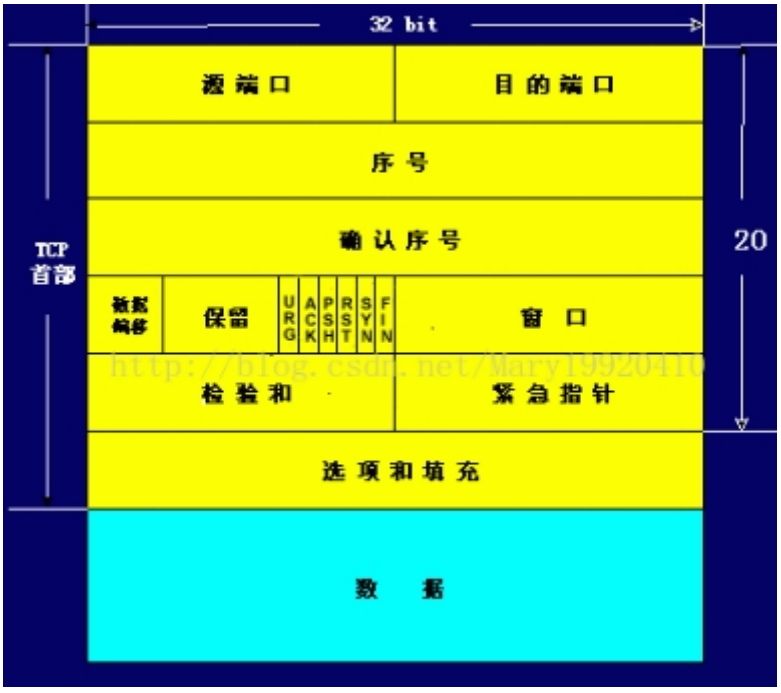
实验(二十一)：TCP段分析实验

一.实验目的

- **理解TCP的基本功能和结构**：通过分析TCP报文段的不同字段，了解如何标识和区分不同的TCP连接，以及TCP如何保证数据传输的可靠性和有序性。
- **掌握TCP连接的建立和拆除过程**：详细分析TCP的三次握手和四次挥手过程，了解这些控制机制如何支持可靠的、全双工的数据传输。
- **使用分析工具观察TCP数据流**：利用Packet Tracer和Wireshark这类工具实时捕捉和分析TCP数据包，理解实际网络环境中TCP的行为。

二.实验原理

- TCP概述
  - TCP是传输层的协议，功能即为在IP的数据报服务之上增加了最基本的服务：复用和分用以及差错检测。TCP 是一个基于连接的四层协议，提供全双工地，可靠地传输系统。它能够保证数据被远程主机接收。并且能够为高层协议提供flow-controlled 服务。空间上，TCP需要在端系统中维护连接状态，需要一定的开销。此连接装入包括接收和发送缓存，拥塞控制参数和序号与确认号的参数。UDP不维护连接状态，也不跟踪这些参数，开销小。空间和时间上都具有优势。
- TCP报文格式
  - TCP报文是TCP层传输的数据单元，也叫报文段。

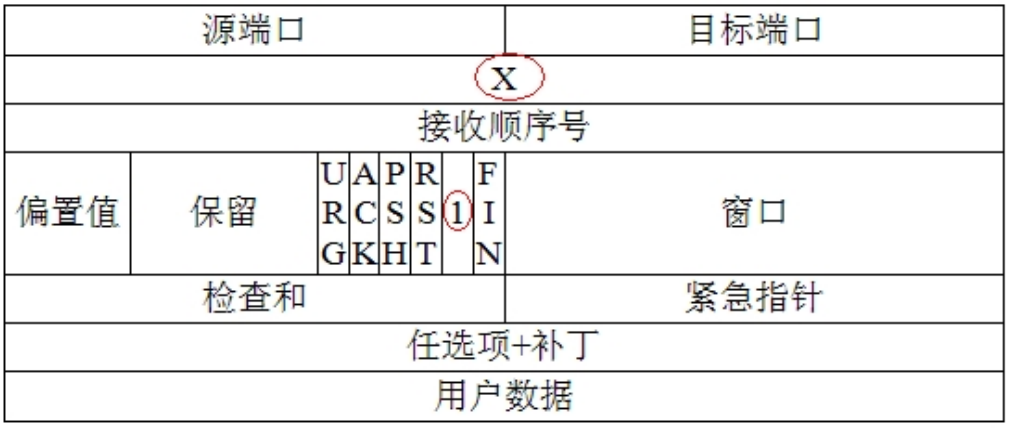


- TCP报文字段
  - 端口号：用来标识同一台计算机的不同的应用进程。
    - 源端口：源端口和IP地址的作用是标识报文的返回地址。
    - 目的端口：端口指明接收方计算机上的应用程序接口。

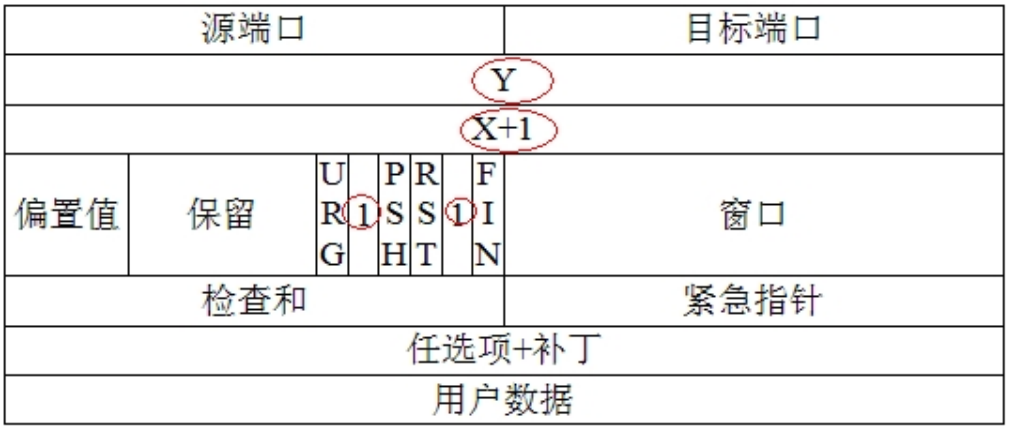
TCP报头中的源端口号和目的端口号同IP数据报中的源IP与目的IP唯一确定一条TCP连接。

- 序号和确认号：是TCP可靠传输的关键部分。序号是本报文段发送的数据组的第一个字节的序号。在TCP传送的流中，每一个字节一个序号。例如：一个报文段的序号为300，此报文段数据部分共有100字节，则下一个报文段的序号为400。所以序号确保了TCP传输的有序性。确认号，即ACK，指明下一个期待收到的字节序号，表明该序号之前的所有数据已经正确无误的收到。确认号只有当ACK标志为1时才有效。比如建立连接时，SYN报文的ACK标志位为0。
- 数据偏移 / 首部长度：4bits。由于首部可能含有可选项内容，因此TCP报头的长度是不确定的，报头不包含任何任选字段则长度为20字节，4位首部长度字段所能表示的最大值为1111，转化为10进制为15，15*32/8 = 60，故报头最大长度为60字节。首部长度也叫数据偏移，是因为首部长度实际上指示了数据区在报文段中的起始偏移值。
- 保留：为将来定义新的用途保留，现在一般置0。
- 控制位：URG ACK PSH RST SYN FIN，共6个，每一个标志位表示一个控制功能。
    - URG：紧急指针标志，为1时表示紧急指针有效，为0则忽略紧急指针。
    - ACK：确认序号标志，为1时表示确认号有效，为0表示报文中不含确认信息，忽略确认号字段。
    - PSH：push标志，为1表示是带有push标志的数据，指示接收方在接收到该报文段以后，应尽快将这个报文段交给应用程序，而不是在缓冲区排队。
    - RST：重置连接标志，用于重置由于主机崩溃或其他原因而出现错误的连接。或者用于拒绝非法的报文段和拒绝连接请求。
    - SYN：同步序号，用于建立连接过程，在连接请求中，SYN=1和ACK=0表示该数据段没有使用捎带的确认域，而连接应答捎带一个确认，即SYN=1和ACK=1。
    - FIN：finish标志，用于释放连接，为1时表示发送方已经没有数据发送了，即关闭本方数据流。
    - 窗口：滑动窗口大小，用来告知发送端接受端的缓存大小，以此控制发送端发送数据的速率，从而达到流量控制。窗口大小时一个16bit字段，因而窗口大小最大为65535。
    - 校验和：奇偶校验，此校验和是对整个的 TCP 报文段，包括 TCP 头部和 TCP 数据，以 16 位字进行计算所得。由发送端计算和存储，并由接收端进行验证。
    - 紧急指针：只有当URG 标志置 1 时紧急指针才有效。紧急指针是一个正的偏移量，和顺序号字段中的值相加表示紧急数据最后一个字节的序号。 TCP 的紧急方式是发送端向另一端发送紧急数据的一种方式。
    - 选项和填充：最常见的可选字段是最长报文大小，又称为MSS（Maximum Segment Size），每个连接方通常都在通信的第一个报文段（为建立连接而设置SYN标志为1的那个段）中指明这个选项，它表示本端所能接受的最大报文段的长度。
    - 选项和填充：选项长度不一定是32位的整数倍，所以要加填充位，即在这个字段中加入额外的零，以保证TCP头是32的整数倍。
    - 数据部分： TCP 报文段中的数据部分是可选的。在一个连接建立和一个连接终止时，双方交换的报文段仅有 TCP 首部。如果一方没有数据要发送，也使用没有任何数据的首部来确认收到的数据。在处理超时的许多情况中，也会发送不带任何数据的报文段。
- TCP连接过程
    - 相对于SOCKET开发者,TCP创建过程和链接折除过程是由TCP/IP协议栈自动创建的。因此开发者并不需要控制这个过程。但是对于理解TCP底层运作机制，相当有帮助。TCP连接过程简单一句话概括：“三次握手四次挥手”。
        - TCP三次握手

