


Last Updated: May 3, 2024 Easy

Application Layer Protocols in Computer Network

 Author
Aayushee ahlawat

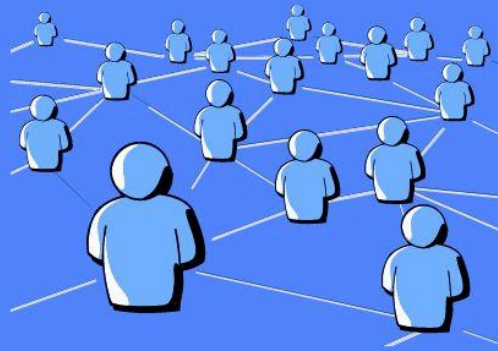
 Share  0 upvote

 Table of contents >

Introduction

Do you know that networking and communication rely on protocols to ensure that devices can communicate effectively? At the application layer of the OSI model, specialized protocols are used to facilitate the transfer of data and support services such as email, file transfer, and web browsing. Whether you are a networking enthusiast or looking to enhance your technical knowledge, read on to discover the fascinating world of application layer protocols!

Application layer protocols



**CODING
NINJAS**

This article will explore different application layer protocols, their functions, and their advantages for communication between devices.

What are Application Layer Protocols?

Understanding [application layer](#) protocols for efficient data transfer and communication in computer networks is crucial.

Applications running on various devices can communicate more easily thanks to a group of protocols known as application layer protocols, which operate at the top layer of the OSI model. Application layer protocols define how applications running on different devices pass messages to each other over a network. These protocols support numerous services, including email, web browsing, and file transfer, and they define the structure and content of the transmitted data. HTTP, FTP, SMTP, POP, TELNET, TFTP, LDAP and DNS are typical application layer protocols.

Each protocol has its own set of rules and specifications that define how messages can be transmitted and received between different devices on a network. Below is the list of application layers protocols.

List of the Application Layer Protocols in Computer Network

1. Hypertext Transfer Protocol (HTTP)

The World Wide Web's foundational protocol is **HTTP**. It supports web client and server communication and loads web pages using hypertext links. An individual, known as the user-agent, requests a server using the client-server protocol known as HTTP. The user agent is typically a web browser.

The protocol outlines the message transmission and reception between the client and server and how to exchange resources like HTML or hypertext documents.

JavaScript programmers can use the **fetch()** API or the Axios library to send HTTP requests. Here is an example of how to make the request using the **fetch()** API:

```
fetch('https://www.example.com/data.json')
  .then(response => response.json())
  .then(data => {
    //Do something with the data
  });
```

2. File Transfer Protocol (FTP)

FTP is an application layer protocol that transfers files between local and remote systems. It runs on TCP/IP and uses separate control and data connections. The end user's computer is the local host in an FTP transaction. FTP can be used to transmit files from one host to another. FTP is often secured with SSL/TLS or replaced with SSH File Transfer Protocol (SFTP) to encrypt the content and protect the username and password.

The FTP protocol is widely used to exchange data between hosts, update websites, and deliver content. ALG (application layer gateway) can be enabled or disabled for the FTP protocol for a DS-Lite configuration.

The FTP protocol can be used with several commands, including **PORT**, **PWD**, **LIST**, **CD**, **PUT**, and **GET**.

Common characteristics of the FTP (File Transfer Protocol) protocol include:

1. FTP operates over TCP/IP and is used for transferring files between a client and a server.

2. It uses separate control and data connections: the control connection for sending commands and receiving responses, and the data connection for transferring files.
3. FTP supports various authentication methods, including username/password authentication and anonymous FTP.
4. It allows for both ASCII and binary file transfer modes, enabling the transfer of text and binary files with appropriate handling of line breaks and character encoding.

3. Domain Name System (DNS)

DNS is a hierarchy and distributed naming system for computers, services, and other Internet resources or IP networks. DNS is responsible for translating domain names into IP addresses for locating and identifying computer services and devices with the underlying network protocols. It is responsible for assigning domain names to authoritative name servers for each domain, providing distributed and fault-tolerant service.

DNS defines the DNS protocol, which specifies the technical functionality of the database service at its core and the data structures and communication exchanges used.

'**dig**', '**host**', and '**nslookup**' are frequently used DNS protocol commands. You can use these commands to ask DNS servers questions and get data on domains, IP addresses, and other related topics. Using the '+short' option with the 'dig' or 'host' command will provide a concise response to a DNS query.

Common characteristics of the DNS protocol include:

- Translates domain names into IP addresses.
- Operates over UDP/TCP on port 53.
- Utilizes hierarchical distributed naming system.
- Provides domain name resolution for internet resources.

4. Simple Mail Transfer Protocol (SMTP)

It is a widely used Internet protocol for sending emails between servers. When sending emails from one mail server to another, **SMTP** uses other protocols like POP3 or IMAP to get the emails to their destinations. SMTP establishes the format for email messages and transmits them using TCP port 25 or 587.

STARTTLS or SSL/TLS are two encryption protocols that can be used with SMTP to protect email transmission. **SMTP** provides a dependable and effective method for sending and receiving emails online. The client and server exchange commands and responses using the text-based SMTP protocol.

