

计算机网络第二次作业

10211900416 郭夏辉

作业内容

第二章习题 P3 P18 P22 P23 P24 P25 P27

P3

P3. 考虑一个要获取给定 URL 的 Web 文档的 HTTP 客户。该 HTTP 服务器的 IP 地址开始时并不知道。在这种情况下，除了 HTTP 外，还需要什么运输层和应用层协议？

应用层：DNS HTTP

运输层：TCP(HTTP 需要) UDP(DNS需要)

P18

P18. 如题：

- 什么是 whois 数据库？
- 使用因特网上的各种 whois 数据库，获得两台 DNS 服务器的名字。指出你使用的是哪个 whois 数据库。
- 你本地机器上使用 nslookup 向 3 台 DNS 服务器发送 DNS 查询：你的本地 DNS 服务器和两台你在 (b) 中发现的 DNS 服务器。尝试对类型 A、NS 和 MX 报告进行查询。总结你的发现。
- 使用 nslookup 找出一台具有多个 IP 地址的 Web 服务器。你所在的机构（学校或公司）的 Web 服务器具有多个 IP 地址吗？
- 使用 ARIN whois 数据库，确定你所在大学使用的 IP 地址范围。
- 描述一个攻击者在发动攻击前，能够怎样利用 whois 数据库和 nslookup 工具来执行对一个机构的侦察。
- 讨论为什么 whois 数据库应当为公众所用。

a.whois数据库是一个记录了域名的详细信息的数据库。比如：域名所有人，域名注册商，域名注册日期，域名过期日期。

b.我使用的whois数据库是[站长之家](#)

查询的是github.com的DNS服务器信息：

DNS	dns1.p08.nsone.net
	dns2.p08.nsone.net
	dns3.p08.nsone.net
	dns4.p08.nsone.net

c.nslookup如果不设置参数默认为A类型

```
C:\Users\tom>nslookup
默认服务器:  moon.ecnu.edu.cn
Address:  202.120.80.2
> _
```

可以看到本地的DNS服务器是 moon.ecnu.edu.cn

本地DNS服务器:

类型A如下图所示:

```
> moon.ecnu.edu.cn
服务器:  moon.ecnu.edu.cn
Address:  202.120.80.2

非权威应答:
名称:      moon.ecnu.edu.cn
Addresses:  2001:da8:8005:80::2
            202.120.80.2
>
```

类型NS如下图所示:

```
> set type=ns
> moon.ecnu.edu.cn
服务器: moon.ecnu.edu.cn
Address: 202.120.80.2

ecnu.edu.cn
    primary name server = yingtao.ecnu.edu.cn
    responsible mail addr = master.ecnu.edu.cn
    serial = 2017092601
    refresh = 14400 (4 hours)
    retry = 7200 (2 hours)
    expire = 604800 (7 days)
    default TTL = 86400 (1 day)
```

类型MX如下图所示:

```
> set type=mx
> moon.ecnu.edu.cn
服务器: moon.ecnu.edu.cn
Address: 202.120.80.2

ecnu.edu.cn
    primary name server = yingtao.ecnu.edu.cn
    responsible mail addr = master.ecnu.edu.cn
    serial = 2017092601
    refresh = 14400 (4 hours)
    retry = 7200 (2 hours)
    expire = 604800 (7 days)
    default TTL = 86400 (1 day)
```

(b)中第一个DNS服务器:dns1.p08.nsone.net

类型A如下图所示:

```
> dns1.p08.nsone.net
服务器: moon.ecnu.edu.cn
Address: 202.120.80.2

非权威应答:
名称: dns1.p08.nsone.net
Address: 198.51.44.8
```

类型NS如下图所示:

```
> set type=ns
> dns1.p08.nsone.net
服务器: moon.ecnu.edu.cn
Address: 202.120.80.2

nsone.net
      primary name server = dns1.p01.nsone.net
      responsible mail addr = hostmaster.nsone.net
      serial = 1666130405
      refresh = 43200 (12 hours)
      retry = 7200 (2 hours)
      expire = 1209600 (14 days)
      default TTL = 3600 (1 hour)
```

