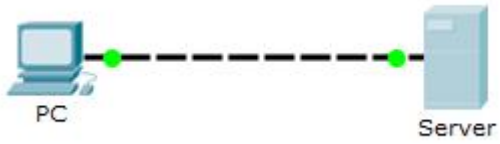
实验六 运输层实验

0. 实验拓扑及配置

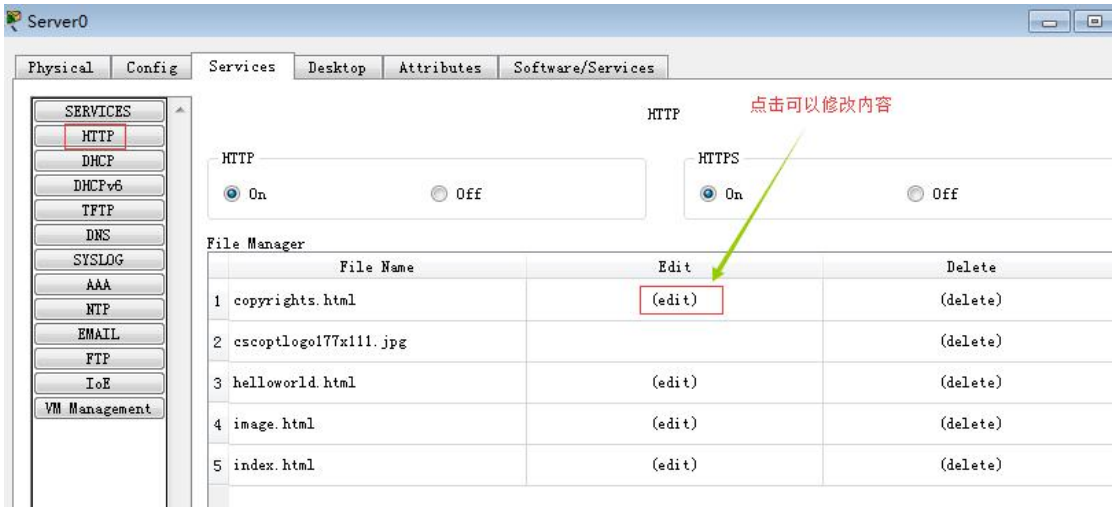


PC 配置：IP:192.168.1.2; Gateway: 192.168.1.254; DNS Server:192.168.1.1

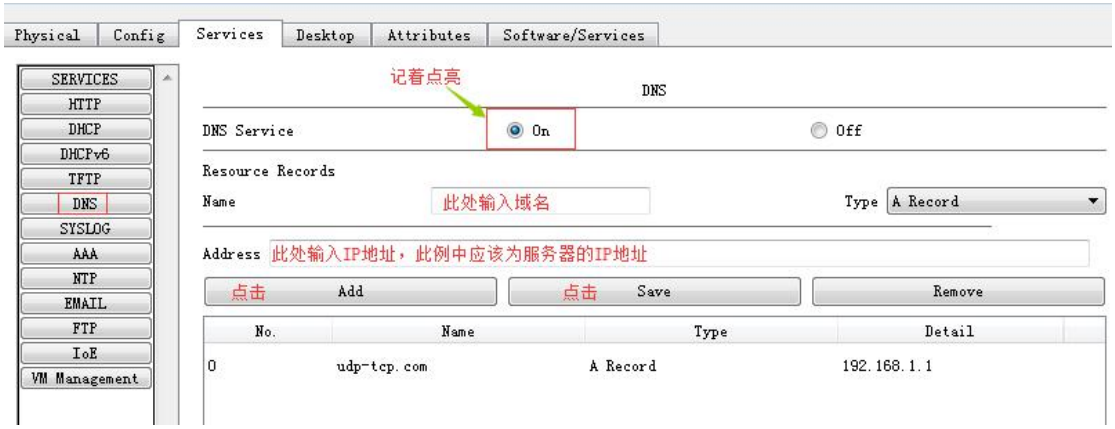
Server 配置：IP:192.168.1.2; Gateway: 192.168.1.254

Server 端还要配置服务：

（1）HTTP



（2）DNS



1. 运输层端口观察实验

端口简介：

从运输层的角度看，通信的真正端点并不是主机而是主机中的进程。运输层解决的就是进程之间的通信问题，即所谓的“端”到“端”的通信。在一个主机中经常有多个应用进程同时分别和另一个主机中的多个应用进程通信，因此，给应用层的每个应用进程赋予一个非常明确的标志是至关重要的。