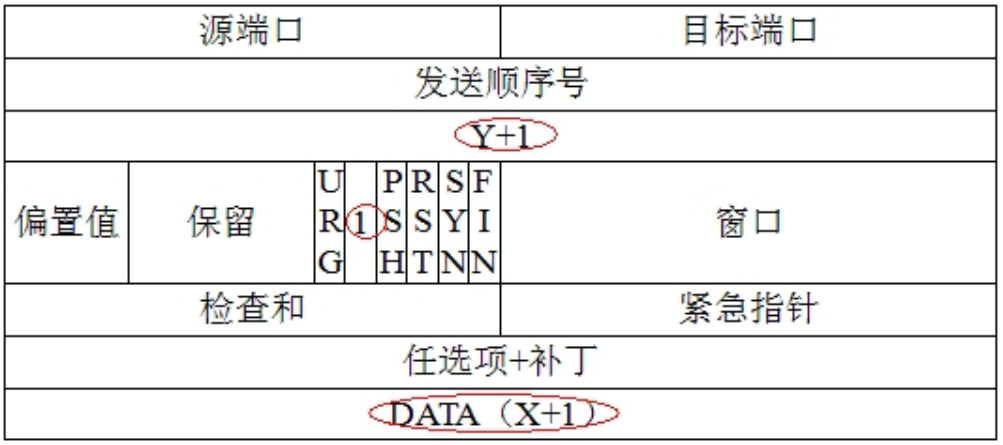
- 所谓三次握手(Three-way Handshake)，是指建立一个TCP连接时，需要客户端和服务器总共发送3个包。三次握手的目的是连接服务器指定端口，建立TCP连接,并同步连接双方的序列号和确认号并交换 TCP 窗口大小信息.在socket编程中，客户端执行connect()时。将触发三次握手。
- 第一次握手:
- 客户端发送一个TCP的SYN标志位置1的包指明客户打算连接的服务器的端口，以及初始序号X,保存在包头的序列号(Sequence Number)字段里。



- 第二次握手:
- 服务器发回确认包(ACK)应答。即SYN标志位和ACK标志位均为1同时，将确认序号(Acknowledgement Number)设置为客户的I S N加以1.即X+1。

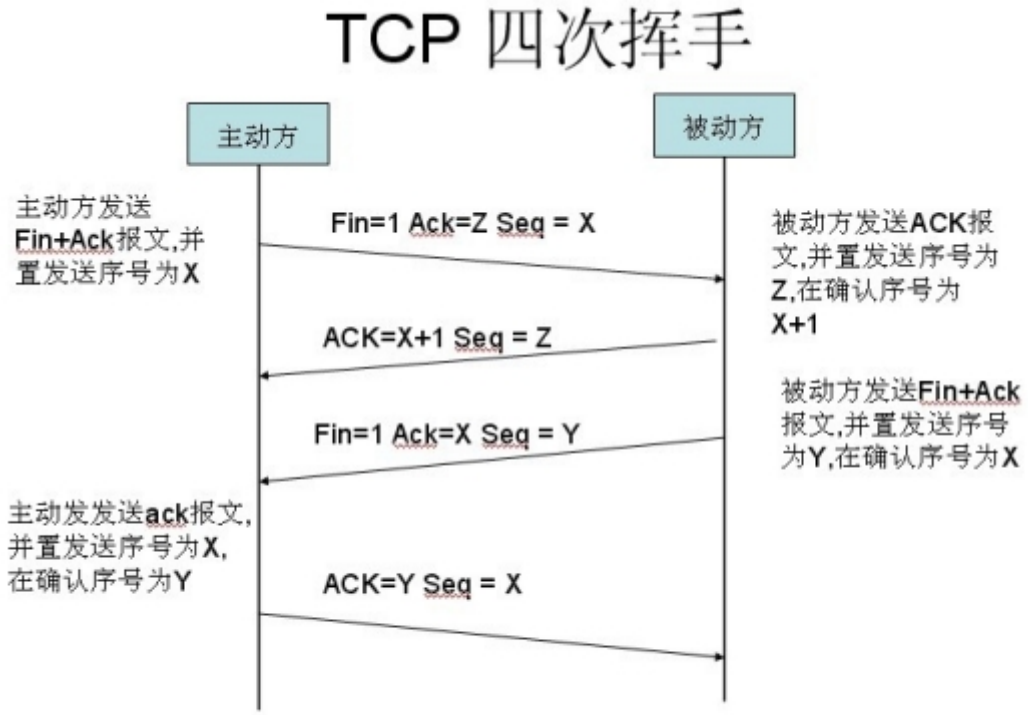


- 第三次握手:
- 客户端再次发送确认包(ACK) SYN标志位为0,ACK标志位为1.并且把服务器发来ACK的序号字段+1,放在确定字段中发送给对方.并且在数据段放写ISN的+1.



- TCP 四次挥手

- TCP的连接的拆除需要发送四个包，因此称为四次挥手(four-way handshake)。客户端或服务器均可主动发起挥手动作，在socket编程中，任何一方执行close()操作即可产生挥手操作。



