LEARNHUB / TECHNICAL COURSE LIBRARY

Computer Networks

Practical examples

Follow data across a network: from layered models and addressing to switching, routing, transport protocols, and the application services people use every day.



Level: Beginner · Category: Networking

Original, beginner-friendly study content designed to accompany the course syllabus and its linked video lesson.

Contents

- How to use this resource
- Learning map
- Chapter 01 · Layers and the OSI model
- Chapter 02 · Addresses, subnetting, and name resolution
- Chapter 03 · Switching, routing, and packet paths
- Chapter 04 · TCP, UDP, and transport behavior
- Chapter 05 · Application protocols and diagnostics
- Summary and next steps

Course purpose: Follow data across a network: from layered models and addressing to switching, routing, transport protocols, and the application services people use every day.

How to use this resource

Read the chapter titles first, then study the explanations and examples. Pause at each practice prompt before reading its suggested answer. Write down questions that remain and revisit them during revision.

Learning map



Chapter 01 · Layers and the OSI model

Problem context

Describe the broad task of the network layer and the transport layer when sending a web request.

Concept to apply

A network layer groups responsibilities such as physical signaling, addressing, transport, or application behavior. The OSI model is a reference framework with seven layers.

Worked example

A browser request is encoded into application data, carried by transport and Internet headers, then framed for the local link.

Walk-through

Layering lets systems evolve one responsibility at a time. Real Internet protocols are commonly described with the TCP/IP model, which groups layers differently. Encapsulation adds headers as data moves down a stack; receiving systems remove them in reverse order.

Check your result

> The network layer routes packets between networks using addresses. The transport layer provides process-to-process delivery, with TCP or UDP behavior depending on the application.

Chapter 02 · Addresses, subnetting, and name resolution

Problem context

Explain what the prefix `/24` tells you and what a default gateway is used for.

Concept to apply

An IP address identifies a network interface. A subnet prefix divides the address into a network portion and a host portion; DNS maps names to records such as IP addresses.

Worked example

A laptop sees a destination outside its local prefix and sends the packet to its default gateway after resolving the service name.

Walk-through

A CIDR prefix states how many leading bits describe the network. Devices use a subnet mask and route table to decide whether a destination is local. DNS resolution happens before a client can contact a named service.

Check your result

A `/24` uses the first 24 address bits as the network prefix. A default gateway is the router used for destinations without a more-specific local route.

Chapter 03 · Switching, routing, and packet paths

Problem context

Trace a packet from a laptop to a remote server through a home gateway, naming the roles involved.

Concept to apply

A switch forwards frames within a local link using link-layer addresses. A router forwards packets between networks using network-layer addresses and routing information.

Worked example

In a home network, a switch connects devices on the local LAN while the wireless router forwards outbound traffic toward the internet provider.

Walk-through

A host first determines the next hop, resolves a local link-layer address, and sends a frame. Routers forward packets hop by hop; each link may use different framing while the IP destination remains meaningful.

Check your result

> The laptop sends a frame to its gateway, the gateway routes the IP packet onward, and intermediate routers select next hops until the destination network receives it.

Chapter 04 · TCP, UDP, and transport behavior

Problem context

For a time-sensitive multiplayer game update, compare TCP and UDP trade-offs without assuming one is always correct.

Concept to apply

TCP provides a reliable, ordered byte stream with connection setup and recovery behavior. UDP sends independent datagrams without TCP's delivery guarantees.

Worked example

A file download usually relies on TCP delivery; a live voice call may prefer low delay and tolerate an occasional lost packet.

Walk-through

Choose transport based on application needs. TCP suits many web and file-transfer tasks where reliable ordered delivery matters. UDP can support low-latency or application-controlled delivery, but the application must handle losses if necessary.

Check your result

> UDP avoids waiting for retransmission of stale data, which can help latency. The game must handle loss, ordering, and security as needed; some systems instead use TCP or modern secure transports.

Chapter 05 · Application protocols and diagnostics

Problem context

A user can ping an IP address but not open a domain. Name a likely first check and one follow-up.

Concept to apply

An application protocol defines message formats and interactions for a service. Examples include HTTP for web transfers, DNS for name lookup, and SMTP for sending email.

Worked example

If a domain resolves but a secure web request fails, DNS may be working while the server, certificate, route, or TLS configuration still needs checking.

Walk-through

When diagnosing a connection, separate name resolution, transport reachability, encryption, and application response. Use authorized tools to inspect requests, status codes, and configuration rather than guessing at the layer where a failure occurred.

Check your result

Check DNS resolution for the domain. If it resolves, test the application port and inspect the browser/network error to distinguish service availability from certificate or HTTP issues.

Summary and next steps

Computer Networks brings together 5 connected topics. Review each definition, explain one example without looking, and then try the practice prompt again.

Key ideas to carry forward

Layers and the OSI model: A network layer groups responsibilities such as physical signaling, addressing, transport, or application behavior. The OSI model is a reference framework with seven layers.

Addresses, subnetting, and name resolution: An IP address identifies a network interface. A subnet prefix divides the address into a network portion and a host portion; DNS maps names to records such as IP addresses.

Switching, routing, and packet paths: A switch forwards frames within a local link using link-layer addresses. A router forwards packets between networks using network-layer addresses and routing information.

TCP, UDP, and transport behavior: TCP provides a reliable, ordered byte stream with connection setup and recovery behavior. UDP sends independent datagrams without TCP's delivery guarantees.

Application protocols and diagnostics: An application protocol defines message formats and interactions for a service. Examples include HTTP for web transfers, DNS for name lookup, and SMTP for sending email.

A short review routine

1. Recall the main idea from each chapter before rereading. 2. Work one example by hand. 3. Explain a common mistake and how you would catch it. 4. Revisit a difficult topic after a short break.

Course outcomes

- Explain the purpose of network layers and common protocols
- Read basic IPv4 addressing and subnet boundaries
- Differentiate switching from routing
- Compare TCP and UDP and trace common application protocols

This practical examples is an original LearnHub learning aid. Use the referenced ideas for study, then consult the course video and the rest of the library for additional practice.