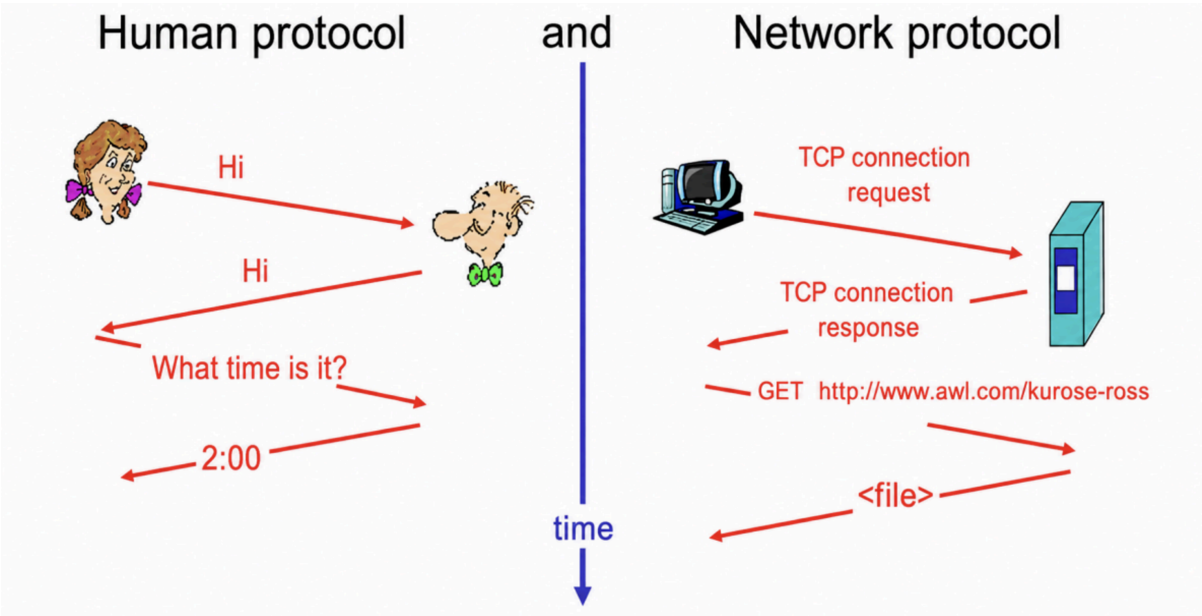
Network protocols

Written by [Febogamedev]

A network protocol is a set of rules that establishes how two or more devices must communicate with one another. These rules define the format of messages, the order in which they are exchanged, and the behavior to follow when errors occur or responses are missing (International Telecommunication Union [ITU], 1994; Kurose & Ross, 2026).
A communication protocol can be defined as a set of rules established to enable correct communication. For example, two people with different native languages might agree to use English to communicate.
In particular, a protocol defines the language formed by the exchanged messages, which must be interpreted correctly (ITU, 1994).