- TCP三次握手会涉及TCP的状态转换图如下

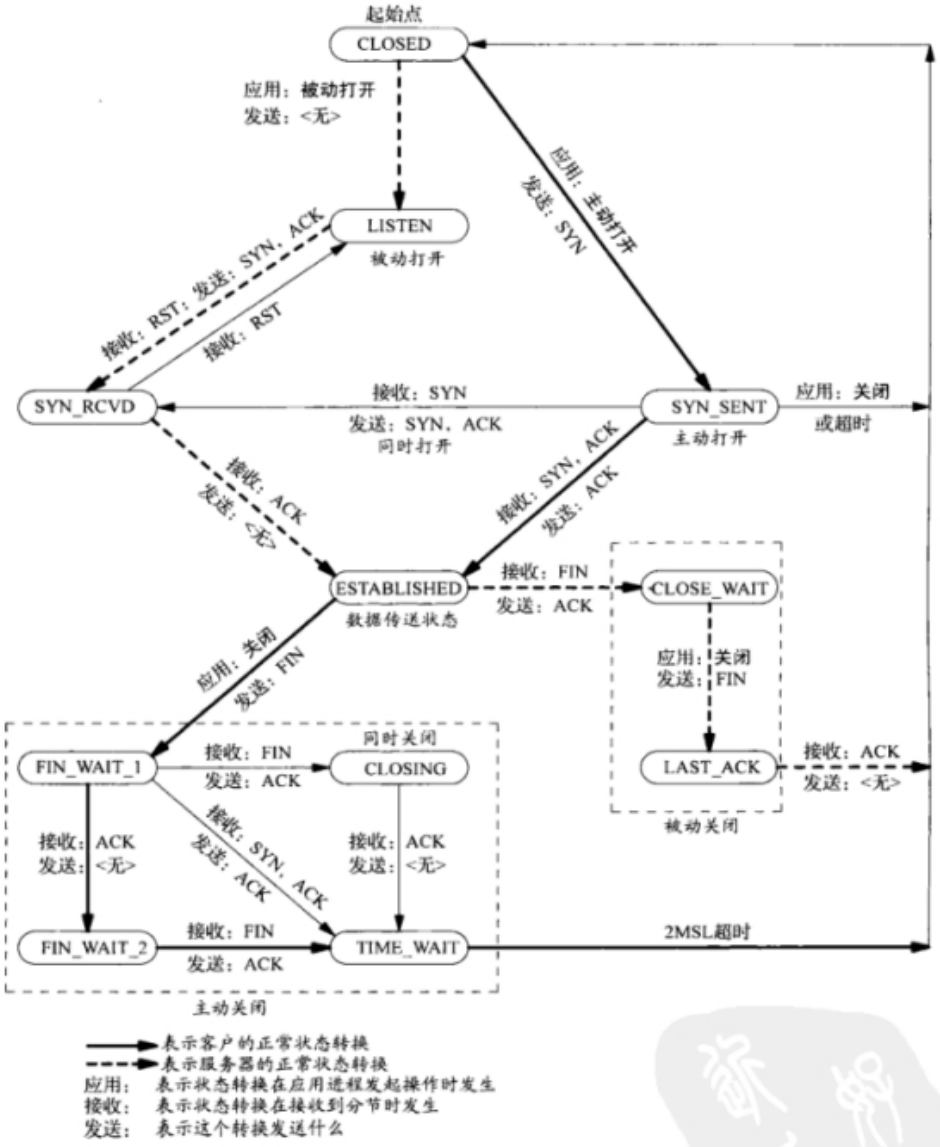


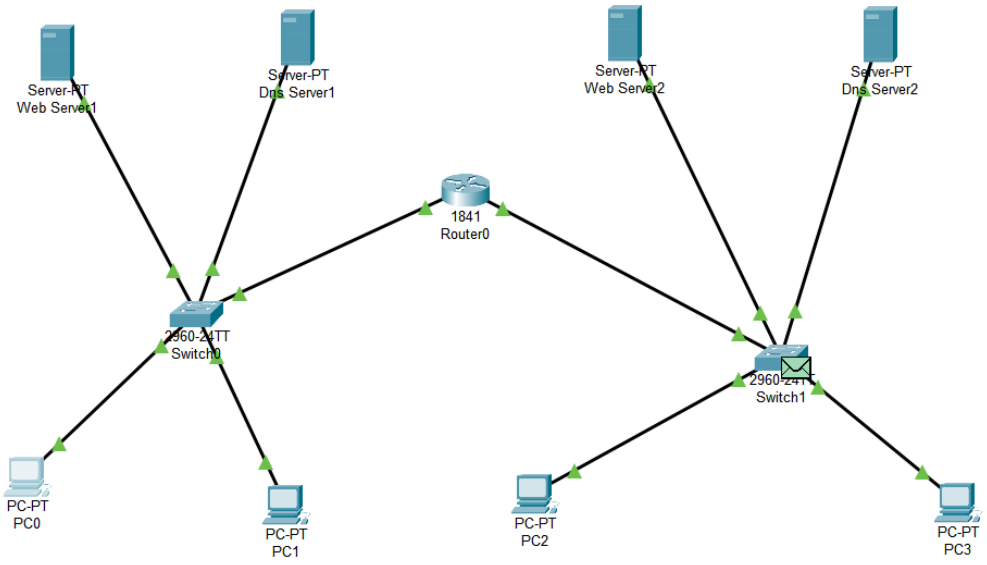
图2-4 TCP状态转换图

三.实验环境

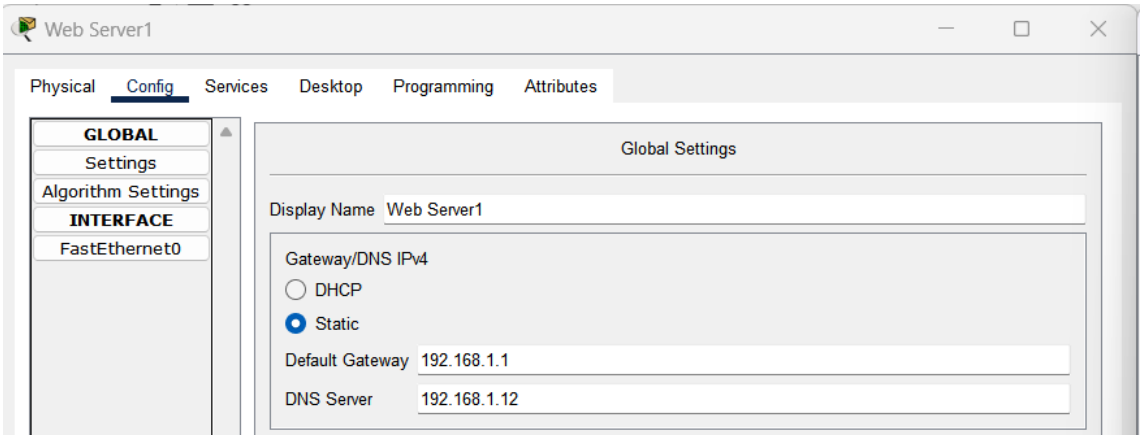
- 操作系统：Windows 11
- 网络环境：局域网
- 软件：Cisco Packet Tracer虚拟实验环境

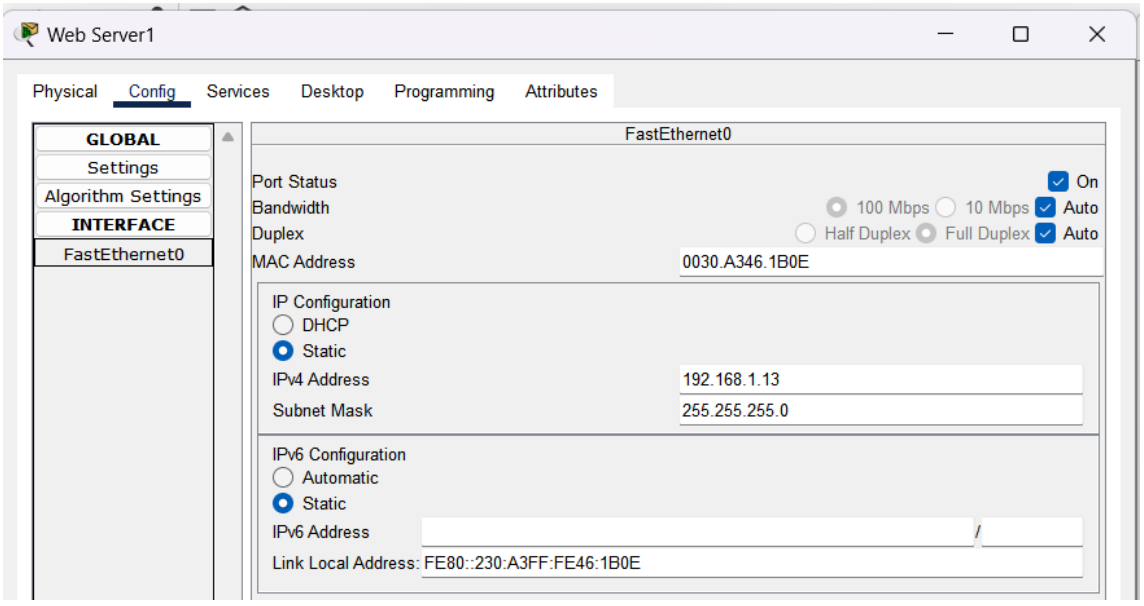
四.实验步骤

- 规划网络地址及拓扑图如下图所示

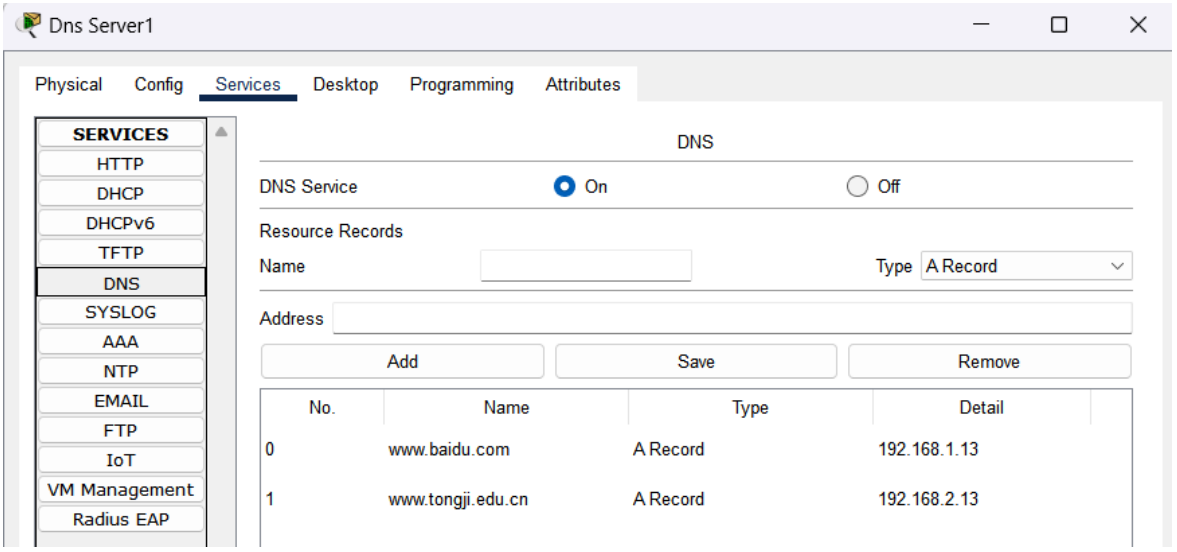


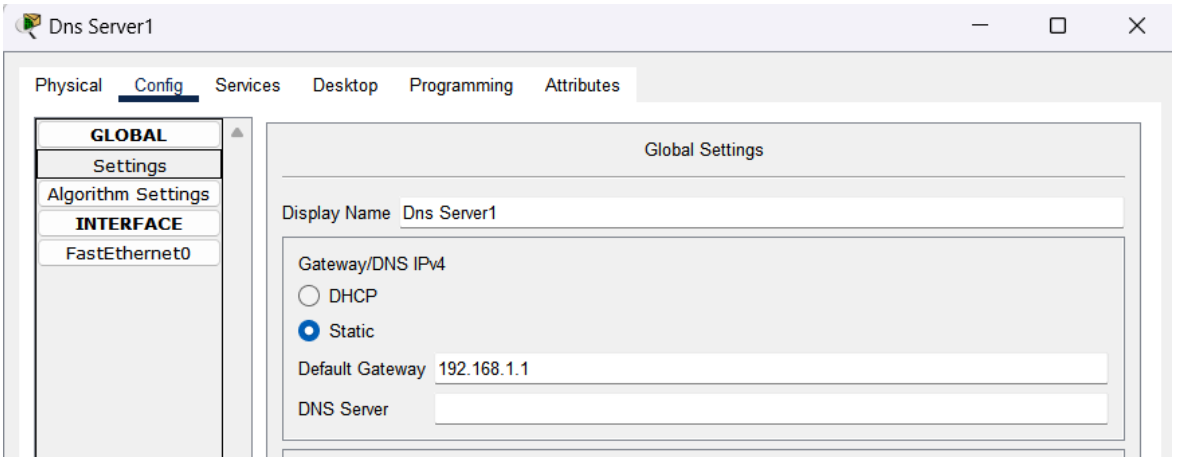
- 配置Web Server和DNS Server（以左侧一组为例，右侧同理）
  - 配置Web Server

