



# Transport Over IP

---

รศ.ดร. อนันต์ พลเพิ่ม

Asso. Prof. Anan Phonphoem, Ph.D.

[anan.p@ku.ac.th](mailto:anan.p@ku.ac.th)

<http://www.cpe.ku.ac.th/~anan>

Computer Engineering Department

Kasetsart University, Bangkok, Thailand



# Outline

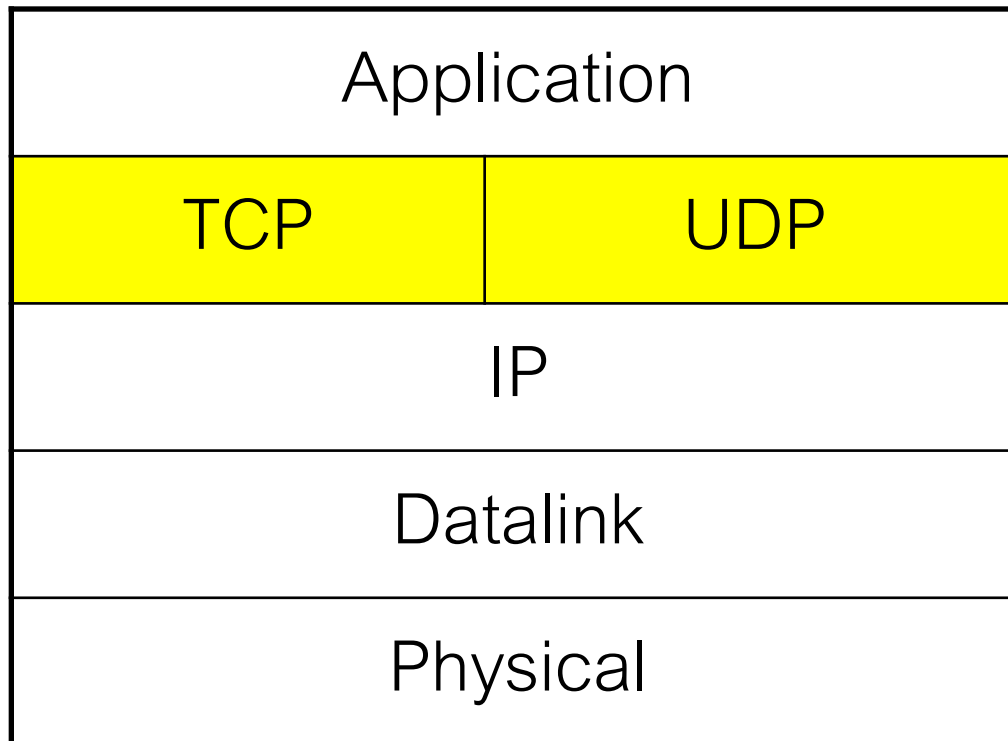
---

- **Transport Layer**
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)
- Stream Control Transmission Protocol (SCTP)
- Real-Time Transport Protocol (RTP)

