Packets

Networking

Sirindhorn International Institute of Technology
Thammasat University

Prepared by Steven Gordon on 2 September 2013
Common/Reports/packets.tex, r676

Contents

Terminology

Example: IP Datagram

Packet Size

Further Example Packets

Networking

Packets

Terminology

Example: IP

Packet Size

Example Packets

Packets

- Most communication protocols group data into separate pieces
- Each piece of data is commonly called a packet
- Information in a packet often separated into parts:

  Header control information at start of packet; used to support protocol operation

  Payload actual data

  Trailer control information at end of packet; used to support protocol operation
- Not all parts in all packets, e.g. Header + Payload; Header + Payload + Trailer; Header only

Packet Terminology

- No standard terminology for packets
- Other names: frame, datagram, segment, package, message
- Differs among protocols and layers, e.g.

Application	message
Transport	TCP segment, UDP datagram
Network	datagram
Data Link	frame

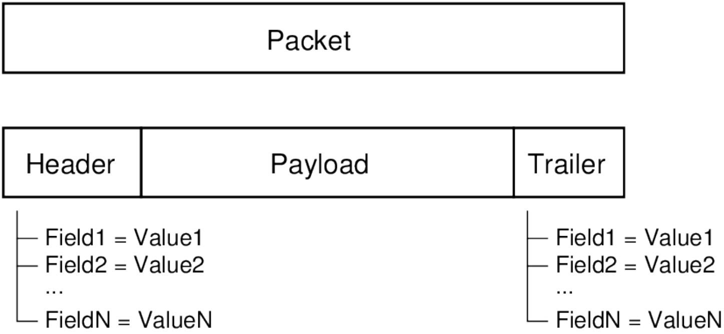
- Standards often measure packet sizes in octets
  - 1 octet = 8 bits (always true)
  - 1 Byte = 8 bits = 1 octet (true in most practical cases today)

Packet Header (and Trailer)

What is Purpose of Header?

- Contains information to support protocol operation
- Sender includes information in header so receiver can correctly process the data and optionally respond
- Information often split into fields; each field has a value
- Number, meaning and size of fields defined in standard
  - RFC 793 defines TCP segment header fields
  - IEEE 802.11 defines wireless LAN frame header and trailer fields
- Many protocols have default, fixed size header, with optional extra fields
  - TCP: 20 bytes required; optional fields allowed
  - IEE 802.11 MAC Data: typically 24 byte header and 4 byte trailer; other sizes possible

General Packet Structure



Packet Header and Trailer

Header vs Trailer

- Trailer also contains information to support protocol operation
- Header before the payload, trailer after the payload
- Devices can process packet as it is received; header then payload then trailer
  - Info in header can be processed before/as data arrives
    - Router can determine where to send the packet before the entire packet has been received
  - Trailer often used when dependent on data, e.g. checksum over data
- Most protocols use header, some use both header and trailer
- (For simplicity, examples often only consider header)

Packet Header (and Trailer)

Example Header Fields

- Source and destination addresses, e.g. IP address, MAC address
- Packet, payload, header lengths
- Sequence numbers, e.g. data sequence, ACK number
- Protocol version, e.g. IPv4
- Checksums, error detection codes
- Packet types, e.g. SYN, ACK, RST
- Flags
  - Single bit values
  - 1: flag is set/true, e.g. feature is on
  - 0: flag is unset/false, e.g. feature is off

Contents

Terminology

Example: IP Datagram

Packet Size

Further Example Packets

Example IP Datagram: Binary

010001010000000000000000111011010000001111000110001000000000000000010000000000000110

00100101011100010000101000000101001100101010000001110010111000000111101000101010010

Read left to right
160 bits = 20 Bytes

Example IP Datagram: Binary

01000101000000000000000111011010000001111000110001000000000000000100000000000110

0010010101110001000010100000101001100101010000011100101110000011110100010101001 0

IETF RFC 791 defines which bits belong to which fields

Version 0100

Header length 0101

. . .

Source Address 00001010000010100110010101000001

Destination Address 11001011100000111101000101010010

Example IP Datagram: Meaning

01000101000000000000000111011010000001111000110001000000000000000100000000000110

00100101011100010000101000001010011001010100000111001011100000111101000101010010

IETF RFC 791 defines meaning of field values

Version 0100 $\rightarrow$ 4

Header length 0101 $\rightarrow 5 \times 4 = 20$ Bytes

...

