计算机网络综合实验

为什么要做网络实验？

- 中国有一句古老的名言：

  “不闻不若闻之，闻之不若见之，见之不若知之，知之不若行之。学至于行而止矣。”

  ——《荀子·儒效》

- 这句话在西方被广为引用：

  “Tell me and I forget. Show me and I remember. Involve me and I understand.”

  —— Chinese Proverb

- 根本原因：

  加深对理论的理解

  提高实践的能力

教学目标

- 使学生通过实验更加深刻的理解计算机网络的基本原理、算法和协议
- 使学生掌握H3C网络设备的配置、组网方法（不同网络设备制造商生产的设备是大同的）
- 为学生参加H3C的H3CNE技术认证考试打下良好的基础（认证的特点和社会认可度；对认证保持正确的态度）

课程主要内容

共计9次实验：

- 网络协议分析工具Wireshark的使用
- 登录交换机与网络操作系统VRP介绍
- 交换机的端口配置与生成树协议配置
- VLAN及VLAN间路由配置
- 登录路由器与路由协议（静态及RIP）配置
- 广域网协议配置
- 防火墙配置与NAT配置
- 路由综合实验（含OSPF）与故障诊断
- 组网综合实验（闭卷实践考核）

教材与参考书

- 教材（推荐）

  新华三大学编著《路由与交换技术》（第1卷）（上、下册）清华大学出版社，2017

- 参考书

  H3C网络设备的操作手册（官方网站下载）

  陈鸣（译）《计算机网络：自顶向下方法》（原书第7版）机械工业出版社 2018

  诸葛建伟（译）.《Wireshark数据包分析实战》（第3版）人民邮电出版社出版社, 2018

  大学霸IT达人编写组编著.《从实践中学习Wireshark数据分析》（第一版）机械工业出版社， 2020年

第一讲　网络协议分析工具Wireshark的使用

第一讲 网络协议分析工具Wireshark的使用

- Wireshark概览
- Wireshark的图形界面
- Wireshark的基本用法
- Wireshark的过滤规则

Wireshark概览

- Wireshark是目前使用最广泛的网络协议分析工具（Network Protocol Analyzer）
- Wireshark能够实时地捕获网络上的包（帧）（所以又被称为是“A Packet Sniffer”），并对包中每个域的细节进行解释
- Wireshark的广为流行主要有以下三个原因：

  功能相当强大

  开源并且免费

  具有非常详细的文档
- 2006年后，Ethereal改名为Wireshark，最新的安装文件和文档可以在http://www.wireshark.org下载 ，Wireshark目前也推出了相应的认证考试和教材。（这是一个所有64位wireshark 版本下载的链接 https://1.as.dl.wireshark.org/win64/all-versions/）

Wireshark概览（WinPcap和Npcap）

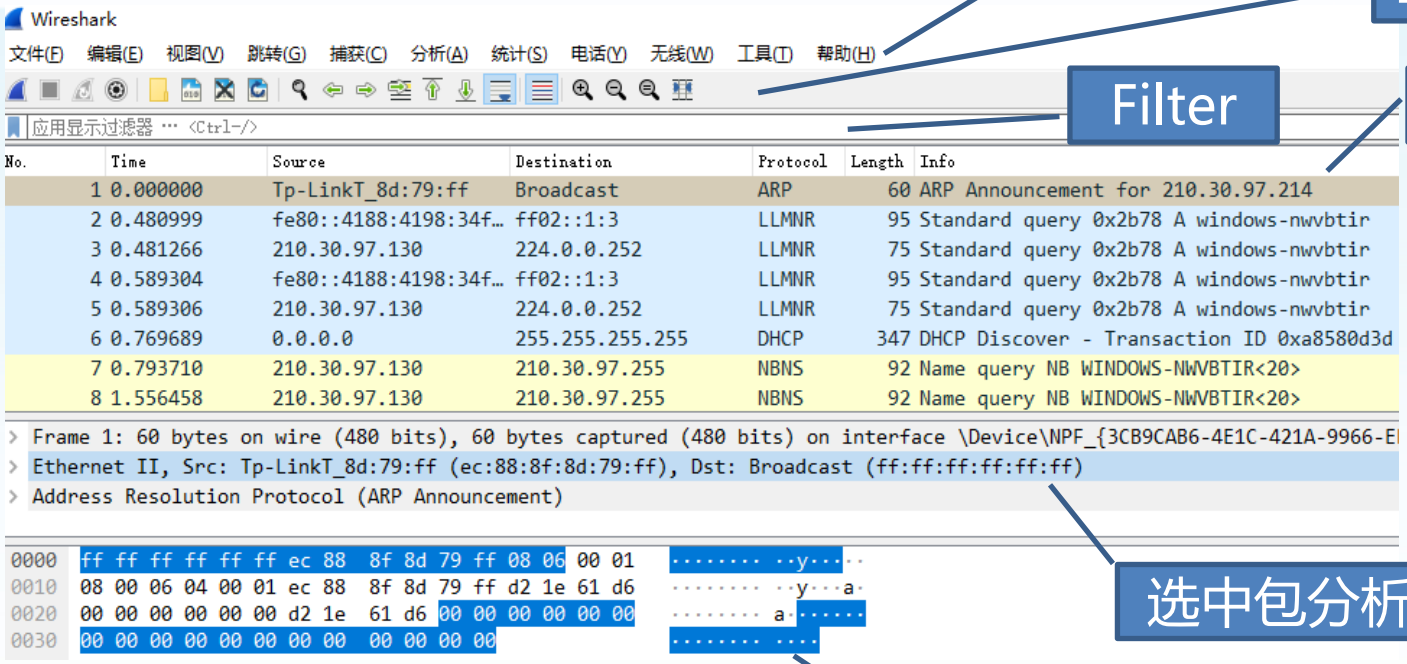
- 多年来，WinPcap被公认为Windows环境中用于链路层网络访问的行业标准工具，它允许应用程序绕过协议栈来捕获和传输网络数据包，包括内核级数据包过滤，网络统计引擎和支持远程数据包捕获。
- WinPcap包含一个扩展操作系统的驱动程序，以提供低级网络访问权限；该库用于轻松访问低级网络层。该库还包含Windows版本的著名libpcap Unix API。
- 凭借其一系列功能，WinPcap已成为许多开源和商业网络工具的数据包捕获和过滤引擎，包括协议分析器，网络监视器，网络入侵检测系统，嗅探器，流量生成器和网络测试器。其中的一些联网工具，例如Wireshark，Nmap，Snort和ntop，在整个联网社区中都是众所周知的并使用。
- Npcap是Windows的Nmap项目的数据包嗅探（和发送）库。它基于已停产的WinPcap库，但具有提高的速度，可移植性，安全性和效率。Wireshark 自3.0以上版本开始使用Npcap。

