一、三次握手四次挥手

1.1 为什么连接的时候是三次握手，关闭的时候却是四次握手？

答：因为当 Server 端收到 Client 端的 SYN 连接请求报文后，可以直接发送 SYN+ACK 报文。其中 ACK 报文是用来应答的，SYN 报文是用来同步的。但是关闭连接时，当 Server 端收到 FIN 报文时，很可能并不会立即关闭 SOCKET，所以只能先回复一个 ACK 报文，告诉 Client 端，”你发的 FIN 报文我收到了”。只有等到我 Server 端所有的报文都发送完了，我才能发送 FIN 报文，因此不能一起发送。故需要四步握手。

1.2 为什么 TIME_WAIT 状态需要经过 2MSL(最大报文段生存时间)才能返回到 CLOSE 状态？

MSL（MaximumSegment Lifetime），TCP 允许不同的实现可以设置不同的 MSL 值。第一，保证客户端发送的最后一个 ACK 报文能够到达服务器，因为这个 ACK 报文可能丢失，站在服务器的角度看来，我已经发送了 FIN+ACK 报文请求断开了，客户端还没有给我回应，应该是我发送的请求断开报文它没有收到，于是服务器又会重新发送一次，而客户端就能在这个 2MSL 时间段内收到这个重传的报文，接着给出回应报文，并且会重启 2MSL 计时器。第二，防止类似与“三次握手”中提到了的“已经失效的连接请求报文段”出现在本连接中。客户端发送完最后一个确认报文后，在这个 2MSL 时间中，就可以使本连接持续的时间内所产生的所有报文段都从网络中消失。这样新的连接中不会出现旧连接的请求报文。

1.3 为什么建立连接是三次握手，关闭连接确是四次挥手呢？

建立连接的时候，服务器在 LISTEN 状态下，收到建立连接请求的 SYN 报文后，把 ACK 和 SYN 放在一个报文里发送给客户端。而关闭连接时，服务器收到对方的 FIN 报文时，仅仅表示对方不再发送数据了但是还能接收数据，而自己也未必全部数据都发送给对方了，所以己方可以立即关闭，也可以发送一些数据给对方后，再发送 FIN 报文给对方来表示同意现在关闭连接，因此，己方 ACK 和 FIN 一般都会分开发送，从而导致多了一次。

1.4 为什么不能用两次握手进行连接？

答：3 次握手完成两个重要的功能，既要双方做好发送数据的准备工作(双方都知道彼此已准备好)，也要允许双方就初始序列号进行协商，这个序列号在握手过程中被发送和确认。现在把三次握手改成仅需要两次握手，死锁是可能发生的。作为例子，考虑计算机 S 和 C 之间的通信，假定 C 给 S 发送一个连接请求分组，S 收到了这个分组，并发送了确认应答分组。按照两次握手的协定，S 认为连接已经成功地建立了，可以开始发送数据分组。可是，C 在 S 的应答分组在传输中被丢失的情况下，将不知道 S 是否已准备好，不知道 S 建立什么样的序列号，C 甚至怀疑 S 是否收到自己的连接请求分组。在这种情况下，C 认为连接还未建立成功，将忽略 S 发来的任何数据分组，只等待连接确认应答分组。而 S 在发出的分组超时后，重复发送同样的分组。这样就形成了死锁。

1.5 如果已经建立了连接，但是客户端突然出现故障了怎么办？

TCP 还设有一个保活计时器，显然，客户端如果出现故障，服务器不能一直等下去，白白浪费资源。服务器每收到一次客户端的请求后都会重新复位这个计时器，

时间通常是设置为 2 小时，若两小时还没有收到客户端的任何数据，服务器就会发送一个探测报文段，以后每隔 75 分钟发送一次。若一连发送 10 个探测报文仍然没反应，服务器就认为客户端出了故障，接着就关闭连接。

