Panel 1

Last time:

IP header format

IP addressing details

Special servers on a LAN:

- DHCP
- Gateway
- DNS
- Firewall

Other network-level protocols

DHCP, DNS, ARP, RARP, ICMP, BOOTP

1

Panel 2

Group 1. -ive

Group 2: DNS

Group 3: DHCP

Group 4: ICMP

Group 5: ARP + RARP

Yuri Abeshkin
Anthony Ambrose
Chris Dubra
Frank Gounello
Jeff Johnson
Michael Malenkov
Steven Malinelli
Ed Mikuszeuski
Thomas Ochenhouse
Stefano Polo

2

Panel 3

New Protocol needed

IP v4 has several limitations

Address space : $2^{32} \approx 4$ Billion

compare with Ethernet ~ 48 bits $\Rightarrow 2^{48} = 218\,000\,000\,000\,000$

New goals: - support more hosts

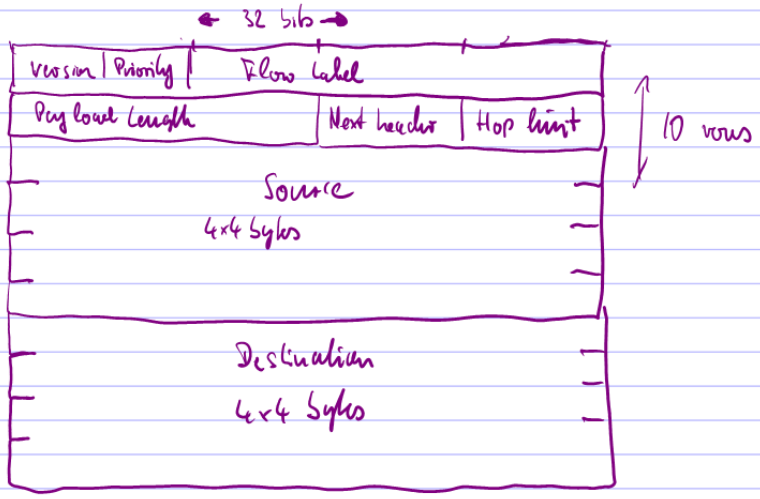
- reduce size of routing tables
- simplify protocol for faster processing
- better security
- attention to type of service
- multicasting + roaming support
- compatibility with old IP v4

