



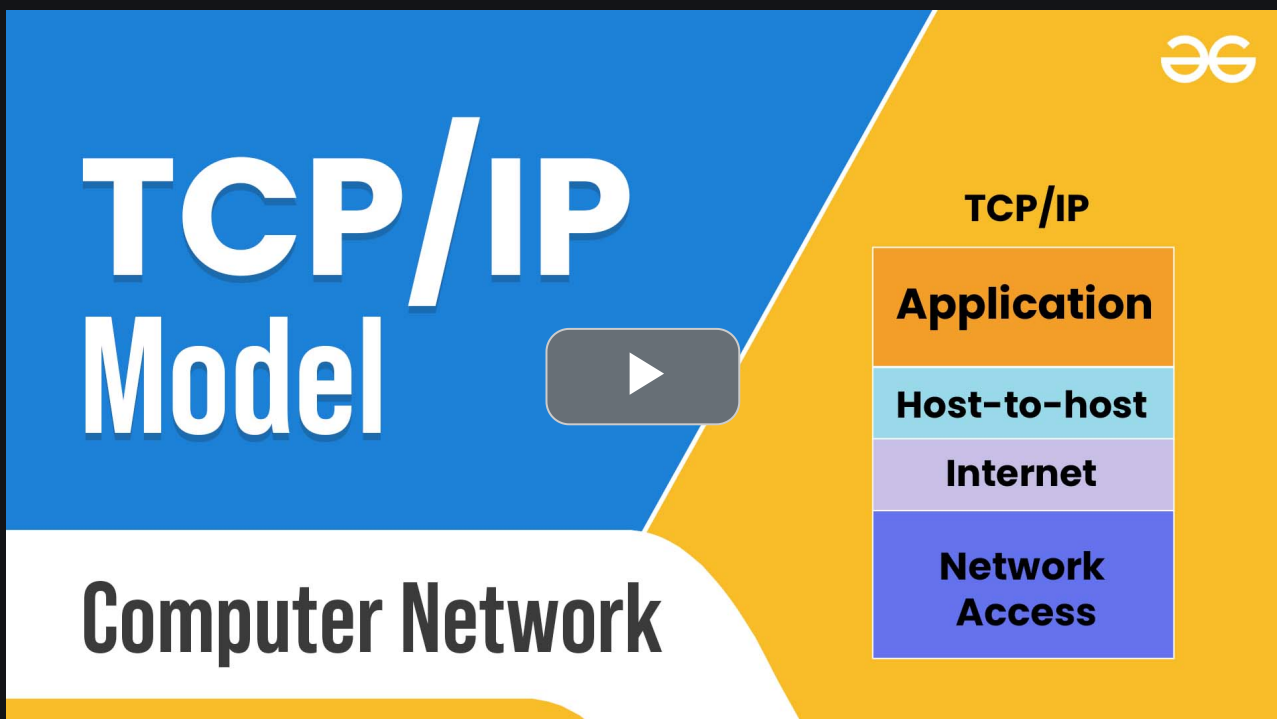
TCP/IP Model

Difficulty Level : Easy • Last Updated : 30 Sep, 2020



Prerequisite – [Layers of OSI Model](#)

The **OSI Model** we just looked at is just a reference/logical model. It was designed to describe the functions of the communication system by dividing the communication procedure into smaller and simpler components. But when we talk about the TCP/IP model, it was designed and developed by Department of Defense (DoD) in 1960s and is based on standard protocols. It stands for Transmission Control Protocol/Internet Protocol. The **TCP/IP model** is a concise version of the OSI model. It contains four layers, unlike seven layers in the OSI model. The layers are:



1. Process/Application Layer
2. Host-to-Host/Transport Layer
3. Internet Layer
4. Network Access/Link Layer



Start Your Coding Journey Now!

[Login](#)
[Register](#)

TCP/IP MODEL

Application Layer

Transport Layer

Internet Layer

Network Access Layer

OSI MODEL

Application Layer

Presentation Layer

Session Layer

Transport Layer

Network Layer

Data Link Layer

Physical Layer

Difference between TCP/IP and OSI Model:

TCP/IP

TCP refers to Transmission Control Protocol.

TCP/IP has 4 layers.

TCP/IP is more reliable

TCP/IP does not have very strict boundaries.

TCP/IP follow a horizontal approach.

TCP/IP uses both session and presentation layer in the application layer itself.

TCP/IP developed protocols then model.

OSI

OSI refers to Open Systems Interconnection.

OSI has 7 layers.

OSI is less reliable

OSI has strict boundaries

OSI follows a vertical approach.

OSI uses different session and presentation layers.

OSI developed model then protocol.

Transport layer in TCP/IP

In OSI model, transport layer



Start Your Coding Journey Now!

[Login](#)[Register](#)

TCP/IP model network layer only provides connection less services.

Connection less and connection oriented both services are provided by network layer in OSI model.

Protocols cannot be replaced easily in TCP/IP model.

While in OSI model, Protocols are better covered and is easy to replace with the change in technology.

The first layer is the Process layer on the behalf of the sender and Network Access layer on the behalf of the receiver. During this article, we will be talking on the behalf of the receiver.

1. Network Access Layer –

This layer corresponds to the combination of Data Link Layer and Physical Layer of the OSI model. It looks out for hardware addressing and the protocols present in this layer allows for the physical transmission of data.

We just talked about ARP being a protocol of Internet layer, but there is a conflict about declaring it as a protocol of Internet Layer or Network access layer. It is described as residing in layer 3, being encapsulated by layer 2 protocols.