类型MX如下图所示:

```
> set type=mx
> dns1.p08.nsone.net
服务器: moon.ecnu.edu.cn
Address: 202.120.80.2

nsone.net
      primary name server = dns1.p01.nsone.net
      responsible mail addr = hostmaster.nsone.net
      serial = 1666130405
      refresh = 43200 (12 hours)
      retry = 7200 (2 hours)
      expire = 1209600 (14 days)
      default TTL = 3600 (1 hour)
```

(b)中第二个DNS服务器:dns2.p08.nsone.net

类型A如下图所示:

```
> dns2.p08.nsone.net
服务器:  moon.ecnu.edu.cn
Address:  202.120.80.2

非权威应答:
名称:     dns2.p08.nsone.net
Address:  198.51.45.8
```

类型NS如下图所示:

```
> set type=ns
> dns2.p08.nsone.net
服务器:  moon.ecnu.edu.cn
Address:  202.120.80.2

nsone.net
      primary name server = dns1.p01.nsone.net
      responsible mail addr = hostmaster.nsone.net
      serial    = 1666130405
      refresh   = 43200 (12 hours)
      retry     = 7200 (2 hours)
      expire    = 1209600 (14 days)
      default TTL = 3600 (1 hour)
```

类型MX如下图所示:

```
> set type=mx
> dns2.p08.nsone.net
服务器:  moon.ecnu.edu.cn
Address:  202.120.80.2

nsone.net
      primary name server = dns1.p01.nsone.net
      responsible mail addr = hostmaster.nsone.net
      serial    = 1666130405
      refresh   = 43200 (12 hours)
      retry     = 7200 (2 hours)
      expire    = 1209600 (14 days)
      default TTL = 3600 (1 hour)
```

要理解结果的差异，首先来回顾一下三种类型:

类型A:A记录是用于名称解析的重要记录，提供标准的主机名到IP的地址映射。可以看到报告中包含主机名和IP地址。

类型NS:NS记录用来指定该域名由哪个DNS服务器来进行解析。注册域名时，总有默认的DNS服务器，每个注册的域名都是由一个DNS域名服务器来进行解析的。简单的说，NS记录返回域中主机IP地址的权威DNS服务器的主机名。可以看到报告中包含域的地址和包含域中IP地址的权威DNS主机名。

类型MX:返回别名为Name对应的邮件服务器的规范主机名。可以看到报告中包含邮件服务器及其主机名。

```
> baidu.com
服务器:  moon.ecnu.edu.cn
Address:  202.120.80.2

d. 非权威应答:
名称:     baidu.com
Addresses: 39.156.66.10
           110.242.68.66
```

我所在的学校之Web服务器具有多个IP地址

```
C:\Users\tom>nslookup www.ecnu.edu.cn
服务器:  moon.ecnu.edu.cn
Address:  202.120.80.2

非权威应答:
名称:     www.ecnu.edu.cn
Addresses: 2001:da8:8005:a492::60
           202.120.92.60
```

e.

"202.120.92.60"

Network: 202.112.0.0 - 202.121.255.255	
Source Registry	APNIC
Net Range	202.112.0.0 - 202.121.255.255
CIDR	202.112.0.0/13 202.120.0.0/15
Name	CERNET-CN
Handle	202.112.0.0 - 202.121.255.255

可以看到范围是202.112.0.0-202.121.255.255

f.攻击者可以利用whois数据库和nslookup来确定被攻击对象的IP地址范围、DNS服务器地址，进而方便发起攻击。

g.whois数据库显然应该为公众所用。举两个例子，个人注册自己的网站或者一些单位注册自己的官方网页时，他们都需要通过whois数据库来查看钟意的域名是否已经被注册过了；同时，面向公众的whois数据库也方便了消费者查询供应商网站的注册状态，使得他们更好地了解供应商的运营状况。