1.6 为什么要三次握手？

保证可靠的核心就是双方都需要确认自己发送和接受信息的功能正常,但因为网络环境的不稳定性,这一秒能收发下一秒可能网络核心就发生严重拥塞,所以世界上不存在完全可靠的通信协议.两次握手会怎样?若建立连接只需两次握手，客户端并没有太大的变化，在获得服务端的应答后进入 ESTABLISHED 状态，即确认自己的发送和接受信息的功能正常.但如果服务端在收到连接请求后就进入 ESTABLISHED 状态,不能保证客户端能收到自己的信息,此时如果网络拥塞，客户端发送的连接请求迟迟到不了服务端，客户端便超时重发请求，如果服务端正确接收并确认应答，双方便开始通信，通信结束后释放连接。此时，如果那个失效的连接请求抵达了服务端，由于只有两次握手，服务端收到请求就会进入 ESTABLISHED 状态，等待发送数据或主动发送数据。但此时的客户端早已进入 CLOSED 状态，服务端将会一直等待下去，这样浪费服务端连接资源。

1.7 为什么要四次挥手？

TCP 连接的释放一共需要四步,因此称为『四次挥手』.我们知道,TCP 连接是双向的,因此在四次挥手中,前两次挥手用于断开一个方向的连接,后两次挥手用于断开另一方向的连接。**第一次挥手:**若 A 认为数据发送完成,则它需要向 B 发送连接释放请求.该请求只有报文头,头中携带的主要参数为:FIN=1,seq=u.此时,A 将进入 FIN-WAIT-1 状态。1，FIN=1 表示该报文段是一个连接释放请求.

2，seq=u,u-1 是 A 向 B 发送的最后一个字节的序号.

第二次挥手:

B 收到连接释放请求后,会通知相应的应用程序,告诉它 A 向 B 这个方向的连接已经释放.此时 B 进入 CLOSE-WAIT 状态,并向 A 发送连接释放的应答,其报文头包含:ACK=1,seq=v,ack=u+1.

ACK=1：除 TCP 连接请求报文段以外,TCP 通信过程中所有数据报的 ACK 都为 1，表示应答.

1，seq=v，v-1 是 B 向 A 发送的最后一个字节的序号.

2，ack=u+1 表示希望收到从第 u+1 个字节开始的报文段,并且已经成功接收了前 u 个字节.A 收到该应答,进入 FIN-WAIT-2 状态,等待 B 发送连接释放请求.

第二次挥手完成后，A 到 B 方向的连接已经释放，B 不会再接收数据，A 也不会再发送数据。但 B 到 A 方向的连接仍然存在，B 可以继续向 A 发送数据。

第三次挥手:

当 B 向 A 发完所有数据后,向 A 发送连接释放请求,请求头中包含:
FIN=1,ACK=1,seq=w,ack=u+1.随后 B 进入 LAST-ACK 状态.

第四次挥手:

A 收到释放请求后,向 B 发送确认应答,此时 A 进入 TIME-WAIT 状态.该状态会持续 2MSL 时间,若该时间段内没有 B 的重发请求的话,就进入 CLOSED 状态,撤销 TCB.当 B 收到确认应答后,也便进入 CLOSED 状态,撤销 TCB。

1.8 为什么 TCP 客户端最后还要发送一次确认呢？

一句话，主要防止已经失效的连接请求报文突然又传送到了服务器，从而产生错误。如果使用的是两次握手建立连接，假设有这样一种场景，客户端发送了第一个请求连接并且没有丢失，只是因为在网络结点中滞留的时间太长了，由于 TCP 的客户端迟迟没有收到确认报文，以为服务器没有收到，此时重新向服务器发送这条报文，此后客户端和服务器经过两次握手完成连接，传输数据，然后关闭连接。此时此前滞留的那一次请求连接，网络通畅了到达了服务器，这个报文本该是失效的，但是，两次握手的机制将会让客户端和服务器再次建立连接，这将导致不必要的错误和资源的浪费。如果采用的是三次握手，就算是那一次失效的报文传送过来了，服务端接受到了那条失效报文并且回复了确认报文，但是客户端不会再次发出确认。由于服务器收不到确认，就知道客户端并没有请求连接。

