

# 数据库管理

**NSD NoSQL**

**DAY01**

# 内容

上午	09:00 ~ 09:30	NoSQL 概述
	09:30 ~ 10:20	部署 Redis 服务
	10:30 ~ 11:20	
	11:30 ~ 12:00	
下午	14:00 ~ 14:50	部署 LNMP+Redis
	15:00 ~ 15:50	
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑



# NoSQL 概述

---

数据库类型

RDBMS

RDBMS 服务软件

NoSQL

NoSQL 服务软件

NoSQL 概述

# 数据库服务软件类型

---

# RDBMS

- 关系数据库管理系统
  - Relational Database Management System
  - 数据按照预先设置的组织结构，存储在物理存储介质上。
  - 数据之间可以做关联操作



# RDBMS 服务软件

- 主流的 RDBMS 软件
  - Oracle
  - DB2
  - WS-sqlserver
  - MySQL



# NoSQL

- NoSQL(NoSQL = Not Only SQL )
  - 意思是 “ 不仅仅是 SQL ”
  - 泛指非关系型数据库
  - 不需要预先定义数据存储结构
  - 表的每条记录都可以有不同的类型和结构



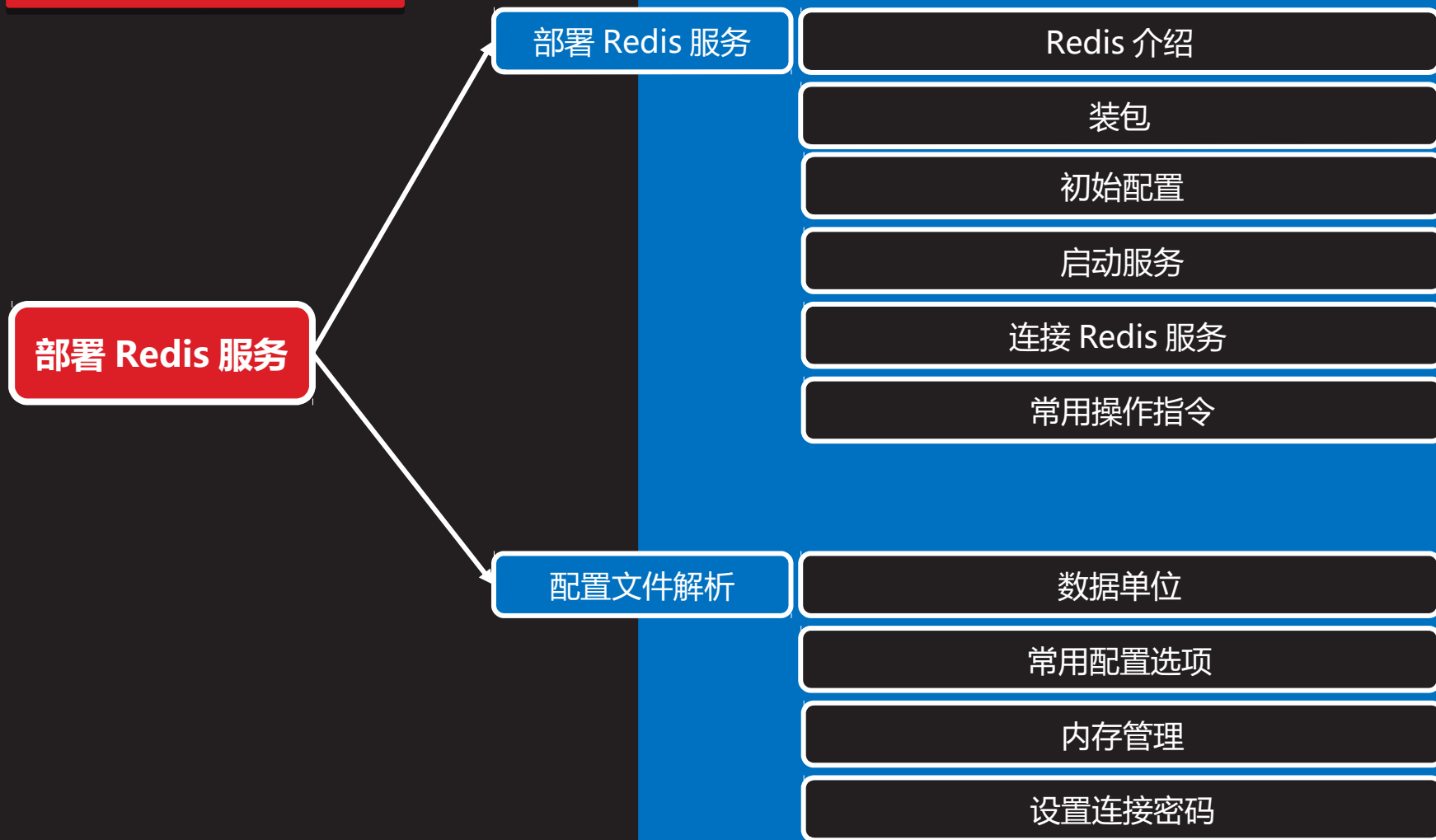
# NoSQL 服务软件

- 主流软件
  - \_ Redis
  - \_ MongoDB
  - \_ Memcached
  - \_ CouchDB
  - \_ Neo4j
  - \_ FlockDB





# 部署 Redis 服务



# 部署 Redis 服务

---

# Redis 介绍

- Redis
  - Remote 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](http://www.redis.cn)



# Redis 介绍 (续 1)

- 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, 同时还支持 list hash set zset 类型
  - 支持 master-slave 模式数据备份



# 装包

- 安装源码包

```
# tar -xzf redis-4.0.8.tar.gz  
# cd redis-4.0.8  
#make  
#make install
```



# 初始化配置

- 配置服务运行参数
  - 端口
  - 主配置文件
  - 数据库目录
  - pid 文件
  - 启动程序

`#!/utils/install_server.sh` // 初始化



# 启动服务

- 启动服务

- \_ # /etc/init.d/redis\_<portnumber> start

- 停止服务

- \_ # /etc/init.d/redis\_<portnumber> stop



# 连接 Redis 服务

- 访问 redis 服务
  - \_ ps -C redis
  - \_ netstat -utnlp | grep redis
  - \_ redis-cli // 连接本机 redis 服务





