1、什么是运维？什么是游戏运维？

1）运维是指大型组织已经建立好的网络软硬件的维护，就是要保证业务的上线与运作的正常，在他运转的过程中，对他进行维护，他集合了网络、系统、数据库、开发、安全、监控于一身的技术

运维又包括很多种，有 DBA 运维、网站运维、虚拟化运维、监控运维、游戏运维等等

2）游戏运维又有分工，分为开发运维、应用运维（业务运维）和系统运维

开发运维：是给应用运维开发运维工具和运维平台的

应用运维：是给业务上线、维护和做故障排除的，用开发运维开发出来的工具给业务上线、维护、做故障排查

系统运维：是给应用运维提供业务上的基础设施，比如：系统、网络、监控、硬件等等

总结：开发运维和系统运维给应用运维提供了"工具"和"基础设施"上的支撑

开发运维、应用运维和系统运维他们的工作是环环相扣的

2、在工作中，运维人员经常需要跟运营人员打交道，请问运营人员是做什么工作的？

游戏运营要做的一个事情除了协调工作以外

还需要与各平台沟通，做好开服的时间、开服数、用户导量、活动等计划

3、现在给你三百台服务器，你怎么对他们进行管理？

管理 3 百台服务器的方式：

1）设定跳板机，使用统一账号登录，便于安全与登录的考量。

2）使用 salt、ansiable、puppet 进行系统的统一调度与配置的统一管理。

3）建立简单的服务器的系统、配置、应用的 cmdb 信息管理。便于查阅每台服务器上的各种信息记录。

4、简述 raid0 raid1 raid5 三种工作模式的工作原理及特点

RAID，可以把硬盘整合成一个大磁盘，还可以在大磁盘上再分区，放数据

还有一个大功能，多块盘放在一起可以有冗余（备份）

RAID 整合方式有很多，常用的：0 1 5 10

RAID 0，可以是一块盘和 N 个盘组合

其优点读写快，是 RAID 中最好的

缺点：没有冗余，一块坏了数据就全没有了

RAID 1，只能 2 块盘，盘的大小可以不一样，以小的为准

10G+10G 只有 10G，另一个做备份。它有 100%的冗余，缺点：浪费资源，成本高

RAID 5 ，3 块盘，容量计算 10*（n-1）,损失一块盘

特点，读写性能一般，读还好一点，写不好

冗余从好到坏：RAID1 RAID10 RAID 5 RAID0

性能从好到坏：RAID0 RAID10 RAID5 RAID1

成本从低到高：RAID0 RAID5 RAID1 RAID10

单台服务器：很重要盘不多，系统盘，RAID1

数据库服务器：主库：RAID10 从库 RAID5RAID0（为了维护成本，RAID10）

WEB 服务器，如果没有太多的数据的话，RAID5,RAID0（单盘）

有多台，监控、应用服务器，RAID0 RAID5

我们会根据数据的存储和访问的需求，去匹配对应的 RAID 级别

5、LVS、Nginx、HAproxy 有什么区别？工作中你怎么选择？

LVS： 是基于四层的转发

HAproxy： 是基于四层和七层的转发，是专业的代理服务器

Nginx： 是 WEB 服务器，缓存服务器，又是反向代理服务器，可以做七层的转发

区别： LVS 由于是基于四层的转发所以只能做端口的转发

而基于 URL 的、基于目录的这种转发 LVS 就做不了

工作选择：

HAproxy 和 Nginx 由于可以做七层的转发，所以 URL 和目录的转发都可以做

在很大并发量的时候我们就要选择 LVS，像中小型公司的话并发量没那么大

选择 HAproxy 或者 Nginx 足已，由于 HAproxy 由是专业的代理服务器

配置简单，所以中小型企业推荐使用 HAproxy

6、Squid、Varinsh 和 Nginx 有什么区别，工作中你怎么选择？

Squid、Varinsh 和 Nginx 都是代理服务器

什么是代理服务器：

能当替用户去访问公网，并且能把访问到的数据缓存到服务器本地，等用户下次再访问相同的资

源的时候，代理服务器直接从本地回应给用户，当本地没有的时候，我代替你去访问公网，我接

收你的请求，我先在我自已的本地缓存找，如果我本地缓存有，我直接从我本地的缓存里回复你

如果我在我本地没有找到你要访问的缓存的数据，那么代理服务器就会代替你去访问公网

区别：

1）Nginx 本来是反向代理/web 服务器，用了插件可以做做这个副业

但是本身不支持特性挺多，只能缓存静态文件