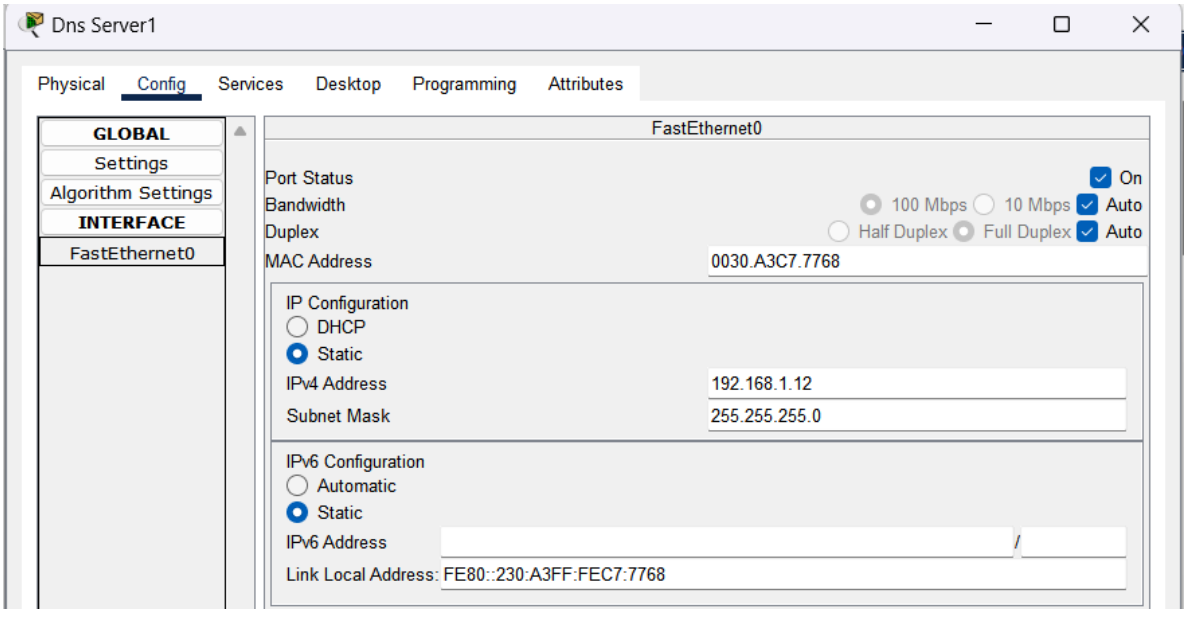




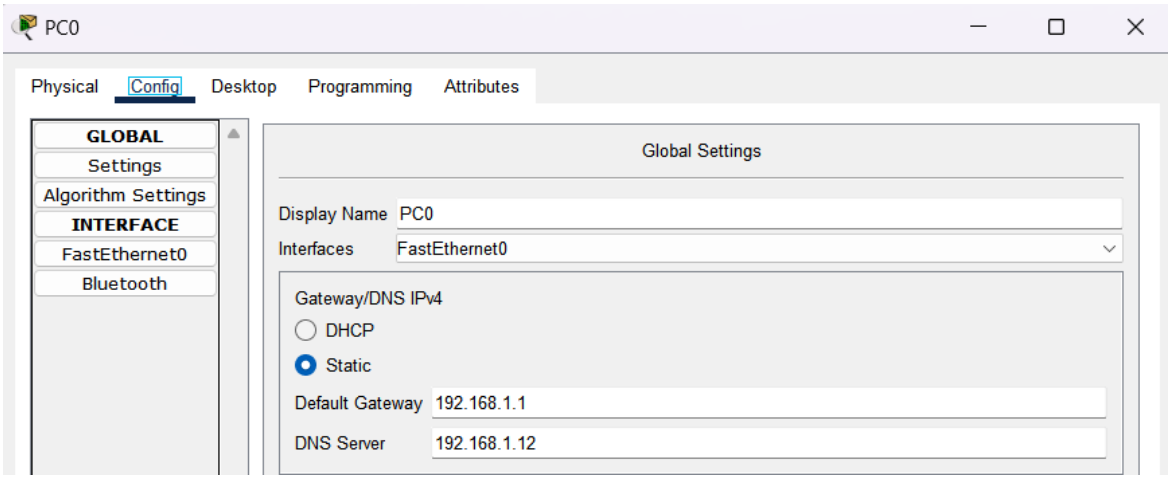
  - 配置DNS Server

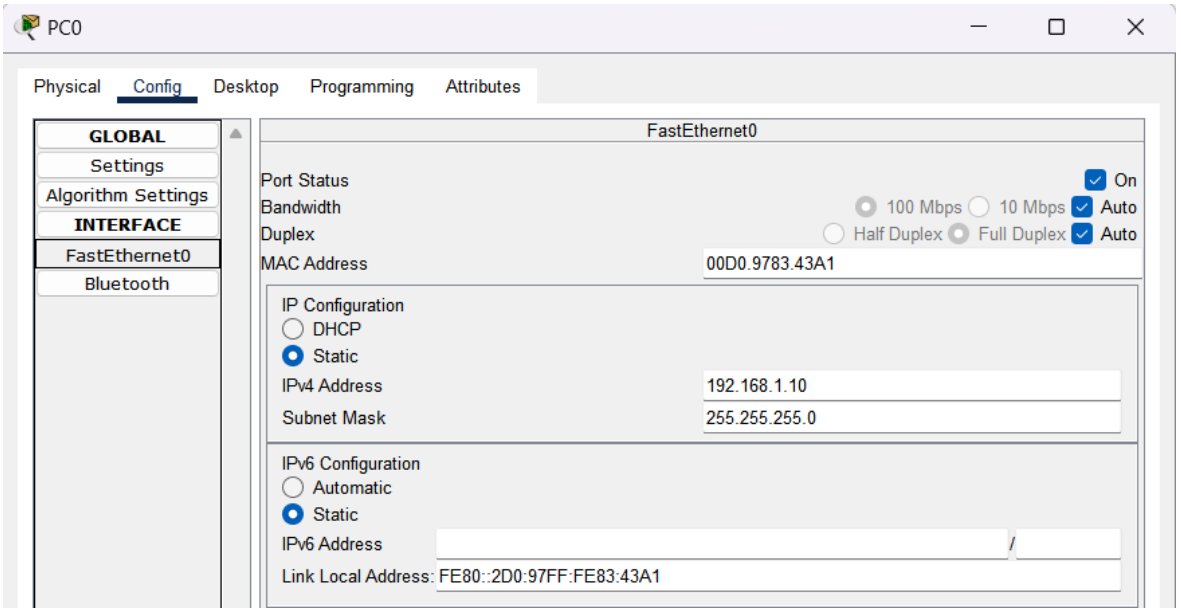


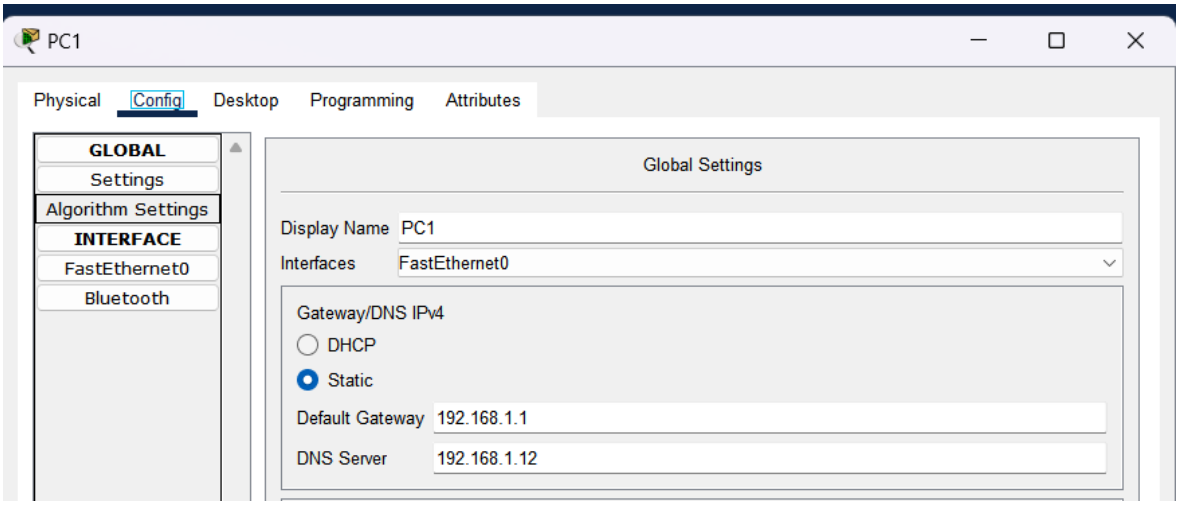


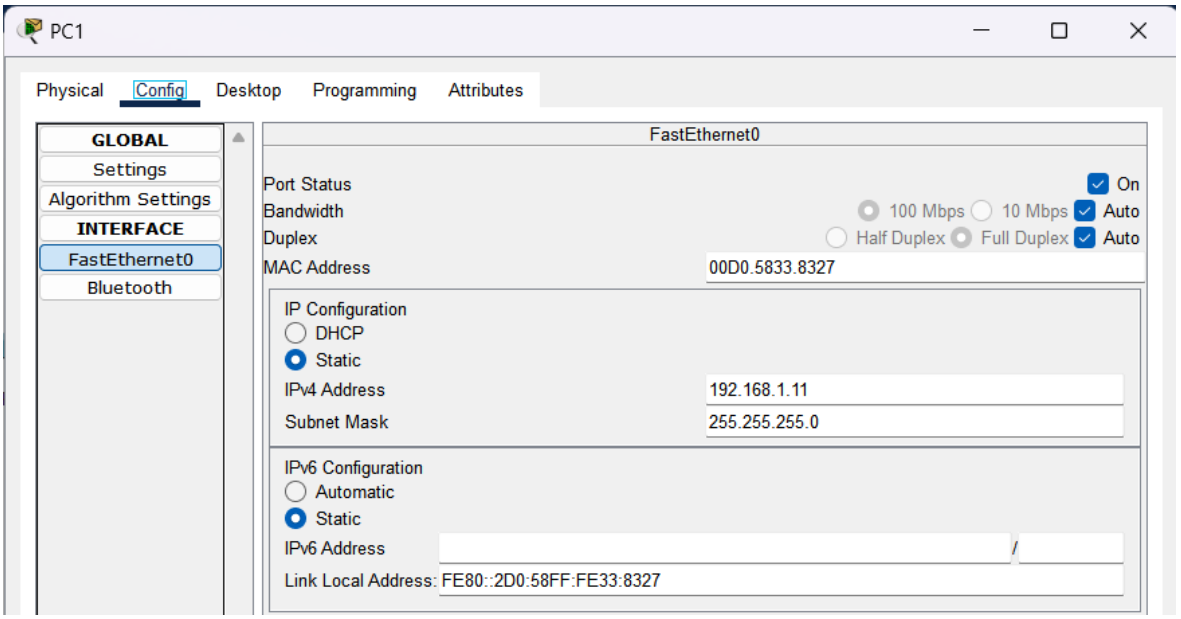


- 配置各个PC（以PC0和PC1为例）

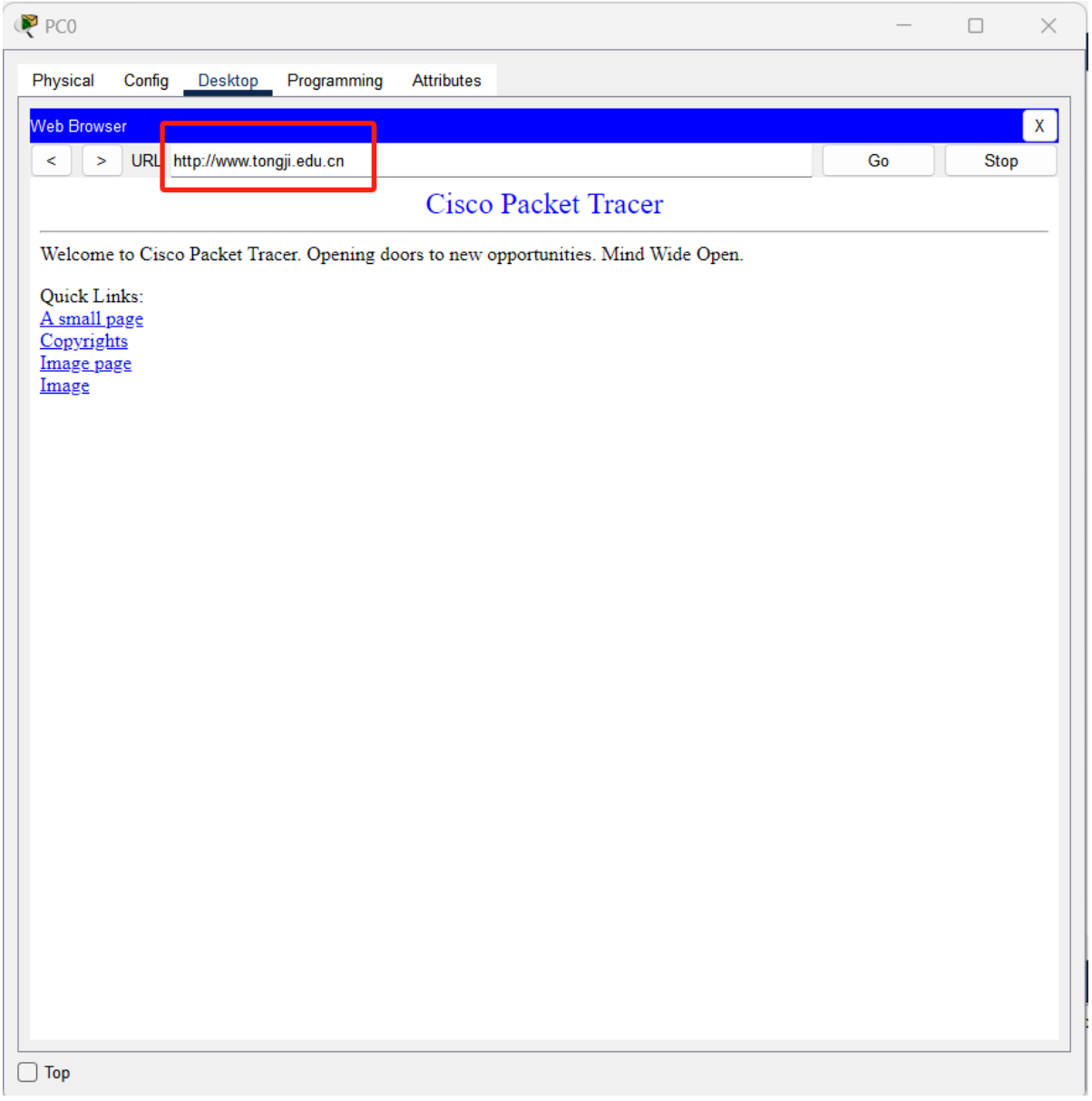

PC0
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
FastEthernet0
Bluetooth
Global Settings
Display Name PC0
Interfaces FastEthernet0
Gateway/DNS IPv4
DHCP
Static
Default Gateway 192.168.1.1
DNS Server 192.168.1.12


PC0
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
FastEthernet0
Bluetooth
FastEthernet0
Port Status On
Bandwidth 100 Mbps 10 Mbps Auto
Duplex Half Duplex Full Duplex Auto
MAC Address 00D0.9783.43A1
IP Configuration
DHCP
Static
IPv4 Address 192.168.1.10
Subnet Mask 255.255.255.0
IPv6 Configuration
Automatic
Static
IPv6 Address
Link Local Address: FE80::2D0:97FF:FE83:43A1