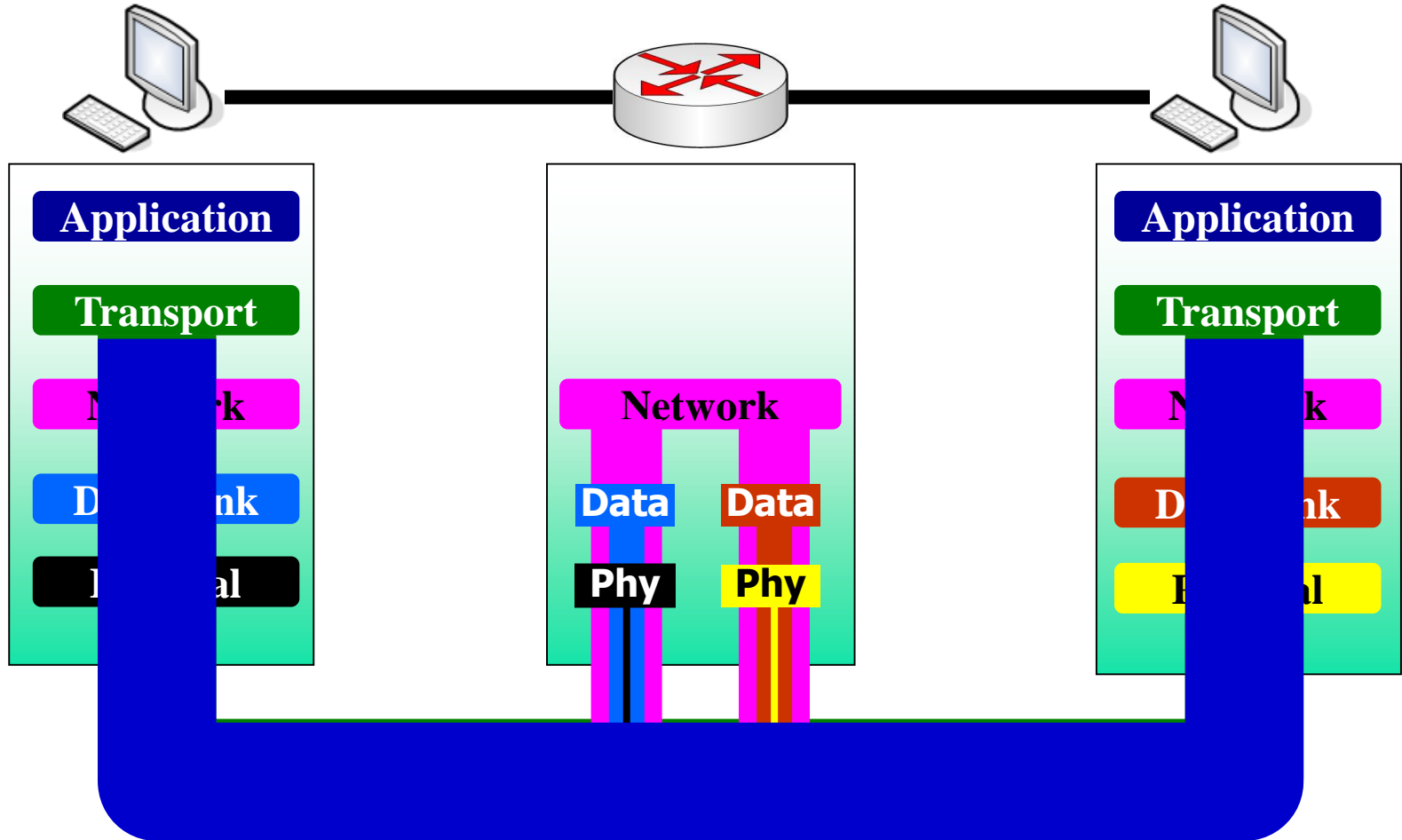


# TCP/IP protocol Suite

---



# TCP/IP protocol Suite





# Issues in Network Layer

---

- No/Little error detection
- No attempt to correct
  - Retransmission
- Connectionless
- No handshaking
- No verification
- No flow control



# Transport Protocol

---

- Set of rules
  - for exchange control message / data
- End-to-End
- Application may by-pass Transport Protocol
  - build functions on top of IP
  - reduce overhead
  - application runs on switch/router that has no Transport Layer



# Transport Protocols

---

- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)
- Stream Control Transmission Protocol (SCTP)
- Real-Time Transport Protocol (RTP)



# Ports and Addresses

---

- IP address is the end point of identifier
  - source / destination
- On same machine (same IP)
  - needs identifier for each application
  - multiple applications (e.g. 3 Ftp sessions)
- “Port”
  - 16-bit number (65,536 ports) for each IP