2）从这些功能上。varnish 和 squid 是专业的 cache 服务，而 nginx 这些是第三方模块完成

3）varnish 本身的技术上优势要高于 squid，它采用了可视化页面缓存技术

在内存的利用上，Varnish 比 Squid 具有优势，性能要比 Squid 高。

还有强大的通过 Varnish 管理端口，可以使用正则表达式快速、批量地清除部分缓存

它是内存缓存，速度一流，但是内存缓存也限制了其容量，缓存页面和图片一般是挺好的

4）squid 的优势在于完整的庞大的 cache 技术资料，和很多的应用生产环境

工作中选择：

要做 cache 服务的话，我们肯定是要选择专业的 cache 服务，优先选择 squid 或者 varnish。

7、Tomcat 和 Resin 有什么区别，工作中你怎么选择？

区别：Tomcat 用户数多，可参考文档多，Resin 用户数少，可考虑文档少

最主要区别则是 Tomcat 是标准的 java 容器，不过性能方面比 resin 的要差一些
但稳定性和 java 程序的兼容性，应该是比 resin 的要好

工作中选择：现在大公司都是用 resin，追求性能；而中小型公司都是用 Tomcat，追求稳定和程序的兼容

8、什么是中间件？什么是 jdk?

中间件介绍：
中间件是一种独立的系统软件或服务程序，分布式应用软件借助这种软件在不同的技术之间共享资源
中间件位于客户机/ 服务器的操作系统之上，管理计算机资源和网络通讯
是连接两个独立应用程序或独立系统的软件。相连接的系统，即使它们具有不同的接口

但通过中间件相互之间仍能交换信息。执行中间件的一个关键途径是信息传递
通过中间件，应用程序可以工作于多平台或 OS 环境。

jdk：jdk 是 Java 的开发工具包
它是一种用于构建在 Java 平台上发布的应用程序、applet 和组件的开发环境

9、讲述一下 Tomcat8005、8009、8080 三个端口的含义？

8005==》 关闭时使用
8009==》 为 AJP 端口，即容器使用，如 Apache 能通过 AJP 协议访问 Tomcat 的 8009 端口
8080==》 一般应用使用

10、什么叫 CDN?

- 即内容分发网络
- 其目的是通过在现有的 Internet 中增加一层新的网络架构，将网站的内容发布到
最接近用户的网络边缘，使用户可就近取得所需的内容，提高用户访问网站的速度

11、什么叫网站灰度发布？

灰度发布是指在黑与白之间，能够平滑过渡的一种发布方式
AB test 就是一种灰度发布方式，让一部用户继续用 A，一部分用户开始用 B
如果用户对 B 没有什么反对意见，那么逐步扩大范围，把所有用户都迁移到 B 上面 来

灰度发布可以保证整体系统的稳定，在初始灰度的时候就可以发现、调整问题，以保证其影响度

12、简述 DNS 进行域名解析的过程?

用户要访问 www.baidu.com，会先找本机的 host 文件，再找本地设置的 DNS 服务器，如果也没有的话，就去网络中找根服务器，根服务器反馈结果，说只能提供一级域名服务器.cn，就去找一级域名服务器，一级域名服务器说只能提供二级域名服务器.com.cn,就去找二级域名服务器，二级域服务器只能提供三级域名服务器.baidu.com.cn，就去找三级域名服务器，三级域名服务器正好有这个网站 www.baidu.com，然后发给请求的服务器，保存一份之后，再发给客户端

13、RabbitMQ 是什么东西?

RabbitMQ 也就是消息队列中间件，消息中间件是在消息的传息过程中保存消息的容器
消息中间件再将消息从它的源中到它的目标中标时充当中间人的作用
队列的主要目的是提供路由并保证消息的传递；如果发送消息时接收者不可用
消息队列不会保留消息，直到可以成功地传递为止，当然，消息队列保存消息也是有期限地

14、讲一下 Keepalived 的工作原理?

在一个虚拟路由器中，只有作为 MASTER 的 VRRP 路由器会一直发送 VRRP 通告信息,
BACKUP 不会抢占 MASTER，除非它的优先级更高。当 MASTER 不可用时(BACKUP 收不到通告信息)
多台 BACKUP 中优先级最高的这台会被抢占为 MASTER。这种抢占是非常快速的(<1s)，以保证服务的连续性
由于安全性考虑，VRRP 包使用了加密协议进行加密。BACKUP 不会发送通告信息，只会接收通告信息

15、讲述一下 LVS 三种模式的工作过程?