1.9 简述 TCP 三次握手的过程？

答：在 TCP/IP 协议中，TCP 协议提供可靠的连接服务，采用三次握手建立一个连接。第一次握手：建立连接时，客户端发送 syn 包(syn=j)到服务器，并进入 SYN_SEND 状态，等待服务器确认。第二次握手：服务器收到 syn 包，必须确认客户的 SYN（ack=j+1），同时自己也发送一个 SYN 包（syn=k），即 SYN+ACK 包，此时服务器进入 SYN_RECV 状态。第三次握手：客户端收到服务器的 SYN＋ACK 包，向服务器发送确认包 ACK(ack=k+1)，此包发送完毕，客户端和服务器进入 ESTABLISHED 状态，完成三次握手。完成三次握手，客户端与服务器开始传送数据简版：首先 A 向 B 发 SYN（同步请求），然后 B 回复 SYN+ACK（同步请求应答），最后 A 回复 ACK 确认，这样 TCP 的一次连接（三次握手）的过程就建立了。**三次握手**我们先明确两个定义：

1，client 为数据发送方

2，server 为数据接收方

好，下面进行三次握手的总结：

1，client 想要向 server 发送数据，请求连接。这时 client 想服务器发送一个数据包，其中同步位（SYN）被置为 1，表明 client 申请 TCP 连接，序号为 j。

2，当 server 接收到了来自 client 的数据包时，解析发现同步位为 1，便知道 client 是想要简历 TCP 连接，于是将当前 client 的 IP、端口之类的加入未连接队列中，并向 client 回复接受连接请求，想 client 发送数据包，其中同步位为 1，并附带确认位 ACK=j+1，表明 server 已经准备好分配资源了，并向 client 发起连接请求，请求 client 为建立 TCP 连接而分配资源。

3，client 向 server 回复一个 ACK，并分配资源建立连接。server 收到这个确认时也分配资源进行连接的建立。

那么问题来了为什么需要第三次握手？第三次握手失败了怎么办？三次握手有什么缺陷可以被黑客利用，用来对服务器进行攻击？

怎么防范这种攻击？

接下来进行一一解答。

1.9.1 为什么需要第三次握手？

答：如果没有第三次握手，可能会出现如下情况：如果只有两次握手，那么 server 收到了 client 的 SYN=1 的请求连接数据包之后，便会分配资源并且向 client 发送一个确认位 ACK 回复数据包。那么，如果在 client 与 server 建立连接的过程中，由于网络不顺畅等原因造成的通信链路中存在着残留数据包，即 client 向 server 发送的请求建立连接的数据包由于数据链路的拥塞或者质量不佳导致该连接请求数据包仍然在网络的链路中，这些残留数据包会造成如下危害危害：当 client 与 server 建立连接，数据发送完毕并且关闭 TCP 连接之后，如果链路中的残留数据包才到达 server，那么 server 就会认为 client 重新发送了一次连接申请，便会回复 ACK 包并且分配资源。并且一直等待 client 发送数据，这就会造成 server 的资源浪费。

1.9.2 第三次握手失败了怎么办？

答：当 client 与 server 的第三次握手失败了之后，即 client 发送至 server 的确认建立连接报文段未能到达 server，server 在等待 client 回复 ACK 的过程中超时了，那么 server 会向 client 发送一个 RTS 报文段并进入关闭状态，即：并不等待 client 第三次握手的 ACK 包重传，直接关闭连接请求，这主要是为了防止泛洪攻击，即坏人伪造许多 IP 向 server 发送连接请求，从而将 server 的未连接队列塞满，浪费 server 的资源。

1.9.3 三次握手有什么缺陷可以被黑客利用，用来对服务器进行攻击？

答：黑客仿造 IP 大量的向 server 发送 TCP 连接请求报文包，从而将 server 的半连接队列（上文所说的未连接队列，即 server 收到连接请求 SYN 之后将 client 加入半连接队列中）占满，从而使得 server 拒绝其他正常的连接请求。即拒绝服务攻击