一台主机可以提供许多服务，这些服务完全可以通过一个 IP 地址来实现，换句话说，IP 地址与网络服务的关系是一对多的关系。显然，主机不能只靠 IP 地址来区分不同的网络服务，而是要通过“IP 地址+端口号”区分的。端口(port)是运输层的应用程序接口，应用层的各个进程都需要通过相应的端口才能与运输实体进行交互。端口通过端口号来标记，TCP/IP 的运输层用一个 16 位端口号来标志一个端口。

UDP 和 TCP 的端口有着本质上的不同，但它们使用相同的端口号表示法。端口号通常分为熟知端口和动态端口，TCP 与 UDP 的 PDU 信息中均包含服务器的熟知端口号，也包含客户端生成的动态端口号。

实验目的：

（1）理解运输层的端口与应用层的进程之间的关系；
（2）了解端口号的划分和分配。

实验步骤：

任务一：通过捕获的 DNS 事件查看并分析 UDP 的端口号

（1）步骤 1：捕获 DNS 事件

（2）步骤 2：查看并分析 UDP 用户数据报中的端口号

本步骤需注意观察并分析以下几项内容：
① DNS 请求包和应答包的源、目的端口号是否发生变化；
② 判断 PC 和 Server 的客户端/服务器角色，分析判断依据。

（3）步骤 3：分析端口号的变化规律

重新回到 PC 机的浏览器窗口单击 Go（转到）按钮再次请求相同的网页，从新捕获的 DNS 事件中观察 DNS 客户端与 DNS 服务器端的端口号是否发生变化。如果没有，分析其原因；如果有，分析其变化的规律。

特别注意：分析完成后不能单击 Reset Simulation（重置模拟）按钮清空原有的事件，同时也不能关闭 PC 的配置窗口。

任务二：通过捕获的 HTTP 事件查看并分析 TCP 的端口号

（1）步骤 1：捕获 HTTP 事件

（2）步骤 2：查看并分析 TCP 报文中的端口号

本步骤需注意观察并分析以下几项内容：
① TCP 报文中的源端口和目的端口值；
② 确定 PC 和 Server 的客户端/服务器角色。
完成后单击 Reset Simulation（重置模拟）按钮，将原有的事件清空。

任务三：分析运输层端口号

（1）步骤 1：分析运输层端口号与应用进程之间的关系

对比任务一中 DNS 服务器端的端口号与任务二中服务器端的端口号是否相同，并分析其原因。

（2）步骤 2：分析运输层动态端口号的分配规律

重新捕获 HTTP 事件以分析 TCP 协议的端口号变化情况。具体操作方法参考任务二中的步骤 2。

该步骤重点观察 HTTP 客户端的端口号，并与任务二中观察到的 HTTP 客户端的端口号进行对比，分析归纳动态端口号的分配规律。

完成后单击 Reset Simulation（重置模拟）按钮，将原有的事件清空。

思考：

（1）运输层如何区分应用层的不同进程？

（2）若使用 Reset Simulation（重置模拟）按钮后再重新进行捕获，端口号如何变化？新的值与重置前有关吗？

2. UDP 协议与 TCP 协议的对比分析

预备知识：

传输控制协议 TCP(Transmission Control Protocol)与用户数据报协议 UDP(User Datagram Protocol)是 TCP/IP 的运输层中的两个主要的协议。

UDP 是一个简单的面向数据报的运输层协议。它有如下几个主要特点：无连接；尽最大努力交付，不提供可靠性；面向报文；支持一对一、一对多、多对一、多对多的交互通信，组播及广播功能强大。它支持的应用层协议主要有：DNS、NFS、SNMP、TFTP 等。

TCP 提供的是面向连接、端到端的、可靠的字节流服务。它的主要特点有：面向连接；提供可靠交付的服务；基于字节流的，而非消息流；不支持多播（multicast）和广播（broadcast）。TCP 支持的应用协议主要有：HTTP、Telnet、FTP、SMTP 等。

