

1. 什么是linux系统负载load average

Linux系统的负载平均值（load average）是指一段时间内系统上运行的进程数量的平均值。具体来说，它是系统处于不可中断（uninterruptible）状态和可中断（interruptible）状态下的进程数量之和与逻辑CPU核心数的比值。通常来说，load average数据中显示的数字分别代表了过去1分钟、5分钟和15分钟内的平均负载

2. 你如何确定服务器的瓶颈所在？有哪些工具可以用于分析系统瓶颈？

确定服务器的瓶颈所在，需要通过一系列的检查和分析来确定，以下是一些可能的检查和分析方法：

1. 检查系统资源利用率：使用top或htop等工具，查看系统中CPU、内存、磁盘I/O等资源的利用情况，寻找可能的瓶颈所在。
2. 检查进程负载情况：使用ps或pidstat等工具，查看系统中运行的进程的情况，寻找相关进程的负载情况，找到可能的瓶颈所在。
3. 检查系统日志和监控信息：使用系统日志和监控工具，查看系统中的异常信息和活动情况，发现可能的性能瓶颈。
4. 进程跟踪：使用strace或perf等工具，跟踪进程的系统调用和性能行为，发现可能的性能瓶颈。

通过以上检查和分析方法，就能发现系统中潜在的性能瓶颈所在，然后根据具体情况来确定采用哪些工具进行优化。

以下是一些常见的系统瓶颈分析工具：

1. vmstat：用于报告虚拟内存（VM）统计信息，包括CPU负载、内存使用情况、磁盘I/O等。
2. iostat：用于报告磁盘 I/O 统计信息，包括磁盘吞吐量、读写请求平均响应时间等指标。
3. sar：用于收集和分析系统活动和性能指标，包括CPU利用率、内存利用率、网络带宽等。
4. netstat：用于报告网络连接和接口统计信息，包括网络带宽、连接数、错误数等。
5. tcpdump：用于捕获和分析网络数据包，可以检查网络流量和连接状况，找到网络瓶颈所在。
6. strace：用于跟踪进程的系统调用和传递参数的行为，用于调试、分析和诊断程序中的问题。
7. perf：用于对Linux系统进行系统性能分析，包括CPU事件分析、线程调度行为、内存访问等

3. 什么是Linux的IO性能？

Linux中的I/O性能是指由CPU发出的I/O请求到达硬件设备，数据被读取或写入的速度和效率

4. free -m，解释 total , used , free , buff/cache , available 之间的关系

```
buff/cache = buff + cache
total = used + free + shared + cache
available = cache + free
```

5. 简述DNS进行域名解析的过程？

用户要访问`http://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`.http://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`http://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

6. 阐述DNS的中迭代查询和递归查询

- 递归查询：
 - 是指`DNS`服务器在收到用户发起的请求时，必须向用户返回一个准确的查询结果。如果`DNS`服务器本地没有存储与之对应的信息，则该服务器需要询问其他服务器，并将返回的查询结果提交给用户。
 - 一般客户机和本地`DNS`服务器之间属于递归查询，即当客户机向`DNS`服务器发出请求后，若`DNS`服务器本身不能解析，则会向另外的`DNS`服务器发出查询请求，得到最终的肯定或否定的结果后转交给客户机。
 - 此查询的源和目标保持不变，为了查询结果只需要发起一次查询。
 - 客户端向`LocalDNS`发起域名查询-->`localDNS`不知道域名对应的`IP`-->但它知道谁知道->他代为帮客户端去查找-->最后再返回最终结果。
- 迭代查询：
 - 是指`DNS`服务器在收到用户发起的请求时，并不直接回复查询结果，而是告诉另一台`DNS`服务器的地址，用户再向这台`DNS`服务器提交请求，这样依次反复，直到返回查询结果。
 - 一般情况下(有例外)本地的`DNS`服务器向其它`DNS`服务器的查询属于迭代查询，如：若对方不能返回权威的结果，则它会向下一个`DNS`服务器(参考前一个`DNS`服务器返回的结果)再次发起进行查询，直到返回查询的结果为止。此查询的源不变，但查询的目标不断变化，为查询结果一般需要发起多次查询。