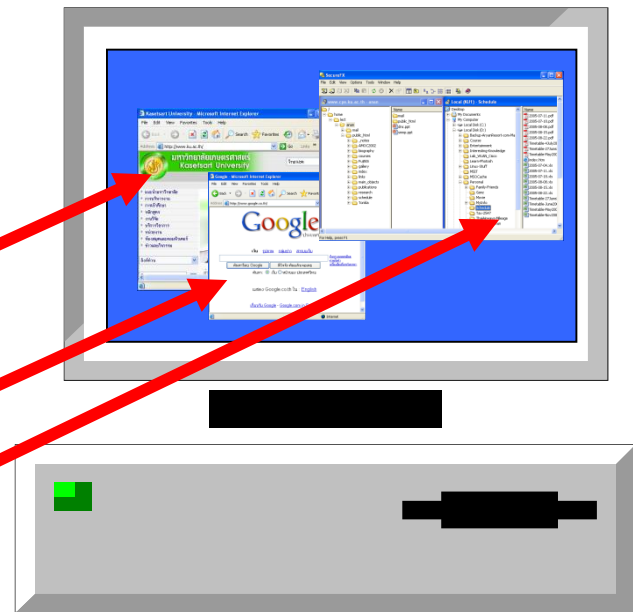
# Ports and Addresses

- Unique connection identifier  
[source IP] + [source port]  
[dest. IP] + [dest. port]

Web #1 [158.108.1.2:80]

Web #2 [158.108.1.2:8080]

Ftp [158.108.1.2:20]



IP address: 158.108.1.2



# Ports and Addresses

---

- Known destination IP
- Concern Issues
  - destination port ?
  - listen port ?
- IANA divides ports into three ranges
  - well-known ports
  - registered ports
  - dynamic / private ports



# Well-known ports

---

- 1 – 1023
- Used by server applications
- Use with restricted privileges (root)
- Assigned by IANA



# Well-known ports: Examples

Port Number	Description
80	WWW
25	SMTP
23	Telnet
22	SSH
21	ftp-control
20	ftp-data
161	SNMP



# Registered Ports

---

- 1024 – 49,151
- Used for server applications
  - Allow client to make contact
- Registered with IANA (for uniqueness)
- On most systems, registered ports can be used by ordinary users
- Ex. 8080: HTTP alternate
  - commonly used for Web proxy and caching server
  - or for running a Web server as a non-root user



# Registered Ports: Examples

Port Number	Description	
1188	hp-webadmin	HP Web Admin
1376	ibm-pps	IBM Person to Person Software
1381	apple-licman	Apple Network License Manager
1741	cisco-net-mgmt	Cisco network managment

# Example from

<http://www.iana.org/assignments/port-numbers>

```
alias          1187/tcp    Alias Service
alias          1187/udp    Alias Service
#             Paul Tokarchuk <ptokarch&alias.com> November 2004
hp-webadmin   1188/tcp    HP Web Admin
hp-webadmin   1188/udp    HP Web Admin
#             Lance Kind <lance_kind&hp.com>
unet          1189/tcp    Unet Connection
unet          1189/udp    Unet Connection
#             Anthony Stahler <imunfair@yahoo.com> November 2004
commlinux-avl 1190/tcp    CommLinx GPS / AVL System
commlinux-avl 1190/udp    CommLinx GPS / AVL System
#             Peter Johnson <peter&commlinux.com.au> November 2004
gpfs          1191/tcp    General Parallel File System
gpfs          1191/udp    General Parallel File System
#             Dave Craft <gpfs&ibm.com> November 2004
caids-sensor  1192/tcp    caids sensors channel
caids-sensor  1192/udp    caids sensors channel
#             Gregory Hostettler <ghostettler&caracal.ch> November 2004
fiveacross    1193/tcp    Five Across Server
fiveacross    1193/udp    Five Across Server
#             Glenn Reid <port-reg&fiveacross.com> November 2004
openvpn       1194/tcp    OpenVPN
openvpn       1194/udp    OpenVPN
#             James Yonan <jim&yonan.net> November 2004
rsf-1         1195/tcp    RSF-1 clustering
```