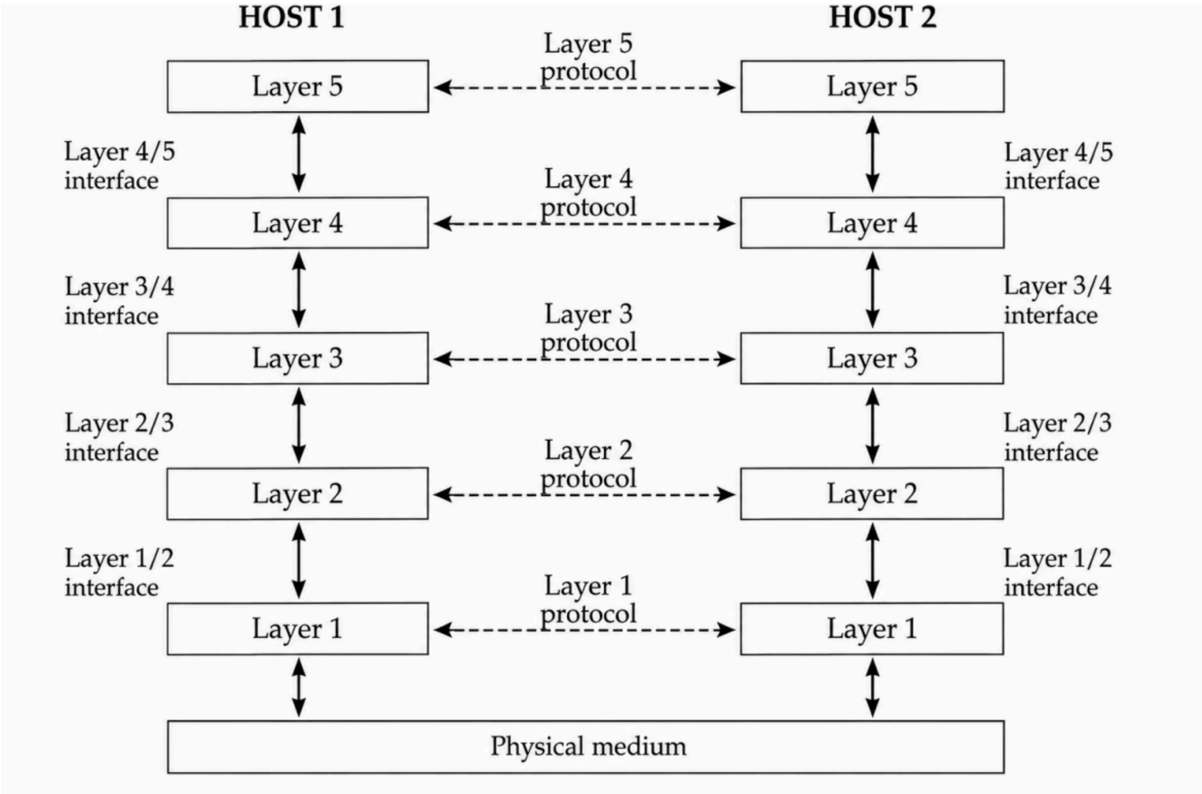


Following protocols ensures that two software programs running on different machines can communicate effectively, even if they were developed independently. In other words, protocols ensure **interoperability** (Braden, 1989; Carpenter, 1996)**.**
Interoperability is the ability of a computer system or product to cooperate and exchange information or services with other systems or products (Braden, 1989).
The purpose of interoperability is therefore to facilitate interaction between different systems.

The implementation of network protocols within a network architecture forms what is known as **network software**, which is implemented through both software and hardware: the operating system, network drivers, applications, and, for certain tasks, the network interface card (Braden, 1989).

To reduce complexity, most networks are organized into a **protocol stack** made up of layers built on top of one another (ITU, 1994; Kurose & Ross, 2026).