1.9.4 怎么防范这种攻击？

1，缩短服务器接收客户端 SYN 报文之后的等待连接时间，即 SYNtimeout 时间，也就是 server 接收到 SYN 报文段，到最后放弃此连接请求的超时时间，将 SYNtimeout 设置的更低，便可以成倍的减少 server 的负荷，但是过低的 SYNtimeout 可能会影响正常的 TCP 连接的建立，一旦网络不通畅便可能导致 client 连接请求失败 2，SYNcookie + SYN proxy 无缝集成（较好的解决方案）SYNcookie：当 server 接收到 client 的 SYN 之后，不立即分配资源，而是根据 client 发送过来的 SYN 包计算出一个 cookie 值，这个 cookie 值用来存储 server 返回给 client 的 SYN+ACK 数据包中的初始序列号，当 client 返回第三次握手的 ACK 包之后进行校验，如果校验成功则 server 分配资源，建立连接。SYNproxy 代理，作为 server 与 client 连接的代理，代替 server 与 client 建立三次握手的连接，同时 SYNproxy 与 client 建立好了三次握手连接之后，确保是正常的 TCP 连接，而不是 TCP 泛洪攻击，那么 SYNproxy 就与 server 建立三次握手连接，作为代理（网关？）来连通 client 与 server。（类似 VPN 了解一下。）

二、路由

2.1 填空题。

1. 静态 路由设定后，若 网络 拓扑结构发生变化，需由系统 管理员 修改路由的 设置 。

2.网络管理的重要任务是： 控制 和 监控 。

3.在安装 Linux 系统中，使用 netconfig 程序对网络进行配置，该安装程序会一步步提示用户输入 主机名 、 域名 、 域名服务器 、 IP 地址 、 网关地址 和 子网掩码 等必要信息。

4. RIP 协议是最为普遍的一种内部协议，一般称为动态路由信息协议。

5. DHCP 可以实现动态 IP 地址分配。

6.网络管理通常由 监测 、 传输 和 管理 三部分组成，其中 管理部分 是整个网络管理的中心。

7. Ping 命令可以测试网络中本机系统是否能到达一台远程主机，所以常常用于 测试网络的连通性 。

8.进行远程登录的命令是 telnet 。

9.DHCP 是动态主机配置协议的简称，其作用是： 为网络中的主机分配 IP 地址。

10.路由选择协议（RIP）的跳数表示到达目的地之前必须通过的网关数，RIP 接受的最长距离是 15 跳。

11.ping 命令用于测试网络的连通性，ping 命令通过 ICMP 协议（internet 控制信息协议）来实现。

2.2 选择题。

12.下面的网络协议中，面向连接的的协议是： A 。

A 传输控制协议

B 用户数据报协议

C 网际协议

D 网际控制报文协议

13.一台主机要实现通过局域网与另一个局域网通信，需要做的工作是 C 。

A 配置域名服务器

B 定义一条本机指向所在网络的路由

C 定义一条本机指向所在网络网关的路由

D 定义一条本机指向目标网络网关的路由

14.局域网的网络地址 192.168.1.0/24，局域网络连接其它网络的网关地址是 192.168.1.1。主机 192.168.1.20 访问 172.16.1.0/24 网络时，其路由设置正确的是 B 。