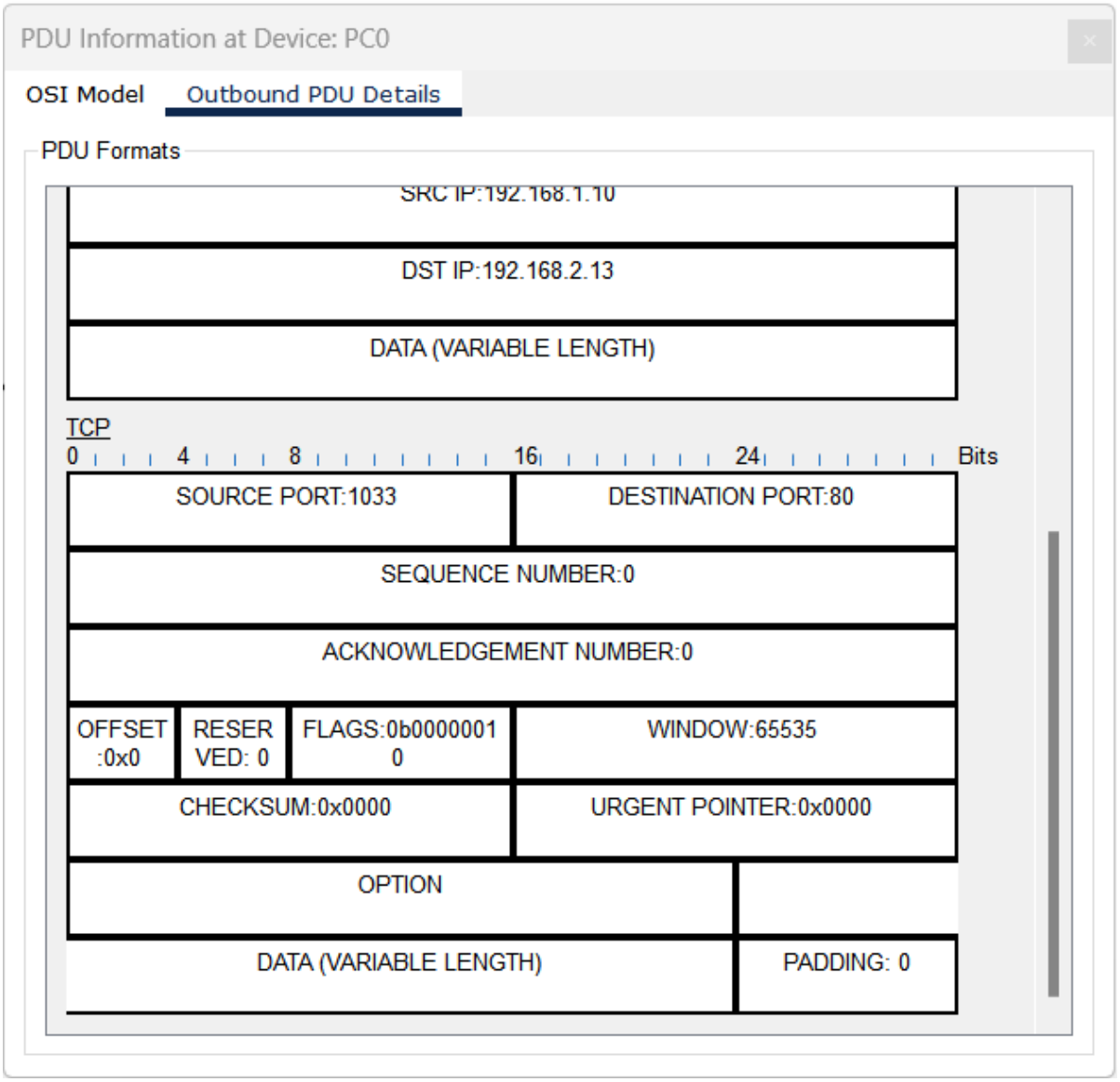

PC1
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
FastEthernet0
Bluetooth
Global Settings
Display Name PC1
Interfaces FastEthernet0
Gateway/DNS IPv4
DHCP
Static
Default Gateway 192.168.1.1
DNS Server 192.168.1.12