Wireshark的图形界面

共分为5个部分：

- 菜单项和工具栏：Wireshark提供的所有功能都可以从这一部分中找到
- Filter: 又称Display Filter，用于在显示被捕获包时对它们进行过滤
- 被捕获包列表：给出了被捕获包的一般信息，例如被捕获的时间，源和目的IP地址，所属的协议类型等
- 选中包分析：解释选中包的每一个域的具体信息，从以太网帧的首部开始，一直到应用层协议的负载
- 选中包原貌：显示选中包的16进制和ASCII表示，以帮助用户来了解一个包的本来的样子

Wireshark的图形界面（续）



菜单项

工具栏

Filter

被捕获包列表

选中包分析

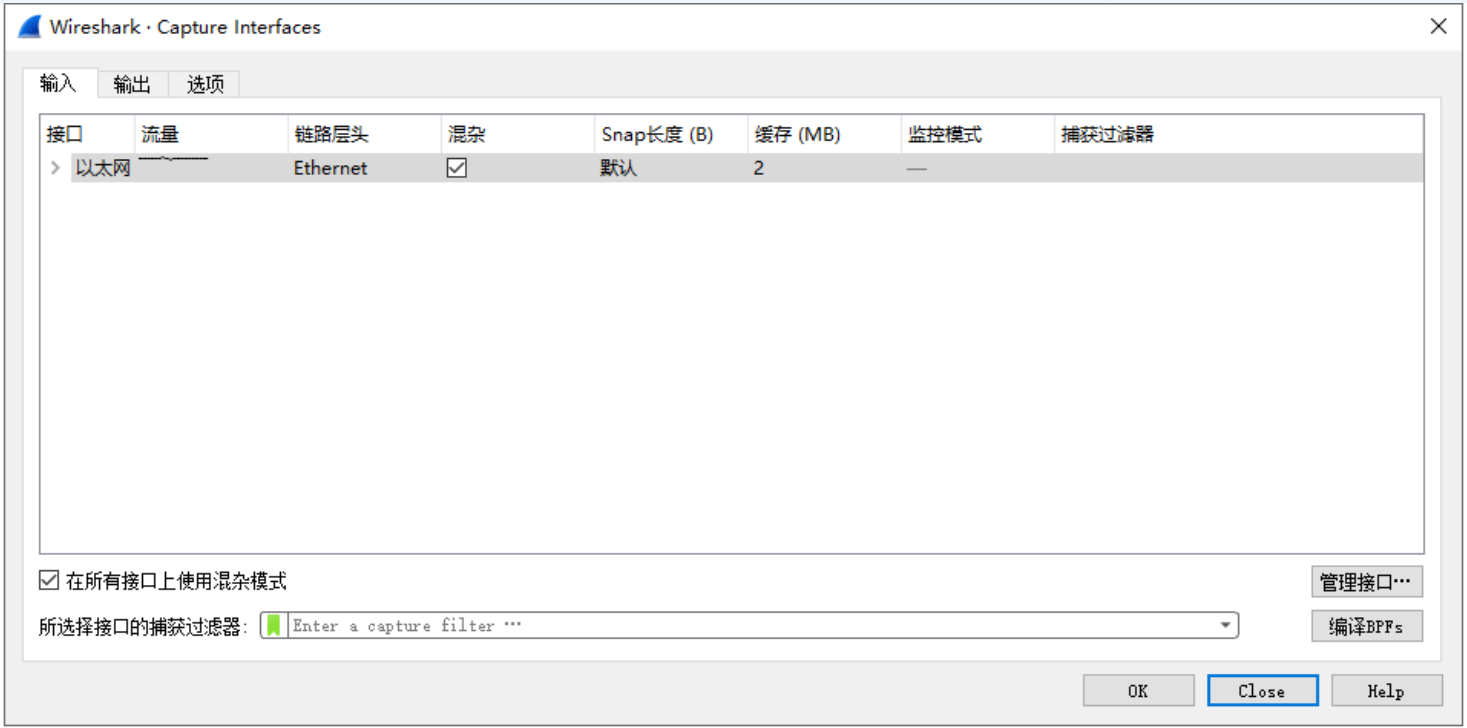
选中包原貌

Wireshark的基本用法 — 捕获包

- 选择”Capture”菜单项中的”Options”
- 配置”Options”对话框：（不同版本的Options对话框有差别）
  - 选择执行”Capture”的网卡（即”Interface”）选对网卡后，IP地址会从Unknow变成有效值。
  - 配置”Capture Filter”
  - 选中“Update list of packets in real time”
- 点击“Start”按钮

Wireshark中options对话框设置

启动wireshark抓帧前的必要设置



Wireshark的基本用法 — 保存结果

注意：不同版本Save操作会有细微差别

- 选择”File”菜单项中的”Save”
- ”Save”对话框：
  - 在“Name:”栏填入保存文件名
  - 在“Save in folder:”栏选择保存路径
- 点击“Save”按钮

Wireshark的过滤规则

- 精通Wireshark使用的重点所在
- Wireshark支持在两处配置过滤规则：
  - Wireshark Display Filter
  - Wireshark Capture Filter
- 两处过滤规则的语法并不相同

Display Filter 过滤规则（1）

- 根据协议来进行包过滤：
  - eth, ip, tcp, http, telnet等

- 根据包格式中的域值来进行包过滤
  - 例：显示以太网地址为 00:d0:f8:00:00:03 的设备的所有帧
    eth.addr==00.d0.f8.00.00.03
  - 例：显示 IP地址为 192.168.10.1 的主机的所有报文
    ip.addr==192.168.10.1
  - 例：显示所有 Web 浏览的报文
    tcp.port==80

Display Filter 过滤规则（2）

- 类似C语言的表达式
  - 关系运算符：==, >, <=, != 等
  - 逻辑连接符：&&, ||, ! 等
  - 还可以用 ( ) 来指定运算顺序
  - 例：显示192.168.10.1除了http外的所有报文
    ip.addr==192.168.10.1 && tcp.port!=80
- 书写过滤规则的两个诀窍：
  - 看Display Filter的背景色
    - 绿色时，说明规则符合语法；红色时，则否。
  - 使用Expression对话框