LVS 有三种负载均衡的模式，分别是 VS/NAT（nat 模式） VS/DR(路由模式) VS/TUN（隧道模式）

一、NAT 模式（VS-NAT）

原理：就是把客户端发来的数据包的 IP 头的目的地址，在负载均衡器上换成其中一台 RS 的 IP 地址

并发至此 RS 来处理,RS 处理完后把数据交给负载均衡器,负载均衡器再把数据包原 IP 地址改为自己的 IP

将目的地址改为客户端 IP 地址即可期间,无论是进来的流量,还是出去的流量,都必须经过负载均衡器

优点：集群中的物理服务器可以使用任何支持 TCP/IP 操作系统，只有负载均衡器需要一个合法的 IP 地址

缺点：扩展性有限。当服务器节点（普通 PC 服务器）增长过多时,负载均衡器将成为整个系统的瓶颈

因为所有的请求包和应答包的流向都经过负载均衡器。当服务器节点过多时

大量的数据包都交汇在负载均衡器那，速度就会变慢！

二、IP 隧道模式（VS-TUN）

原理：首先要知道，互联网上的大多 Internet 服务的请求包很短小，而应答包通常很大

那么隧道模式就是，把客户端发来的数据包，封装一个新的 IP 头标记(仅目的 IP)发给 RS

RS 收到后,先把数据包的头解开,还原数据包,处理后,直接返回给客户端,不需要再经过

负载均衡器。注意,由于 RS 需要对负载均衡器发过来的数据包进行还原,所以说必须支持

IPTUNNEL 协议，所以,在 RS 的内核中,必须编译支持 IPTUNNEL 这个选项

优点：负载均衡器只负责将请求包分发给后端节点服务器，而 RS 将应答包直接发给用户

所以，减少了负载均衡器的大量数据流动，负载均衡器不再是系统的瓶颈，就能处理很巨大的请求量

这种方式，一台负载均衡器能够为很多 RS 进行分发。而且跑在公网上就能进行不同地域的分发。

缺点：隧道模式的 RS 节点需要合法 IP，这种方式需要所有的服务器支持"IP Tunneling"

(IP Encapsulation)协议，服务器可能只局限在部分 Linux 系统上

三、直接路由模式（VS-DR）

原理：负载均衡器和 RS 都使用同一个 IP 对外服务但只有 DR 对 ARP 请求进行响应

所有 RS 对本身这个 IP 的 ARP 请求保持静默也就是说,网关会把对这个服务 IP 的请求全部定向给 DR

而 DR 收到数据包后根据调度算法,找出对应的 RS,把目的 MAC 地址改为 RS 的 MAC（因为 IP 一致）

并将请求分发给这台 RS 这时 RS 收到这个数据包,处理完成之后，由于 IP 一致，可以直接将数据返给客户

则等于直接从客户端收到这个数据包无异,处理后直接返回给客户端

由于负载均衡器要对二层包头进行改换,所以负载均衡器和 RS 之间必须在一个广播域

也可以简单的理解为在同一台交换机上

优点：和 TUN（隧道模式）一样，负载均衡器也只是分发请求，应答包通过单独的路由方法返回给客户端

与 VS-TUN 相比，VS-DR 这种实现方式不需要隧道结构，因此可以使用大多数操作系统做为物理服务器。

缺点：（不能说缺点，只能说是不足）要求负载均衡器的网卡必须与物理网卡在一个物理段上。

16、mysql 的 innodb 如何定位锁问题，mysql 如何减少主从复制延迟?

mysql 的 innodb 如何定位锁问题:

在使用 show engine innodb status 检查引擎状态时，发现了死锁问题

在 5.5 中，information_schema 库中增加了三个关于锁的表（MEMORY 引擎）

innodb_trx ## 当前运行的所有事务

innodb_locks ## 当前出现的锁

innodb_lock_waits ## 锁等待的对应关系

mysql 如何减少主从复制延迟:

如果延迟比较大，就先确认以下几个因素：

1. 从库硬件比主库差，导致复制延迟
2. 主从复制单线程，如果主库写并发太大，来不及传送到从库

   就会导致延迟。更高版本的 mysql 可以支持多线程复制
3. 慢 SQL 语句过多
4. 网络延迟
5. master 负载

主库读写压力大，导致复制延迟，架构的前端要加 buffer 及缓存层

6. slave 负载

一般的做法是，使用多台 slave 来分摊读请求，再从这些 slave 中取一台专用的服务器

只作为备份用，不进行其他任何操作.另外， 2 个可以减少延迟的参数:

–slave-net-timeout=seconds 单位为秒 默认设置为 3600 秒

#参数含义：当 slave 从主数据库读取 log 数据失败后，等待多久重新建立连接并获取数据

–master-connect-retry=seconds 单位为秒 默认设置为 60 秒

#参数含义：当重新建立主从连接时，如果连接建立失败，间隔多久后重试

通常配置以上 2 个参数可以减少网络问题导致的主从数据同步延迟

MySQL 数据库主从同步延迟解决方案

最简单的减少 slave 同步延时的方案就是在架构上做优化，尽量让主库的 DDL 快速执行

还有就是主库是写，对数据安全性较高，比如 sync_binlog=1 ，innodb_flush_log_at_trx_commit

= 1 之类的设置，而 slave 则不需要这么高的数据安全，完全可以讲 sync_binlog 设置为 0 或者关闭 binlog

innodb_flushlog 也可以设置为 0 来提高 sql 的执行效率。另外就是使用比主库更好的硬件设备作为 slave

17、如何重置 mysql root 密码?

一、 在已知 MYSQL 数据库的 ROOT 用户密码的情况下，修改密码的方法：

1、 在 SHELL 环境下，使用 mysqladmin 命令设置：

mysqladmin –u root –p password “新密码” 回车后要求输入旧密码

2、 在 mysql>环境中,使用 update 命令，直接更新 mysql 库 user 表的数据：

Update mysql.user set password=password(‘新密码’) where user=’root’;

flush privileges;

注意：mysql 语句要以分号”；”结束

3、 在 mysql>环境中，使用 grant 命令，修改 root 用户的授权权限。

grant all on *.* to root@'localhost' identified by '新密码';

二、 如查忘记了 mysql 数据库的 ROOT 用户的密码，又如何做呢？方法如下：

1、 关闭当前运行的 mysqld 服务程序：service mysqld stop（要先将 mysqld 添加为系统服务）

2、 使用 mysqld_safe 脚本以安全模式（不加载授权表）启动 mysqld 服务

/usr/local/mysql/bin/mysqld_safe --skip-grant-table &

3、 使用空密码的 root 用户登录数据库，重新设置 ROOT 用户的密码

mysql -u root

Mysql> Update mysql.user set password=password('新密码') where user='root';

Mysql> flush privileges;

18、lvs/nginx/haproxy 优缺点

Nginx 的优点是：

1、工作在网络的 7 层之上，可以针对 http 应用做一些分流的策略，比如针对域名、目录结构它的正则规则比 HAProxy 更为强大和灵活，这也是它目前广泛流行的主要原因之一

Nginx 单凭这点可利用的场合就远多于 LVS 了。

2、Nginx 对网络稳定性的依赖非常小，理论上能 ping 通就就能进行负载功能，这个也是它的优势之一

相反 LVS 对网络稳定性依赖比较大，这点本人深有体会；

3、Nginx 安装和配置比较简单，测试起来比较方便，它基本能把错误用日志打印出来

LVS 的配置、测试就要花比较长的时间了，LVS 对网络依赖比较大。

4、可以承担高负载压力且稳定，在硬件不差的情况下一般能支撑几万次的并发量，负载度比 LVS 相对小些。

5、Nginx 可以通过端口检测到服务器内部的故障，比如根据服务器处理网页返回的状态码、超时等等，并且会把返回错误的请求重新提交到另一个节点，不过其中缺点就是不支持 url 来检测。比如用户正在上传一个文件，而处理该上传的节点刚好在上传过程中出现故障，Nginx 会把上传切到另一台服务器重新处理，而 LVS 就直接断掉了
如果是上传一个很大的文件或者很重要的文件的话，用户可能会因此而不满。

6、Nginx 不仅仅是一款优秀的负载均衡器/反向代理软件，它同时也是功能强大的 Web 应用服务器
LNMP 也是近几年非常流行的 web 架构，在高流量的环境中稳定性也很好。

7、Nginx 现在作为 Web 反向加速缓存越来越成熟了，速度比传统的 Squid 服务器更快，可考虑用其作为反向代理加速器

8、Nginx 可作为中层反向代理使用，这一层面 Nginx 基本上无对手，唯一可以对比 Nginx 的就只有 lighttpd 了
不过 lighttpd 目前还没有做到 Nginx 完全的功能，配置也不那么清晰易读，社区资料也远远没 Nginx 活跃

9、Nginx 也可作为静态网页和图片服务器，这方面的性能也无对手。还有 Nginx 社区非常活跃，第三方模块也很多

Nginx 的缺点是：
1、Nginx 仅能支持 http、https 和 Email 协议，这样就在适用范围上面小些，这个是它的缺点
2、对后端服务器的健康检查，只支持通过端口来检测，不支持通过 url 来检测
不支持 Session 的直接保持，但能通过 ip_hash 来解决

