高级计算机网络
Advanced Computer Networks

Instructor: 黄倩怡
huangqy89@mail.sysu.edu.cn

2026.09

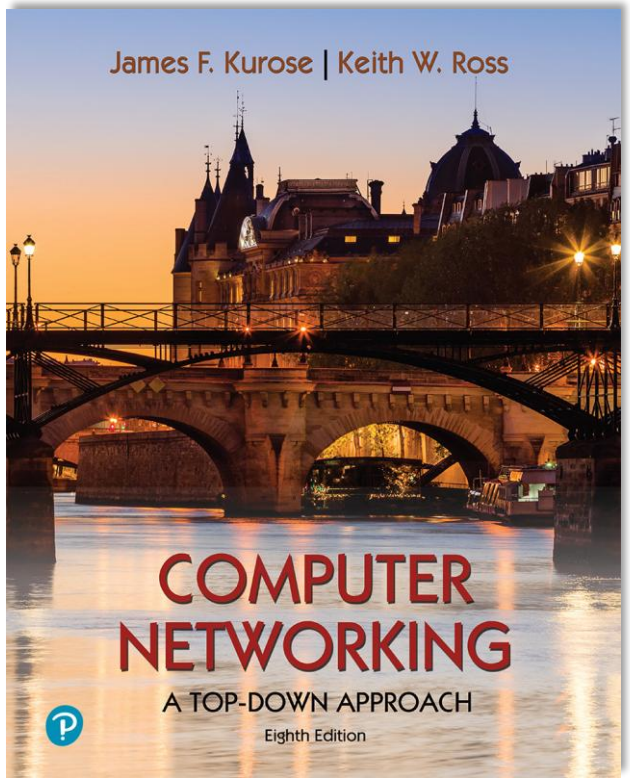
Lecturer

- Qianyi Huang
- Associate Professor from School of Computer Science and Engineering
- Email: huangqy89@mail.sysu.edu.cn
- Office: Supercomputing Center 529F

- Research Topics
  - Mobile Deep learning/Mobile LLM/Mobile Agents
  - Mobile Communication (5G/6G, UAV)
  - Embodied AI

Chapters (2023)

- Chapter 1: Introduction
- Chapter 2: The Application Layer
- Chapter 3: The Transport Layer
- Chapter 4: The Network Layer: Data Plane
- Chapter 5: The Network Layer: Control Plane
- Chapter 6: The Link Layer and LANs
- Chapter 7: Wireless and Mobile Networks
- Chapter 8: Security
- Chapter 9: Multimedia Networking



Main Contents (2023)

- Wireless Networks:
  - WiFi, Cellular Networks (4G/5G), Internet of Things
- Multi-media Networks:
  - Video streaming, VoIP, QoS, CDN, network support for multimedia
- Network Security:
  - Application Layer, Transport Layer, Network Layer
- Cloud & Edge Computing:
  - DC architecture and energy/cost optimization

Main Contents (2026)

Research-oriented

- Networking for AI:
  - Agents (Codex, Claude Code, WorkBuddy, DeepSeek)
  - LLM Serving
- AI for Networking
- Satellite Communication
- 5G/6G
- Edge computing

Class

- Time: Wednesday 14:20-17:15, Week 1-17
- Room: B502

- 14:20-16:00 Lectures
- 16:30-17:15 Paper Presentations

Grading

- Attendance: 10%
- Class presentation: 30%
  - Presenter: 20-30 (Email Me: huangqy89@mail.sysu.edu.cn)
  - Audience: Ask questions/Comment, maximum 20 points
- Final project: 60%
  - Tiny Research: contribution (insights/algorithm/etc)
  - Report & Presentation

