Introduction to TCP/IP protocols

TCP/IP protocols (a.k.a Internet protocol suite) is the conceptual model and set of communications protocols used on the Internet and similar computer networks. It provides end-to-end data communication specifying how data should be packetized, addressed, transmitted, routed, and received.

The history of TCP/IP protocols

The TCP/IP protocols came from the ARPANET(he Advanced Research Projects Agency Network), developed by ARPA（Advanced Research Project Agency），which is dedicated to provide a stable interconnection between independent network. First, the data can still be delivered even if some part of the system is attacked and break down. Second, it can work through all the platforms.

The technical standards are maintained by Internet Engineering Task Force (IETF). All the details about TCP/IP could be found in Request For Comment (RFC), which could be download in ds.internic.net or <ftp://nic.merit.edu/internet/documents/rfc/>

The character of TCP/IP

Connectionless Packet Delivery Service

Which means all the packet delivery doesn’t require a connection between the hosts. TCP/IP divides the data into small packets and deliver them separately. We can’t the make sure there will be no loss, error insertion, misdelivery, duplication, or out-of-sequence delivery of the packet. So, we need mechanism to ensure that. But this way has an advantage which is it allows for multicast and broadcast. That will be more effective compared to connection-oriented communication.

Reliable Stream Transport Service

This solve the question of reliability. We need a protocol to ensure that the data has been received. TCP does this work. The basic method is to packetize the packets, number them, deliver them and wait for the acknowledgement from the destination, which will be fully explained later.

Network Technology Independent

Among all the packet-switched network, TCP/IP is independent with the hardware, which means it will works through all the networks.

Universal Interconnection

Once TCP/IP is used to connect the internet, the computer will have one the only discriminant address, which we call normally IP address. In all the packet-switch process, It all depends on this address. No matter which routes passes by the packets, it will finally be routed to the specific address.

End-to-End Acknowledgements

TCP/IP use End-to-End Acknowledgements, which means we don’t need to care about the devices participating in the packet-switch process.

Application Protocol Standards

TCP/IP not only provide basic deliver service, it has also provide many application protocol standards, which could be referenced by the developer in avoid of overlapping development.

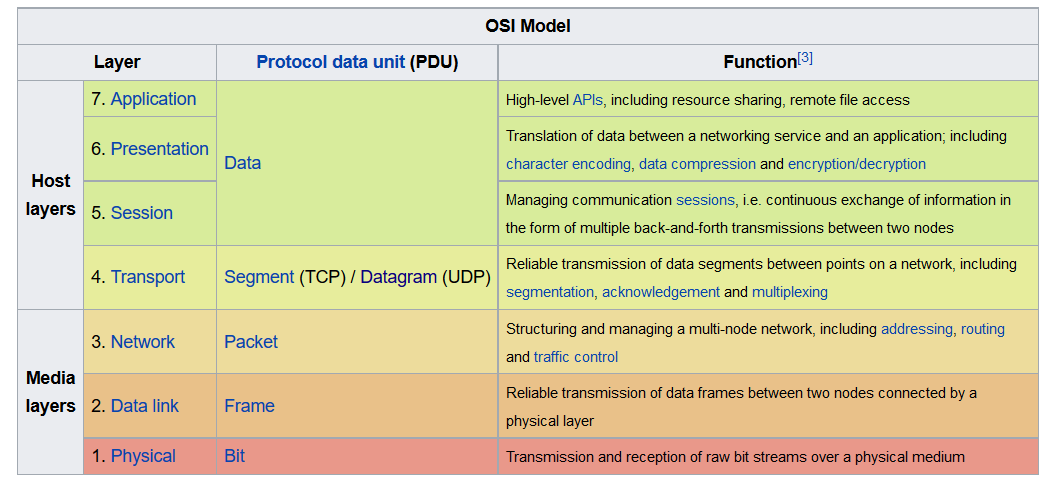
Open System Interconnection (OSI)

If you want to understand TCP/IP, first we need to understand Open System Interconnection (OSI model), which is a conceptual model that characterizes and standardizes the communication functions of a telecommunication or computing system without regard to their underlying structure and technology. Its goal is the interoperability of diverse communication systems with standard protocols. The model partitions a communication system into 7 abstraction layers. Functionally, it could be divided into two groups: host layers group and Media layers group.

They are

|  |
| --- |
| Application |
| Presentation |
| Session |
| Transport |
| Network |
| Data Link |
| Physical |

These seven layers are used to comprehend other models. Every layer does its own job, at the same time cooperate with the two or one layer next to it. Each one receives the information from the lower layer, and transport them to the upper layer after dealing with them, which means each layer works separately. It doesn’t need to know how other layers function. All it need to do is to do his own job and everything well function well. The advantage of that is once we find a more efficient way of a specific layer, we don’t need to change all the protocol.