    - 查询协议或域的关键字
    - wireshark支持的全部协议及协议字段可查看
      https://www.wireshark.org/docs/dfref/

Capture Filter 的过滤规则（1）

- 过滤规则共有两种形式：
  - 一个原语（Primitive）；
  - 用”and”，“or”，“not” 关系运算符，以及括号”()”将原语组合起来而构成的表达式；

- 因此，我们只要掌握了原语的写法，再用关系运算符把它们组合起来，就可以写出满足各种不同要求的过滤规则了

Capture Filter 的过滤规则（2）

- ether [src|dst] host <mac_addr>
  - 捕获所有源或目的MAC地址是“08:00:1B:D3:D3:61”的以太网帧

    ether host 08:00:1B:D3:D3:61
  - 捕获所有源MAC地址是“08:00:1B:D3:D3:61”的以太网帧

    ether src host 08:00:1B:D3:D3:61

- [src|dst] host <ip_addr>
  - 捕获所有源或目的IP地址是“210.30.97.53”的报文

    host 210.30.97.53
  - 捕获所有目的IP地址是“210.30.97.53”的报文

    dst host 210.30.97.53

Capture Filter 的过滤规则（3）

- [tcp|udp] [src|dst] port <number>
  - 捕获所有源或目的端口号是80的TCP协议的包

    tcp port 80
  - 捕获所有目的端口号是53的UDP协议的包

    udp dst port 53

- arp | ip | icmp | udp | tcp 等
  - 捕获所有ICMP协议的包

    icmp
  - 捕获所有ARP协议的包

    arp

Capture Filter 的过滤规则（4）

用关系运算符组合原语举例:

例1：捕获主机 192.168.0.10 发出或收到的，除HTTP协议之外的所有网络包

host 192.168.0.10 and not tcp port 80

注：HTTP协议通常使用TCP 端口号 80。

例2：记主机 192.168.0.10为 A，捕获 A 与主机 192.168.0.20 或 A 与主机 192.168.0.30 之间的网络包

host 192.168.0.10 and (host 192.168.0.20 or host 192.168.0.30)

附录一：traceroute 和 tracert

- traceroute
  - 请参见课程网站上的文章“Traceroute-from-Linux.doc”
- tracert
  - 请参见课程网站上的文章 “Tracert-from-Microsoft.doc”

附录二：DNS（Domain Name System）

- 起因
- 历史
- 域名 (Domain Names)
- 域名服务器 (Name Servers)
- 域名解析 (Name Resolution)
- 资源纪录 (Resource Records)
- DNS消息

