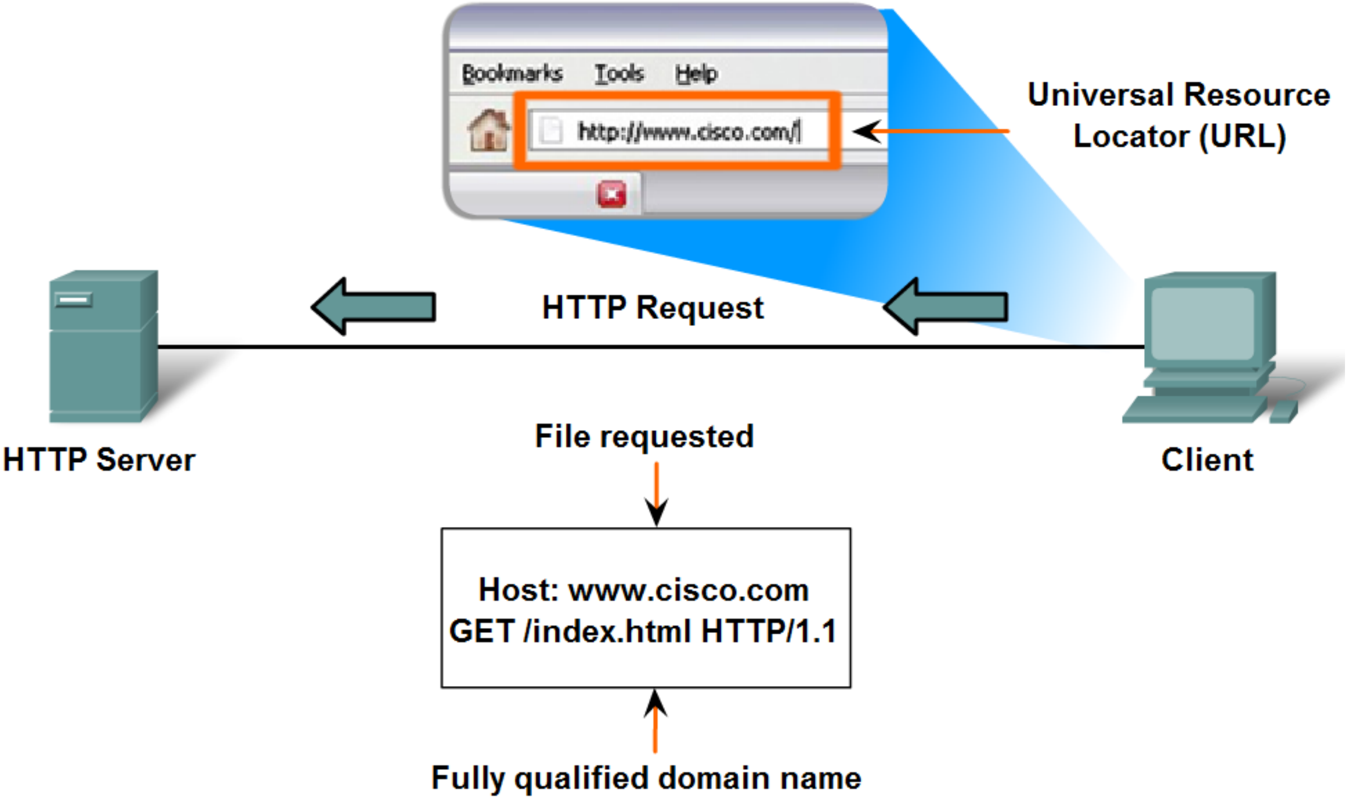
POST and PUT are used to send messages that upload data to the web server. For example, when the user enters data into a form embedded in a web page, POST includes the data in the message sent to the server.

PUT uploads resources or content to the web server.

HTTP is not a secure protocol. The POST messages upload information to the server in plain text that can be intercepted and read. Similarly, the server responses, typically HTML pages, are also unencrypted.

For secure communication across the Internet, the Secure HTTP (HTTPS) protocol is used.

HTTP



Universal Resource Locator (URL)

HTTP Request

HTTP Server

Client

File requested

Host: www.cisco.com
GET /index.html HTTP/1.1

Fully qualified domain name

Entering 'http://www.cisco.com' in the address bar of a web browser generates the HTTP 'GET' Message.