**Layer 1: Physical Layer**

In Physical layer, we define the electrical and physical specifications of the data connection.

Physical transmission medium:

an electrical cable, an optical fiber cable, or a radio frequency link etc…

Which are different in pins, voltages, line impedance, cable specifications, signal timing and similar characteristics for connected devices and frequency (5 GHz or 2.4 GHz etc.) for wireless devices

Transmission mode

simplex, half duplex, and full duplex

Network topology

Bus(ethernet) , mesh, or ring

Networking decives

network adapters, repeaters, network hubs, modems, and fiber media converters

### Layer 2: Data Link Layer

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Preamble | Destination | Source | Message Type | Data | Frame check sequence | |
| 8 bytes | 6 bytes | 6 bytes | 2 bytes | 46-1500 bytes | | 4 bytes |

In the physical layer, the protocol data unit(PDU) is bit, But in data link layer, we should define the message unit we are going to use, their format and how they gonna be transport through the internet. In TCP/IP we use Frame as the message unit, which will be like this.

Data link layer always have following functions:

Managing the MAC address of the network adapter

Establishment of the connection of Virtual Circuit and Logical Link and disconnection.

The transportation of the frame and error detection.

Managing the order of the frames.

Build the frames and regroup them.

Reception of acknowledge frame, resend them once they got lost.

Transform the frames to Physical Layer

Check the MAC address of the destination.

In a line, the datalink layer try to make sure a correct physical data transportation.

### Layer 3: Network Layer

The network layer provides the functional and procedural means of transferring variable length data sequences (called datagrams) from one node to another connected in "different networks"

1. If the packet is not for the node in the same network, it will be delivered to router.
2. Traffic control: When the buffer zone is full, the router will inform other device to change the way.
3. When the volume of the packet is too big, the router will regroup them and transfer
4. Manage the explanation of the MAC address and the Internet Address(IP)

The Network will hide all the lower layers to the host layer. That’s way we could use different hardware.

### Layer 4: Transport Layer

The transport layer provides the functional and procedural means of transferring variable-length data sequences from a source to a destination host via one or more networks, while maintaining the quality of service functions.

1. To transfer the data from the upper layer into packets. Segmentation/Desegmentation.
2. Provides the acknowledgement of the successful data transmission
3. Error detection and recovery

### Layer 5: Session Layer

The session layer controls the dialogues (connections) between computers

1. Establish or terminate the connection computers
2. Allow computer to register an Internet address
3. Manage the sync between the computer, supervise the connection between computers, react to the errors.

### Layer 6: Presentation Layer

Two computers may use different syntax and semantics. The presentation layer help to transforms data into the form that the other computer may accept.

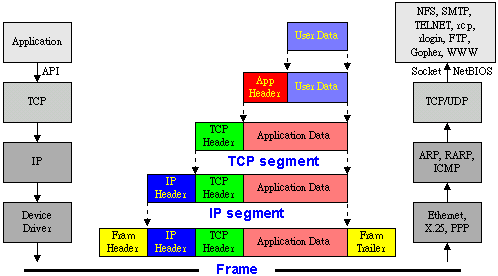
Example: Unicode to ASCII

### Layer 7: Application Layer

It defines all kinds of API, how the application connect to the internet.

|  |  |
| --- | --- |
| **Open System Interconnection** | **Internet Protocol Suite** |
|  |  |
| Application | Application |
| Presentation |
| Session |
| Transport | Transport |
| Network | Network |
| Data Link | Physical |
| Physical |

Encapsulation



During the data transfer process between the different layer, each layer builds a protocol data unit (PDU) by adding a header (and sometimes trailer) containing control information to the PDU from the layer above. That’s what we call encapsulation. Each layer encapsulate the data from previous layer, without being able to identify which part of the data is the header or trailer from the previous layer. When the PDUs arrive at the destination, it will be decapsulated by each layer and become the acceptable data to the application.

ARP

After we know about the TCP/IP models we can talk about the protocols. Let’s begin with the ARP protocol.

As we know in the Network Layer, router help us to select the next destination of our packets, but what we got here is only an Internet address (IP address). Even the packets are broadcasted to the network, every node won’t receive except it has a corresponding MAC address to itself. So the problem here is that we need a mapping between the IP address and the MAC address.

So how we do it?