Course Website

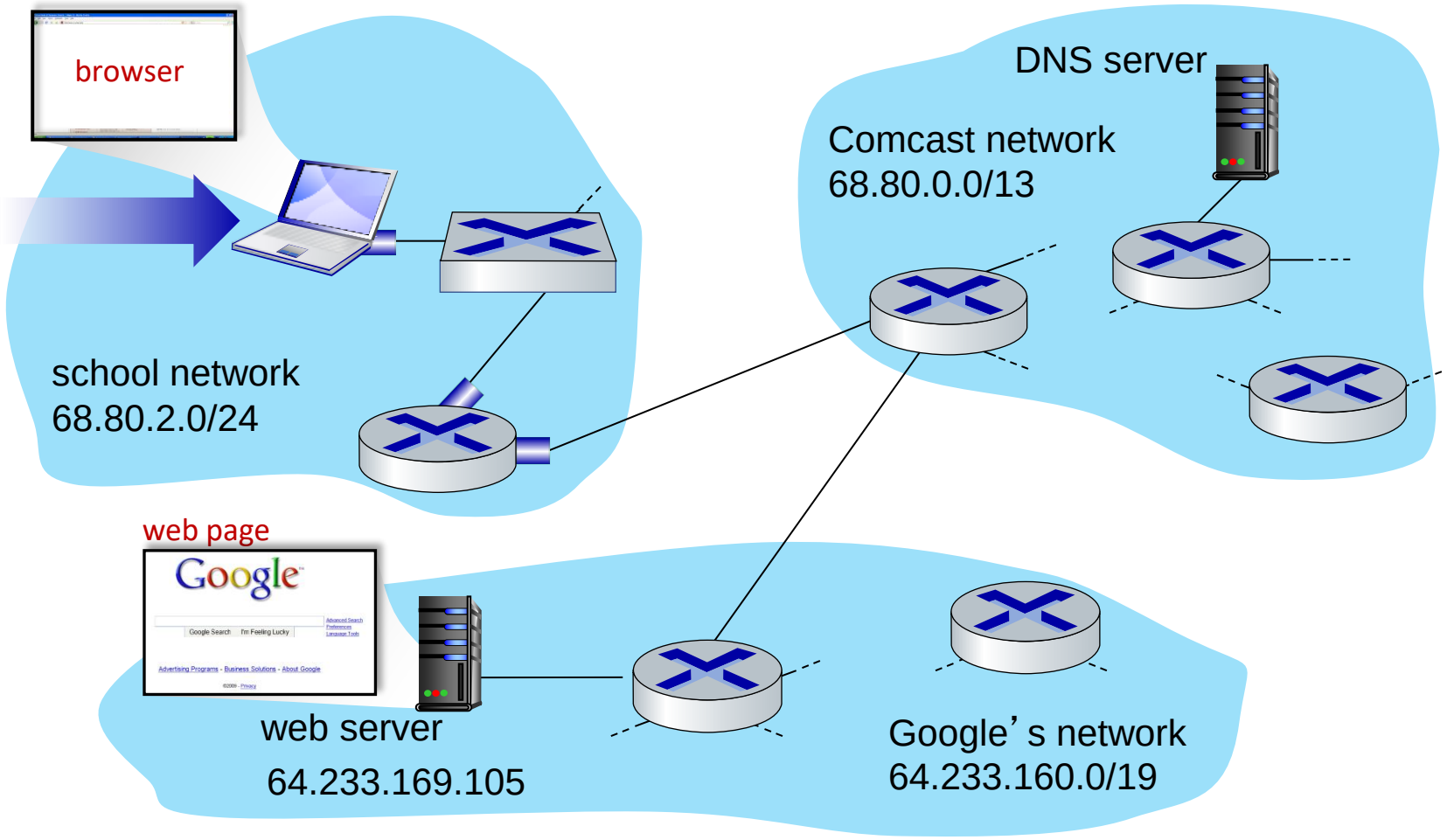
- https://huangqy89.github.io/acn-2026/

- Lecture notes
- Reading lists
- Presentation schedule

Synthesis: a day in the life of a web request

- our journey down the protocol stack is now complete!
  - application, transport, network, link
- putting-it-all-together: synthesis!
  - *goal:* identify, review, understand protocols (at all layers) involved in seemingly simple scenario: requesting www page
  - *scenario:* student attaches laptop to campus network, requests/receives www.google.com