# Dynamic / Private ports

---

- 49,152 – 65,535
- Ephemeral ports (one day, temporary)
  - as-needed basis
  - freed up when done



# Ephemeral

- Nature
  - “An **ephemeral** waterbody is a wetland, spring, stream, river, pond or lake that only exists for a short period following precipitation or snowmelt.”
- Bio
  - “Many plants are adapted to an **ephemeral** lifestyle, in which they spend most of the year or longer as seeds before conditions are right for a brief period of growth and reproduction.”



Staircase Falls in [Yosemite National Park](#) only flows after heavy rainfall or snowmelt



# Ephemeral

---

- Network
  - “In computer networking technology, an **ephemeral port** is a TCP, UDP or SCTP port which is dynamically assigned to a client application for a short period of time (the duration of time the application is running).”
  - “This is in contrast to the “**well known**” ports which are typically statically assigned to a specific application or service.”



# Dynamic / Private ports

---

- 49,152 – 65,535
- Ephemeral ports (one day, temporary)
  - as-needed basis
  - freed up when done
- Assigned by local machine
  - Not register with IANA
- Never used for destination port **at start**
  - used by initiator as the return add.



# Connection-Oriented Transport

---

- Advantages
  - easy to communicate (if receiver is there!)
  - verify the real receiver
  - agree on some protocols
  - error correction
- Disadvantages
  - heavyweight protocol
  - sophisticated implementation (keep state)
  - consume high bandwidth for management
- TCP



# Connectionless Transport

---

- Datagram approach
- Advantages
  - rapid data transfer
  - no connection setup
  - no need for connection maintenances
- Disadvantages
  - not reliable (or fail)
  - out of order packets
  - application more complex (takes precautions)
- UDP



# Outline

---



## Transport Layer

- **User Datagram Protocol (UDP)**
- Transmission Control Protocol (TCP)
- Stream Control Transmission Protocol (SCTP)
- Real-Time Transport Protocol (RTP)