3

Panel 4

IP v6

After much discussion → IP v6

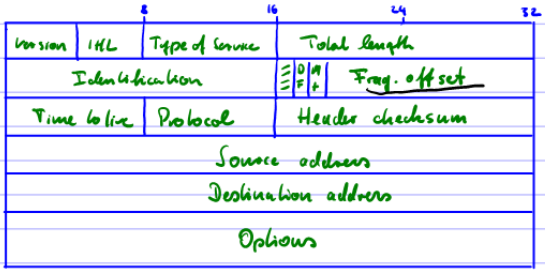


No header checksum → faster processing

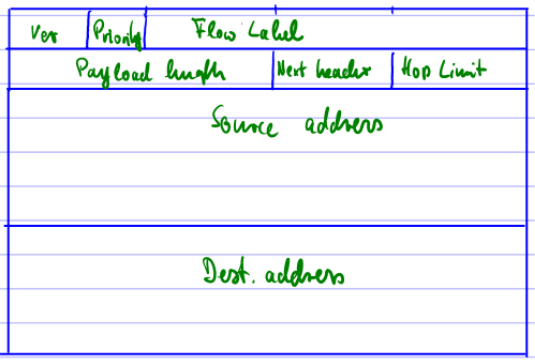
4

Panel 5

20 = 4 × 5 byte IPv4 header

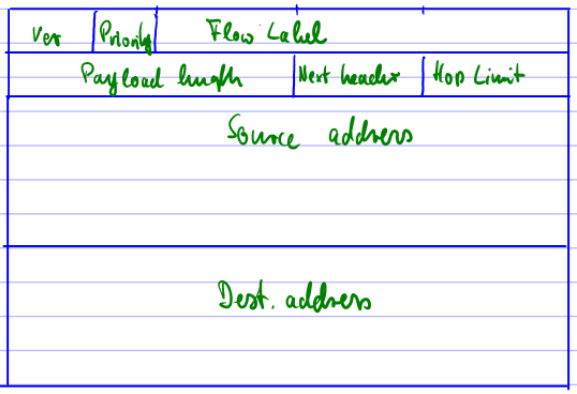


40 = 4 × 10 byte IPv6 header



5

Panel 6

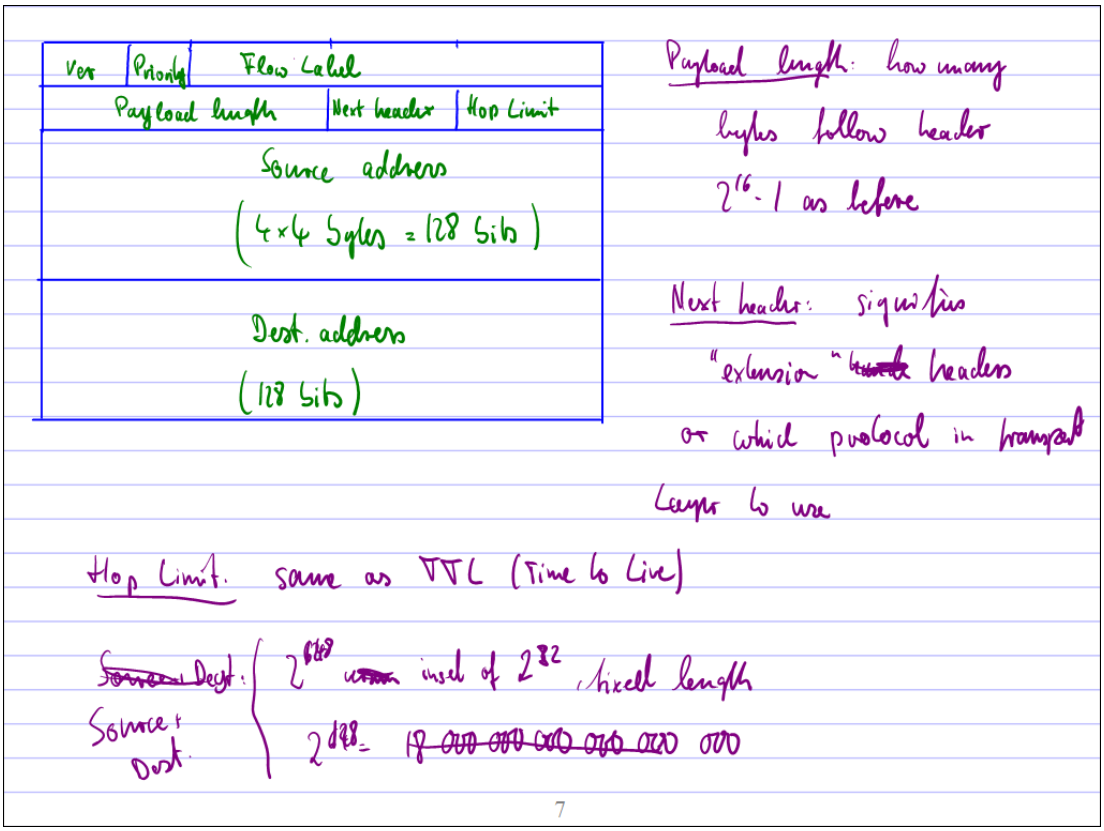


Priority: 0-7 for transmission with flow control, 8-15 for constant rate, real-time traffic (with potential loss) QoS

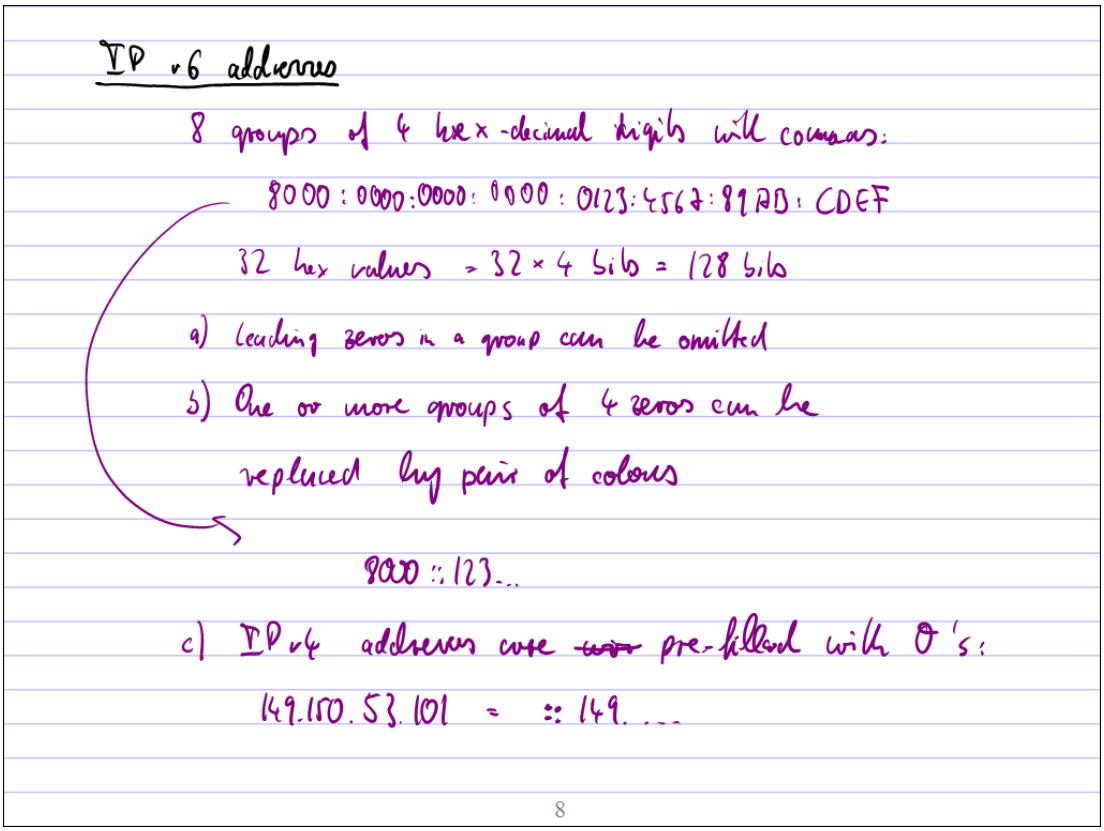
Flow label: experimental to try to setup virtual circuits