Source Address 10.10.101.65

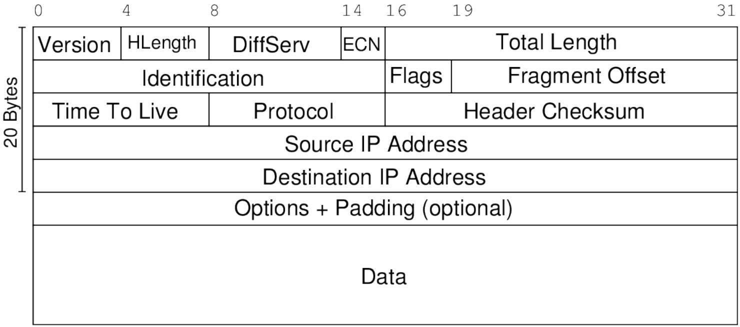
Destination Address 203.131.209.82

Example IP Datagram: Re-arranged Binary

01000101 00000000 00000001 11011010
00000111 10001100 01000000 00000000
01000000 00000110 00100101 01110001
00001010 00001010 01100101 01000001
11001011 10000011 11010001 01010010

32 bits per row
5 rows = 160 bits = 20 Bytes

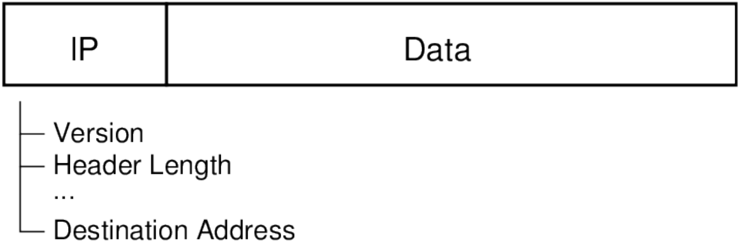
IP Datagram Structure



Although packets are just sequence of bits, for convenience headers and header fields often drawn row-by-row

Simplified IP Datagram Structure

Another way to draw packets . . .



Example IP Datagram: Meaning

Version 4

Header length 20 Bytes

Differentiated Services Codepoint Default (000000)

Explicit Congestion Notification Not-ECT (00)

Total Length 474 Bytes

Identification 0x078c

Flags 0x02

- 0... .... = Reserved bit: Not set
- .1.. .... = Don't fragment: Set
- ..0. .... = More fragments: Not set

Fragment offset 0

Time to live 64

Protocol TCP (6)

Header checksum 0x2571

Source 10.10.101.65

Destination 203.131.209.82

Contents

Terminology

Example: IP Datagram

Packet Size

Further Example Packets

Packet Size

- Standards define the size of headers (and trailers)
- Normally a default header size (cannot be smaller) and optional extra fields (depending on the protocol features in use)
- Headers and trailers are overhead from users perspective
- How big should a packet be?
    - Depends on many factors; tradeoff between different performance criteria
    - Standards often define a maximum packet/payload size
        - E.g. Ethernet frame: 14 Byte header, 4 Byte trailer, maximum payload of 1500 Bytes
- Given fixed header/trailer and maximum payload, what size payload is optimal?
    - Large payload: minimize overhead of header/trailer
    - Small payload: minimize overhead of retransmissions
    - Small payload: make efficient use of buffers
    - Small payload: provide fairness when multiple users sharing medium

Packet Overhead

Assume 2000 Bytes of user data to be sent; each packet contains 20 Bytes of header

Payload size = 1000 B 2 packets; 40 Bytes of total overhead

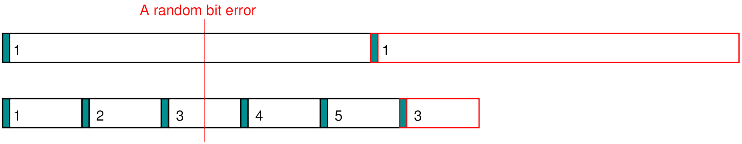
Payload size = 200 B 10 packets; 200 Bytes of overhead

Larger packet/payload → less overhead of headers/trailers

Packet Retransmit

Assume 1 bit error occurs randomly

Payload size = 1000 B 1020 Bytes retransmitted; 2040 Bytes sent to deliver 1000 B of data



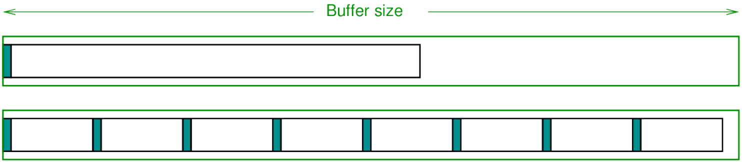
Payload size = 200 B 220 Bytes retransmitted; 1320 Bytes sent to deliver 1000 B of data