HTTP request methods

GET Requests a representation of the specified resource. Requests using GET should only retrieve data and should have no other effect.

POST Requests that the server accept the entity enclosed in the request as a new subordinate of the web resource identified by the URI. The data POSTed might be, for example, an annotation for existing resources; a message for a bulletin board

PUT Requests that the enclosed entity be stored under the supplied URI. If the URI refers to an already existing resource, it is modified; if the URI does not point to an existing resource, then the server can create the resource with that URI

DELETE Deletes the specified

TRACE Echoes back the received request so that a client can see what (if any) changes or additions have been made by intermediate servers

HEAD Asks for the response identical to the one that would correspond to a GET request, but without the response body.

HTTP request

Example of HTTP client request:

```
GET / HTTP/1.1
Host: www.google.com
```

Server response:

```
HTTP/1.1 200 OK
Content-Length: 3059
Server: GWS/2.0 Date: Sat, 12 Jan 2013 14:49:31 GMT
Content-Type: text/html
Cache-control: private
Set-Cookie:
PREF=ID=73d4aef52e57bae9:TM=1042253044:LM=1042253044
:S=SMCc_HRPCQiqy X9j; expires=Sun, 17-Jan-2038
19:14:07 GMT; path=/; domain=.google.com
Connection: keep-alive
```

FTP protocol

The File Transfer Protocol (FTP) was developed to allow for file transfers between a client and a server. An FTP client is an application that runs on a computer that is used to push and pull files from a server running the FTP daemon (FTPd).

To successfully transfer files, FTP requires two connections between the client and the server: one for commands and replies, the other for the actual file transfer.

The client establishes the first connection to the server on TCP port 21. This connection is used for control traffic, consisting of client commands and server replies.

FTP

The client establishes the second connection to the server over TCP port 20. This connection is for the actual file transfer and is created every time there is a file transferred.

The file transfer can happen in either direction. The client can download (pull) a file from the server or, the client can upload (push) a file to the server.

Transfer methods

- **Block**. File is transferred by blocks.
- **Stream**. Data are transferred as stream of bits.
- **ASCII**. File is transferred as text.
- **IMAGE**. Stream of packets (8 bits).

ICMP

Internet Control Message Protocol (ICMP) is one of the main protocols of the Internet Protocol Suite. It is used by network devices, like routers, to send error messages indicating, for example, that a requested service is not available or that a host or router could not be reached. ICMP can also be used to relay query messages.

ICMP differs from transport protocols such as TCP and UDP in that it is not typically used to exchange data between systems, nor is it regularly employed by end-user network applications (with the exception of some diagnostic tools like ping and traceroute).

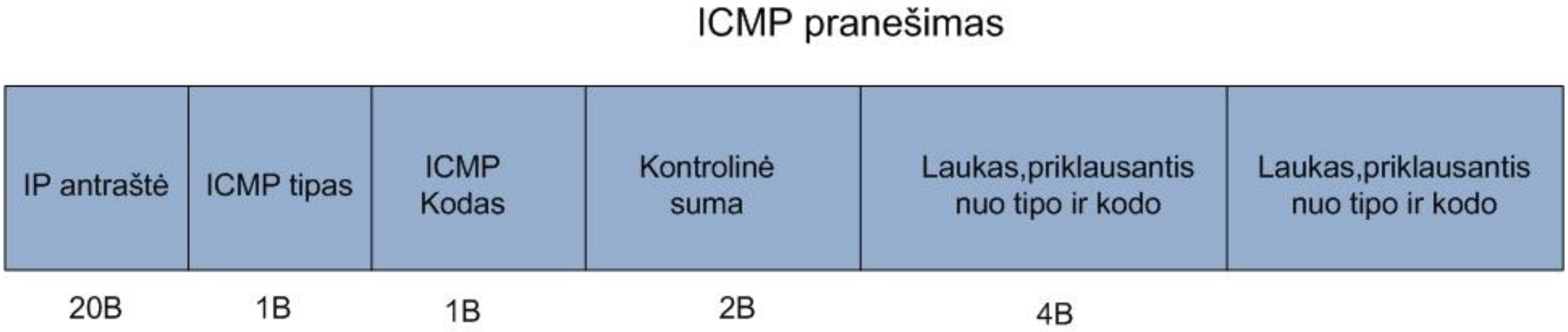
ICMP

ICMP is used to:

- Discover route of the IP packets
- Evaluate network throughput
- Measure round trip time
- Identify destination IP address

ICMP message

Structure of ICMP message:



ICMP pranešimas

IP antraštė	ICMP tipas	ICMP Kodas	Kontrolinė suma	Laukas,priklausantis nuo tipo ir kodo	Laukas,priklausantis nuo tipo ir kodo
20B	1B	1B	2B	4B	

Tipas – pranešimo tipo kodas (0 – echo atsakymas, 3 – mazgas nepasiekiamas, 8 – echo užklausa, 11-baigėsi gyvavimo laikas, 12 – paketo parametrų problema ir t.t.)

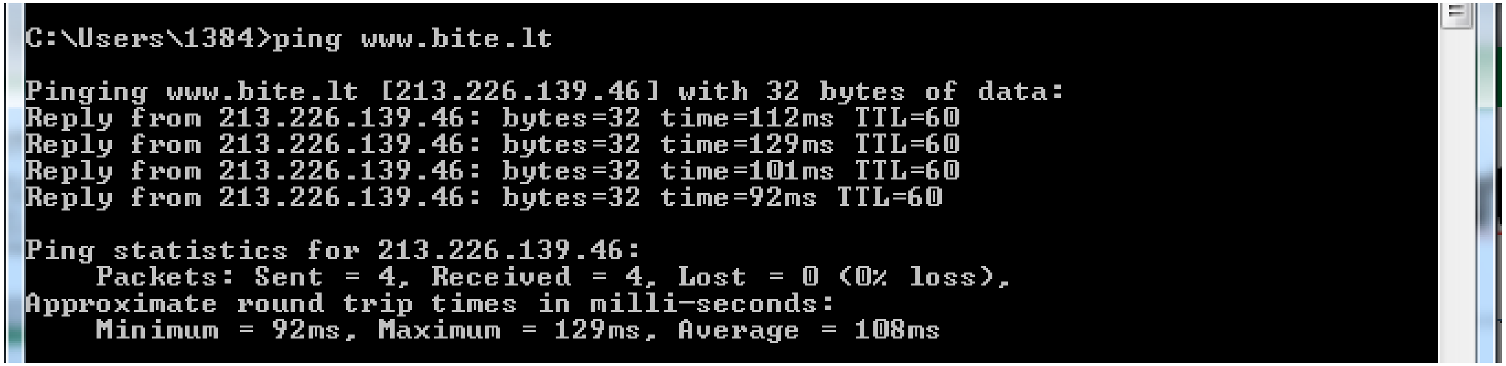
Kodas – smulkesnis pranešimo tipo apibūdinimas – kodas. (0 – tinklas nepasiekiamas, 1 – mazgas nepasiekiamas, 2 – protokolas nepasiekiamas, 3 – prievadas nepasiekamas ir t.t.)

Kontrolinė suma – ICMP pranešimo kontrolinė suma

ping

Ping is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network and to measure the round-trip time for messages sent from the originating host to a destination computer.

Ping operates by sending ICMP *echo request* packets to the target host and waiting for an ICMP response. In the process it measures *round-trip time* and records any packet loss. The results of the test are printed in the form of a statistical summary of the response packets received, including the minimum, maximum, and the mean round-trip times, and sometimes the standard deviation of the mean.