P22

P22. 考虑向 N 个对等方分发 $F = 15\text{Gb}$ 的一个文件。该服务器具有 $u_s = 30\text{Mbps}$ 的上载速率，每个对等方具有 $d_i = 2\text{Mbps}$ 的下载速率和上载速率 u 。对于 $N = 10$ 、 100 和 1000 并且 $u = 300\text{kbps}$ 、 700kbps 和

2Mbps ，对于 N 和 u 的每种组合绘制出确定最小分发时间的图表。需要分别针对客户 - 服务器分发和 P2P 分发两种情况制作。

在C-S分发情况下：

$$D_{CS} = \max\{\frac{NF}{u_s}, \frac{F}{d_{min}}\} = \max\{512N, 7680\}$$

$u \backslash N$	10	100	1000
300kbps	7680	51200	512000

$U \backslash N$	10	100	1000
700kbps	7680	51200	512000
2Mbps	7680	51200	512000

在P2P分发情况下:

$$D_{P2P} = \max\left\{\frac{F}{u_s}, \frac{F}{d_{\min}}, \frac{NF}{u_s + \sum_{i=1}^N u_i}\right\} = \max\left\{512, 7680, \frac{15 \cdot 2^{10} \cdot 2^{10} N}{30 \cdot 2^{10} + Nu}\right\}$$

(这里面u的单位以kbps记录)

$U \backslash N$	10	100	1000
300kbps	7680	25904	47559
700kbps	7680	15616	21525
2Mbps	7680	7680	7680

P23

P23. 考虑使用一种客户 - 服务器体系结构向 N 个对等方分发一个 F 比特的文件。假定一种某服务器能够同时向多个对等方传输的流体模型, 只要组合速率不超过 u_s , 则以不同的速率向每个对等方传输。

- 假定 $u_s/N \leq d_{\min}$ 。定义一个具有 NF/u_s 分发时间的分发方案。
- 假定 $u_s/N \geq d_{\min}$ 。定义一个具有 F/d_{\min} 分发时间的分发方案。
- 得出最小分发时间通常是由 $\max\{NF/u_s, F/d_{\min}\}$ 所决定的结论。

a. 服务器向每个客户端并行发送文件, 上载速率为 $\frac{u_s}{N}$ 。由于 $\frac{u_s}{N} \leq d_{\min}$, 因此客户端下载的速率也是 $\frac{u_s}{N}$, 这样每个客户端下载完成的时间为 $\frac{F}{\frac{u_s}{N}} = \frac{NF}{u_s}$

b. 服务器向每个客户端并行发送文件, 上载速率为 d_{\min} , 由于 $\frac{u_s}{N} \geq d_{\min}$, 因此 $u_s \geq Nd_{\min}$, 服务器可以承受此速率, 这样每个客户端以 d_{\min} 速率下载, 接收时间为 $\frac{F}{d_{\min}}$

c. 若 $\frac{u_s}{N} \leq d_{\min}$, $\frac{NF}{u_s} \geq \frac{F}{d_{\min}}$, 此时 $t = \frac{NF}{u_s} = \max\left\{\frac{NF}{u_s}, \frac{F}{d_{\min}}\right\}$

若 $\frac{u_s}{N} \geq d_{\min}$, $\frac{NF}{u_s} \leq \frac{F}{d_{\min}}$, 此时 $t = \frac{F}{d_{\min}} = \max\left\{\frac{NF}{u_s}, \frac{F}{d_{\min}}\right\}$

综上所述, $t = \max\left\{\frac{NF}{u_s}, \frac{F}{d_{\min}}\right\}$

P24

P24. 考虑使用 P2P 体系结构向 N 个用户分发 F 比特的一个文件。假定一种流体模型。为了简化起见，假定 d_{\min} 很大，因此对等方下载带宽不会成为瓶颈。

a. 假定 $u_s \leq (u_s + u_1 + \dots + u_N)/N$ 。定义一个具有 F/u_s 分发时间的分发方案。

b. 假定 $u_s \geq (u_s + u_1 + \dots + u_N)/N$ 。定义一个具有 $NF/(u_s + u_1 + \dots + u_N)$ 分发时间的分发方案。

c. 得出最小分发时间通常是由 $\max\{F/u_s, NF/(u_s + u_1 + \dots + u_N)\}$ 所决定的结论。

为了讨论方便 $u = \sum_{i=1}^N u_i$

这道题目还是非常复杂的，我采用的是逆向思路法。

a.