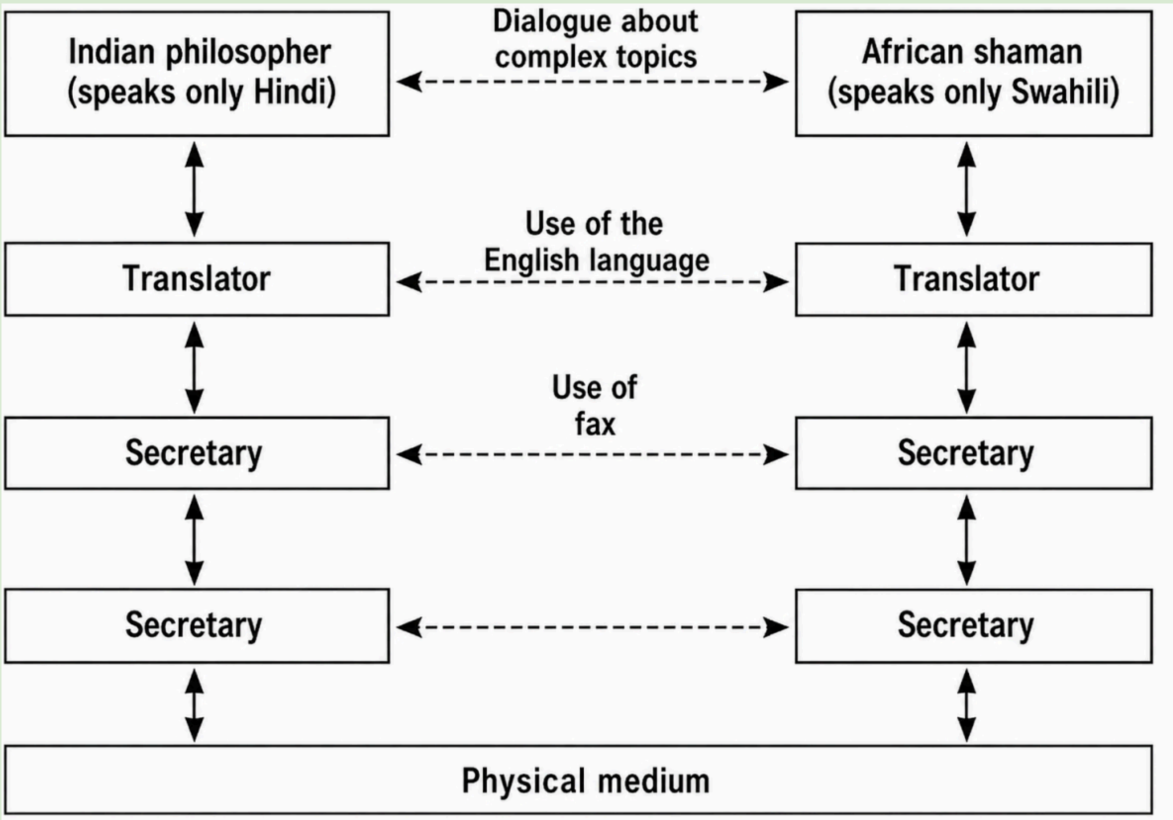
Each layer provides services to the layer above it and hides the technical details of how those services are implemented (ITU, 1994).



When layer n in one computer communicates with layer n in another computer, the rules and conventions used in this communication are collectively known as the protocols of that layer (ITU, 1994).
The entities belonging to corresponding layers on different computers are called **peers** (ITU, 1994).
In other words, peers communicate with one another by using a protocol (ITU, 1994).
If layer n of one computer needs to communicate with layer n of another computer, the sender passes the information down to layer n−1 and continues until it reaches the lowest layer, usually the physical layer, where the information is actually transmitted. At the receiver, the information then moves progressively up through the layers until it reaches layer n (ITU, 1994).
The complete set of layers and protocols is called a **network architecture** (ITU, 1994)**.**

Example
To understand the basic mechanisms of a network architecture, consider the following analogy: an Indian philosopher wants to speak with an African philosopher, but the two do not speak the same language.
Each philosopher therefore relies on a translator, and the two translators agree to use English as a common language.
Each translator, in turn, gives the text to a secretary, and the two secretaries agree to transmit it by fax.

Communication therefore takes place as follows: the Indian philosopher expresses an idea to the translator; the translator converts it into English and passes it to the secretary; the secretary sends it by fax.
At the other end, the secretary receives the fax and gives it to the translator, who translates the text from English into the African philosopher's language and delivers it.
Each layer ideally communicates with its peer: the philosophers exchange ideas, the translators exchange texts in English, and the secretaries exchange faxes. In reality, however, no layer communicates directly with its counterpart at the other end. Each layer passes the message to the layer below it until physical transmission occurs at the lowest layer.
Communication could not take place without an agreement between the peers at each layer: English between the translators and fax transmission between the secretaries. These agreements are the protocols. In addition, each layer can change its own agreement without affecting the others. If the translators switched from English to French, or the secretaries from fax to email, the philosophers would not even notice (ITU, 1994).



