The Application Layer

Computer Networking, A Top-Down Approach, 5th Edition

任志诚 2023212020

2025-06-07 15:22:34

Contents

End-of-chapter exercises

R.1

List five nonproprietary Internet applications and the application-layer protocols that they use.

Type	Protocol(s)
Email	SMTP, IMAP, POP3
Web Browser	HTTP
File Transfer	FTP
Domain Name Resolution	DNS
Remote Terminal Access	SSH, Telnet

R.2

What is the difference between network architecture and application architecture?（不太理解这里的 network architecture，默认和 application architecture 一样，都指的是在一个 OSI layer）

Aspect	Network Architecture	Application Architecture
Definition	Describes the organization of network layers and components for data transmission.	Describes how application components interact to achieve specific functionalities.
Focus	Focuses on data transmission methods, routing, switching, and protocol stacks.	Focuses on the logical structure and communication patterns of applications.
Scope	Concerned with the entire network, including physical, data link, and network layers.	Concerned with the application layer and its communication between processes.
Examples	Virtual circuit networks, datagram networks.	Client-server model, P2P model (e.g., Skype, HTTP).

R.6

Suppose you wanted to do a transaction from a remote client to a server as fast as possible. Would you use UDP or TCP? Why?

- 对于简单、小型且允许失败的事务（如状态查询、监控数据上报），可以选择 **UDP**
- 对于大多数商业事务（如金融交易、数据库操作），应选择 **TCP**，因为：
  - 事务的完整性和正确性通常比速度更重要
  - TCP 的可靠性保障减少了应用层的复杂度
  - 虽然 TCP 建立连接有开销，但对于事务的整体成功率和效率更有保障
  - 在现代网络环境中，**TCP** 连接建立的时延相对事务处理总时间通常可以接受
- 但题目要求 **as fast as possible**，所以还是用 **UDP**。

R.17

Print out the header of an e-mail message you have recently received. How many `Received:` header lines are there? Analyze each of the header lines in the message.

```
Received: from codeforces.com (mx2.codeforces.com [77.234.215.195])
          by newxmmxszgpub6-1.qq.com (NewMX) with SMTP id BF70781A
          for <1xx575xxxx@qq.com>; Sat, 17 May 2025 00:47:55 +0800
Received: from localhost (gauss.codeforces.com [192.168.10.103])
          by codeforces.com (Postfix) with ESMTP id D7D9F322729C1
          for <1xx575xxxx@qq.com>; Fri, 16 May 2025 18:48:56 +0300 (
              MSK)
From: "Codeforces@codeforces.com" <Codeforces@codeforces.com>
To: "1xx575xxxx@qq.com" <1xx575xxxx@qq.com>
Subject: Codeforces Round 1025 (Div. 2)
```

There are 2 `Received:` header lines.

The first part is:

```
Received: from codeforces.com (mx2.codeforces.com [77.234.215.195])
          by newxmmxszgpub6-1.qq.com (NewMX) with SMTP id BF70781A
          for <1xx575xxxx@qq.com>; Sat, 17 May 2025 00:47:55 +0800
```

- `Received: from codeforces.com (mx2.codeforces.com [77.234.215.195])`: The email was sent from the Codeforces mail server with public IP.

- by newxmmxszgpub6-1.qq.com (NewMX): Received by QQ Mail’s mail server.
- Using **SMTP** protocol, Date\Time: Sat, 17 May 2025 00:47:55 +0800 .
- 其中的 mx2 指的是 **Mail eXchanger 2**，即 codeforces 的第二台邮件交换 server。

The second part is:

```
Received: from localhost (gauss.codeforces.com [192.168.10.103])
          by codeforces.com (Postfix) with ESMTP id D7D9F322729C1
          for <1xx575xxxx@qq.com>; Fri, 16 May 2025 18:48:56 +0300 (
              MSK)
```

- Received: from localhost (gauss.codeforces.com [192.168.10.103]) : The email originated from the local server named gauss.codeforces.com (internal IP).
- by codeforces.com (Postfix): Received by the main Codeforces mail server using Postfix.
- Using **ESMTP** protocol.