A day in the life: scenario

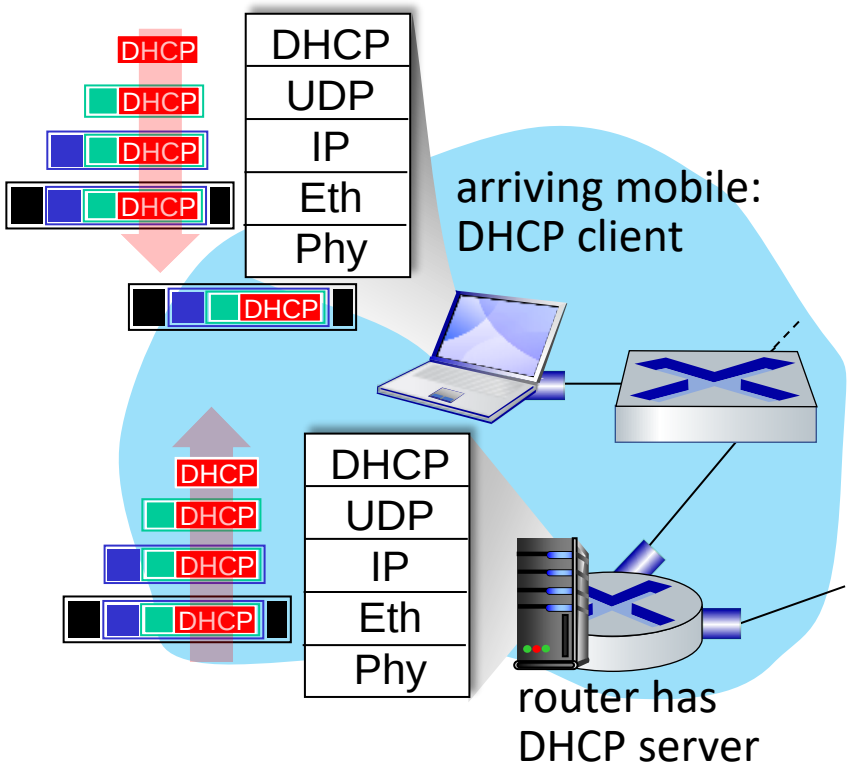


scenario:

- arriving mobile client attaches to network …
- requests web page: www.google.com

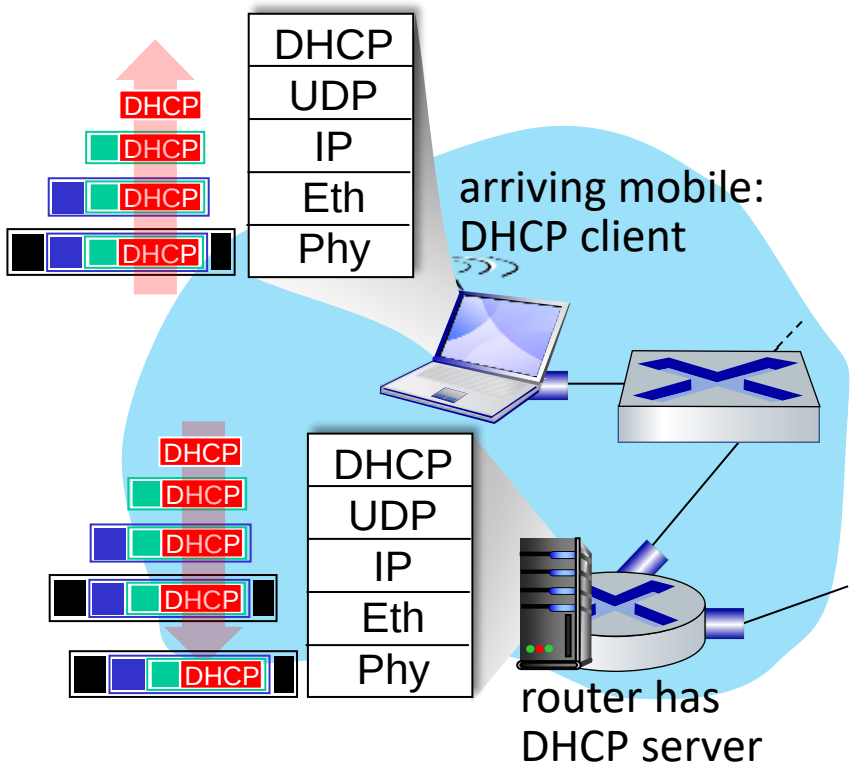
Sounds simple!

