TCP/IP协议

应用层

- 功能说明：将应用层、表示层和会话层功能合并，为用户提供网络服务（如HTTP、FTP、SMTP等协议）。
- 数据格式：数据（Data）
- （1）Data：包含用户名、密码的加密请求内容及请求头，格式化为 HTTP/HTTPS 请求。

Data举例：

如 {"username": "user123", "password": "pass123"}。

应用层数据 传给 传输层时，会把数据进行加密处理，变成：Encrypted Data: 3F 2B 7A C9 5E 12 AB 43 1A 96 ...

1）应用层 ==> 传输层

传输层

- 功能说明：提供端到端的通信，负责分段、传输和重组。常见协议有TCP（传输控制协议）和UDP（用户数据报协议）。
- 数据格式：段（Segment）/报文（Message）
- （2）段：加密请求被分为段，包含端口号、序列号、确认号。

TCP规则：分段、封装、传输

=== 分段过程：

密文数据较大的话，会被分成多个小段。

传递的是：【加密Data数据】

=== 封装过程：

1）每一小段被称为一个 TCP 报文段，传输层会为每个 TCP 报文段附加 TCP 头部信息；

2）头部信息包含必要的控制信息，如 源端口、目标端口、序列号、确认号、窗口大小等。这些信息用于管理和校验传输中的每个段。

3）封装后的格式：
TCP Header + Encrypted Data Segment

举例说明：
假设原始加密数据较大，被分成两个 TCP 报文段，每个段的内容如下（简化表示）

TCP报文段1：

TCP Header:
Source Port: 56324
Destination Port: 443
Sequence Number: 1000
Ack Number: 0
Flags: SYN
Window Size: 64240
Checksum: <calculated>
Encrypted Data: 3F 2B 7A C9 5E 12 ...

- 序列号是 1000，表示这是传输数据的起始序号
- 确认号是 0，通常表示初始连接请求时的状态。
- 标志位是 SYN，表示这是一个建立连接的请求报文段（SYN，同步序列编号）。
- 可能是：{"username": "user123"} 这一部分。

TCP报文段2：

TCP Header:
Source Port: 56324
Destination Port: 443
Sequence Number: 2460
Ack Number: 1001
Flags: ACK
Window Size: 64240
Checksum: <calculated>
Encrypted Data: AB 43 1A 96 ...

- 序列号是 2460，表示它承载的是从 2460 开始的后续数据。这意味着每个报文段携带了大约 1460 字节的数据，依次叠加。
- 确认号是 1001，表示它期望对方已经收到第一个报文段，接收端可以按此序号来确认数据的完整性。
- 标志位是 ACK，表示这是一个应答报文段，用于确认收到之前的数据。
- 可能是：包含 {"password": "pass123"} 这一部分。

备注：

1）标志位中的 SYN 用于连接建立，ACK 用于数据传输过程中的确认应答。

2）传输层 ==> 网络层

网络层

- 功能说明：负责路由选择和数据包转发。确保数据到达目标主机。主要协议是IP协议，辅助协议包括ICMP和ARP等。
- 数据格式：数据包（Packet）
- （3）数据包：传输层段被封装为 IP 数据包，包含源/目标 IP 地址，用于路由转发。

【IP数据包】的格式如下：

IP Header:
Version: 4
IHL: 5
Total Length: <length of entire packet>
Protocol: TCP (6)
Source IP: 192.168.1.1
Destination IP: 203.0.113.5
TTL: 64
Checksum: <calculated>

Payload (TCP Segment):
Source Port: 56324
Destination Port: 443
Sequence Number: 2460
Ack Number: 1001
Flags: ACK
Window Size: 64240
Checksum: <calculated>
Encrypted Data: AB 43 1A 96 ...

蓝色字体为：TCP报文段数据

粉色字体为：新加的 IP头部

增加了 IP 头部信息，用于路由和寻址。

传递的是：【报文段】

=== 封装步骤如下：

1）封装：将 TCP 报文段作为数据部分（Payload）封装到 IP 数据包中。

2）添加 IP 头部：为了能够通过网络正确转发，IP 数据包需要添加 IP 头部。这个头部包含源 IP 地址、目的 IP 地址等信息，帮助路由器确定如何将数据包转发到正确的目标。

3）路由和转发：IP 层的路由器根据目标 IP 地址确定下一跳并将数据包发送到目标网络。

IP协议：对【报文段】进行再次封装处理。

在网络层，传输层的 TCP 报文段 会被封装进 IP 数据包 中。

3）网络层 ==> 链路层

链路层

- 功能说明：负责设备之间的数据传输，管理物理地址（MAC地址）及链路连接，主要用于本地网络的数据传输。
- 数据格式：帧（Frame）
- （4）帧：网络层数据包被封装为数据帧，附加源/目标 MAC 地址，用于本地网络设备传输。

【数据帧】的格式如下：

以太网帧：
Ethernet Header:
Destination MAC: 00:1A:2B:3C:4D:5E
Source MAC: 00:6F:7G:8H:9I:0J
Type: 0x0800

Payload (IP Packet):
IP Header:
Version: 4
IHL: 5
Total Length: <length of entire packet>
Protocol: TCP (6)
Source IP: 192.168.1.1
Destination IP: 203.0.113.5
TTL: 64
Checksum: <calculated>

TCP Segment:
Source Port: 56324
Destination Port: 443
Sequence Number: 2460
Ack Number: 1001
Flags: ACK
Window Size: 64240
Checksum: <calculated>
Encrypted Data: AB 43 1A 96 ...

Ethernet Trailer:
Frame Check Sequence (FCS): <calculated>

蓝色字体为： IP数据包数据

粉色字体为：新加的 链路层头部/尾部

在 链路层，数据从 IP 数据包 被封装成了 以太网帧，加入了链路层头部（包含 MAC 地址）和尾部（用于校验）。

传递的是：【IP 数据包】

=== 封装步骤如下：

1）帧封装：将网络层的 IP 数据包封装进链路层的 帧 中。

2）添加链路层头部和尾部：链路层会在数据帧前面添加 头部（Frame Header），包含源和目标的 MAC 地址，并在帧的末尾添加 尾部（Frame Trailer），用于数据完整性检查。

3）帧的传输：链路层负责通过物理链路传输帧，使数据可以在相邻节点之间进行传输。

对【 IP数据包】进行再次封装处理。

在链路层，网络层的 IP数据包 会被封装进 数据帧 中，以便在物理网络上传输。经过链路层的处理，数据变成了一个 数据帧，其中包含 链路层的头部、IP 数据包（即 Payload），以及链路层的尾部。