1. 邮件头传输顺序说明：邮件头按照邮件传输的相反顺序排列（最新的记录在最上面）。因此第二个Received行实际上是邮件传输的起始点，第一个是最后一跳。
2. 时区分析：注意到两个头部行的时间戳不同:
   - 第一个记录: Sat, 17 May 2025 00:47:55 +0800(中国时区)
   - 第二个记录: Fri, 16 May 2025 18:48:56 +0300(莫斯科时区 MSK) 这说明邮件确实是从俄罗斯发往中国的，时间差符合时区差异。
3. **ESMTP vs SMTP** 的区别：第二个头部使用 ESMTP(扩展 SMTP) 而不是普通 SMTP，这表明使用了更多高级功能 (如身份验证、加密等)。

R.22

What is an overlay network? Does it include routers? What are the edges in the overlay network? How is the query-flooding overlay network created and maintained?

An **overlay network** is a virtual network built on top of an existing physical network. It consists of logical connections (or “edges”) between nodes, which are typically end systems or hosts. These logical connections are established using the underlying physical network infrastructure.

- **Does it include routers?**
  No, an overlay network does not include physical routers. Instead, the nodes in the overlay network are typically end systems (e.g., computers, servers) that communicate directly with each other using logical links. The physical routers are part of the underlying network and are not explicitly represented in the overlay.
- **What are the edges in the overlay network?**
  The edges in an overlay network are logical connections between nodes. These connections are established using the underlying physical network but are abstracted away from the physical topology. For example, in a peer-to-peer (P2P) network, the edges represent direct communication paths between peers.
- **How is the query-flooding overlay network created and maintained?**
  A query-flooding overlay network is created by connecting nodes in a logical topology where each node knows a subset of other nodes (its neighbors). When a query is initiated, it is broadcasted (or “flooded”) to all neighboring nodes, which in turn forward the query to their neighbors, and so on.
  Maintenance of the overlay involves:

  1. **Node discovery:** New nodes join the network by discovering existing nodes and establishing connections.
  2. **Topology updates:** Nodes periodically update their neighbor lists to reflect changes in the network (e.g., nodes joining or leaving).
  3. **Failure handling:** Mechanisms are implemented to detect and recover from node or connection failures to ensure the overlay remains functional.

R.28

For the client-server application over TCP described in Section 2.7, why must the server program be executed before the client program? For the clientserver application over UDP described in Section 2.8, why may the client program be executed before the server program?

- **For the TCP client-server application (Section 2.7):** The server program must be executed before the client because the server needs to create a socket, bind it to a port, and listen for incoming connections. If the client starts first, it will try to connect to the server’s port, but if the server isn’t running and listening yet, the connection will fail.

- **For the UDP client-server application (Section** 2.8**):** The client program may be executed before the server because UDP is connectionless. The client can send a datagram to the server’s address and port even if the server isn’t running yet; the datagram may be lost, but the client doesn’t need to establish a connection first. When the server starts, it can immediately receive any new datagrams sent to its port.

P.4

Consider the following string of ASCII characters that were captured by Wireshark when the browser sent an HTTP `GET` message (i.e., this is the actual content of an HTTP `GET` message). The characters `<cr><lf>` are carriage return and line-feed characters (that is, the italized character string `<cr>` in the text below represents the single carriage-return character that was contained at that point in the HTTP header). Answer the following questions, indicating where in the HTTP GET message below you find the answer.

```
GET /cs453/index.html HTTP/1.1<cr><lf>Host: gai
a.cs.umass.edu<cr><lf>User-Agent: Mozilla/5.0 (
Windows;U; Windows NT 5.1; en-US; rv:1.7.2) Gec
ko/20040804 Netscape/7.2 (ax) <cr><lf>Accept:ex
t/xml, application/xml, application/xhtml+xml, text
/html;q=0.9, text/plain;q=0.8,image/png,*/*;q=0.5
<cr><lf>Accept-Language: en-us,en;q=0.5<cr><lf>Accept-
Encoding: zip,deflate<cr><lf>Accept-Charset: ISO
-8859-1,utf-8;q=0.7,*;q=0.7<cr><lf>Keep-Alive: 300<cr>
<lf>Connection:keep-alive<cr><lf><cr><lf>
```

Questions:

a. What is the URL of the document requested by the browser?
b. What version of HTTP is the browser running?
c. Does the browser request a non-persistent or a persistent connection?
d. What is the IP address of the host on which the browser is running?
e. What type of browser initiates this message? Why is the browser type needed in an HTTP request message?

Answers:

a. What is the URL of the document requested by the browser? - `http://gaia.cs.`

umass.edu/cs453/index.html.

b. What version of HTTP is the browser running? - HTTP/1.1

c. Does the browser request a non-persistent or a persistent connection? - Connection :keep-alive: a persistent connection.

d. What is the IP address of the host on which the browser is running? - The IP address of the host is not explicitly provided in the HTTP GET message. It would typically be determined by examining the network layer (IP) headers in the packet capture, which are not included in the provided data.

e. What type of browser initiates this message? Why is the browser type needed in an HTTP request message?