A day in the life: connecting to the Internet



- connecting laptop needs to get its own IP address, addr of first-hop router, addr of DNS server: use DHCP
- DHCP request encapsulated in UDP, encapsulated in IP, encapsulated in 802.3 Ethernet
- Ethernet frame broadcast (dest: FFFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet de-muxed to IP de-muxed, UDP de-muxed to DHCP

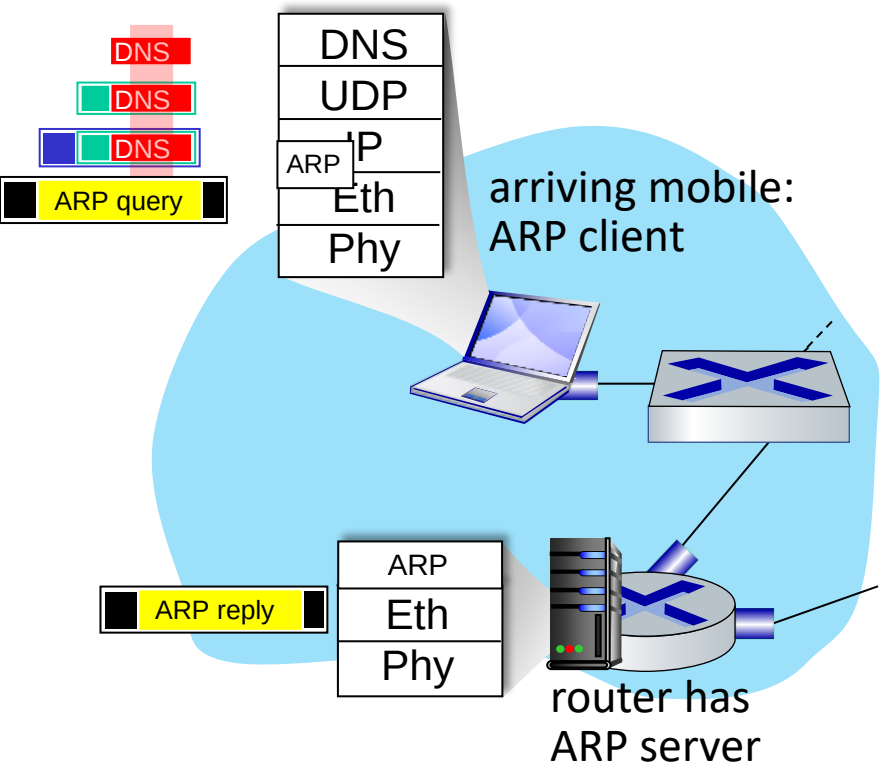
A day in the life: connecting to the Internet



- DHCP server formulates DHCP ACK containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server
- encapsulation at DHCP server, frame forwarded (switch learning) through LAN, demultiplexing at client
- DHCP client receives DHCP ACK reply