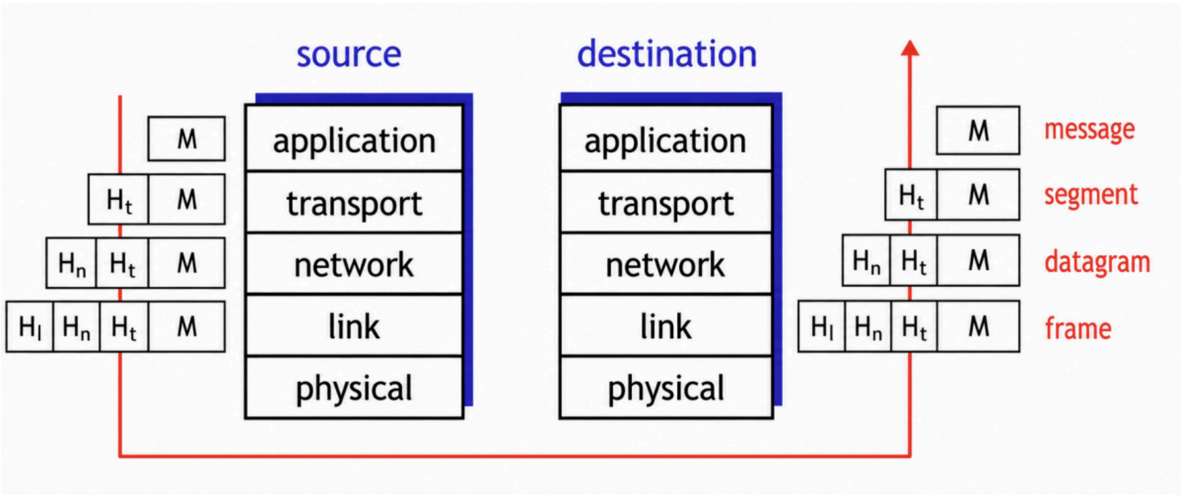
The Internet is organized into the following layers, in this order:

- **Application layer.**
- **Transport layer.**
- **Network layer.**
- **Link layer.**
- **Physical layer.**

When data moves from one layer to the layer below it, it is encapsulated. This means that each layer takes the data received from the layer above and adds its own header, an initial section containing control information (Kurose & Ross, 2026).

The content received from the layer above becomes the payload of the new data unit. In this way, each layer adds the information required to perform its own task (Kurose & Ross, 2026).

This new unit becomes the packet passed to the layer below (Kurose & Ross, 2026).



Supplementary materials

Key points

- **A network protocol** is a set of rules that defines how devices communicate.
- It establishes the format and order of messages, as well as the behavior to follow in case of errors or missing responses.
- Protocols ensure **interoperability** between different systems and software.
- To reduce complexity, networks are organized into **layers**.
- Each layer provides services to the layer above it and hides the details of its own operation.
- Entities belonging to the same layer on different devices are called **peers** and ideally communicate using the same protocol.
- The complete set of layers and protocols forms **the network architecture**.
- The Internet uses five layers: **application**, **transport**, **network**, **link**, and **physical**.
- Data travels down through the layers to the physical layer and then moves up through the layers of the receiving device.

Exercises

Two Programs That Cannot Communicate
Two companies independently develop two applications that need to exchange data over the Internet.

- Application A sends messages in the following format: COMMAND → IDENTIFIER → DATA
- Application B instead expects: IDENTIFIER → DATA → COMMAND

In addition, A immediately sends a new message if it does not receive a response within five seconds, while B interprets silence as the end of the communication.

Analyze the situation.

- Explain why the simple fact that the two computers are connected to the network does not guarantee that the applications can communicate correctly.
- Identify at least two communication rules on which the two programs must agree.
- Explain the function performed by a network protocol in the situation described.
- Relate the problem to the concept of interoperability.
- Explain why two programs developed by different companies can still communicate if they correctly implement the same protocol.
- Propose a minimum set of rules that would allow the two applications to communicate correctly.

The Message Travels Through the Layers
A browser needs to send a request to a web server.
Consider the architecture described in the learning material:
Application → Transport → Network → Link → Physical

Reconstruct the journey of the data.

- Identify the layer at which the request generated by the browser originates.
- Explain why the application layer does not transmit data directly through the physical medium.
- Describe what happens as the data is progressively passed to the lower layers.
- Explain what encapsulation means.
- Explain the difference between a header and a payload during the encapsulation process.
- Imagine that the original message is M: represent, in a simplified way, how it may change as the headers of the different layers are added.
- Once the data reaches the destination computer, describe the direction in which it must pass through the layers to reach the server application.
- Explain why, even though the client's application layer ideally communicates with the server's application layer, the data must actually pass through all the lower layers.

Changing a Technology Without Changing the Entire System

An application works correctly on two computers connected through a particular network technology.