- The browser type is Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US ; rv:1.7.2)Gecko/20040804 Netscape/7.2 (ax).
- The browser type is included in the User-Agent header. It is needed in an HTTP request message to allow the server to tailor its response based on the browser’s capabilities, such as supported features, rendering engine, or platform-specific optimizations.

P.5

The text below shows the reply sent from the server in response to the HTTP GET message in the question above. Answer the following questions, indicating where in the message below you find the answer.

```
HTTP/1.1 200 OK<cr><lf>Date: Tue, 07 Mar 2008
12:39:45GMT<cr><lf>Server: Apache/2.0.52 (Fedora)
<cr><lf>Last-Modified: Sat, 10 Dec2005 18:27:46
GMT<cr><lf>ETag: "526c3-f22-a88a4c80"<cr><lf>Accept-
Ranges: bytes<cr><lf>Content-Length: 3874<cr><lf>
Keep-Alive: timeout=max=100<cr><lf>Connection:
Keep-Alive<cr><lf>Content-Type: text/html; charset=
ISO-8859-1<cr><lf><cr><lf><!doctype html public "-
//w3c//dtd html 4.0 transitional//en"><lf><html><lf>
<head><lf> <meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1"><lf> <meta
name="GENERATOR" content="Mozilla/4.79 [en] (Windows NT
5.0; U) Netscape"><lf> <title>CMPSCI 453 / 591 /
NTU-ST550A Spring 2005 homepage</title><lf></head><lf>
<much more document text following here (not shown)>
```

Questions and Answers:

a. Was the server able to successfully find the document or not? What time was the document reply provided? - `200 OK`: successfully find the document; `Tue, 07 Mar2008 12:39:45`

b. When was the document last modified? - `Last-Modified: Sat, 10 Dec2005 18:27:46`

c. How many bytes are there in the document being returned? - `Ranges: bytes<cr><lf>Content-Length: 3874`: 3874 bytes.

d. What are the first 5 bytes of the document being returned? Did the server agree to a persistent connection?

- **First 5 bytes of the document:** `<!doc` (from the document content starting with `<!doctype html public...`). 所选的 HTTP 报文内容声明了 `Content-Type: text/html; charset=ISO-8859-1`，该编码是单字节编码（每个字符 1 字节）
- **Persistent connection:** Yes, the server agreed to a persistent connection as indicated by the header `Connection: Keep-Alive`.

P.9

Consider Figure 2.12, for which there is an institutional network connected to the Internet. Suppose that the average object size is $850,000$ bits and that the average request rate from the institution’ s browsers to the origin servers is 16 requests per second. Also suppose that the amount of time it takes from when the router on the Internet side of the access link forwards an HTTP request until it receives the response is 3 seconds on average (see Section 2.2.5). **Model the total average response time as the sum of the average access delay (that is, the delay from Internet router to institution router) and the average Internet delay**. For the average access delay, use $\Delta/(1-\Delta\beta)$, where Δ is the average time required to send an object over the access link and β is the arrival rate of objects to the access link.

Complement: 15 Mbps access link and 100 Mbps LAN.