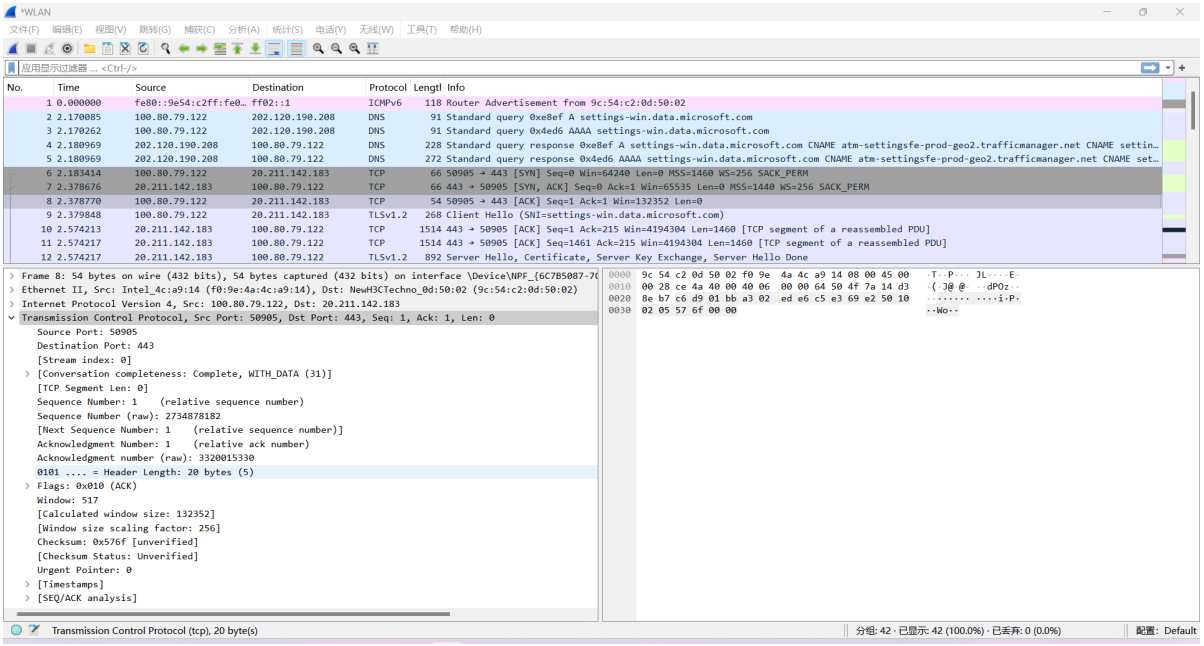
- 打开PC0浏览器，输入配置Web服务器的Web地址，产生TCP数据报文



- 观察并分析TCP数据报文

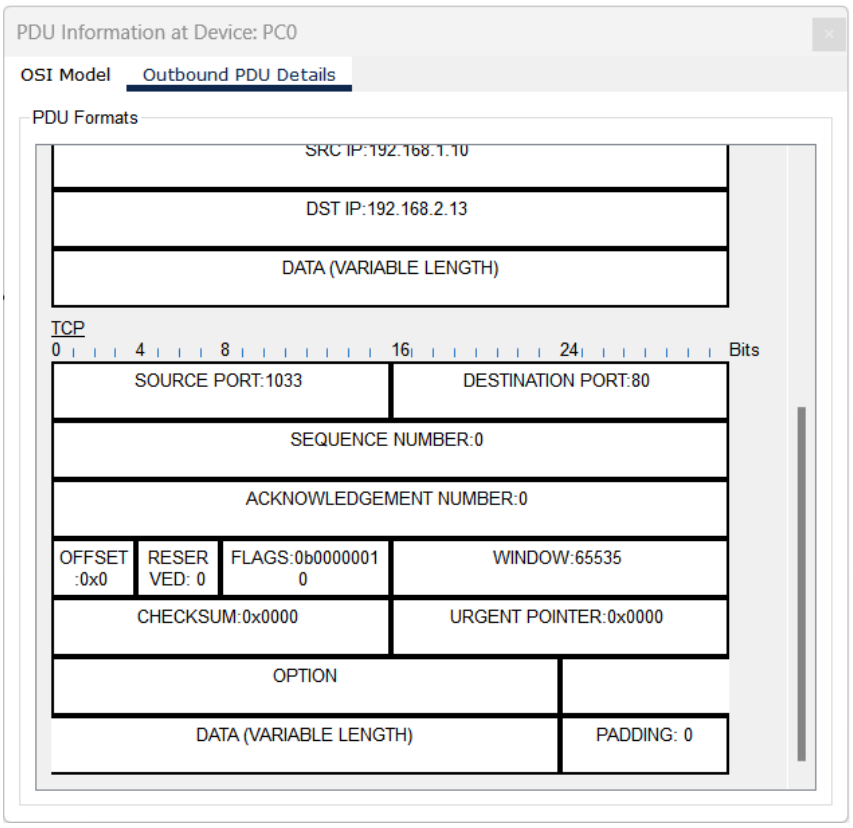


- WireShark抓取TCP报文并分析

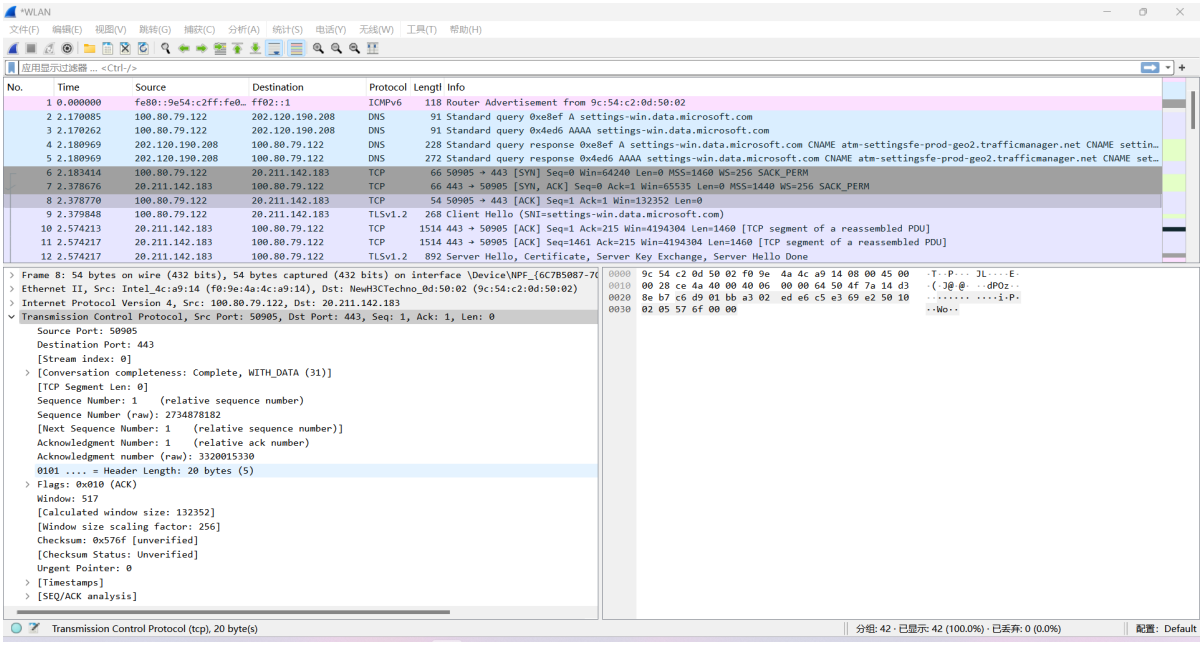


五、实验现象

- 以Packet Tracer中捕获到的一个TCP数据为例

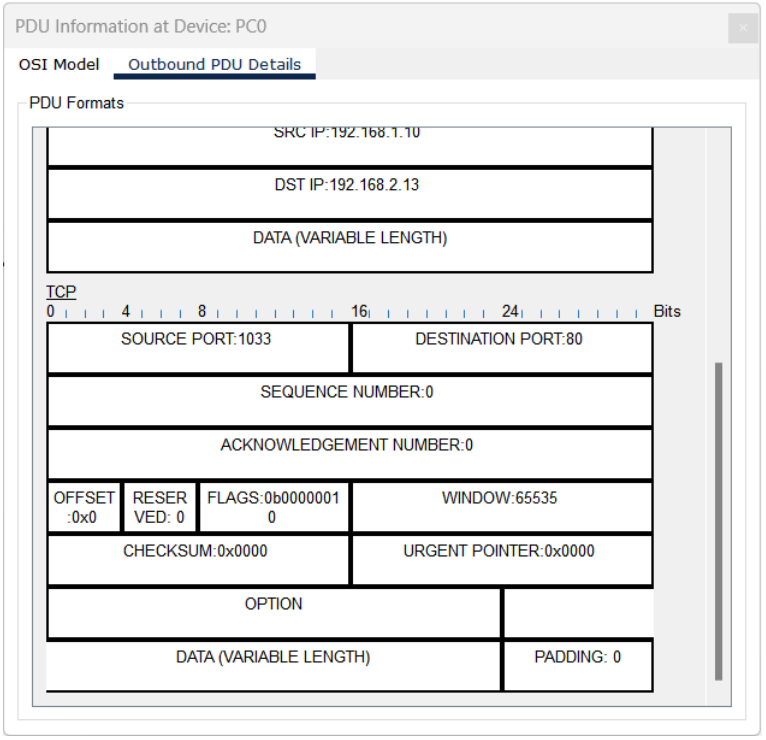


- 在该报文中：
  - 源端口号为1033
  - 目标端口为80
  - 初始序号为0
  - 接收顺序号为0
  - 偏置为0
  - SYN标志为1，表明客户端希望连接服务器端口(第一次握手)
  - 窗口为65535
  - 其余字段均为0
- 用WireShark抓取IP数据包，选择WLAN抓包

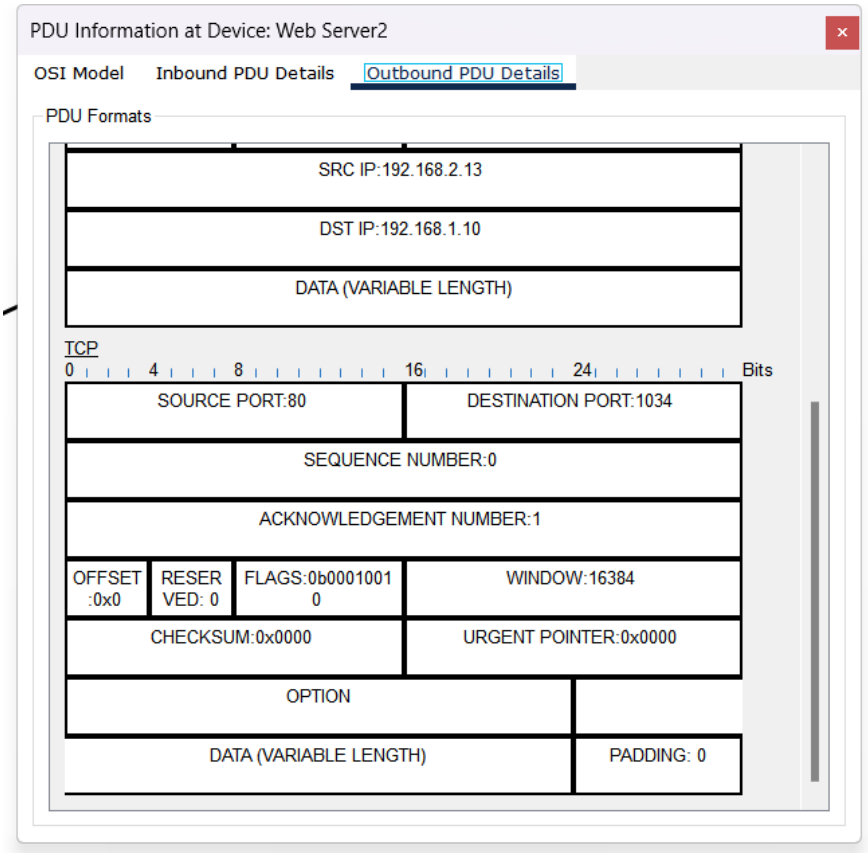


- 由上图我们可以得到：
  - 报文的关键信息如下：
    1. **源端口与目标端口**：
       - 源端口 (Source Port): 59095
       - 目标端口 (Destination Port): 443 (HTTPS服务)
    2. **序列号与确认号**：
       - 序列号 (Sequence Number): 2734878182
       - 确认号 (Acknowledgement Number): 3320015330
       - 报文段长度 (TCP Segment Len): 0，表示没有数据被传输。
    3. **标志位**：