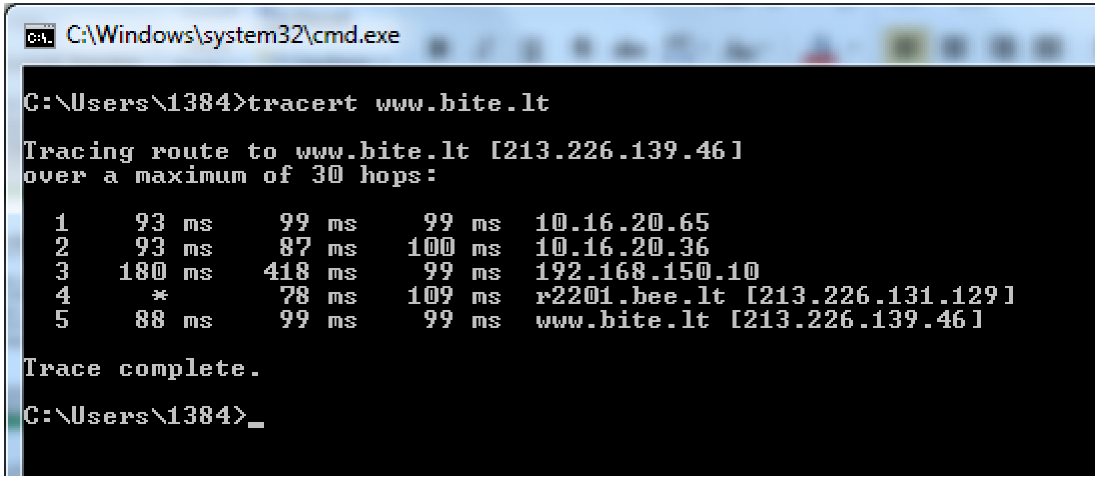
```
C:\Users\1384>ping www.bite.lt

Pinging www.bite.lt [213.226.139.46] with 32 bytes of data:
Reply from 213.226.139.46: bytes=32 time=112ms TTL=60
Reply from 213.226.139.46: bytes=32 time=129ms TTL=60
Reply from 213.226.139.46: bytes=32 time=101ms TTL=60
Reply from 213.226.139.46: bytes=32 time=92ms TTL=60

Ping statistics for 213.226.139.46:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 92ms, Maximum = 129ms, Average = 108ms
```

tracert (traceroute)

traceroute is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network. The history of the route is recorded as the round-trip times of the packets received from each successive host (remote node) in the route (path); the sum of the mean times in each hop indicates the total time spent to establish the connection. Traceroute proceeds unless all (three) sent packets are lost more than twice, then the connection is lost and the route cannot be evaluated.



```
C:\Users\1384>tracert www.bite.lt

Tracing route to www.bite.lt [213.226.139.46]
over a maximum of 30 hops:

  1    93 ms    99 ms    99 ms  10.16.20.65
  2    93 ms    87 ms   100 ms  10.16.20.36
  3   180 ms   418 ms    99 ms  192.168.150.10
  4     *       78 ms   109 ms  r2201.bee.lt [213.226.131.129]
  5    88 ms    99 ms    99 ms  www.bite.lt [213.226.139.46]

Trace complete.

C:\Users\1384>_
```

SNMP

Simple Network Management Protocol (SNMP) is an Internet-standard protocol for managing devices on IP networks.

Devices that typically support SNMP include routers, switches, servers, workstations, printers, modem racks and more.

SNMP is used mostly in network management systems to monitor network-attached devices for conditions that warrant administrative attention.