Smaller packet/payload → less to retransmit if errors

Packet Buffering

Assume receiver has buffer of 1800 Bytes

Payload size = 1000 B 1 packet can fit in buffer; 780 Bytes of buffer space wasted

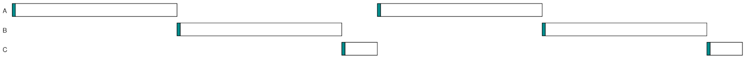


Payload size = 200 B 8 packets can fit in buffer; 40 Bytes of buffer space wasted

Smaller packet/payload → less waste of buffer space

Packet Fairness

Assume 3 users take in turns transmitting packets; 2 users have large packets, 1 user has only small payload to send



Aim of a fair system: each user can transmit 33% of time
User sending small packet has to wait long time for other users to complete transmissions

Payload size = 1000 B Spend 1020 out of 2260 time units transmitting; 45% of time

Payload size = 200 B Spend 220 out of 2260 time units transmitting; 10% of time; large time waiting for other users

Smaller packet/payload → fairer for all users

Contents

Terminology

Example: IP Datagram

Packet Size

Further Example Packets

Example Maximum Packet Sizes

IEEE 802.3 Ethernet frame 1500 Bytes payload

IEEE 802.11 WLAN frame 2312 Bytes payload

SDH (STM-1/OC-3) 2430 Bytes (including header)

ATM Cell 5 Byte header, 48 Byte payload (fixed)

IP datagram 65535 Bytes (including header)

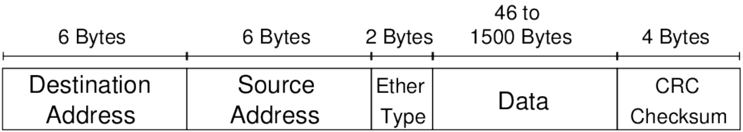
UDP datagram 65535 Bytes (including header)

TCP segment 65535 Bytes (including header)

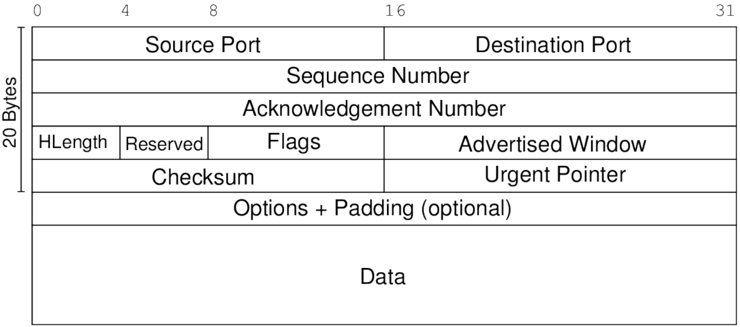
- Typically source and destination negotiate a Maximum Segment Size (MSS) such that the IP datagram carrying the TCP segment will not have to be split to fit into data link layer packets
- Maximum payload allowed by data link layer called Maximum Transmission Unit (MTU)
- TCP over IP over IEEE 802.3: MTU = 1500 Bytes; MSS = 1460 Bytes

HTTP request message no hard limit; browsers may implement limits, e.g. 8 kB

Other Example Packets: Ethernet Frame

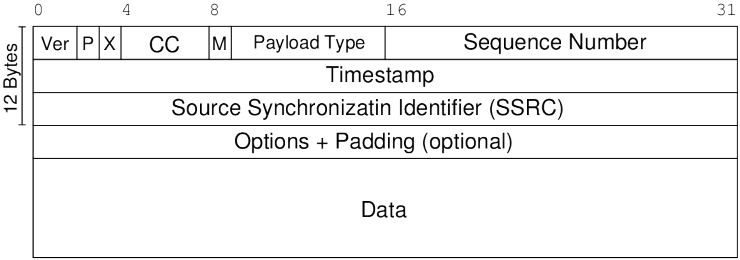


Other Example Packets: TCP Segment



Other Example Packets: UDP Datagram

Other Example Packets: RTP Packet



Other Example Packets: HTTP Message

Text formatted message

- Generic HTTP message format:

```
Start line
Optional header lines
<empty line>
Optional message body
```

- Start line differs for request and response, e.g. request:
  - Start line: `Method URL Version`
  - Methods:
    - GET: retrieve the resource at the specific URL
    - POST: asks server to accept and process the attached data at the resource
    - ...
  - Version: version of HTTP, e.g. HTTP/1.0, HTTP/1.1
- Header format: `field-name: value`
  - Date: data and time of message generation
  - Content-Length: length of message body in bytes
  - User-Agent: indicates information about the client
  - Host: domain name of host of resource