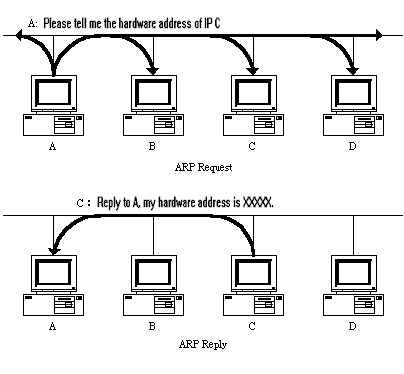
The first thing come to our mind is that why don’t we build a table of the mapping between the IP and the MAC for every computer. This good idea but the problem is : there are so many computers in the internet, and the IPs of them change very often, this table is going to hard to maintain. It possible that some devices don’t even have a disk to restore this table.

So we need ARP

ARP protocol stands for Address Resolution Protocol, is a communications protocol used for resolution of Internet layer addresses into link layer addresses(MAC, Hardware address), a critical function in the Internet protocol suite.

So how the address Resolution Protocol function.

1. Every host in the internet will build a table in their ARP Cache, to restore the mapping between the IP and MAC, every terms from this table have its own life time and will be removed once it’s done.
2. When a host got IP address want to send a packet to this address, it will first look for it through the table. If it does, then no problem, the packet will be broadcasted. But if it doesn’t the host will broadcast a ARP request packet demand for it, which include its own IP and MAC
3. Every host in the network will receive this packet. Ignore it if it’s not its IP address, if it is, first, the host will check the IP address and MAC of the sender, compared them with the one in the table, update them if they are different. And then the host broadcast its ARP replying packet to the network to inform the sender.
4. If the sender receives the reply packet, it will update the table and continue to deliver the packet.
5. If the sender doesn’t receive the reply packet, then it declares the failure of searching.



The ARP table only exist when the TCP/IP is active in the host, once the protocol is removed of host is shut down. The ARP table will be removed and will be rebuilt after the host turn on or the protocol is active again. At that time the host will send several ARP request to the network to make sure everyone has the right information.

RARP

Sometimes in a diskless workstation, there is nowhere to restore a IP address, without a IP address it couldn’t get connected to the network. To solve this problem, we have RARP, just like ARP but instead of demanding the IP address of the others, it demand the IP address of itself.

The form of a data link frame

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Preamble | Destination | Source | Message Type | Data | Frame check sequence | |
| 8 bytes | 6 bytes | 6 bytes | 2 bytes | 46-1500 bytes | | 4 bytes |

We could see that in the data field it has a room of 46-1500 bytes. Our ARP packet is encapsulated in this field.

The form of a data link frame

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 8 | 16 | 31 |

|  |  |  |
| --- | --- | --- |
| Hardware Type | | Protocol Type |
| HLEN | PLEN | Operation |
| Sender HA (Byte 0~3) | | |
| Sender HA (Byte 4~5) | | Sender IP (Byte 0~1) |
| Sender IP (Byte 2~3) | | Target HA (Byte 0~1) |
| Target HA (Byte 2~5) | | |
| Target IP (Byte 0~3) | | |

**HARDWARE TYPE**

that value indicates the type of the internet adapter. 1 means Ethernet

**PROTOCOL TYPE**

That value indicates the type of the protocole. 0x0800 mean IP

**HLEN**

The length of hardware address(byte)﹐in Ethernet the value is 6

**PLEN**

The length of protocal address(byte)﹐the value of IP is 4。

**OPERATION**

The type of packet. We have 4 types

1. ARP Request
2. ARP Reply
3. RARP Request
4. RARP Reply

**SENDER HA**

The hardware address of the sender.

**SENDER IP**

The IP address of the sender.

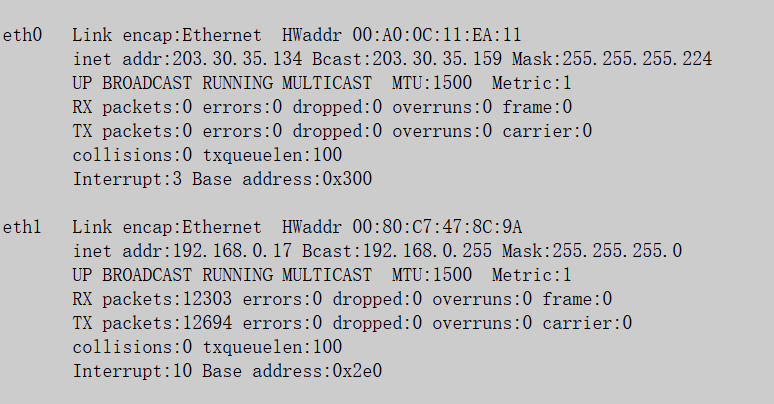
**TARGET HA**

The hardware address of the target.

**TARGET IP**

The IP address of the target.

In linux, we could use ifconfig to check our IP or MAC



We could also use arp to check our ARP tables