LVS：使用 Linux 内核集群实现一个高性能、高可用的负载均衡服务器
它具有很好的可伸缩性(Scalability)、可靠性(Reliability)和可管理性(Manageability)

LVS 的优点是：

1、抗负载能力强、是工作在网络 4 层之上仅作分发之用，没有流量的产生

这个特点也决定了它在负载均衡软件里的性能最强的，对内存和 cpu 资源消耗比较低

2、配置性比较低，这是一个缺点也是一个优点，因为没有可太多配置的东西

所以并不需要太多接触，大大减少了人为出错的几率

3、工作稳定，因为其本身抗负载能力很强，自身有完整的双机热备方案

如 LVS+Keepalived，不过我们在项目实施中用得最多的还是 LVS/DR+Keepalived

4、无流量，LVS 只分发请求，而流量并不从它本身出去，这点保证了均衡器 IO 的性能不会收到大流量的影响。

5、应用范围较广，因为 LVS 工作在 4 层，所以它几乎可对所有应用做负载均衡，包括 http、数据库、在线聊天室等

LVS 的缺点是：

1、软件本身不支持正则表达式处理，不能做动静分离

而现在许多网站在这方面都有较强的需求，这个是 Nginx/HAProxy+Keepalived 的优势所在

2、如果是网站应用比较庞大的话，LVS/DR+Keepalived 实施起来就比较复杂了

特别后面有 Windows Server 的机器的话，如果实施及配置还有维护过程就比较复杂了

相对而言，Nginx/HAProxy+Keepalived 就简单多了。

HAProxy 的特点是：

1、HAProxy 也是支持虚拟主机的。

2、HAProxy 的优点能够补充 Nginx 的一些缺点，比如支持 Session 的保持，Cookie 的引导

同时支持通过获取指定的 url 来检测后端服务器的状态

3、HAProxy 跟 LVS 类似，本身就只是一款负载均衡软件

单纯从效率上来讲 HAProxy 会比 Nginx 有更出色的负载均衡速度，在并发处理上也是优于 Nginx 的

4、HAProxy 支持 TCP 协议的负载均衡转发，可以对 MySQL 读进行负载均衡

对后端的 MySQL 节点进行检测和负载均衡，大家可以用 LVS+Keepalived 对 MySQL 主从做负载均衡

5、HAProxy 负载均衡策略非常多，HAProxy 的负载均衡算法现在具体有如下 8 种：

①roundrobin，表示简单的轮询，这个不多说，这个是负载均衡基本都具备的；

② static-rr，表示根据权重，建议关注；

③leastconn，表示最少连接者先处理，建议关注；

④ source，表示根据请求源 IP，这个跟 Nginx 的 IP_hash 机制类似

我们用其作为解决 session 问题的一种方法，建议关注；

⑤ri，表示根据请求的 URI；

⑥ rl_param ，表示根据请求的 URl 参数 ’ balance url_param’ requires an URL parameter name；

⑦hdr(name)，表示根据 HTTP 请求头来锁定每一次 HTTP 请求；

⑧rdp-cookie(name)，表示根据据 cookie(name)来锁定并哈希每一次 TCP 请求。

19、mysql 数据备份工具

mysqldump 工具

mysqldump 是 mysql 自带的备份工具，目录在 bin 目录下面：/usr/local/mysql/bin/mysqldump

支持基于 innodb 的热备份，但是由于是逻辑备份，所以速度不是很快，适合备份数据比较小的场景

Mysqldump 完全备份+二进制日志可以实现基于时间点的恢复。

基于 LVM 快照备份

在物理备份中，有基于文件系统的物理备份（LVM 的快照），也可以直接用 tar 之类的命令对整个数据库目录

进行打包备份，但是这些只能进行冷备份，不同的存储引擎备份的也不一样，myisam 自动备份到表级别

而 innodb 不开启独立表空间的话只能备份整个数据库。

tar 包备份

percona 提供的 xtrabackup 工具

支持 innodb 的物理热备份，支持完全备份，增量备份，而且速度非常快，支持 innodb 存储引起的数据在不同

数据库之间迁移，支持复制模式下的从机备份恢复备份恢复，为了让 xtrabackup 支持更多的功能扩展

可以设立独立表空间，打开 innodb_file_per_table 功能，启用之后可以支持单独的表备份

20、keepalive 的工作原理和如何做到健康检查

keepalived 是以 VRRP 协议为实现基础的,VRRP 全称 Virtual Router Redundancy Protocol，即虚拟路由冗余协议。