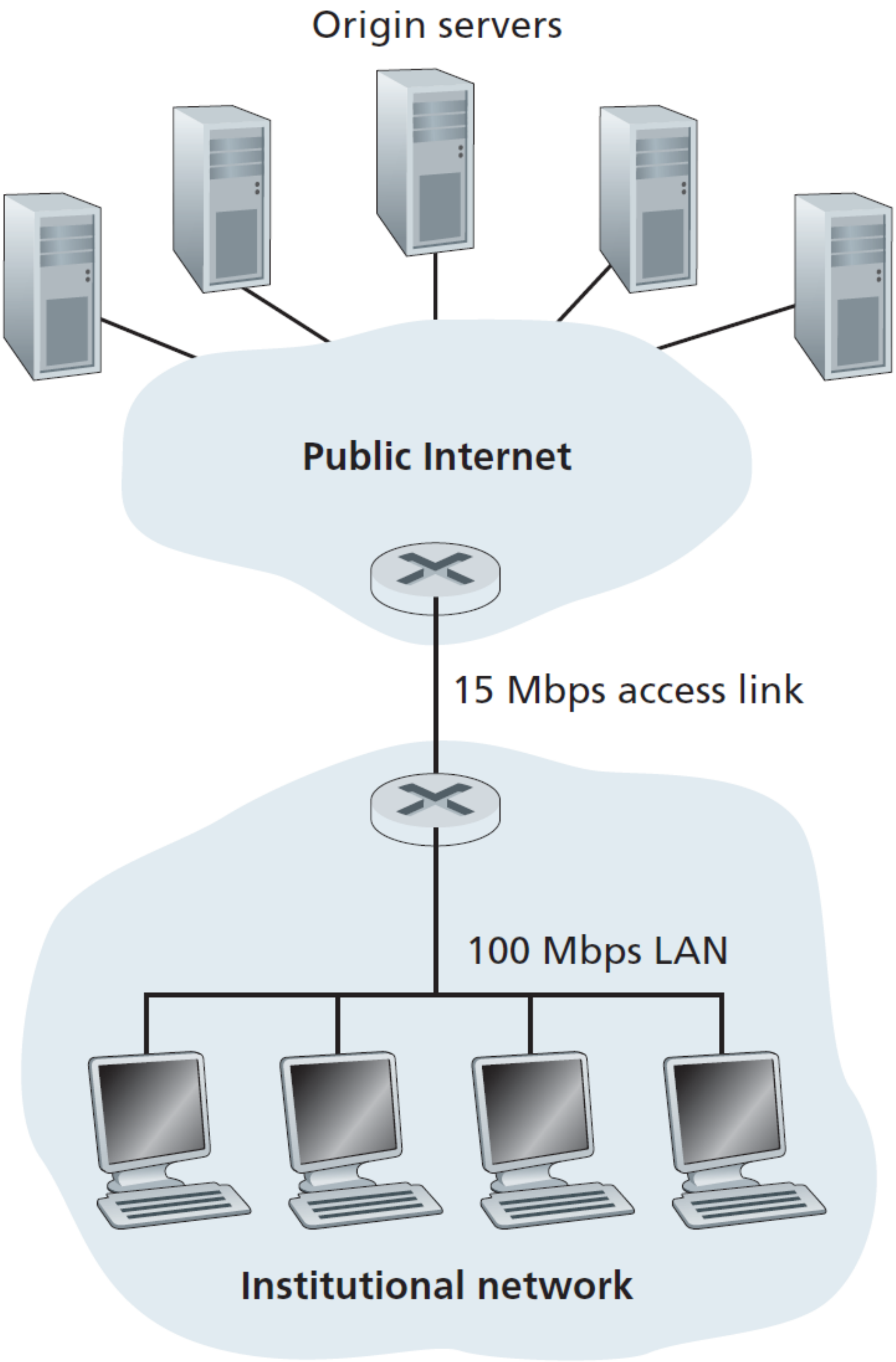

Figure 1: Bottleneck between an institutional network and the Internet

Questions and Answers:

a. Find the total average response time.

The total average response time is the sum of the **average access delay** and the **average Internet delay**.

1. **Given data:**
   - Average object size: $L = 850,000$ bits
   - Access link rate: $R = 15$ Mbps
   - Request rate: $\beta = 16$ requests/second
   - Average Internet delay: 3 seconds
2. **Calculate Δ:**
$$\Delta = \frac{L}{R} = \frac{850,000}{15 \times 10^6} = 0.0567\,\text{seconds}$$
3. **Calculate average access delay:**
$$\text{Access delay} = \frac{\Delta}{1 - \Delta\beta} = \frac{0.0567}{1 - (0.0567 \times 16)} = \frac{0.0567}{1 - 0.9072} = \frac{0.0567}{0.0928} \approx 0.611\,\text{seconds}$$
4. **Total average response time:**
$$\text{Total response time} = \text{Access delay} + \text{Internet delay} = 0.611 + 3 = 3.611\,\text{seconds}$$

b. Now suppose a cache is installed in the institutional LAN. Suppose the miss rate is 0.4**. Find the total response time.**

1. **Given data:**
   - Miss rate: 0.4
   - Hit rate: $1 - 0.4 = 0.6$
   - Access delay (from part a): 0.611 seconds
   - Internet delay: 3 seconds
2. **Calculate total response time with caching:**
$$\text{Total response time} = (\text{Hit rate} \times \text{Access delay}) + (\text{Miss rate} \times (\text{Access delay} + \text{Internet delay}))$$