The SMTP protocol's standard commands include **EHLO**, **MAIL FROM**, **RCPT TO**, **DATA**, and **QUIT**. These commands are used in the above order to open communication between the client and the email server, send an email, and end the session.

Common characteristics of the SMTP protocol include:

- Used for sending email messages between servers.
- Operates over TCP on port 25.
- Follows a client-server model for email transmission.
- Supports basic email delivery and forwarding functionalities.

5. Post Office Protocol (POP)

Email can be retrieved from a server and delivered to a local client using this application layer protocol. Users can manage and read email messages locally on their devices using **POP**. POP3 is the most recent iteration of POP. For unencrypted transmission, it uses

TCP port 110, and for encrypted communication, it uses port 995.

IMAP downloads a copy of the email and leaves the original on the server, whereas POP3 downloads and deletes it from the server. POP3 is a text-based protocol that involves requests from the client and answers from the server.

The POP3 protocol has commands such as **USER**, **PASS**, **LIST**, **RETR**, and **QUIT**.

Common characteristics of the POP protocol include:

- Retrieves email from a server for local storage.
- Operates over TCP on port 110.
- Typically uses POP3 for downloading messages to a client device.
- Supports simple email retrieval and deletion.

6. Telnet

"Teletype network" is what **Telnet** stands for. A remote terminal access application layer protocol is used. Telnet enables users to sign in to a distant computer as if they were physically nearby. TCP port 23 is used for transmission.

Telnet is a text-based protocol where the client and server exchange commands and answers. Telnet is insecure because it transmits data and login credentials in clear text, leaving it open to interception.

The central command to use Telnet is to connect to a remote server with **telnet <serverIP>** command.

Common characteristics of the Telnet protocol include:

- Telnet is a protocol used for remotely accessing and managing devices over a network.
- Operates over TCP on port 23.
- Provides a text-based interface for logging into remote systems and executing commands.
- Lacks encryption and security features, making it vulnerable to eavesdropping and interception.
- Historically widely used for remote administration of servers and network devices but now considered insecure due to its lack of encryption, often replaced by more secure protocols like SSH (Secure Shell).

7. Trivial File Transfer Protocol (TFTP)

To transfer files between network devices, **TFTP** is used. It's a condensed form of FTP, and user authentication is unnecessary. TFTP uses UDP port 69 for transmission. It is a text-based protocol where the client and server exchange commands and answers. TFTP is frequently used to back up network devices, transfer firmware updates, and start diskless workstations.

The TFTP protocol command is simply typing **"tftp"** followed by the server's IP address. After entering this command, you can use specific TFTP commands like "get" and "put" to transfer files between the local computer and the remote TFTP server.

Common characteristics of the TFTP protocol include:

- Simplified version of FTP used for basic file transfers.
- Operates over UDP on port 69.
- Lacks authentication and error checking features.
- Primarily used for bootstrapping devices over a network.

8. Lightweight Directory Access Protocol (LDAP)

An application layer protocol called **LDAP** is used to access and manage distributed directory information services. It enables network-wide sharing of data about users, systems, networks, services, and applications. It is a widely used, vendor-neutral protocol that transmits data over TCP port 389 and supports SSL/TLS for secure communication.

LDAP employs a client-server architecture with requests and responses and stores data hierarchically.

Common LDAP commands include *ldapsearch*, *ldapmodify*, *ldapadd*, and *ldapdelete*. With the help of these commands, you can browse, edit, add, and delete entries in a directory service.

Common characteristics of the LDAP protocol include:

- Provides directory services for accessing and managing directory information.
- Operates over TCP on port 389.
- Supports queries and updates to directory services.
- Utilizes a hierarchical data structure for storing directory information.

9. Dynamic Host Configuration Protocol (DHCP)

The network protocol known as [Dynamic Host Configuration Protocol](#) (DHCP) automates the distribution of IP addresses, subnet masks, gateways, and DNS servers to devices connected to a network. It manages and distributes these configurations in a dynamic manner, streamlining network administration and enabling smooth device connection and communication.

Common characteristics of the DHCP protocol include:

- Assigns IP addresses dynamically to devices on a network.
- Operates over UDP on port 67/68.
- Automates the configuration of network parameters such as IP addresses, subnet masks, and default gateways.
- Supports lease management for IP address allocation.

10. Simple Network Management Protocol (SNMP)

The [Simple Network Management Protocol](#) (SNMP) is a widely used protocol for controlling and keeping track of network devices. Administrators can use it to access data and change settings on servers, switches, and other network equipment. Through centralized network administration made possible by SNMP, problems can be quickly identified and effectively fixed.

Common characteristics of the SNMP protocol include:

- Manages network devices and monitors their performance.
- Operates over UDP on port 161/162.
- Uses a manager-agent architecture for network management.
- Provides a standardized framework for collecting and organizing information about network devices.

Frequently Asked Questions

What is the security of the application layer?

The security of the application layer involves implementing measures to protect applications and their data from unauthorized access, manipulation, and other threats. This includes encryption, authentication, access control, input validation, and measures to prevent common web application vulnerabilities like SQL injection and cross-site scripting (XSS).