DNS – 起因

- 网络设备使用IP地址，但IP地址难于记忆
- 人们容易记忆有意义的名字
  - www.dlut.edu.cn
  - www.huawei-3com.com.cn
- DNS (Domain Name System) 就是用于在IP地址和主机名字之间进行相互转换的查询系统。

DNS – 历史

- 在DNS产生之前，人们使用一个文本文件来记录IP地址和主机名字之间的映射
  - /etc/hosts
  - 今天仍然在使用

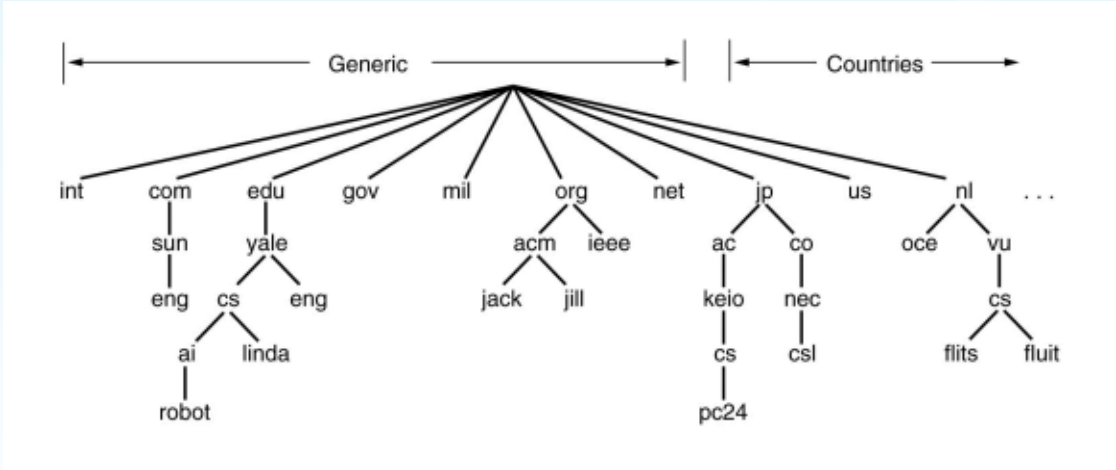
- 1983年之后，人们逐渐开始使用DNS.
  - RFC 882, 883　1983
  - RFC 1034, 1035　1987 -- today

DNS – 域名

- Internet上的主机数目极其巨大 (4.4亿个, Jul 2006), 如何管理数目如此巨大的名字呢？

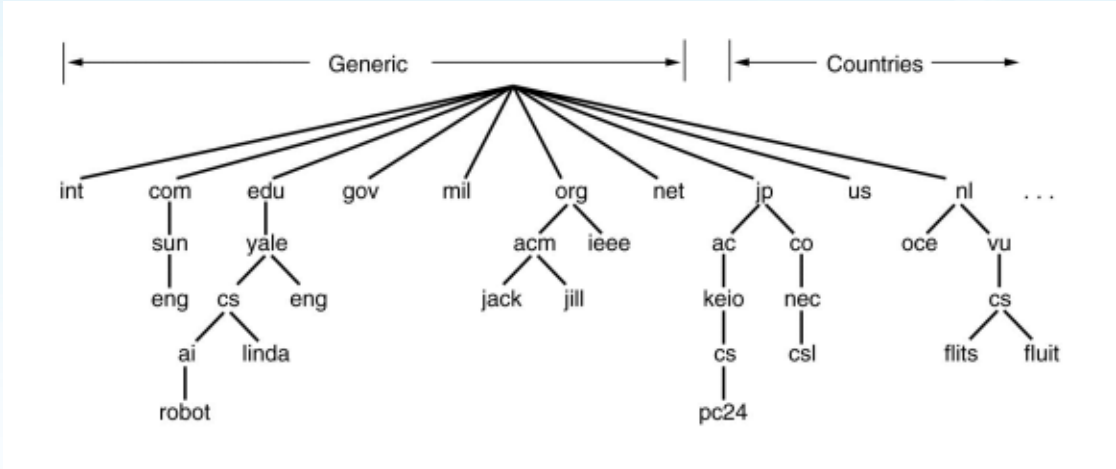
- 方法：把Internet上的主机名字组织成分层的树状结构。
  - 模仿地名系统 ：国家 -> 省份 -> 城市 -> 街道