Substituting values:

$$\text{Total response time} = 0.6 \times \frac{850000}{100 \times 10^6} + 0.4 \times (0.611 + 3)$$

$$\text{Total response time} = 0.0051 + 0.4 \times 3.611 = 1.4495\,\text{s}$$

P.15

Question:

Read RFC 5321 for SMTP. What does MTA stand for? Consider the following received spam email (modified from a real spam email). Assuming only the originator of this spam email is malacious and all other hosts are honest, identify the malacious host that has generated this spam email.

```
From - Fri Nov 07 13:41:30 2008
Return-Path: <tennis5@pp33head.com>
Received: from barmail.cs.umass.edu
(barmail.cs.umass.edu [128.119.240.3]) by cs.umass.edu
(8.13.1/8.12.6) for <hg@cs.umass.edu>; Fri, 7 Nov 2008
13:27:10 -0500
Received: from asusus-4b96 (localhost [127.0.0.1]) by
barmail.cs.umass.edu (Spam Firewall) for
<hg@cs.umass.edu>; Fri, 7 Nov 2008 13:27:07 -0500
(EST)
Received: from asusus-4b96 ([58.88.21.177]) by
barmail.cs.umass.edu for <hg@cs.umass.edu>; Fri,
07 Nov 2008 13:27:07 -0500 (EST)
Received: from [58.88.21.177] by
inbnd55.exchangeddd.com; Sat, 8 Nov 2008 01:27:07 +0700
From: "Jonny" <tennis5@pp33head.com>
To: <hg@cs.umass.edu>
Subject: How to secure your savings
```

Answer:

最底部的 **Received**：记录代表邮件的最初来源，即发件人最初连接的主机。

- **What does MTA stand for?**
  MTA stands for **Mail Transfer Agent**. It is a software application used to transfer email messages from one server to another using protocols such as SMTP.

- **Identify the malicious host:**
  To identify the malicious host, we analyze the Received headers in reverse order (from

bottom to top), as each `Received` header represents a hop in the email’s journey.

1. `Received: from [58.88.21.177] by inbnd55.exchangeddd.com`
   - This indicates that the email originated from the IP address `58.88.21.177`.
2. `Received: from asusus-4b96 ([58.88.21.177])by barmail.cs.umass.edu`
   - This confirms that the email was sent from the same IP address `58.88.21.177`.
3. `Received: from asusus-4b96 (localhost [127.0.0.1])by barmail.cs.umass.edu`
   - This shows that the email passed through a local host (`127.0.0.1`) on the `barmail.cs.umass.edu` server.
4. `Received: from barmail.cs.umass.edu (barmail.cs.umass.edu [128.119.240.3])by cs.umass.edu`
   - This indicates that the email was forwarded by `barmail.cs.umass.edu` to `cs.umass.edu`.

Based on the analysis, the **malicious host** is the originator of the email, which is the IP address **`58.88.21.177`**. This is the source of the spam email.

P.18

Questions and Answers:

a. What is a whois database?

A **whois database** is a publicly accessible database that contains information about the registered owners of domain names and IP address blocks. It is maintained by domain registrars and regional internet registries (RIRs). The database provides details such as:

- The name and contact information of the domain owner or organization.
- The domain’s registration and expiration dates.
- The domain’s associated name servers.
- The registrar responsible for the domain.

The **whois database** is commonly used for administrative purposes, such as verifying domain ownership, resolving technical issues, or investigating malicious activities.

b. Use various whois databases on the Internet to obtain the names of two DNS servers. Indicate which whois databases you used.

DNS server 的名字，通常指的是域名形式的主机名。

通过在 ICANN Lookup 查询 `BiliBili.com` 得到其 Nameservers - `NS3.DNSV5.COM` - `NS4.DNSV5.COM`

通过在 DomainTools 查询 `Baidu.com` 得到其 Nameservers - `NS1.BAIDU.COM (has 805 domains)` - `NS2.BAIDU.COM (has 805 domains)` - `NS3.BAIDU.COM (has 805 domains)` - `NS4.BAIDU.COM (has 805 domains)` - `NS7.BAIDU.COM (has 805 domains)`

c. Use nslookup on your local host to send DNS queries to three DNS servers: your local DNS server and the two DNS servers you found in part (b). Try querying for Type A, NS, and MX reports. Summarize your findings.