虚拟路由冗余协议，可以认为是实现路由器高可用的协议，即将 N 台提供相同功能的路由器组成一个路由器组

这个组里面有一个 master 和多个 backup，master 上面有一个对外提供服务的 vip（该路由器所在局域网内

其他机器的默认路由为该 vip），master 会发组播，当 backup 收不到 vrrp 包时就认为 master 宕掉了

这时就需要根据 VRRP 的优先级来选举一个 backup 当 master。这样就可以保证路由器的高可用了

keepalived 主要有三个模块，分别是 core、check 和 vrrp。core 模块为 keepalived 的核心，负责主进程的启动、维护

及全局配置文件的加载和解析。check 负责健康检查，包括常见的各种检查方式，vrrp 模块是来实现 VRRP 协议的

Keepalived 健康检查方式配置

HTTP_GET|SSL_GET

HTTP_GET | SSL_GET

{

url {

```
path  /#  HTTP/SSL  检查的 url 可以是多个
digest  <STRING>  #  HTTP/SSL  检查后的摘要信息用工具 genhash 生成
status_code  200#  HTTP/SSL  检查返回的状态码
}
connect_port  80  #  连接端口
bindto<IPADD>
connect_timeout  3  #  连接超时时间
nb_get_retry  3  #  重连次数
delay_before_retry  2  #连接间隔时间
}
```

21、统计 ip 访问情况，要求分析 nginx 访问日志，找出访问页面数量在前十位的 ip

```
cat access.log  |  awk '{print  $1}'|  uniq -c  |  sort -rn  |  head -10
```

22、使用 tcpdump 监听主机为 192.168.1.1，tcp 端口为 80 的数据，同时将输出结果保存输出到 tcpdump.log

```
tcpdump  'host  192.168.1.1  and  port  80' >  tcpdump.log
```

23、如何将本地 80 端口的请求转发到 8080 端口，当前主机 IP 为 192.168.2.1

```
iptables -A PREROUTING -d 192.168.2.1 -p tcp -m tcp -dport 80 -j DNAT-to-destination
192.168.2.1:8080
```