DNS – 域名（续）



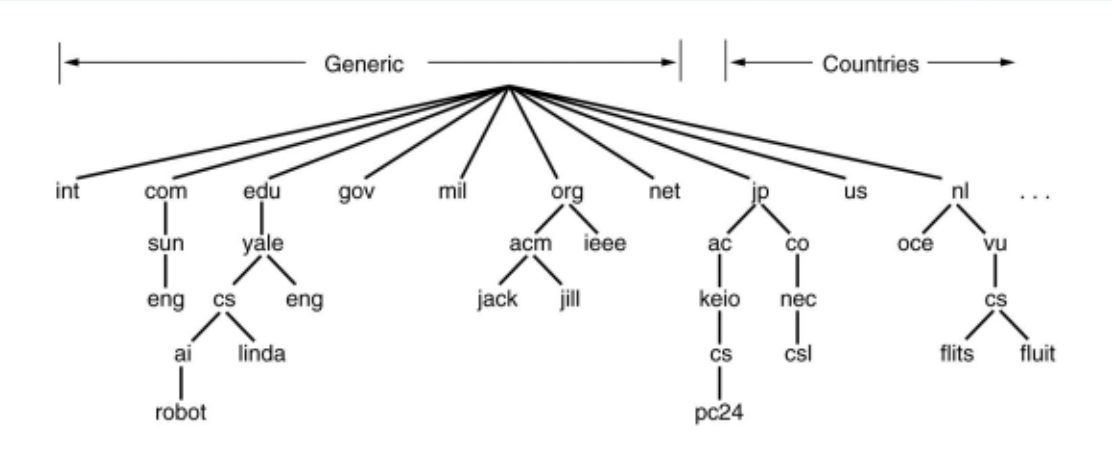
- 树状结构上的每一个非叶节点都对应于一个域 (domain)。
- 树状结构上的每一个叶节点都对应于一个主机名字。
- 一个域可以包含子域或主机名字。

DNS – 域名 (续)



- 树状结构上的每一个节点都具有一个标记，这个标记被称为该节点的 label。
- 兄弟节点不可以具有相同的 label, 但非兄弟节点可以。

DNS – 域名 (续)



- 域的名字或主机的名字被称为域名 (Domain Name)。
- 域名由该节点到根节点路径上的每个节点的 label 连接而成，并且这些 label 之间用 ‘.’ 分开。

  例如：acm.org，robot.ai.cs.yale.edu, 等

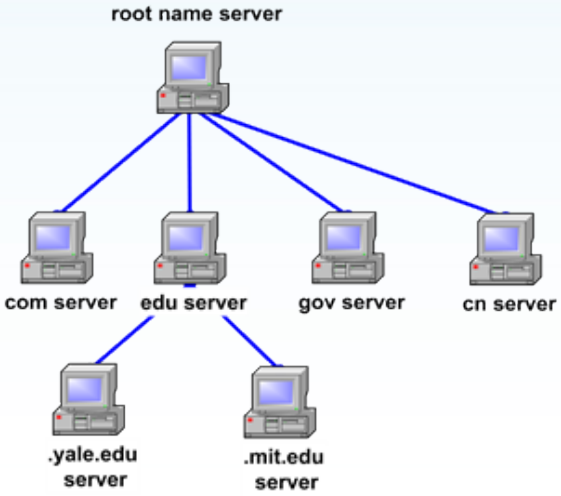
DNS – 域名 (续)

- ICANN (Internet Corporation For Assigned Names and Numbers 互联网名称与数字地址分配机构)具有对顶级域的管理权
  - http://www.icann.org/
  - 2000年，ICANN新增加了四个顶级域：biz (business), info (information), name (people’ s names), pro (professions)

- 每个单位或组织拥有对属于自己的域的管理权。
  - 子域划分
  - 主机或子域命名

DNS – 域名服务器

- DNS由三种域名服务器 (Name Servers)组成：
- Root Name Servers
- TLD Name Servers
- Authoritative Name Servers



DNS – 域名服务器 (续)

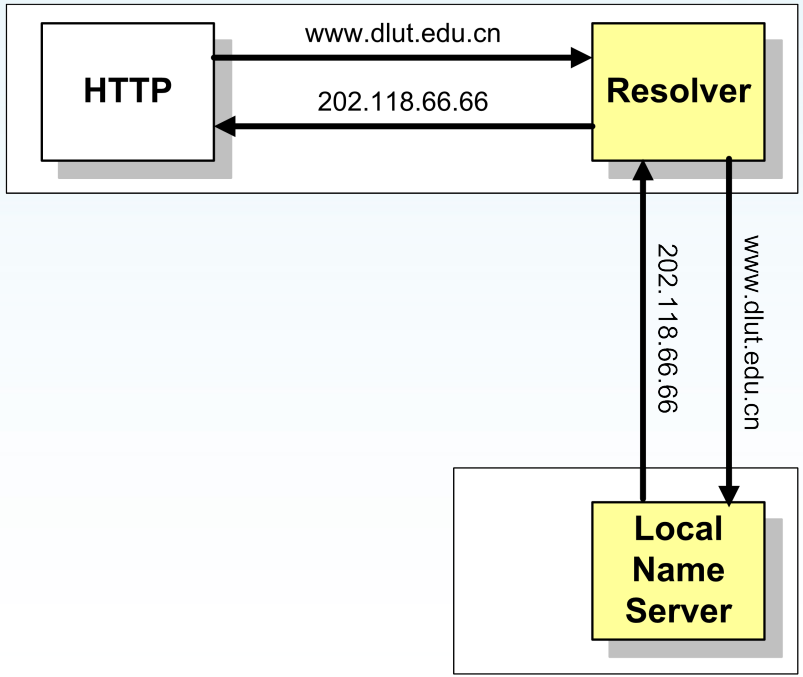
- 拥有域的机构至少要有一个 primary name server 和一个或多个 secondary name servers。
- Primary name server中存储该机构中域名和IP地址对应的最新信息。
- Secondary name server中存储的是primary name server中信息的拷贝。
- 当primary name server不能正常运行时，查询请求会被送到secondary name server。