我在终端输入的语句及其显示结果如下：

```
(base) PS C:\Users\17657\Desktop\Github\HEXO> nslookup
默认服务器:  UnKnown
Address:  10.3.9.5

> set type=A
> www.baidu.com
服务器:  UnKnown
Address:  10.3.9.5

非权威应答:
名称:    www.a.shifen.com
Addresses:  220.181.111.232
          220.181.111.1
Aliases:  www.baidu.com

> set type=NS
> baidu.com
服务器:  UnKnown
Address:  10.3.9.5

非权威应答:
baidu.com       nameserver = ns2.baidu.com
baidu.com       nameserver = ns7.baidu.com
baidu.com       nameserver = ns3.baidu.com
baidu.com       nameserver = dns.baidu.com
baidu.com       nameserver = ns4.baidu.com

ns2.baidu.com   internet address = 220.181.33.31
ns7.baidu.com   internet address = 180.76.76.92
dns.baidu.com   internet address = 110.242.68.134
ns3.baidu.com   internet address = 36.155.132.78
ns3.baidu.com   internet address = 153.3.238.93
ns4.baidu.com   internet address = 14.215.178.80
ns4.baidu.com   internet address = 111.45.3.226
> set type=MX
> baidu.com
服务器:  UnKnown
Address:  10.3.9.5

非权威应答:
baidu.com       MX preference = 20, mail exchanger = mx.baidu.com
baidu.com       MX preference = 10, mail exchanger = mx.maillb.baidu.com
> server ns1.baidu.com
默认服务器:  ns1.baidu.com
Address:  110.242.68.134

> set type=A
> www.baidu.com
服务器:  ns1.baidu.com
Address:  110.242.68.134

非权威应答:
名称:    www.a.shifen.com
Addresses:  220.181.111.1
          220.181.111.232
Aliases:  www.baidu.com

> set type=NS
> baidu.com
服务器:  ns1.baidu.com
Address:  110.242.68.134

非权威应答:
baidu.com       nameserver = dns.baidu.com
baidu.com       nameserver = ns3.baidu.com
baidu.com       nameserver = ns4.baidu.com
baidu.com       nameserver = ns2.baidu.com
baidu.com       nameserver = ns7.baidu.com

ns7.baidu.com   internet address = 180.76.76.92
ns4.baidu.com   internet address = 14.215.178.80
ns4.baidu.com   internet address = 111.45.3.226
ns2.baidu.com   internet address = 220.181.33.31
ns3.baidu.com   internet address = 36.155.132.78
ns3.baidu.com   internet address = 153.3.238.93
```

```
dns.baidu.com   internet address = 110.242.68.134
> set type=MX
> baidu.com
服务器:  ns1.baidu.com
Address:  110.242.68.134

非权威应答:
baidu.com       MX preference = 20, mail exchanger = mx.baidu.com
baidu.com       MX preference = 10, mail exchanger = mx.maillb.baidu.com
> server ns3.dnsv5.com
默认服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

> set type=A
> baidu.com
服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

非权威应答:
名称:    baidu.com
Addresses:  182.61.201.211
          182.61.244.181

> set type=NS
> baidu.com
服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

非权威应答:
baidu.com       nameserver = ns4.baidu.com
baidu.com       nameserver = ns7.baidu.com
baidu.com       nameserver = ns2.baidu.com
baidu.com       nameserver = ns3.baidu.com
baidu.com       nameserver = dns.baidu.com

ns7.baidu.com   internet address = 180.76.76.92
ns4.baidu.com   internet address = 14.215.178.80
ns4.baidu.com   internet address = 111.45.3.226
ns2.baidu.com   internet address = 220.181.33.31
ns3.baidu.com   internet address = 36.155.132.78
ns3.baidu.com   internet address = 153.3.238.93
dns.baidu.com   internet address = 110.242.68.134
> set type=MX
> baidu.com
服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

非权威应答:
baidu.com       MX preference = 10, mail exchanger = mx.maillb.baidu.com
baidu.com       MX preference = 20, mail exchanger = mx.baidu.com
```

总结：- `www.baidu.com` 和 `baidu.com` 不是同一个东西。具体来说后者涵盖范围更广。- 一个 Nameserver 能有多个 Internet address。- `type=A` 模式返回的是域名的 IPv4 地址。- `type=NS` 模式返回的是 Nameserver 的名字和其 internet address。- `type=MS` 模式返回的是该域名的邮件服务器主机名及优先级。

d. Use nslookup to find a Web server that has multiple IP addresses. Does the Web server of your institution (school or company) have multiple IP addresses?