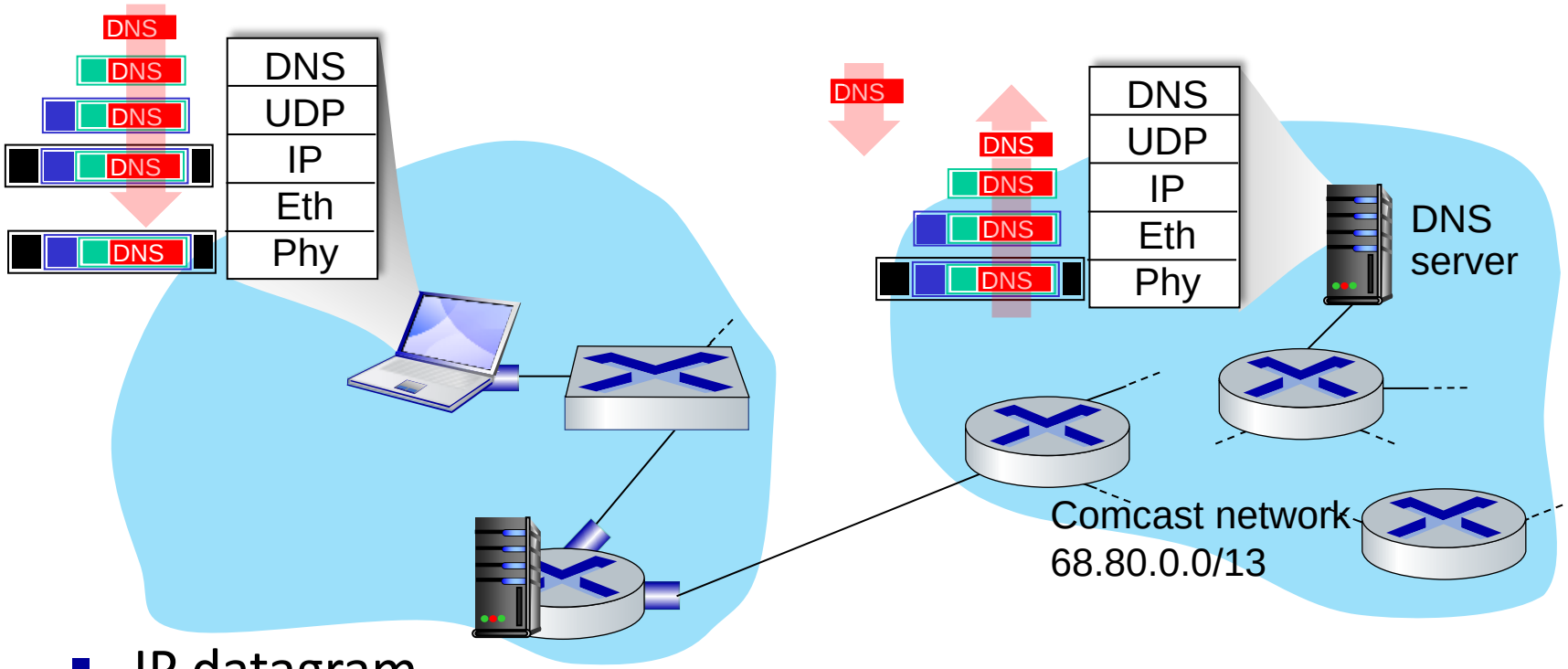
Client now has IP address, knows name & addr of DNS server, IP address of its first-hop router

A day in the life… ARP (before DNS, before HTTP)



- before sending HTTP request, need IP address of www.google.com: DNS
- DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. To send frame to router, need MAC address of router interface: ARP
- ARP query broadcast, received by router, which replies with ARP reply giving MAC address of router interface
- client now knows MAC address of first hop router, so can now send frame containing DNS query