24、简述 raid0 raid1 raid5 三种工作模式的工作原理及特点

RAID 0：带区卷，连续以位或字节为单位分割数据，并行读/写于多个磁盘上，因此具有很高的数据传输率

但它没有数据冗余，RAID 0 只是单纯地提高性能，并没有为数据的可靠性提供保证

而且其中的一个磁盘失效将影响到所有数据。因此，RAID 0 不能应用于数据安全性要求高的场合

RAID 1：镜像卷，它是通过磁盘数据镜像实现数据冗余，在成对的独立磁盘上产生互为备份的数据

不能提升写数据效率。当原始数据繁忙时，可直接从镜像拷贝中读取数据，因此 RAID1 可以提高读取性能

RAID 1 是磁盘阵列中单位成本最高的，镜像卷可用容量为总容量的 1/2，但提供了很高的数据安全性和可用性

当一个磁盘失效时，系统可以自动切换到镜像磁盘上读写，而不需要重组失效的数据

RAID5：至少由 3 块硬盘组成，分布式奇偶校验的独立磁盘结构，它的奇偶校验码存在于所有磁盘上

任何一个硬盘损坏，都可以根据其它硬盘上的校验位来重建损坏的数据（最多允许 1 块硬盘损坏）

所以 raid5 可以实现数据冗余，确保数据的安全性，同时 raid5 也可以提升数据的读写性能

25、你对现在运维工程师的理解和以及对其工作的认识

运维工程师在公司当中责任重大，需要保证时刻为公司及客户提供最高、最快、最稳定、最安全的服务

运维工程师的一个小小的失误，很有可能会对公司及客户造成重大损失

因此运维工程师的工作需要严谨及富有创新精神

26、实时抓取并显示当前系统中 tcp 80 端口的网络数据信息，请写出完整操作命令

```
tcpdump -nn tcp port 80
```

27、服务器开不了机怎么解决一步步的排查

A、造成服务器故障的原因可能有以下几点：

B、如何排查服务器故障的处理步骤如下：

28、Linux 系统中病毒怎么解决

1）最简单有效的方法就是重装系统

2）要查的话就是找到病毒文件然后删除

- 中毒之后一般机器 cpu、内存使用率会比较高
- 机器向外发包等异常情况，排查方法简单介绍下

top 命令找到 cpu 使用率最高的进程

一般病毒文件命名都比较乱，可以用 ps aux 找到病毒文件位置

rm -f 命令删除病毒文件

检查计划任务、开机启动项和病毒文件目录有无其他可以文件等

3）由于即使删除病毒文件不排除有潜伏病毒，所以最好是把机器备份数据之后重装一下

29、发现一个病毒文件你删了他又自动创建怎么解决

公司的内网某台 linux 服务器流量莫名其妙的剧增,用 iftop 查看有连接外网的情况

针对这种情况一般重点查看 netstat 连接的外网 ip 和端口。

用 lsof -p pid 可以查看到具体是那些进程，哪些文件

经查勘发现/root 下有相关的配置 conf.n hhe 两个可疑文件，rm -rf 后不到一分钟就自动生成了

由此推断是某个母进程产生的这些文件。所以找到母进程就是找到罪魁祸首

查杀病毒最好断掉外网访问，还好是内网服务器，可以通过内网访问

断了内网，病毒就失去外联的能力，杀掉它就容易的多

怎么找到呢，找了半天也没有看到蛛丝马迹，没办法只有 ps axu 一个个排查

方法是查看可以的用户和和系统相似而又不是的冒牌货，果然，看到了如下进程可疑

看不到图片就是/usr/bin/.sshd

于是我杀掉所有.sshd 相关的进程，然后直接删掉.sshd 这个可执行文件

然后才删掉了文章开头提到的自动复活的文件

总结一下，遇到这种问题，如果不是太严重，尽量不要重装系统

一般就是先断外网，然后利用 iftop,ps,netstat,chattr,lsof,pstree 这些工具顺藤摸瓜

一般都能找到元凶。但是如果遇到诸如此类的问题

/boot/efi/EFI/redhat/grub.efi: Heuristics.Broken.Executable FOUND，个人觉得就要重装系统了

30、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”

31、你常用的 Nginx 模块，用来做什么

rewrite 模块，实现重写功能

access 模块：来源控制

ssl 模块：安全加密

ngx_http_gzip_module：网络传输压缩模块

ngx_http_proxy_module 模块实现代理

ngx_http_upstream_module 模块实现定义后端服务器列表

ngx_cache_purge 实现缓存清除功能

32、请列出你了解的 web 服务器负载架构

Nginx

Haproxy

Keepalived

LVS

33、查看 http 的并发请求数与其 TCP 连接状态

```
netstat -n | awk '/^tcp/ {++S[$NF]} END {for(a in S) print a, S[a]}'
```

还有 ulimit -n 查看 linux 系统打开最大的文件描述符，这里默认 1024

不修改这里 web 服务器修改再大也没用，若要用就修改很几个办法，这里说其中一个：

修改/etc/security/limits.conf

```
*  soft  nofile  10240
*  hard  nofile  10240
```

重启后生效

34、用 tcpdump 嗅探 80 端口的访问看看谁最高

```
tcpdump -i eth0 -tnn dst port 80 -c 1000 | awk -F"." '{print $1"."$2"."$3"."$4}'|
sort | uniq -c | sort -nr |head -20
```

35、写一个脚本，实现判断 192.168.1.0/24 网络里，当前在线的 IP 有哪些，能 ping 通则认为在线

```
#!/bin/bash
for ip  in `seq 1  255`
do
{

ping -c  1  192.168.1.$ip  >  /dev/null 2>&1
if [  $?  -eq 0  ];  then
echo 192.168.1.$ip  UP
else
```

```
echo 192.168.1.$ip  DOWN
fi
}&
done
wait
```

36、已知 apache 服务的访问日志按天记录在服务器本地目录/app/logs 下，由于磁盘空间紧张现在要求只能保留最近 7 天的访问日志！请问如何解决？ 请给出解决办法或配置或处理命令

创建文件脚本：

```
#!/bin/bash
for n in `seq 14`
do

date -s "11/0$n/14"
touch access_www_`(date +%F)`.log
done
```

解决方法：

```
# pwd/application/logs
# ll
-rw-r--r--. 1 root root 0 Jan  1 00:00 access_www_2015-01-01.log
-rw-r--r--. 1 root root 0 Jan  2 00:00 access_www_2015-01-02.log
-rw-r--r--. 1 root root 0 Jan  3 00:00 access_www_2015-01-03.log
-rw-r--r--. 1 root root 0 Jan  4 00:00 access_www_2015-01-04.log
-rw-r--r--. 1 root root 0 Jan  5 00:00 access_www_2015-01-05.log
-rw-r--r--. 1 root root 0 Jan  6 00:00 access_www_2015-01-06.log
-rw-r--r--. 1 root root 0 Jan  7 00:00 access_www_2015-01-07.log
-rw-r--r--. 1 root root 0 Jan  8 00:00 access_www_2015-01-08.log
-rw-r--r--. 1 root root 0 Jan  9 00:00 access_www_2015-01-09.log
```