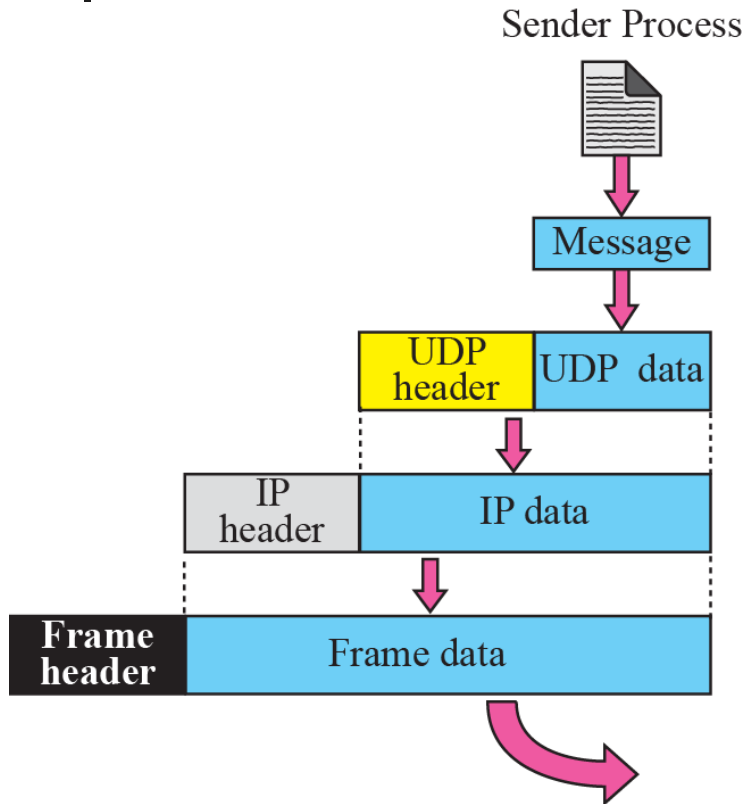


# User Datagram Protocol (UDP)

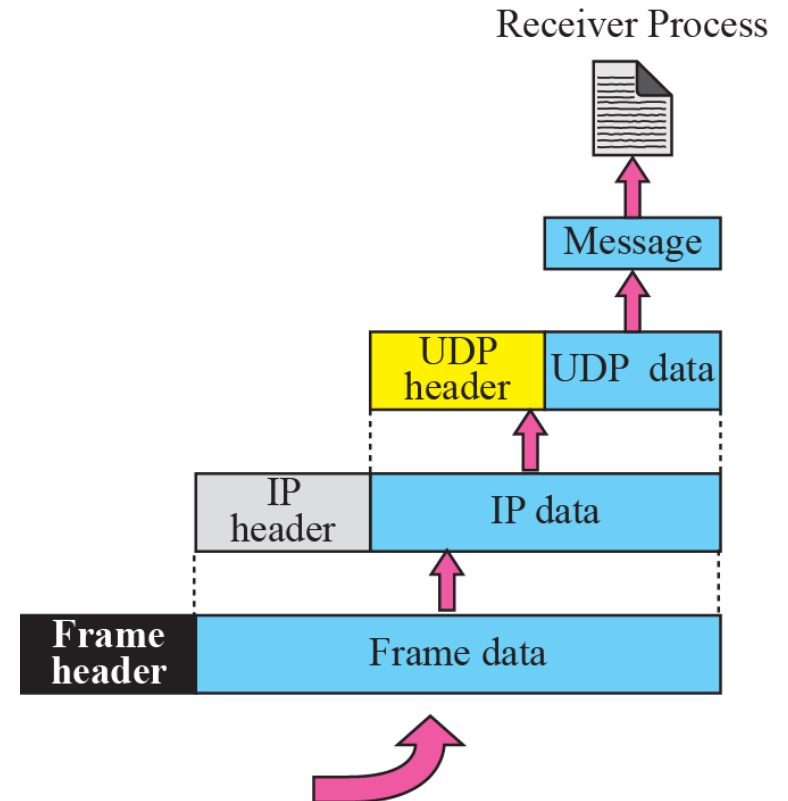
---

- Characteristics
  - No error detection/recovery
  - No flow control
  - No checking for existing of destination
- Simple services
- Very useful protocol