A route add –net 192.168.1.0 gw 192.168.1.1 netmask 255.255.255.0metric1

B route add –net 172.16.1.0 gw 192.168.1.1 netmask 255.255.255.255metric1

C route add –net 172.16.1.0 gw 172.16.1.1 netmask 255.255.255.0metric 1

D route add default 192.168.1.0 netmask 172.168.1.1 metric 1

15.下列提法中，不属于 ifconfig 命令作用范围的是 D 。

A 配置本地回环地址

B 配置网卡的 IP 地址

C 激活网络适配器

D 加载网卡到内核中

16.在局域网络内的某台主机用 **ping** 命令测试网络连接时发现网络内部的主机都可以连同，而不能与公网连通，问题可能是 **C** 。

A 主机 IP 设置有误

B 没有设置连接局域网的网关

C 局域网的网关或主机的网关设置有误

D 局域网 DNS 服务器设置有误

17.下列文件中，包含了主机名到 **IP** 地址的映射关系的文件是： **B** 。

A /etc/HOSTNAME

B /etc/hosts

C /etc/resolv.conf

D /etc/networks

18.在 **TCP/IP** 模型中，应用层包含了所有的高层协议，在下列的一些应用协议中， **B** 是能够实现本地与远程主机之间的文件传输工作。

A telnet

B FTP

C SNMP

D NFS

19.当我们与某远程网络连接不上时，就需要跟踪路由查看，以便了解在网络的什么位置出现了问题，满足该目的的命令是 **C** 。

A ping

B ifconfig

C traceroute

D netstat

20.**DNS** 域名系统主要负责主机名和 **A** 之间的解析。

A IP 地址

B MAC 地址

C 网络地址

D 主机别名

21.**WWW** 服务器是在 **Internet** 上使用最为广泛，它采用的是 **B** 结构。

A 服务器/工作站

B B/S

C 集中式

D 分布式

22.网络管理具备以下几大功能：配置管理、 **A** 、性能管理、安全管理和计费管理等。

A 故障管理

B 日常备份管理

C 升级管理

D 发送邮件

23.关于代理服务器的论述，正确的是 **A** 。

A 使用 internet 上已有的公开代理服务器，只需配置客户端。

B 代理服务器只能代理客户端 http 的请求。

C 设置好的代理服务器可以被网络上任何主机使用。

D 使用代理服务器的客户端没有自己的 ip 地址。

24.实现从 **IP** 地址到以太网 **MAC** 地址转换的命令为： **C** 。

A ping

B ifconfig

C arp

D traceroute

25.在 **DNS** 系统测试时，设 **named** 进程号是 **53**，命令 **D** 通知进程重读配置文件。

A kill –USR2 53

B kill –USR1 53

C kill -INT 63

D kill –HUP 53

26.在 **DNS** 配置文件中，用于表示某主机别名的是： **B** 。

A NS

B CNAME

C NAME

D CN

27.为保证在启动服务器时自动启动 **DHCP** 进程，应对 **B** 文件进行编辑。

A /etc/rc.d/rc.inet2

B /etc/rc.d/rc.inet1

C /etc/dhcpd.conf

D /etc/rc.d/rc.S

2.3 简答题。

28. 写一条 **192.168.10.0** 网段从网关 **192.168.9.1** 出去的路由

答：routeadd -net 192.168.10.0/24gw 192.168.9.1

29.给主机 **host：172.16.0.2** 增加 **gateway10.0.0.1**

答：routeadd 172.16.0.2 gw 10.0.0.1 或者网卡配置文件更改

30.网站出现 **500,502,400,403,404** 都是什么意思，怎么排查和解决

答：500：服务器内部错误，因为服务器上的程序写的有问题，需要打开错误日志，查看日志，分析错误信息。502：网关错误，服务器作为网关或代理，从上游服务器收到无效响应。Nginx 出现最多，出现 502 要么是 nginx 配置的不对，要么是 php-fpm 资源不够，可以分析 php-fpm 的慢执行日志，优化 php-fpm 的执行速度。400：错误请求，服务器不理解请求的语法。这可能是用户发起的请求不合理，需要检查客户端的请求。

403：服务器拒绝请求。检查服务器配置，是不是对客户端做了限制。

404：未找到请求的资源。检查服务器上是否存在请求的资源，看是否是配置问题。

32.简述 **DNS** 进行域名解析的过程。

参考答案：首先，客户端发出 DNS 请求翻译 IP 地址或主机名。DNS 服务器在收到客户机的请求后：（1）检查 DNS 服务器的缓存，若查到请求的地址或名字，即向客户机发出应答信息；（2）若没有查到，则在数据库中查找，若查到请求的地址或名字，即向客户机发出应答信息；

（3）若没有查到，则将请求发给根域 DNS 服务器，并依序从根域查找顶级域，由顶级查找二级域，二级域查找三级，直至找到要解析的地址或名字，即向客户机所在网络的 DNS 服务器发出应答信息，DNS 服务器收到应答后现在缓存中存储，然后，将解析结果发给客户机。