       - ACK (Acknowledgment): 设置，表示这是一个确认报文。
       - 在标志位中，Acknowledgement为1表示接收确认序号有效，Push标志位为1表示指示接收方在接收到该报文段以后，应尽快将这个报文段交给应用程序，而不是在缓冲区排队
    4. **窗口大小**：
       - 窗口大小 (Window): 517
       - 计算窗口大小 (Calculated window size): 132352，说明已应用窗口缩放因子。
    5. **窗口缩放因子**：
       - 窗口缩放因子 (Window size scaling factor): 256，这是在TCP连接建立时协商的，用于支持较大的窗口大小，从而提高网络性能。
    6. **校验和**：
       - 校验和 (Checksum): 0x576f，显示为未验证状态。

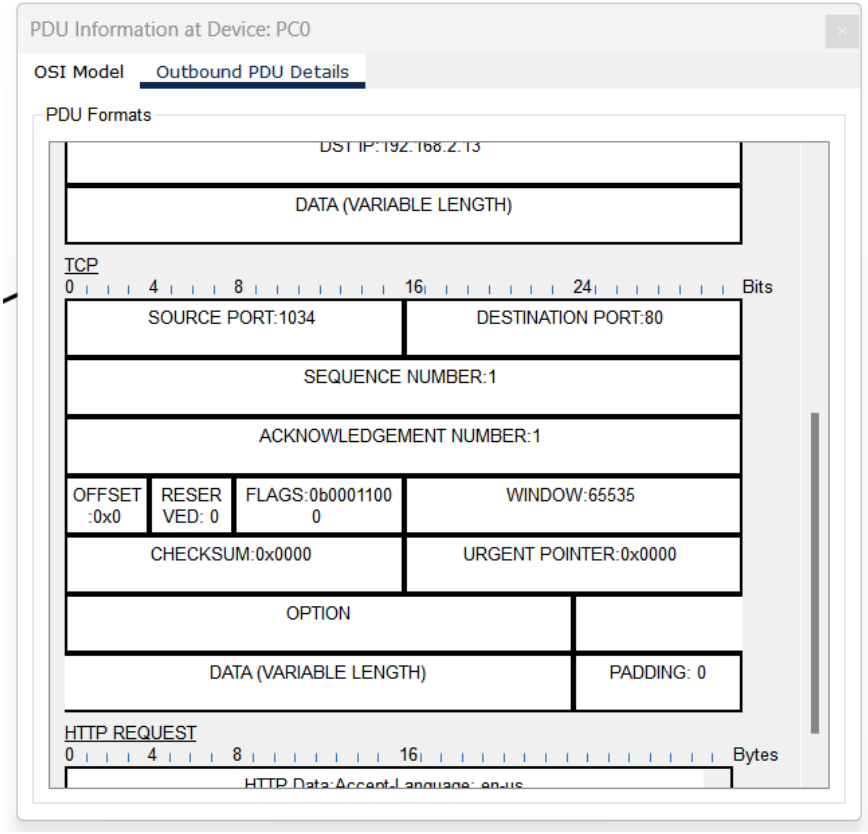
- TCP连接建立过程数据报文，这里使用PT软件中的TCP报文进行分析。
  - TCP报文段重要字段的内容
    - **序号**：表示发送的数据字节流，确保TCP传输有序，对每个字节编号
    - **确认序号**：发送方期待接收的下一序列号，接收成功后的数据字节序列号加 1。只有ACK=1时才有效。
    - **ACK**：确认序号的标志，ACK=1表示确认号有效，ACK=0表示报文不含确认序号信息
    - **SYN**：连接请求序号标志，用于建立连接，SYN=1表示请求连接
    - **FIN**：结束标志，用于释放连接，为1表示关闭本方数据流
  - 第一次握手，PC0发送了一个SYN位为1的包，表明PC0打算与服务器建立连接，发送序列号X为0