DNS – 域名解析

- 应用程序通过一个系统函数调用来查询一个主机名字所对应的IP地址，这个系统函数调用被称为“resolver”.
  - 例如：gethostbyname()
- Resovler会把域名解析请求传送给本地域名服务器，并把由本地域名服务器收到的回答返回给应用程序。
- 如果本地域名服务器知道被查询域名所对应的IP地址，那么它就会直接把这个IP地址回答给resovler。否则，它会去查询其它的域名服务器

DNS – 域名解析 (续)

- 举例：


HTTP
www.dlut.edu.cn
202.118.66.66
Resolver
202.118.66.66
www.dlut.edu.cn
Local
Name
Server

DNS – 根域名服务器

- 本地域名服务器在不知道被查询域名所对应的IP地址的情况下，它一般会把这个查询请求传递给根域名服务器 (Root Name Servers)。
- 根域名服务器知道每个顶级域名服务器（例如，edu, com, cn, jp, 等）的地址。
- 从而这个查询请求能够依次向下传递。

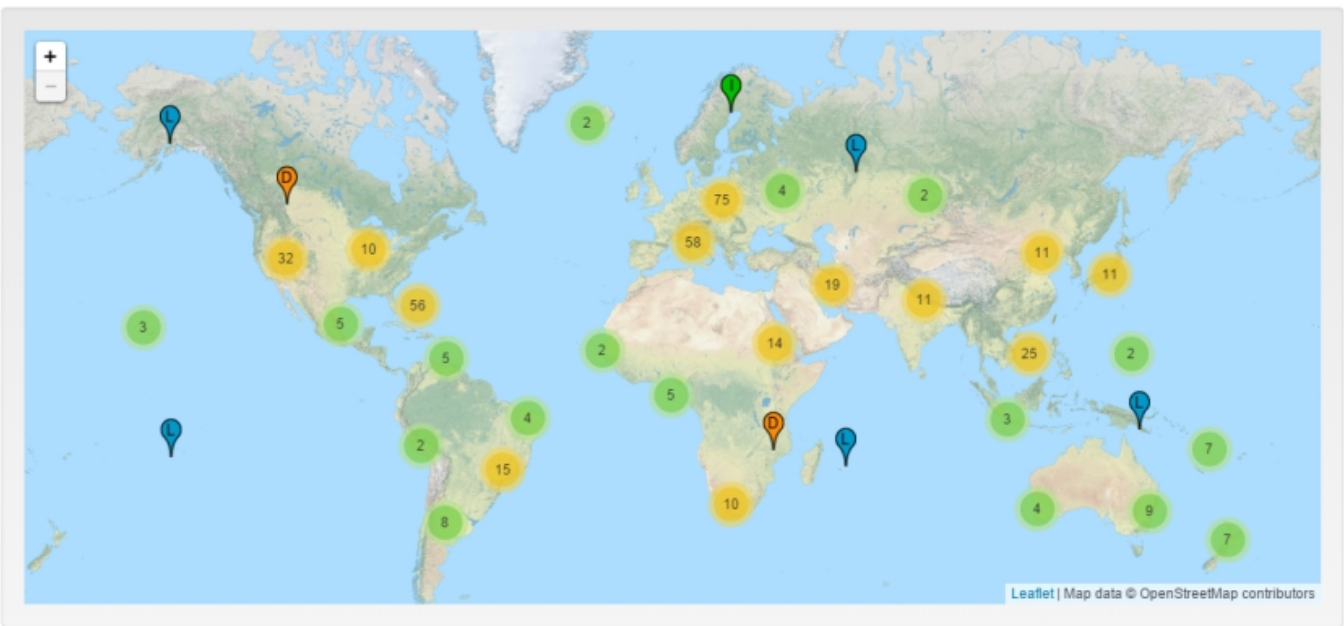
DNS – 根域名服务器 (续)

全球一共有13组根域名服务器（a – m):

察看镜像情况：http://www.root-servers.org/

截止2020年4月，全球共有1091组根域名服务器运行实例（含镜像）

DNS – 全球根域名服务器及其镜像



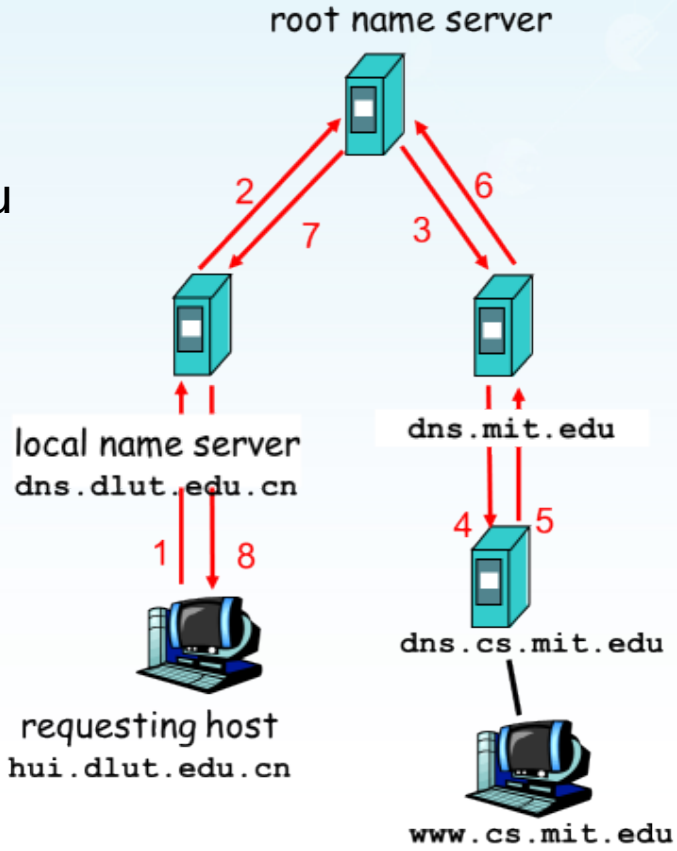
数据来源 http://www.root-servers.org 2020中国大陆目前有13组镜像服务器，香港有7组镜像服务器 ，澳门有2组镜像服务器