实验目的：

（1）熟悉 UDP 与 TCP 协议的主要特点及支持的应用协议；

（2）理解 UDP 的无连接通信与 TCP 的面向连接通信；

（3）熟悉 TCP 报文段和 UDP 报文的数据封装格式。

实验步骤：

任务一：观察 UDP 无连接的工作模式

（1）步骤 1：捕获 UDP 事件

（2）步骤 2：分析 UDP 无连接的工作过程

本步骤仅查看第 4 层中 UDP 报文段的内容。注意观察并分析以下几项内容：

① 运输层的 UDP 发送 DNS 的请求之前是否有先建立连接；

② 记录 UDP 的用户数据报首部中的 LENGTH 字段的值，分析该报文的首部及数据部分的长度。

分析完成后单击 Reset Simulation（重置模拟）按钮，将原有的事件全部清空。

任务二：观察 TCP 面向连接的工作模式

（1）步骤 1：捕获 TCP 事件

（2）步骤 2：分析 TCP 面向连接的工作过程

本步骤仅查看第 4 层中 TCP 报文段的内容。注意观察并分析以下几项内容：

① 在捕获到的第一个 HTTP 事件之前及最后一个 HTTP 事件之后是否有 TCP 事件；

② 第一个以及最后一个 HTTP 事件对应的 TCP 报文中的 sequence number（序号）、ACK number（确认号）的值以及它们与 data length（数据长度）的关系；

③ 并查看 TCP 报文首部中固定部分的长度。

分析完成后单击 Reset Simulation（重置模拟）按钮，将原有的事件全部清空。

思考：

（1）TCP 报文首部中的序号和确认号有什么作用？

（2）无连接的 UDP 和面向连接的 TCP 各有什么优缺点？

3. TCP 的连接管理

TCP 连接管理简介：

TCP 是面向连接的协议，运输连接有三个阶段：连接建立、数据通信、连接释放。连接管理的目标就是使连接的建立和释放都能正常进行。TCP 连接的建立采用客户服务器的方式，主动发起连接建立请求的应用进程叫做客户（client），而被动等待连接建立的应用进程叫做服务器（server）。

TCP 协议通过三次握手（three-way handshake）完成连接的建立。完成三次握手，客户端与服务器开始传送数据。连接可以由任一方或双方发起，一旦连接建立，数据就可以双向对等地流动，而没有所谓的主从关系。三次握手协议可以完成两个重要功能：它确保连接双方做好传输准备，并使双方统一了初始顺序号，两台机器仅仅使用三个握手报文就能协商好各自的数据流的顺序号。

当一对 TCP 连接的双方数据通信完毕，任何一方都可以发起连接释放请求。TCP 采用和三次握手类似的方法即四次握手（或称为两个二次握手）的方式释放连接。释放连接的操作可以看成是由两个方向上分别释放连接的操作构成。

实验目的：

（1）熟悉 TCP 通信的三个阶段；

（2）理解 TCP 连接建立过程和 TCP 连接释放过程。

实验步骤：

任务一：捕获 TCP 事件

任务二：分析 TCP 连接建立阶段的三次握手

注意观察任务一中捕获到的 TCP 事件，完成以下几项内容：

(1) 分析 TCP 连接建立阶段的三次握手的过程；
(2) 查看 TCP 报文段首部中的各项字段的值，包括 SYN 字段、ACK 字段、PSH 字段、FIN 字段、sequence number（序号）字段、ACK number（确认号）字段、窗口大小、选项字段 MSS（最大报文段长度）、报文段长度等；
(3) 分析三次握手过程中 TCP 连接状态的变迁。

任务三：分析 TCP 连接释放阶段的四次握手

继续观察任务一中捕获到的 TCP 事件，完成以下几项内容：

(1) 分析 TCP 连接释放阶段的四次握手的过程；
(2) 查看 TCP 报文段首部中的各项字段的值，包括 SYN 字段、ACK 字段、PSH 字段、FIN 字段、sequence number（序号）字段、ACK number（确认号）字段、窗口大小、选项字段 MSS（最大报文段长度）、报文段长度等；
(3) 分析四次握手过程中 TCP 连接状态的变迁。

思考：

(1) 连接建立阶段的第一次握手是否需要消耗一个序号？其 SYN 报文段是否携带数据？为什么？第二次握手呢？
(2) 本实验中连接释放过程的第二、三次握手是同时进行的还是分开进行的？这两次握手何时需要分开进行？
(3) 本实验中连接释放阶段的第四次握手，PC 向 Server 发送最后一个 TCP 确认报文段后，为什么不是直接进入 CLOSED（已关闭）连接状态，而是进入 CLOSING（正在关闭）连接状态？
(4) 本实验中 TCP 连接建立后的数据通信阶段，PC 向 Server 发送的了多少数据？Server 向 PC 发送的数据呢？