# UDP Encapsulation



a. Encapsulation



b. Decapsulation





# UDP Basic Functions

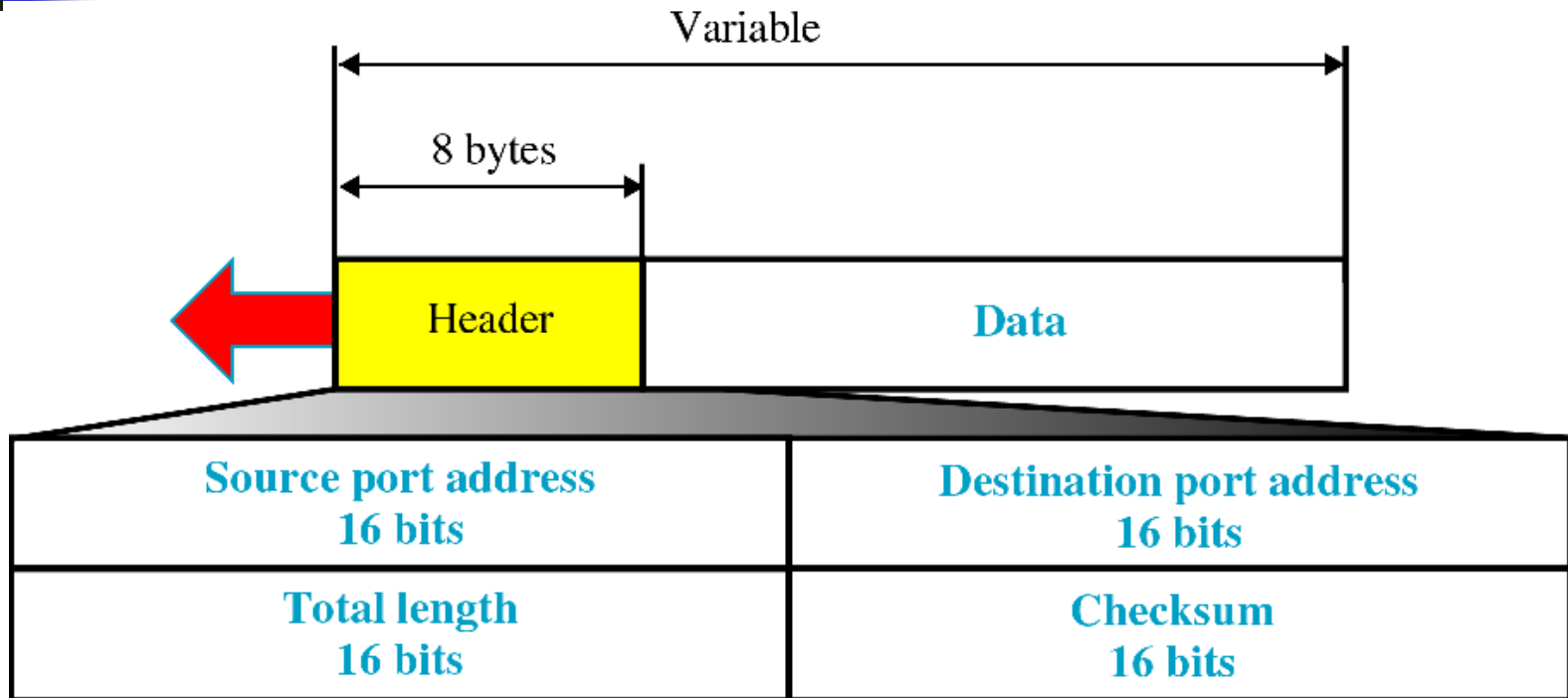
---

- provide basic functions above raw IP
  - Destination port – to specific application
  - Source port – for send back info
  - Data integrity verification – checksum
  - Data reassembly – segmented packets