Later, the infrastructure is modified: a technology used in the lower layers is changed, but the application continues to work without being rewritten.

Analyze why this is possible.

- Explain the advantage of organizing a network as a protocol stack.
- Explain what it means for each layer to provide services to the layer above while hiding the details of its own implementation.
- Relate the situation to the analogy of philosophers, translators, and secretaries used in the learning material.
- In the analogy, imagine that the secretaries replace the fax with another communication system that is compatible between them. Explain why the philosophers can continue to communicate without knowing about this change.
- Explain what is meant by a peer within a layered architecture.
- Identify which components are peers if we consider the transport layer on two different computers.
- Explain why peers at the same layer ideally communicate with each other through a protocol, while the actual transfer of data takes place by passing through the lower layers.
- Use the case described to explain why dividing a network into layers reduces the complexity of designing and modifying a network.

Bibliography

Braden, R. (Ed.). (1989). *Requirements for Internet Hosts—Communication Layers* (RFC 1122). RFC Editor. https://doi.org/10.17487/RFC1122

Carpenter, B. (Ed.). (1996). *Architectural principles of the Internet* (RFC 1958). RFC Editor. https://doi.org/10.17487/RFC1958

International Telecommunication Union. (1994). *Information technology—Open Systems Interconnection—Basic Reference Model: The basic model* (Recommendation ITU-T X.200). https://www.itu.int/rec/T-REC-X.200-199407-I/en

Kurose, J. F., & Ross, K. W. (2026). *Computer networking: A top-down approach* (9th ed.). Pearson. https://www.pearson.com/en-us/subject-catalog/p/computer-networking-a-top-down-approach/P200000013385/9780135415603

Further reading

- James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach, 9th ed., Pearson, 2026 — Provides a clear introduction to network protocols and layered architectures, explaining how the application, transport, network, link, and physical layers cooperate, how protocols define communication between peer entities, and how data is encapsulated as it moves through the protocol stack.
- Andrew S. Tanenbaum, Nick Feamster, and David J. Wetherall, Computer Networks, 6th ed., Pearson, 2022 — Offers a systematic explanation of network architectures, protocols, layers, interfaces, and services, showing how complex communication systems can be divided into independent layers and how different networking technologies fit within this architecture.
- Behrouz A. Forouzan, Data Communications and Networking with TCP/IP Protocol Suite, 6th ed., McGraw Hill, 2022 — Uses a bottom-up approach in which each layer of the TCP/IP protocol suite is built on the services provided by the layer below, making it particularly useful for understanding protocol layering, encapsulation, and the relationships among the physical, data-link, network, transport, and application layers.
- Larry L. Peterson and Bruce S. Davie, Computer Networks: A Systems Approach, 6th ed., Morgan Kaufmann, 2021 — Explores networking through real examples of network and protocol design, emphasizing how individual protocols and network components interact as parts of a larger system and how layered abstractions help manage the complexity of communication.
- Douglas E. Comer, Internetworking with TCP/IP, Volume One, 6th ed., Pearson, 2014 — Provides a detailed conceptual treatment of TCP/IP protocols and Internet architecture, with dedicated coverage of protocol layering, the TCP/IP five-layer model, packet formats, encapsulation, addressing, and the principles that allow different networks and systems to interoperate.

Editorial notes

In producing these materials, I have used generative artificial intelligence tools to support the writing process, particularly to improve the form of the text, reorganize content, refine wording, and speed up certain editorial tasks.

As I work independently on the production of these materials, I try to automate any activities that can reasonably be automated, so that I can devote more time to research, design, and content development.

However, artificial intelligence does not determine the content of this work: the choice of topics, structure, ideas, interpretations, examples, and teaching approach are developed by me. AI is therefore used primarily as a tool to support production and formal revision, while authorship and responsibility for the design and content remain mine.

Unless otherwise indicated, this material is distributed under the Creative Commons Attribution–NonCommercial–ShareAlike 4.0 International license (CC BY-NC-SA 4.0).

You are therefore free to share, redistribute, adapt, and create derivative works based on this material, provided that appropriate credit is given, the material is not used for commercial purposes, and any modified or derivative versions are distributed under the same license.