查询 `www.bilibili.com` 得到结果如下，其有两个 IP address。

```

> set type=A
> www.bilibili.com
服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

```

```
非权威应答:
名称:    a.w.bilicdn1.com
Addresses:  121.194.11.73
          121.194.11.72
Aliases:  www.bilibili.com
```

貌似我们学校的 web server 只有一个 IP Address

```

> ucloud.bupt.edu.cn
服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

非权威应答:
名称:    vn.bupt.edu.cn
Address:  10.3.19.2
Aliases:  ucloud.bupt.edu.cn

> auth.bupt.edu.cn
服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

非权威应答:
名称:    vn.bupt.edu.cn
Address:  10.3.19.2
Aliases:  auth.bupt.edu.cn

> www.bupt.edu.cn
服务器:  ns3.dnsv5.com
Addresses:  1.12.0.18
          1.12.0.17
          43.140.237.52
          111.13.203.52
          36.155.149.211
          101.227.168.52
          220.196.136.52

非权威应答:
```

```
名称:    vn46.bupt.edu.cn
Address:  10.3.19.2
Aliases:  www.bupt.edu.cn
```

e. Use the ARIN whois database to determine the IP address range used by your university.

操作：

```

nslookup www.bupt.edu.cn
服务器:  UnKnown
Address:  10.3.9.5

非权威应答:
名称:    vn46.bupt.edu.cn
Addresses:  2001:da8:215:4038::161
          10.3.19.2
Aliases:  www.bupt.edu.cn
```

这里的IPv4地址是子网地址吧？我在rain上查询查到了一个美国机构，而且网页也提醒我了 **These addresses are in use by many millions of independently operated networks, which might be as small as a single computer connected to a home gateway, and are automatically configured in hundreds of millions of devices.** 所以我用的是IPv6的地址，这个地址能查到，显示：Net Range `2001:da8:: - 2001:da8:ffff:ffff:ffff:ffff:ffff:ffff`。

f. Describe how an attacker can use whois databases and the nslookup tool to perform reconnaissance on an institution before launching an attack.

An attacker 可以用 whois 和 nslookup 干如下的事情：

- **whois** 数据库：公开的域名/IP 注册信息数据库，可查询域名所有者、联系方式、DNS 服务器、IP 地址段等信息。
- **nslookup** 工具：DNS 查询工具，可用来获取域名解析记录（如 A、NS、MX、CNAME 等），进一步了解目标机构的网络结构和服务部署。

从而可以：

- 查询目标机构的域名，获取注册人、联系方式、注册商、DNS 服务器、IP 地址段等信息。
- 通过 whois 查询 IP 地址，了解目标机构的公网 IP 范围、网络归属、可能的子网划分。

- 利用这些信息，攻击者可以锁定攻击目标、寻找潜在的弱点（如联系邮箱、技术负责人等）。
- 查询目标机构域名的 A 记录，获取 Web 服务器等主机的 IP 地址。
- 查询 NS 记录，了解目标机构使用的权威 DNS 服务器，判断是否存在 DNS 攻击面。
- 查询 MX 记录，获取邮件服务器信息，可能用于钓鱼邮件、垃圾邮件攻击。
- 查询 CNAME、TXT 等记录，发现隐藏的服务、第三方集成、邮件安全策略等。
- 通过对不同子域名的批量查询，发现更多内部服务和主机。

进而：

- 绘制目标机构的网络拓扑和服务分布图。
- 寻找潜在的攻击入口（如暴露的服务器、邮件系统、DNS 服务等）。
- 为后续的漏洞扫描、社工攻击、钓鱼邮件等攻击手段做准备。

g. Discuss why whois databases should be publicly available.

whois 数据库作为互联网基础设施的重要组成部分，其公开可用性具有多方面的价值与意义：

1. 互联网透明度与问责制
    - 提供域名和 IP 地址资源的所有权透明度，确保资源分配可追溯
    - 建立互联网资源使用的公开记录，减少匿名滥用可能性
    - 符合互联网作为公共资源的基本属性，保障公众知情权
2. 技术协调与故障排除
    - 网络管理员能迅速找到技术联系人解决网络问题
    - 跨组织网络协作时提供必要的联络信息
    - 在安全事件、网络中断等紧急情况下提供快速响应渠道
3. 法律与知识产权保护
    - 协助商标持有者保护其在线知识产权
    - 为域名争议解决提供必要的所有权信息
    - 帮助执法机构打击网络犯罪和识别不法行为
4. 历史与文化因素
    - 符合互联网早期建立的开放共享精神
    - 继承了学术网络环境下的信任与协作文化