（4）若没有找到，则返回错误信息。

33.什么是静态路由，其特点是什么？什么是动态路由，其特点是什么？

参考答案：静态路由是由系统管理员设计与构建的路由表规定的路由。适用于网关数量有限的场合，且网络拓朴结构不经常变化的网络。其缺点是不能动态地适用网络状况的变化，当网络状况变化后必须由网络管理员修改路由表。动态路由是由路由选择协议而动态构建的，路由协议之间通过交换各自所拥有的路由信息实时更新路由表的内容。动态路由可以自动学习 网络的拓朴结构，并更新路由表。其缺点是路由广播更新信息将占据大量的网络带宽。

34.**linux** 下常用的 **DNS** 服务软件是什么，举出几种常用的 **DNS** 记录，如果域名 **abc.com** 配置好了一台邮件服务器,**IP** 地址为 **202.106.0.20**，我该如何做相关的解析？是否了解 **bind** 的智能解析，如果了解请简述一下其原理。

答案：1)常用的 DNS 软件是 bind2)A 记录地址记录 MX 记录邮件交换记录

CNAME 记录别名域记录

3)修改 abc.com 域名的配置文件，增加以下记录

INMX 10 mail.abc.com.

mailIN A 202.106.0.20

4)bind 根据请求解析客户端的 IP 地址，做出不同的解析，其原理是在配置文件中，设定了 view，在每个 view 都有客户端的 IP 地址段，bind 服务器根据请求解析客户端的 IP 地址，匹配不同的 view,再根据该 view 的配置，到相应的配置文件进行查询，将结果返回给请求的客户端。

35.**AB 网络是通的，最少列出五种传输文件的服务**

nfs，ftp，scp，rsync，samba，http://

36.我们都知道，dns 既采用了 tcp 协议，又采用了 udp 协议，什么时候采用 tcp 协议？什么时候采用 udp 协议？为什么要这么设计？

答：这个题需要理解的东西比较的多，分一下几个方面 a，从数据包大小上分：UDP 的最大包长度是 65507 个字节，响应 dns 查询的时候数据包长度超过 512 个字节，而返回的只要前 512 个字节，这时名字解释器通常使用 TCP 从发原来的请求。b，从协议本身来分：大部分的情况下使用 UDP 协议，大家都知道 UDP 协议是一种不可靠的协议，dns 不像其它的使用 UDP 的 Internet 应用(如：TFTP，BOOTP 和 SNMP 等)，大部分集中在局域网，dns 查询和响应需要经过广域网，分组丢失和往返时间的不确定性在广域网比局域网上更大，这就要求 dns 客户端需要好的重传和超时算法，这时候使用 TCP。

三、七层模型

3.1 说说 TCP/IP 的七层模型

应用层 (Application)：网络服务与最终用户的一个接口。协议有：HTTP FTP TFTP SMTP SNMP DNS TELNET HTTPS POP3 DHCP 表示层（Presentation Layer）：

数据的表示、安全、压缩。（在五层模型里面已经合并到了应用层）

格式有，JPEG、ASCll、DECOIC、加密格式等

会话层（Session Layer）：

建立、管理、终止会话。（在五层模型里面已经合并到了应用层）

对应主机进程，指本地主机与远程主机正在进行的会话

传输层 (Transport)：

定义传输数据的协议端口号，以及流控和差错校验。

协议有：TCP UDP，数据包一旦离开网卡即进入网络传输层

网络层 (Network)：

进行逻辑地址寻址，实现不同网络之间的路径选择。

协议有：ICMP IGMP IP（IPV4 IPV6） ARP RARP

数据链路层 (Link)：

建立逻辑连接、进行硬件地址寻址、差错校验等功能。（由底层网络定义协议）

将比特组合成字节进而组合成帧，用 MAC 地址访问介质，错误发现但不能纠正。

物理层（Physical Layer）：

是计算机网络 OSI 模型中最低的一层。

物理层规定：为传输数据所需要的物理链路创建、维持、拆除而提供具有机械的，电子的，功能的和规范的特性。

简单的说，物理层确保原始的数据可在各种物理媒体上传输。局域网与广域网皆属第 1、2 层。物理层是 OSI 的第一层，它虽然处于最底层，却是整个开放系统的基础。物理层为设备之间的数据通信提供传输媒体及互连设备，为数据传输提供可靠的环境。如果您想要用尽量少的词来记住这个第一层，那就是“信号和介质”。