### DNS服务器的作用

域名解析，为客户机提供“域名--IP”的信息查询服务

-权威DNS：经过互联网域名管理机构授权，至少负责一个DNS区域，只提供对少量域名的查询

-缓存DNS：不要求负责任何DNS区域，无需授权，但是能代理客户机的请求去查询几乎所有的域名

- 端口号：53

-地址库存放路径：/var/named

DNS资源解析记录

正向解析 域名--->IP地址

反向解析 IP---->域名 不常用

**常用的DNS记录**

-A记录 地址记录

-MX记录 邮件交换记录

-CNAME记录 别名域记录

-NS记录

-AAAA记录 地址记录（IPv6）

DNS递归和迭代查询的作用

对于一台DNS服务器来说：

若允许递归，则当客户端请求解析的域名非本DNS管辖时，本DNS会向其他DNS服务器代询；

若不允许递归，则当客户端请求解析的域名非本DNS管辖时，本DNS会放弃代询 —— 但是，如果目标地址位于已知的某个授权子域，本DNS会告知客户端对应的子DNS服务器的地址信息（即迭代）

DNS分离解析（智能DNS、CDN项目）

作用：不同的客户端访问相同的域名，得到不同的解析结果

缓存DNS

不要求负责任何DNS区域，无需授权，但是能代理客户机的请求去查询几乎所有的域名

适用场景：公司内部

当LAN内没有条件建立DNS服务器，但又想让局域网内的用户可以使用计算机名互相访问时，应配置（/etc/hosts）文件

benet公司使用RHEL5系统构建了一台DNS服务器，以便当有客户发送邮件到admin@benet.com时，最终会由mail.benet.com这台邮件服务器来进行处理。在这个过程中，DNS服务器的作用是（ ）　（选择二项）

A、为mail.benet.com添加名为benet.com的别名记录

B、为benet.com域设置MX记录，指向mail.benet.com

C、为主机mail.benet.com提供正确的域名解析

D、将收到的邮件转发至mail.benet.com服务器

DNS实际上是分布在internet上的主机信息的数据库，其作用是实现ip与域名之间的转换

DNS服务器的进程命名为named，当其启动时，自动装载 /etc目录下的named.conf文件中定义的DNS分区数据库文件

**/var/named：地址库文件存放路径**

**##########################################################################**

**MySQL体系结构的组成，每个组成部分的作用**

主要包括8个部分：

连接池： 进程数限制、内存检查、缓存检查等。

SQL接口：用户通过sql客户端发过来的命令，由sql接口接收，sql操作(DML数据操作语言：查询、修改、升级数据等；DDL数据定义语言：创建一个新的数据库、新的索引、删除一个用户等；存储过程、视图触发器。

分析器：分析查询语句 事务处理 对象访问权限。

优化器： 优化访问路径 、 生成执行树。

缓存和缓冲：保存sql查询结果。

存储引擎：用于管理存储的文件系统，将逻辑结构转换为物理结构的程序；不同的存储引擎有不同的功能和存储方式。

管理工具：备份，恢复，安全，移植，集群等，这些工具一般和文件系统打交道，不需要和mysql-server打交道，它们对应的都是命令。

物理存储设备(文件系统)。

**MySQL数据库访问的执行过程**

1）客户端发出请求。

2）服务器端开辟线程响应客户端请求。

3）客户端发起sql语句查询数据库。

4）查询缓存：记录用户的sql查询语句，如果查询内容相同，直接从查询缓存回复。

5）如果缓存没有进入分析器。

6）分析器：分析用户命令语法是否正确，将用户的命令进行切片，一个词一个词用空格隔开，获得用户要查询的表、内容、用户的权限等。

7）优化器：执行路径的选择，生成执行树。（每个SQL语句都有很多执行路径，优化的目的就是在这些执行路径里选择最优的执行路径）。

8）存储引擎：用于管理存储的文件系统，不同的存储引擎有不同的功能和存储方式。

###############################################################################

**Redis 特点**

– 支持数据持久化，可以把内存里数据保存到硬盘中

– 不仅仅支持 key/values 类型的数据，并且支持比较丰富的数据类型

– 支持 master-salve 模式数据备份

**redis数据类型**

字符、列表、hash表、集合、有序集合

**Redis 介绍**

– Remode DIctionary Server( 远程字典服务器 )

– 使用 C 语言编写的，遵守 BSD 的开源软件

– 是一款高性能的 (Key/Values) 分布式内存数据库

– 并支持数据持久化的 NoSQL 数据库服务软件

– 中文网站 www.redis.cn

**redis常用操作指令**

– set keyname keyvalue 存储

– get keyname 获取

– select 数据库编号 0-15 切换库

– keys \* 打印所以变量

– keys a? 打印指定变量

– exits keyname 测试是否存在

– ttl keyname 查看生存时间

– type keyname 查看类型

**redis哨兵服务的作用**

监控当前的主服务器，当主服务宕机后，在所有的从服务器中选择一个新的主服务器（数据最接近主的服务器）

**redis主从复制工作原理**

• 工作原理

– slave 向 master 发送 sync 命令

– master 启动后台存盘进程，同时收集所有修改数据命令

– master 执行完后台存盘进程后，传送整个数据文件到 slave 。

– slave 接收数据文件后，将其存盘并加载到内存中完成 首次完全同步

– 后续有新数据产生时， master 继续将新的所以收集到 的修改命令依次传给 slave ，完成同步。

**redis主从复制缺点**

– 网络繁忙，会产生数据同步延时问题

– 系统繁忙，会产生数据同步延时问题

**Redis支持持久化的方式**

- RDB

- AOF

**RDB 介绍**

• 全称 Reids DataBase

– 数据持久化方式之一

– 在指定时间间隔内，将内存中的数据集快照写入硬盘 。

– 术语叫 Snapshot 快照。

– 恢复时，将快照文件直接读到内存里。

**RDB 优点缺点有哪些？**

• RDB 优点

– 持久化时， Redis 服务会创建一个子进程来进行持久 化，会先将数据写入到一个临时文件中，待持久化过 程都结束了，再用这个临时文件替换上次持久化好的 文件；整个过程中主进程不做任何 IO 操作，这就确保 了极高的性能。

– 如果要进程大规模数据恢复，且对数据完整行要求不是非常高，使用 RDB 比 AOF 更高效。

• RDB 的缺点

– 意外宕机，最后一次持久化的数据会丢失。

**AOF 介绍**

• 只追加操作的文件

– Append Only File

– 记录 redis 服务所有写操作。

– 不断的将新的写操作，追加到文件的末尾。

– 使用 cat 命令可以查看文件内容

**AOF 优点 / 缺点**

• RDB 优点

– 宕机时，仅可能丢失 1 秒的数据

• RDB 的缺点

–AOF 文件的体积通常会大于 RDB 文件的体积。执行 rs ync 策略时的速度可能会比 RDB 慢