- 反映了互联网治理中的多方参与模式

5. 安全与风险的平衡

    - 虽然公开信息存在被滥用的风险，但安全通过隐蔽不是可持续策略
    - 现代 whois 服务已引入数据隐私保护机制（如代理注册服务）
    - 信息公开带来的集体安全收益通常超过个体风险

总之，whois 数据库的公开可用反映了互联网基于透明、协作和问责的核心价值观，在保护隐私和维护网络健康运行之间寻求平衡。尽管存在被攻击者利用的风险，但其对互联网正常运行、问题排除和资源管理的价值仍然超过潜在风险。

P.20

Question and Answer: Suppose you can access the caches in the local DNS servers of your department. Can you propose a way to roughly determine the Web servers (outside your department) that are most popular among the users in your department? Explain.

To determine the most popular external Web servers among the users in my department, I would propose the following method:

1. **Access the local DNS server’s cache:**

    - The local DNS server maintains a cache of recently resolved domain names and their corresponding IP addresses.
    - By accessing this cache, I can retrieve a list of domain names that users in my department have recently accessed.

2. **Filter out internal domain names:**

    - Remove any domain names that belong to the local department or organization.

3. **Count the frequency of external domain names:**

    - For each external domain name in the cache, count how many times it appears.
    - This will give an estimate of how frequently users in the department access each external Web server.

4. **Identify the most popular Web servers:**

    - Sort the external domain names by their access frequency.

- The domain names with the highest counts represent the most popular external Web servers among the users in the department.

Explanation: This method works because the local DNS server’s cache reflects the browsing behavior of users in the department. By analyzing the cache, we can infer which external Web servers are most frequently accessed. However, this method has limitations, as it only provides a rough estimate and may not account for caching mechanisms in user devices or browsers.

P.22

Question:

Consider distributing a file of $F = 15$ Gbits to N peers. The server has an upload rate of $u_s = 30$ Mbps, and each peer has a download rate of $d_i = 2$ Mbps and an upload rate of u. For $N = 10$, 100, and $1,000$ and $u = 300$ Kbps, 700 Kbps, and 2 Mbps, prepare a chart giving the minimum distribution time for each of the combinations of N and u for both client-server distribution and P2P distribution.

Answer:

To calculate the minimum distribution time for both client-server distribution and P2P distribution, we use the following formulas:

1. **Client-Server Distribution:**

$$t_{cs} = \max\left\{\frac{N \cdot F}{u_s}, \frac{F}{d_i}\right\}$$

2. **P2P Distribution:**

$$t_{p2p} = \max\left\{\frac{F}{u_s}, \frac{F}{d_i}, \frac{N \cdot F}{u_s + \sum_i u_i}\right\}$$

N	u Kbps	t_{cs} seconds	t_{p2p} seconds
10	300	$\max\{5000, 7500\} = 7500$	$\max\{500, 7500, 4545\} = 7500$
10	700	$\max\{5000, 7500\} = 7500$	$\max\{500, 7500, 4054\} = 7500$
10	$2,000$	$\max\{5000, 7500\} = 7500$	$\max\{500, 7500, 3000\} = 7500$

N	u Kbps	t_{cs} seconds	t_{p2p} seconds
100	300	$\max\{50000, 7500\} = 50000$	$\max\{500, 7500, 25000\} = 25000$
100	700	$\max\{50000, 7500\} = 50000$	$\max\{500, 7500, 15000\} = 15000$
100	$2,000$	$\max\{50000, 7500\} = 50000$	$\max\{500, 7500, 6522\} = 7500$
$1,000$	300	$\max\{500000, 7500\} = 500000$	$\max\{500, 7500, 45455\} = 45455$
$1,000$	700	$\max\{500000, 7500\} = 500000$	$\max\{500, 7500, 20548\} = 20548$
$1,000$	$2,000$	$\max\{500000, 7500\} = 500000$	$\max\{500, 7500, 7389\} = 7500$

1. 当节点数量较少（N=10）时，无论使用何种上传速率，两种分发方式所需时间相同，均受限于节点的下载速率。
2. 当节点数量增加时，客户端-服务器模式的分发时间显著上升，而 P2P 模式在节点上传速率足够高时效率更高。
3. 当节点上传速率达到 **2 Mbps** 时，P2P 模式的分发时间在各种节点数下都可以保持在较低水平，这显示了 P2P 架构在大规模分发时的优势。