6

Panel 7



Panel 8



Panel 9

Extension Headers

Options can immediately follow header

Controversies

Many discussions preceded IPv6 — 20 byte header / 12 byte header

Hop count : 255 is enough?

Max packet size - jumbo-grams via extensions

CRC checksum - remove breaks from core
but still in data-link layer

Security: in network layer would be convenient, but nobody would rely on it for real!
what type MD5 ? DES ?

9

Panel 10

The Transport Layer

Provide reliable, cost-effective data transport from source to destination.

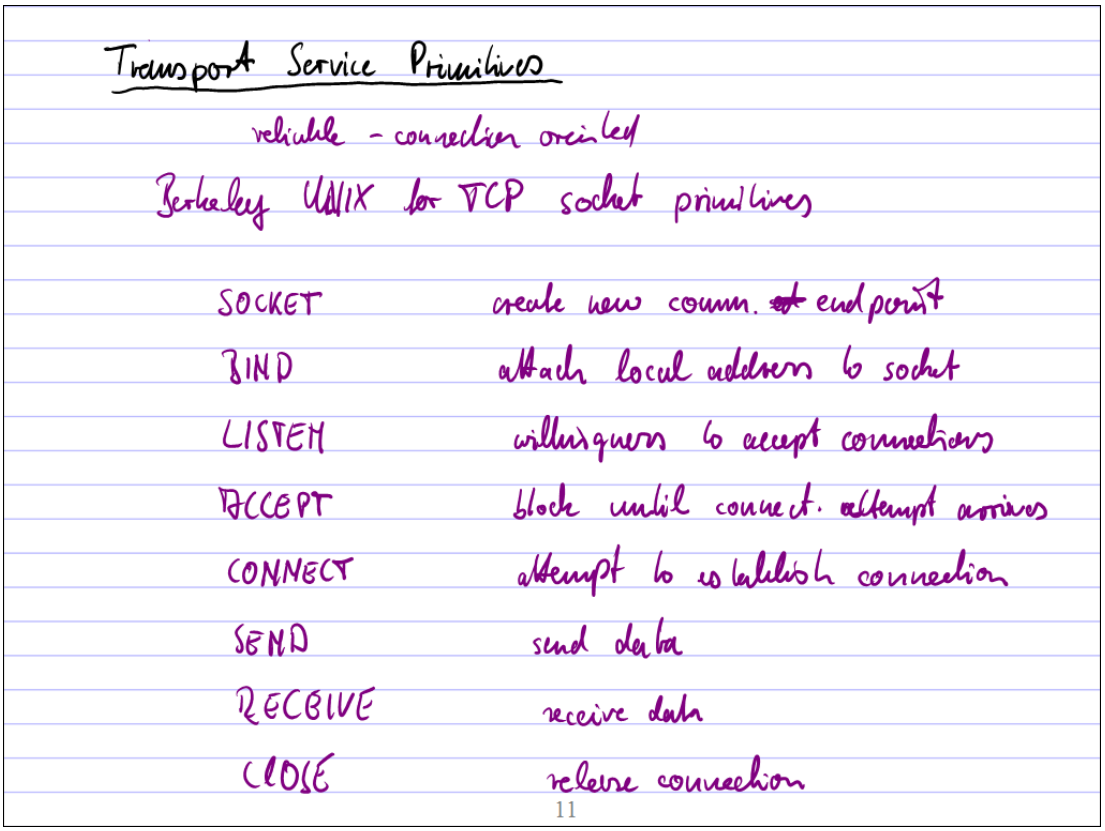
Network: routing + addressing

Transport: reliable

TPDU = Transport Protocol Data Unit

10

Panel 11


Transport Service Primitives
reliable - connection oriented
Berkeley UNIX for TCP socket primitives
SOCKET
create new comm. endpoint
BIND
attach local address to socket
LISTEN
willingness to accept connections
ACCEPT
block until connect. attempt arrives
CONNECT
attempt to establish connection
SEND
send data
RECEIVE
receive data
CLOSE
release connection
11