分发方案：

将文件分为 N 个块，第 i 个块大小为 $\frac{u_i}{u}F$ 比特，服务器并行地向第 i 个客户端以速率 $\frac{u_i}{u}u_s$ 传送第 i 个块，每个对等方向其他所有 $N-1$ 个对等方以速率 $\frac{u_i}{u}u_s$ 传送自己拥有的文件块。

采用这个方案，由于 $\frac{u_1}{u}u_s + \frac{u_2}{u}u_s + \dots + \frac{u_N}{u}u_s = \frac{u}{u}u_s = u_s$ ，服务端向客户端发送数据的总上载速率没有超过其限度。由于下载速率 d_{\min} 很大， $\frac{F}{d_{\min}}$ 很小，下载链路不会成为主要制约因素，因此可以认为第 i 个客户端接收到来自服务端的文件所需时间为 $\frac{\frac{u_i}{u}F}{\frac{u_i}{u}u_s} = \frac{F}{u_s}$ 。

每一个客户端在接收到来自服务端的数据块的同时，也在向其他所有用户传输自己所拥有的数据，对于第 i 个客户端，它发送给另外一个客户端的速率都是 $\frac{u_i}{u}u_s$ （和服务端传数据给它的速率一致，达到最快了）。总共要给其他 $N-1$ 个客户端发送自己的数据，这样总的传输速率是 $(N-1)\frac{u_i}{u}u_s$ 。

根据题意，可以知道 $u_s \leq \frac{u_s+u}{N}$ ，即 $\frac{(N-1)u_s}{u} \leq 1$ ，即 $(N-1)\frac{u_i}{u}u_s \leq u_i$ ，可以知道该客户端的总上载速率并没有超过其极限，这样是可行的。

对于第 i 个客户端而言，它需要多少时间获得来自其他客户端的数据呢？拆分的分析是不容易的，我采用了聚合分析的方法。其他客户端发送给第 i 个客户端的聚合速率是 $\sum_{k=1}^N \frac{u_k}{u}u_s - \frac{u_i}{u}u_s$ ，然后服务器给第 i 个客户端发送的速率是 $\frac{u_i}{u}u_s$ ，两者相加得到聚合速率 $\sum_{k=1}^N \frac{u_k}{u}u_s = u_s$ 。需要发送的数据量是 F 比特。同时，由于下载速率 d_{\min} 很大，下载链路不会成为主要制约因素，可以知道这个客户端能在 $\frac{F}{u_s}$ 时间内获取所有来自其他客户端的数据。

综上，这个方案能达到要求。

b.

分发方案：将文件分为 $N+1$ 个块，第 i 块的大小是 $\frac{Nu_i}{(u_s+u)(N-1)}F$ ， $i = 1, 2, \dots, N$ ，第 $N+1$ 块的大小是 $\frac{Nu_s - u_s - u}{(u_s+u)(N-1)}F$

服务器并行地向所有客户端以速率 $\frac{u_s - \frac{u}{N-1}}{N}$ 传送第 $N+1$ 个块，与此同时并行地向第 i 个客户端以上载速率 $\frac{u_i}{N-1}$ 传送第 i 个数据块，每个对等方向其他所有 $N-1$ 个对等方以速率 $\frac{u_i}{N-1}$ 传送自己拥有的文件块。