DNS – 解析方式

- 域名服务器解析域名的方式一共有两种：
  - Recursive: 被查询域名服务器承担起继续查询的任务
  - Iterative: 被查询域名服务器返回另外一个域名服务器的地址

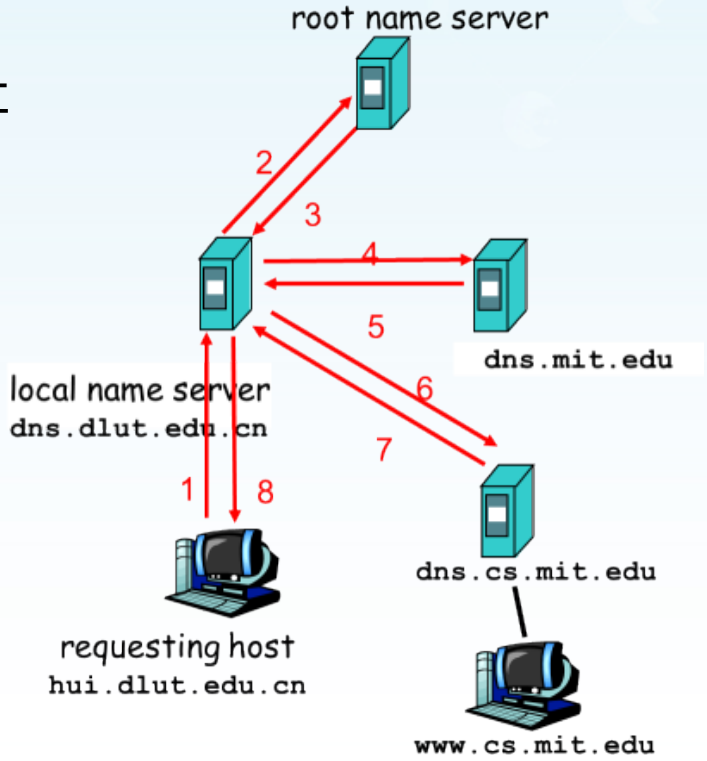
DNS – 解析方式 (续)

- Recursive:
  - 举例：解析地址 www.cs.mit.edu



DNS – 解析方式 (续)

- Iterative:
  - 举例：解析地址 www.cs.mit.edu



DNS – Caching

- 域名服务器把每次解析的结果都暂时存放在缓存 (Cache)里面
- Cache中的结果都具有一定的生存期 (Time-to-live)；如果生存期结束，该结果就被删除。
- 由缓存中返回的结果被称为“non-authoritative answer”
- 由非缓存中（即由负责该域名的服务器中）返回的结果被称为“authoritative answer”

DNS – Resource Record

- 域名服务器中存储域名信息的基本单位被称为Resource Record（RR）.
- 每个Resource Record都由一个五元组构成
  - Domain_name
  - Time_to_live
    - 单位是秒，例如：86400（一天）
  - Class
    - 一般为“IN”，表示Internet
  - Type
  - Value

DNS – Resource Record (续)

- Type的类型主要有以下几种（所对应的Value取值情况也一并列出）：

Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept e-mail
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

DNS – Resource Record 分析

```
; Authoritative data for cs.vu.nl
cs.vu.nl.        86400  IN  SOA     star boss (952771,7200,7200,2419200,86400)
cs.vu.nl.        86400  IN  TXT     "Divisie Wiskunde en Informatica."
cs.vu.nl.        86400  IN  TXT     "Vrije Universiteit Amsterdam."
cs.vu.nl.        86400  IN  MX      1 zephyr.cs.vu.nl.
cs.vu.nl.        86400  IN  MX      2 top.cs.vu.nl.

flits.cs.vu.nl.  86400  IN  HINFO   Sun Unix
flits.cs.vu.nl.  86400  IN  A       130.37.16.112
flits.cs.vu.nl.  86400  IN  A       192.31.231.165
flits.cs.vu.nl.  86400  IN  MX      1 flits.cs.vu.nl.
flits.cs.vu.nl.  86400  IN  MX      2 zephyr.cs.vu.nl.
flits.cs.vu.nl.  86400  IN  MX      3 top.cs.vu.nl.
www.cs.vu.nl.    86400  IN  CNAME   star.cs.vu.nl
ftp.cs.vu.nl.    86400  IN  CNAME   zephyr.cs.vu.nl

rowboat                 IN  A       130.37.56.201
                        IN  MX      1 rowboat
                        IN  MX      2 zephyr
                        IN  HINFO   Sun Unix

little-sister           IN  A       130.37.62.23
                        IN  HINFO   Mac MacOS

laserjet                IN  A       192.31.231.216
                        IN  HINFO   "HP Laserjet IIISi" Proprietary
```