-rw-r--r--. 1 root root 0 Jan 10 00:00 access_www_2015-01-10.log

-rw-r--r--. 1 root root 0 Jan 11 00:00 access_www_2015-01-11.log

-rw-r--r--. 1 root root 0 Jan 12 00:00 access_www_2015-01-12.log

-rw-r--r--. 1 root root 0 Jan 13 00:00 access_www_2015-01-13.log

-rw-r--r--. 1 root root 0 Jan 14 00:00 access_www_2015-01-14.log

find /application/logs/ -type f -mtime +7 -name "*.log"|xargs rm –f

##也可以使用-exec rm -f {} ;进行删除

ll

-rw-r--r--. 1 root root 0 Jan 7 00:00 access_www_2015-01-07.log

-rw-r--r--. 1 root root 0 Jan 8 00:00 access_www_2015-01-08.log

-rw-r--r--. 1 root root 0 Jan 9 00:00 access_www_2015-01-09.log

-rw-r--r--. 1 root root 0 Jan 10 00:00 access_www_2015-01-10.log

-rw-r--r--. 1 root root 0 Jan 11 00:00 access_www_2015-01-11.log

-rw-r--r--. 1 root root 0 Jan 12 00:00 access_www_2015-01-12.log

-rw-r--r--. 1 root root 0 Jan 13 00:00 access_www_2015-01-13.log

-rw-r--r--. 1 root root 0 Jan 14 00:00 access_www_2015-01-14.log

37、如何优化 Linux 系统（可以不说太具体）？

1. 不用 root，添加普通用户，通过 sudo 授权管理
2. 更改默认的远程连接 SSH 服务端口及禁止 root 用户远程连接
3. 定时自动更新服务器时间
4. 配置国内 yum 源
5. 关闭 selinux 及 iptables（iptables 工作场景如果有外网 IP 一定要打开，高并发除外）
6. 调整文件描述符的数量
7. 精简开机启动服务（crond rsyslog network sshd）

8. 内核参数优化（/etc/sysctl.conf）

9. 更改字符集，支持中文，但建议还是用英文字符集，防止乱码

10. 锁定关键系统文件

11. 清空/etc/issue，去除系统及内核版本登录前的屏幕显示

38、请执行命令取出 linux 中 eth0 的 IP 地址(请用 cut，有能力者也可分别用 awk,sed 命令答)

cut 方法 1：

```
# ifconfig eth0|sed -n '2p'|cut -d ":" -f2|cut -d " " -f1
192.168.20.130
```

awk 方法 2：

```
# ifconfig eth0|awk 'NR==2'|awk -F ":" '{print $2}'|awk '{print $1}'
192.168.20.130
```

awk 多分隔符方法 3：

```
# ifconfig eth0|awk 'NR==2'|awk -F "[: ]+" '{print $4}'
192.168.20.130
```

sed 方法 4：

```
# ifconfig eth0|sed -n '/inet addr/p'|sed -r 's#^.*ddr:(.*)Bc.*$##g'
192.168.20.130
```

39、请写出下面 linux SecureCRT 命令行快捷键命令的功能？

Ctrl + a

Ctrl + c

Ctrl + d

Ctrl + e

Ctrl + l

Ctrl + u

Ctrl + k

tab

Ctrl+shift+c

Ctrl+shift+v

解答：

Ctrl + a —->光标移动到行首

Ctrl + e —->光标移动到行尾

Ctrl + c —->终止当前程序

Ctrl + d —->如果光标前有字符则删除，没有则退出当前中断

Ctrl + l —->清屏

Ctrl + u —->剪切光标以前的字符

Ctrl + k —->剪切光标以后的字符

Ctrl + y —->复制 u/k 的内容

Ctrl + r —->查找最近用过的命令

tab —->命令或路径补全

Ctrl+shift+c —->复制

Ctrl+shift+v —->粘贴

40、每天晚上 12 点，打包站点目录/var/www/html 备份到/data 目录下（最好每次备份按时间生成不同的备份包）

cat a.sh

```
#/bin/bash
cd /var/www/ && /bin/tar zcf /data/html-`date +%m-%d%H`.tar.gz html/
```

crontab –e

00 00 * * * /bin/sh /root/a.sh