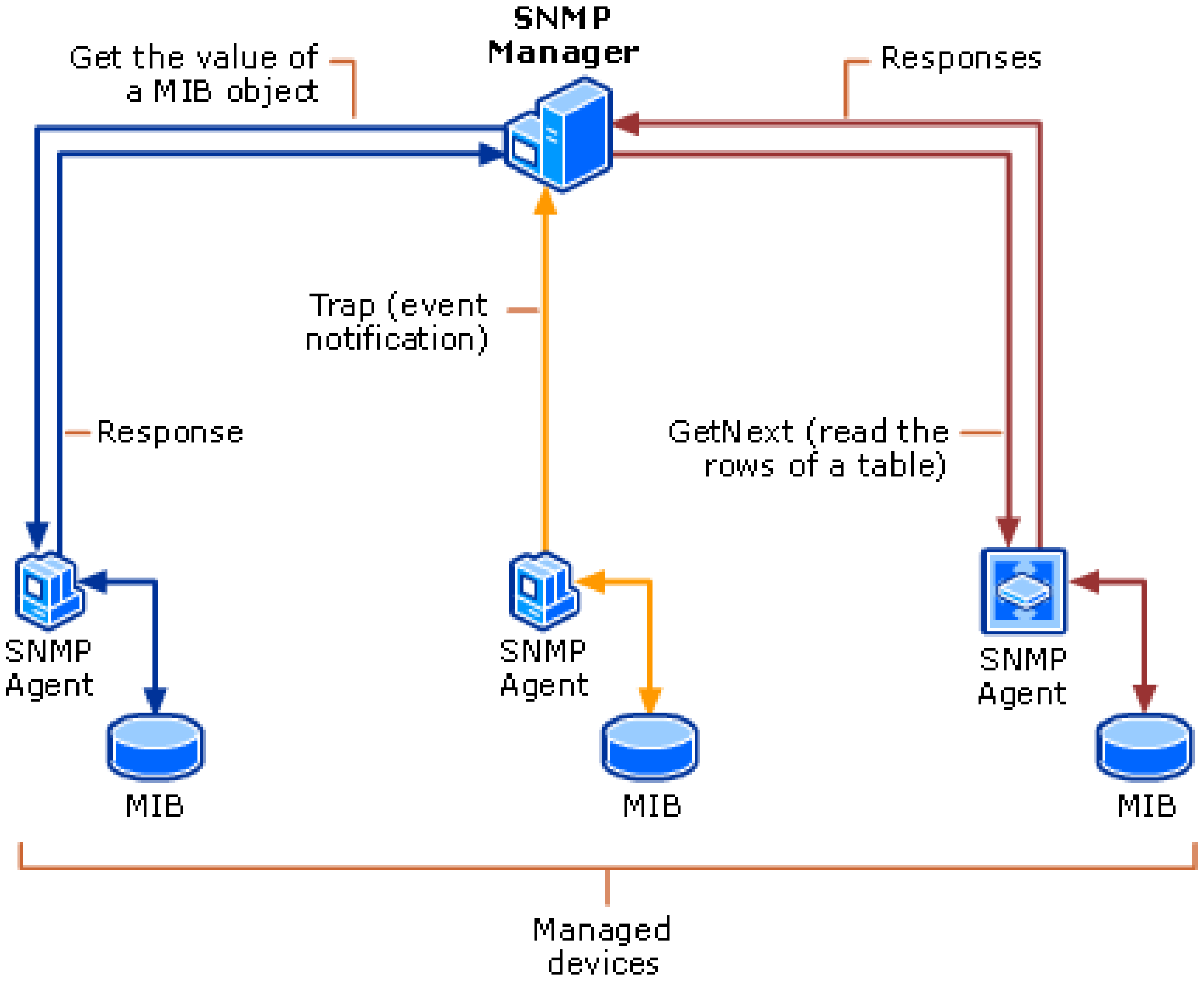
SNMP

An SNMP-managed network consists of three key components:

- Managed device that has agent (software which runs on managed devices)
- Network management station (NMS) — software which runs on the manager
- MIB - Management Information Base that runs on managed device.

MIBs describe the structure of the management data of a device subsystem; they use a hierarchical namespace containing object identifiers (OID). Each OID identifies a variable that can be read or set via SNMP. MIBs use the notation defined by Structure of Management Information Version 2.0.

SNMP


SNMP
Manager
Get the value of
a MIB object
Responses
Trap (event
notification)
Response
GetNext (read the
rows of a table)
SNMP
Agent
SNMP
Agent
SNMP
Agent
MIB
MIB
MIB
Managed
devices

Telnet

Telnet is an application protocol used on the Internet or local area networks to provide a bidirectional interactive text-oriented communication facility using a virtual terminal connection.

User data is interspersed in-band with Telnet control information in an 8-bit byte oriented data connection over the TCP.

Telnet was developed in 1968 beginning with RFC 15, extended in RFC 854, and standardized as Internet Engineering Task Force (IETF) Internet Standard STD 8, one **of the first Internet standards**.

RDP

Remote Desktop Protocol (RDP) is a proprietary protocol developed by Microsoft, which provides a user with a graphical interface to connect to another computer over a network connection. The user employs RDP client software for this purpose, while the other computer must run RDP server software.

By default, the server listens on TCP port 3389 and UDP port 3389.

Microsoft currently refers to their official RDP server software as Remote Desktop Connection, formerly "Terminal Services Client”

SSH

Secure Shell, or SSH, is a cryptographic (encrypted) network protocol for initiating **text-based** shell sessions on remote machines in a secure way.

This allows a user to run commands on a machine's command prompt without them being physically present near the machine. It also allows a user to establish a secure channel over an insecure network in a client-server architecture, connecting an SSH client application with an SSH server.

Common applications include remote command-line login and remote command execution, but any network service can be secured with SSH.

The protocol specification distinguishes between two major versions, referred to as SSH-1 and SSH-2.