# UDP well-known ports

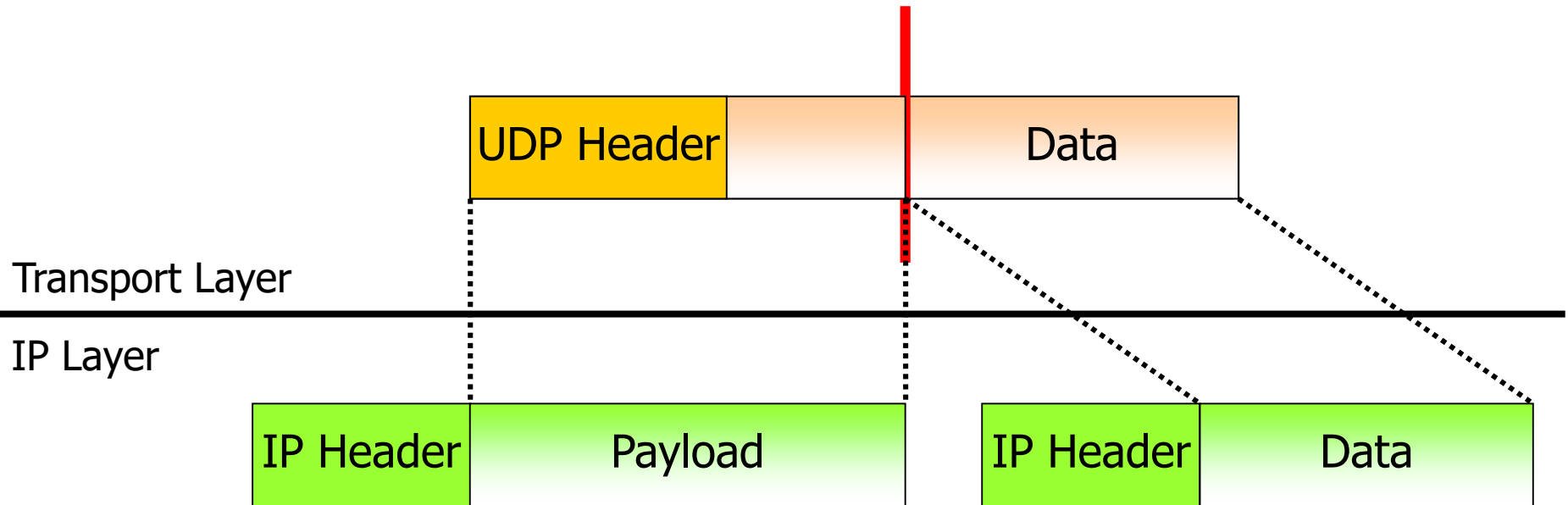
<i>Port</i>	<i>Protocol</i>	<i>Description</i>
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
53	Domain	Domain Name Service (DNS)
67	Bootsps	Server port to download bootstrap information
68	Bootpc	Client port to download bootstrap information
69	TFTP	Trivial File Transfer Protocol
111	RPC	Remote Procedure Call
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management Protocol
162	SNMP	Simple Network Management Protocol (trap)

# UDP Message Format



Total length = UDP header+ Payload  $\leq 2^{16}$  bytes

# UDP Message Fragmentation





# UDP Checksum

---

- Why needed it ?
  - Data link already provided CRC
  - No corrupted frame accepted
- Connectionless !
  - loss packets due to route
- Sender can choose not to use – set to 0
  - Optional
  - performance issue



# UDP Checksum

---

- Compare to other checksums
  - different – UDP **pseudo header**
  - same – 16-bit one's complement
- At sender
  - create pseudo header
  - Add zero to checksum field
  - calculating checksum of pseudo header + regular header + message
  - discard pseudo header when finished



# UDP Checksum

---

- At receiver
  - create pseudo header
  - save received checksum
  - change checksum to zero
  - calculating checksum (pseudo+regular header+message)
  - compare the calculate and save value