2. Internet Layer –

This layer parallels the functions of OSI's Network layer. It defines the protocols which are responsible for logical transmission of data over the entire network. The main protocols residing at this layer are :

1. **IP** – stands for Internet Protocol and it is responsible for delivering packets from the source host to the destination host by looking at the IP addresses in the packet headers. IP has 2 versions:
IPv4 and IPv6. IPv4 is the one that most of the websites are using currently. But IPv6 is growing as the number of IPv4 addresses are limited in number when compared to the number of users.
2. **ICMP** – stands for Internet Control Message Protocol. It is encapsulated within IP datagrams and is responsible for providing hosts with information about network



Start Your Coding Journey Now!

[Login](#)[Register](#)

address of a host from a known IP address. ARP has several types: Reverse ARP, Proxy ARP, Gratuitous ARP and Inverse ARP.


3. Host-to-Host Layer –

This layer is analogous to the transport layer of the OSI model. It is responsible for end-to-end communication and error-free delivery of data. It shields the upper-layer applications from the complexities of data. The two main protocols present in this layer are :

1. **Transmission Control Protocol (TCP)** – It is known to provide reliable and error-free communication between end systems. It performs sequencing and segmentation of data. It also has acknowledgment feature and controls the flow of the data through flow control mechanism. It is a very effective protocol but has a lot of overhead due to such features. Increased overhead leads to increased cost.
2. **User Datagram Protocol (UDP)** – On the other hand does not provide any such features. It is the go-to protocol if your application does not require reliable transport as it is very cost-effective. Unlike TCP, which is connection-oriented protocol, UDP is connectionless.

4. Application Layer –

This layer performs the functions of top three layers of the OSI model: Application, Presentation and Session Layer. It is responsible for node-to-node communication and controls user-interface specifications. Some of the protocols present in this layer are: HTTP, HTTPS, FTP, TFTP, Telnet, SSH, SMTP, SNMP, NTP, DNS, DHCP, NFS, X Window, LPD. Have a look at [Protocols in Application Layer](#) for some information about these protocols. Protocols other than those present in the linked article are :

1. **HTTP and HTTPS** – HTTP stands for Hypertext transfer protocol. It is used by the World Wide Web to manage communications between web browsers and servers. HTTPS stands for HTTP-Secure. It is a combination of HTTP with SSL(Secure Socket Layer). It is efficient in cases where the browser need to fill out forms, sign in, authenticate and carry out bank transactions.
-  **SSH** – SSH stands for Secure Shell. It is a terminal emulations software similar to Telnet. The reason SSH is more preferred is because of its ability to maintain the encrypted connection. It sets up a secure session over a TCP/IP connection.

Start Your Coding Journey Now!

[Login](#)[Register](#)

bank transactions. Assume the following situation without the presence of NTP. Suppose you carry out a transaction, where your computer reads the time at 2:30 PM while the server records it at 2:28 PM. The server can crash very badly if it's out of sync.

This article is contributed by **Achiv Chauhan** and **Palak Jain**. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Interview Series

Prepare for free
Every Sunday | 7 - 8:30 PM IST



Like 313

[< Previous](#)[Next >](#)

Start Your Coding Journey Now!

[Login](#)[Register](#)

RECOMMENDED ARTICLES

Page : [1](#) [2](#) [3](#)

01 Difference between E-R Model and Relational Model in DBMS

16, Apr 20

05 Generalization, Specialization and Aggregation in ER Model

11, Oct 17

02 Similarities between TCP/IP model and OSI model

01, May 21

06 Enhanced ER Model

24, Oct 17

03 This is exactly why we still use the OSI model when we have TCP/IP Model

01, Jun 21

07 Mathematics | Hypergeometric Distribution model

08, Oct 18

04 Introduction of ER Model

13, Oct 15

08 Attributes to Relationships in ER Model

28, Jan 19



Article Contributed By :



GeeksforGeeks

Vote for difficulty

Current difficulty : [Easy](#)[Easy](#)[Normal](#)[Medium](#)[Hard](#)[Expert](#)Improved By : [Palak Jain 5](#), [ashushrma378](#), [rahulhindochoa05](#), [mohitmm](#), [satyajitdebnath87](#)

Start Your Coding Journey Now!

[Login](#)[Register](#)[Improve Article](#)[Report Issue](#)

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

[Load Comments](#)

 A-143, 9th Floor, Sovereign Corporate Tower,
Sector-136, Noida, Uttar Pradesh - 201305

 feedback@geeksforgeeks.org



Company

- [About Us](#)
- [Careers](#)
- [In Media](#)
- [Contact Us](#)
- [Privacy Policy](#)
- [Copyright Policy](#)

News

- [Top News](#)
- [Technology](#)
- [Work & Career](#)
- [Business](#)

Learn

- [Algorithms](#)
- [Data Structures](#)
- [SDE Cheat Sheet](#)
- [Machine learning](#)
- [CS Subjects](#)
- [Video Tutorials](#)
- [Courses](#)

Languages

- [Python](#)
- [Java](#)
- [CPP](#)
- [Golang](#)



Start Your Coding Journey Now!

[Login](#)[Register](#)

Knowledge

Kotlin

Web Development

[Web Tutorials](#)[Django Tutorial](#)[HTML](#)[JavaScript](#)[Bootstrap](#)[ReactJS](#)[NodeJS](#)

Contribute

[Write an Article](#)[Improve an Article](#)[Pick Topics to Write](#)[Write Interview Experience](#)[Internships](#)[Video Internship](#)

@geeksforgeeks , Some rights reserved