DNS – 消息 (Messages)

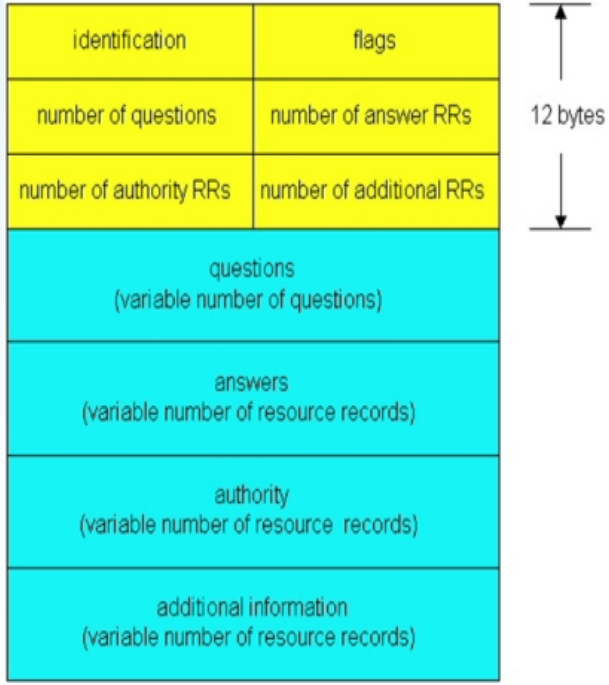
- DNS消息交换一般用UDP来实现，使用UDP端口号53。

- DNS消息一共只有两种类型：
  - DNS Query
  - DNS Reply

DNS – 消息 (续)

头部共有12个字节：

- Identification (2 bytes): 用于匹配 query 和 reply
- Flags (2 bytes): 内容主要包括：
  - query or reply
  - recursion desired
  - reply is authoritative



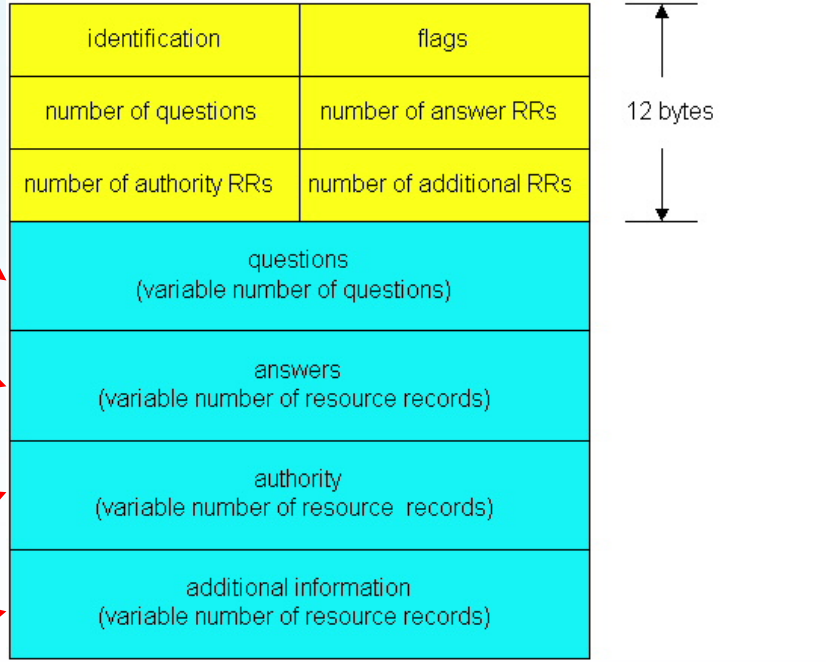
DNS – 消息 (续)

查询请求，包括
Name, type 等域

查询结果的Resource Records

Resource Records for authoritative servers

其它有用的信息



DNS – 总结

- DNS是一个分布式的系统。
  - 避免了 Single Point of Failure Problem
  - 避免了 Overloading Problem
  - 符合对域名进行管理的需要

- DNS提供的域名解析服务是Internet上最重要的服务之一
  - 域名解析几乎是我们使用Internet上任何其它服务的第一步
  - 如果DNS瘫痪，Internet将会怎样？

- 除域名解析外，DNS还起到其它一些作用，例如：
  - 使更换IP地址更加容易
  - 在多个服务器之间作负载平衡

实验注意事项

- 1 认真阅读实验指导书，看清并了解实验要求，严格依据实验指导书操作步骤及注意事项进行操作。
- 2 编写实验指导书问题回答部分时，看清题目，如果要求根据实验现象进行回答，则应该图文并茂，根据实验时的截图进行解释，只有截图或者只有文字都不能得分。截图应该简洁明了，只保留有效部分。