9.实验题答案

准备8台机器

10.0.0.201	client
10.0.0.202	localdns
10.0.0.203	forwarddns
10.0.0.204	rootdns
10.0.0.204	orgdns
10.0.0.206	magedumaster
10.0.0.207	mageduslave
10.0.0.208	webserver

在复杂架构中，我们进行配置，要保证每次配置都能够得到验证，防止全部配置后，出现问题无法排错，因此，这里我们从下往上搭建，先搭建主从复制

客户端client配置

```
# 更改网卡配置，将DNS指向10.0.0.202
# Ubuntu将resolve和网卡DNS对齐
rm -rf /etc/resolv.conf
ln -s /run/systemd/resolve/resolv.conf /etc/resolv.conf

# 保证所有主机都能ping通
```

配置webserver

```
# 下载nginx
apt install -y nginx
systemctl enable --now nginx
echo www.magedu.org > /var/www/html/index.html
```

搭建主MasterDNS服务器

```
# 服务器: 10.0.0.206

vim /etc/bind/name.conf.default-zones

# 添加
zone "magedu.org" {
    type master;
    file "/etc/bind/db.magedu.org.zone"; # 写绝对路径
};

vim /etc/bind/db.magedu.org.zone

$TTL 1D

@   IN   SOA   master  admin.magedu.org. (1 1D 1H 1W 3H)

                NS    master
                NS    slave

master  A      10.0.0.206
slave   A      10.0.0.207
www     A      10.0.0.208

# 授权
chown bind:bind db.magedu.org.zone

# 检查语法命令
named-checkzone magedu.org /etc/bind/db.magedu.org.zone
# 检查bind配置
named-checkconf
```

实现从节点DNS服务器

```
vim /etc/bind/named.conf.default-zones

zone "magedu.org" {
    type slave;
    masters {10.0.0.206;};
    file "/etc/bind/slaves/magedu.org.zone"; #建议绝对路径
};

mkdir /etc/bind/slaves
chown -R bind:bind /etc/bind/slaves

# 给权限
```

```
vim /etc/apparmor.d/usr.sbin.named

# 添加
/etc/bind/slaves/** rw,
/etc/bind/slaves/ rw,

# 重启
systemctl restart named
```

配置magedu的父域

```
# 服务器: 10.0.0.205, orgdns
apt install -y bind9

vim /etc/bind/named.conf.default-zones

zone "org." {
    type master;
    file "/etc/bind/db.org";
}

vim db.org

$TTL 86400

@ IN SOA master admin.magedu.org. ( 123 1D 1H 1W 3D )

    NS master

magedu NS mageduns1
magedu NS mageduns2

master A 10.0.0.205
mageduns1 A 10.0.0.206
mageduns2 A 10.0.0.207

# 重启
rndc reload
```

实现根域的主DNS服务器

```
# 服务器: 10.0.0.204 rootdns
# 修改配置
vim named.conf.default-zones

zone "." {
    type master;
    file "/etc/bind/root.zone";
};

vim root.zone

$TTL 1D
```

```
@ IN SOA master admin.magedu.org. ( 1 1D 1H 1W 3D )

      NS      master
org NS      orgns

master A 10.0.0.204
orgns  A 10.0.0.205
```

实现转发目标的DNS服务器

```
# 转发DNS: 10.0.0.203 forwarddns

apt install -y bind9

vim /usr/share/dns/root.hints
.                3600000      NS      A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000      A      10.0.0.204

# 重启服务
systemctl restart named

# ubuntu, 关闭安全加密
dnssec-validation no;

# 如果是rocky, 将安全加密关闭
dnssec-enable no;
dnssec-validation no;

# 测试
nslookup www.magedu.org 10.0.0.203
```

配置本地localDNS服务器

```
# 服务器: 10.0.0.202 localdns

# 设置转发, 并取消安全加固
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forward first;
    forwarders { 10.0.0.203; };

    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys.  See https://www.isc.org/bind-keys
```

```
//=====
dnssec-validation no;

listen-on-v6 { any; };

};
```

在客户端测试

```
# 在客户端10.0.0.201测试
[root@client ~]$ curl www.magedu.org
www.magedu.org

# 成功
```