采用这个方案, 服务器给客户端的上载速率之和为 $N \frac{u_s - \frac{u}{N-1}}{N} + \sum_{i=1}^N \frac{u_i}{N-1} = u_s - \frac{u}{N-1} + \frac{u}{N-1} = u_s$, 服务端向客户端发送数据的总上载速率没有超过其限度。由于下载速率 d_{min} 很大, $\frac{F}{d_{min}}$ 很小, 下载链路不会成为主要制约因素, 因此可以认为第 i 个客户端接收到来自服务端的数据所需时间为 $\max\left\{\frac{\frac{N u_s - u_s - u}{(u_s + u)(N-1)} F}{\frac{u_s - \frac{u}{N-1}}{N}}, \frac{\frac{N u_i}{(u_s + u)(N-1)} F}{\frac{u_i}{N-1}}\right\} = \frac{NF}{u_s + u}$ 。

每一个客户端在接收到来自服务端的数据块的同时, 也在向其他所有用户传输自己所独有的数据, 对于第 i 个客户端, 它发送给另外一个客户端的速率都是 $\frac{u_i}{N-1}$ (和服务端传数据给它的速率一致, 达到最快了)。总共要给其他 $N-1$ 个客户端发送自己的独有的数据, 这样总的传输速率是 u_i , 可以知道该客户端的总上载速率并没有超过其极限, 这样是可行的。

对于第 i 个客户端而言, 它需要多少时间获得来自其他客户端的数据呢? 类似于 a 题我采用了聚合分析的方法。其他客户端发送给第 i 个客户端的聚合速率是 $\frac{1}{N-1}(u - u_i)$, 服务器向这个客户端发送的速率是 $\frac{u_s - \frac{u}{N-1}}{N} + \frac{u_i}{N-1}$, 这两者相加, 得到总的聚合上载速率是 $\frac{u}{N-1} + \frac{u_s - \frac{u}{N-1}}{N}$, 需要发送的数据量是 F 比特。同时, 由于下载速率 d_{min} 很大, 下载链路不会成为主要制约因素, 可以知道这个客户端能在 $\frac{NF}{u_s + u}$ 时间内获取所有来自其他客户端的数据。

综上, 这个方案能达到要求。

c.

根据课本前面的理论, 可知 $D_{P2P} = \max\left\{\frac{F}{u_s}, \frac{F}{d_{min}}, \frac{NF}{u_s + \sum_{i=1}^N u_i}\right\}$

但是在下载速率 d_{min} 很大的情况下, $\frac{F}{d_{min}}$ 很小, 可以忽略不计

综上, 结合 a, b 题中证明的相关理论, 可以知道最小分发时间在这样的情况下通常是由 $\max\left\{\frac{F}{u_s}, \frac{NF}{u_s + \sum_{i=1}^N u_i}\right\}$ 所决定的, 问题得证。

P25

P25. 考虑在一个有 N 个活跃对等方的覆盖网络中, 每对对等方有一条活跃的 TCP 连接。此外, 假定该 TCP 连接通过总共 M 台路由器。在对应的覆盖网络中, 有多少节点和边?

在这个网络中, 有 N 个结点。

边数为 $1 + 2 + \dots + (N-1) = \frac{(N-1)N}{2}$

P27

P27. 考虑一个具有 N 个视频版本 (具有 N 个不同的速率和质量) 和 N 个音频版本 (具有 N 个不同的速率和质量) 的 DASH 系统。假设我们想允许播放者在任何时间选择 N 个视频版本和 N 个音频版本之一:

- 如果我们生成音频与视频混合的文件, 因此服务器在任何时间仅发送一个媒体流, 该服务器将需要存储多少个文件 (每个文件有一个不同的 URL)?
- 如果该服务器分别发送音频流和视频流并且与客户同步这些流, 该服务器将需要存储多少个文件?

a.根据题目，我们按照质量和速率降序一对一地去匹配音频和视频，生成了音频和视频混合的文件就行了，所以这种情况下服务器要存储 N 个文件。

b.根据题目，服务器要分别发送音频和视频文件，这样的话服务器要存储 N 个视频文件和 N 个音频文件，总计要存储 $2N$ 个文件。