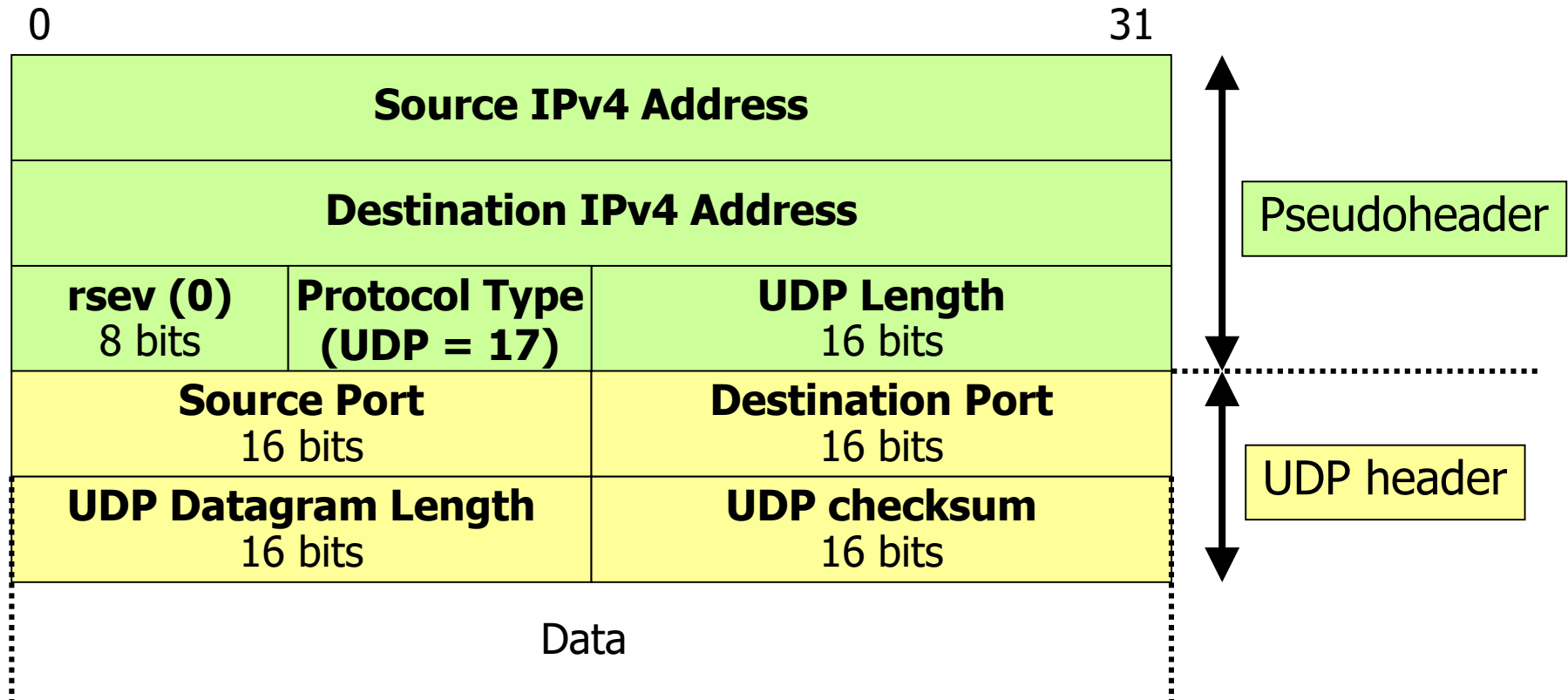
# Why Pseudo Header is needed?

---

- UDP Header
  - Only source/destination ports in the header
  - No Destination IP Address
    - IP Header contains IP add.
- Reduce size of UDP Header
- Need to check for correct destination



# UDP Pseudo Header





# UDP Checksum

---

- If error found
  - discard packet
  - responsibility of application
- IPv6
  - Pseudo header is v6
  - checksum is mandatory



# Protocols that use UDP

---

- TFTP (trivial FTP)
- BOOTP (for diskless station)
- Network Time Protocol (NTP)
  - need basic delivery and checksum
- Network File System (NFS)
  - need low overhead
- Link Management Protocol (LMP)
  - original use raw IP (has its own error process)
  - move to UDP (preserve scarce IP protocol identifier)
- Label Distribution Protocol (LDP)
  - need to support multicast and broadcast
  - discovery mechanism



# UDP Lite

---

- Ultra-lightweight version
- Codecs for voice / video
  - good at handle error in data streams
  - need UDP not to discard damage packets (RF error message) based on checksum
  - remove checksum for better performance
- UDP Lite applies checksum to part of packet
  - **Sensitive part** (e.g. address ; not payload)
  - whole packet sensitive → regular UDP packet

# UDP Lite Header

<b>Source Port</b> 16 bits	<b>Destination Port</b> 16 bits
<b>UDP Datagram Length</b> 16 bits	<b>UDP checksum</b> 16 bits

regular  
UDP header

<b>Source Port</b> 16 bits	<b>Destination Port</b> 16 bits
<b>Checksum Coverage Field</b> 16 bits	<b>UDP Lite checksum</b> 16 bits

UDP Lite  
header

# of bytes in UDP header that included in the checksum  
0 = all bytes not included  
UDP datagram Length = same as regular UDP