A day in the life… using DNS

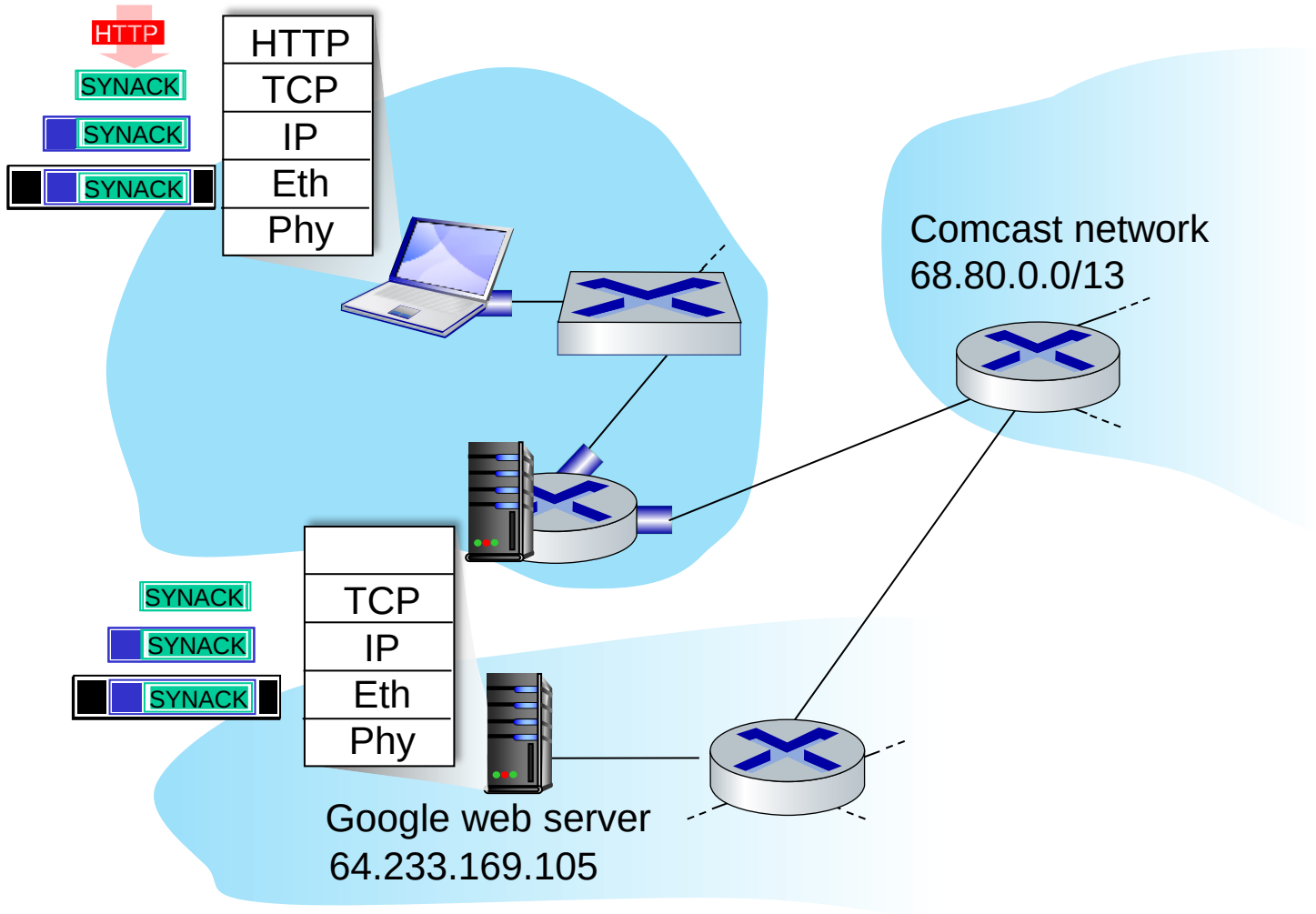


- de-muxed to DNS
- DNS replies to client with IP address of www.google.com

- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router

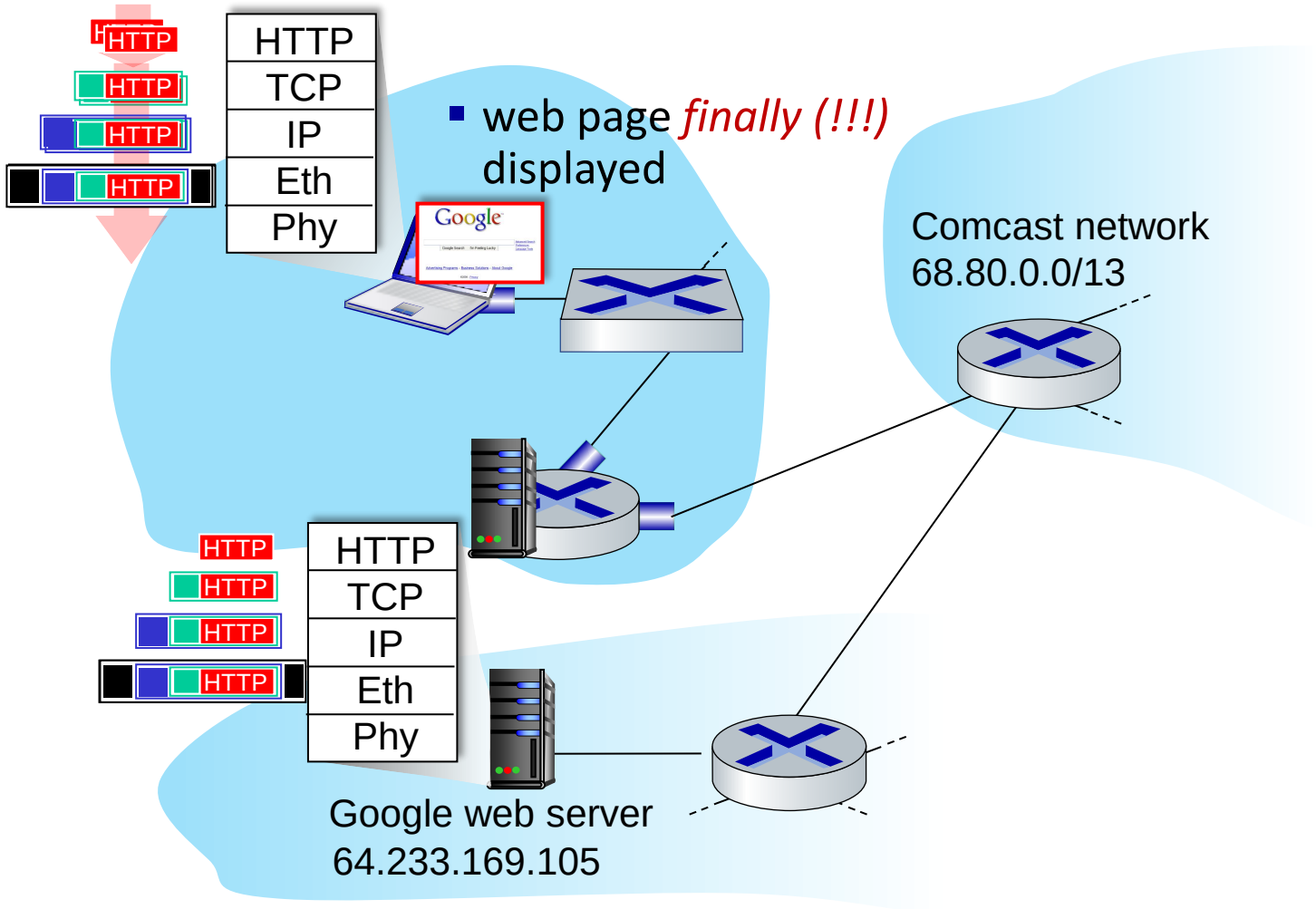
- IP datagram forwarded from campus network into Comcast network, routed (tables created by RIP, OSPF, IS-IS and/or BGP routing protocols) to DNS server

A day in the life…TCP connection carrying HTTP



- to send HTTP request, client first opens TCP socket to web server
- TCP SYN segment (step 1 in TCP 3-way handshake) inter-domain routed to web server
- web server responds with TCP SYNACK (step 2 in TCP 3-way handshake)
- TCP connection established!

A day in the life… HTTP request/reply



- HTTP request sent into TCP socket
- IP datagram containing HTTP request routed to www.google.com
- web server responds with HTTP reply (containing web page)
- IP datagram containing HTTP reply routed back to client