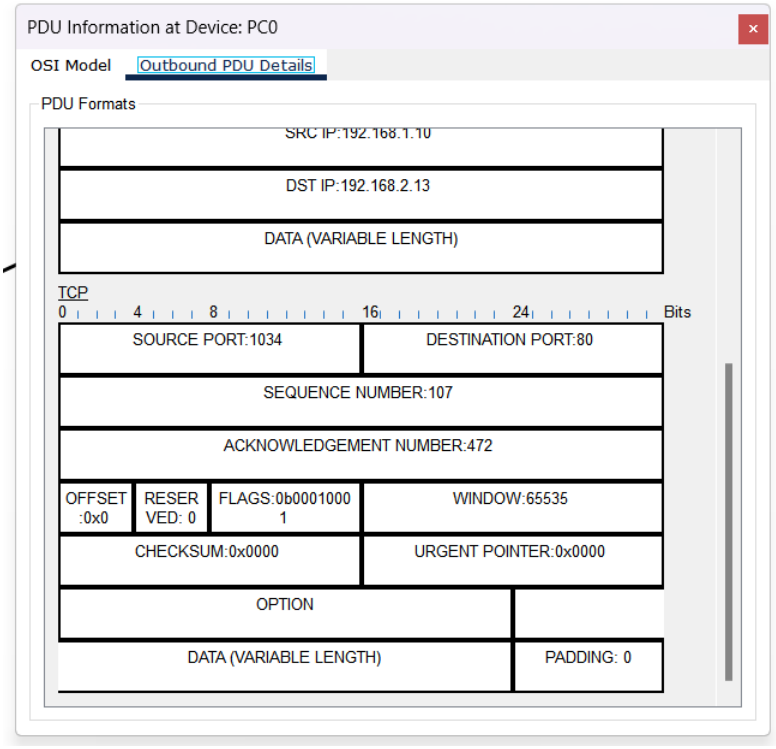
- 这是服务器收到P0的包后向PC0发送的TCP报文。在这个报文中，接受序列号为X+1，即为1，接受序列号Y为0.标识位中，SYN和ACK为都被置为1。



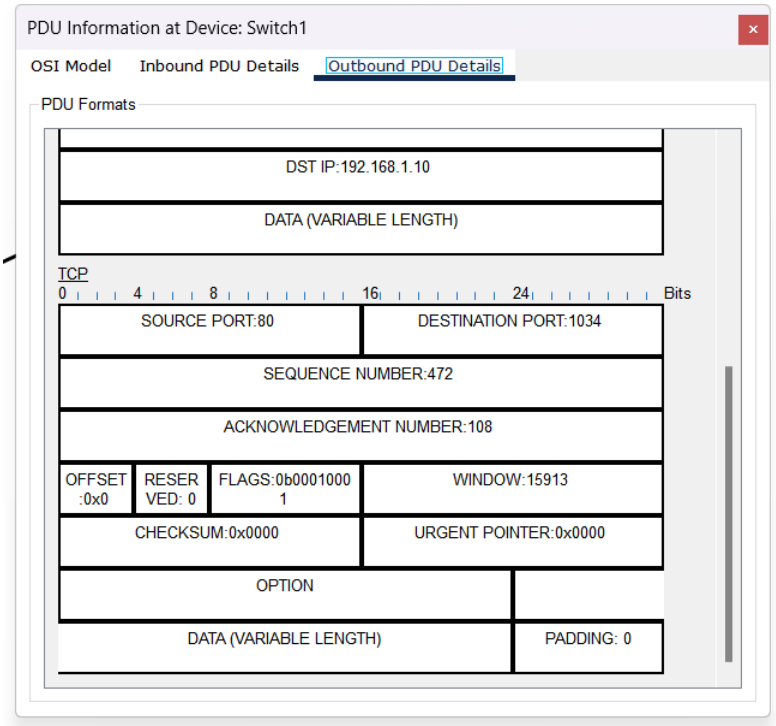
- 该包为第三次握手，由PC0向服务器发送的数据包，接收顺序号为Y+1即为1，标识位中只有ACK被置为1



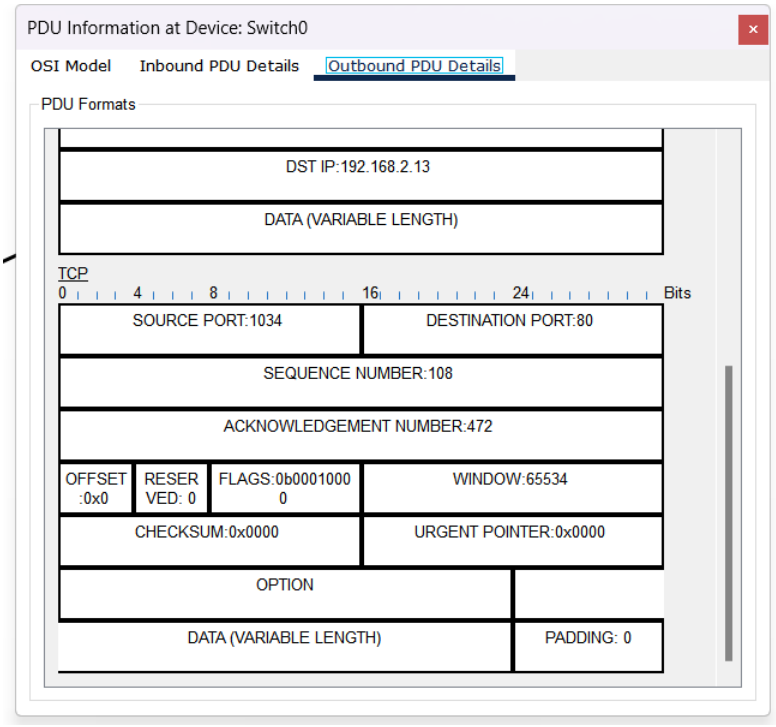
- TCP拆链过程数据报文
  - 第一次挥手，PC0为主动方，发送Fin+Ack报文，将标识位中这两位置为1，发送序号X为107，接受序号Z为472.



  - 第二次和第三次挥手，这个地方十分重要。这里我一开始很奇怪，不是四次挥手吗，怎么服务器只发了一个包。后来上网查询得知，这里把四次挥手简化为了三次挥手，第二次和第三次挥手合成了一个包，接受序列号为X+1即108，发送序列号为Y即472，并且Fin和Ack位都置为了1。



- 第四次挥手，主动方即PC0发送ACK报文，置发送序列号为X=108，接收序列号为Y=472。完成拆链过程。



六、实验结论

- 根据TCP段分析实验的内容和您提供的TCP报文段的分析图，可以归纳以下实验结论：

1. **TCP协议的有效性**：
   - TCP通过序列号和确认号确保了数据的有序性和可靠性传输。在提供的报文段中，确认号的设置表明接收端已成功接收前序数据，且通过ACK标志进行了确认。
2. **窗口大小与流量控制**：
   - 窗口大小和窗口缩放因子的应用显示了TCP协议在适应不同网络条件下如何动态管理流量。窗口大小的调整对于维持网络的高效运作和防止拥塞非常关键。
3. **连接的安全性**：

    - 目标端口443表示TCP连接被用于HTTPS，这强调了TCP协议在安全传输（如加密网站数据）中的重要作用。
4. **网络分析工具的作用**：
    - 使用如Wireshark这样的网络分析工具可以有效地抓取和分析网络通信中的TCP报文，提供了深入了解网络协议操作和性能状况的可能。
5. **协议栈的自动化**：
    - TCP协议的操作大多是自动化的，如ACK的发送通常由协议栈自动处理，而不需人为干预，这有助于降低网络通信的复杂性并提高效率。

- 综上所述，实验验证了TCP协议在确保网络数据可靠传输、动态调整流量、以及提供安全通信方面的关键角色。同时，实验也展示了网络分析工具在理解和优化这些机制中的重要性。