# 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 \* // 打印所有变量
- \_ 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



## 常用操作指令（续 1）

- \_ move keyname dbname // 移动变量
- \_ expire keyname 10 // 设置有效时间
- \_ del keyname // 删除变量
- \_ flushall // 删除所有变量
- \_ save // 保存变量
- \_ shutdown // 关闭服务



# 案例 1：搭建 Redis 服务器

具体要求如下：

- 在主机 192.168.4.51 上运行 redis 服务
- 设置变量 test，值为 123
- 查看变量 test 的值



# 配置文件解析

---

# 数据单位

- 数据单位

```
# 1k => 1000 bytes
# 1kb => 1024 bytes
# 1m => 1000000 bytes
# 1mb => 1024*1024 bytes
# 1g => 1000000000 bytes
# 1gb => 1024*1024*1024 bytes
#
# units are case insensitive so 1GB 1Gb 1gB are all the same.
```



# 常用配置选项

- port 6379 // 端口
- bind 127.0.0.1 //IP 地址
- tcp-backlog 511 //tcp 连接总数
- timeout 0 // 连接超时时间
- tcp-keepalive 300 // 长连接时间
- daemonize yes // 守护进程方式运行
- databases 16 // 数据库个数
- logfile /var/log/redis\_6379.log //pid 文件
- maxclients 10000 // 并发连接数量
- dir /var/lib/redis/6379 // 数据库目录



# 内存管理

- 内存清除策略
  - volatile-lru 最近最少使用（针对设置了过期时间的 key）
  - allkeys-lru 删除最少使用的 key
  - volatile-random 在设置了过期的 key 里随机移除
  - allkeys-random 随机移除 key
  - volatile-ttl (minor TTL) 移除最近过期的 key
  - noeviction 不删除 写满时报错



# 内存管理（续 1）

- 选项默认设置
  - \_ maxmemory <bytes> // 最大内存
  - \_ maxmemory-policy noeviction // 定义使用的策略
  - \_ maxmemory-samples 5 // 选取模板数据的个数  
(针对 lru 和 ttl 策略)





# 设置连接密码

- 设置密码

- `grep -n requirepass /etc/redis/6379.conf`  
`501:requirepass 123456`
- `[root@localhost redis-4.0.8]# redis-cli`
- `127.0.0.1:6379> ping`
- `(error) NOAUTH Authentication required.`
- `127.0.0.1:6379> auth 123456` // 输入密码
- `OK`
- `127.0.0.1:6379> ping`
- `PONG`
- `127.0.0.1:6379>`



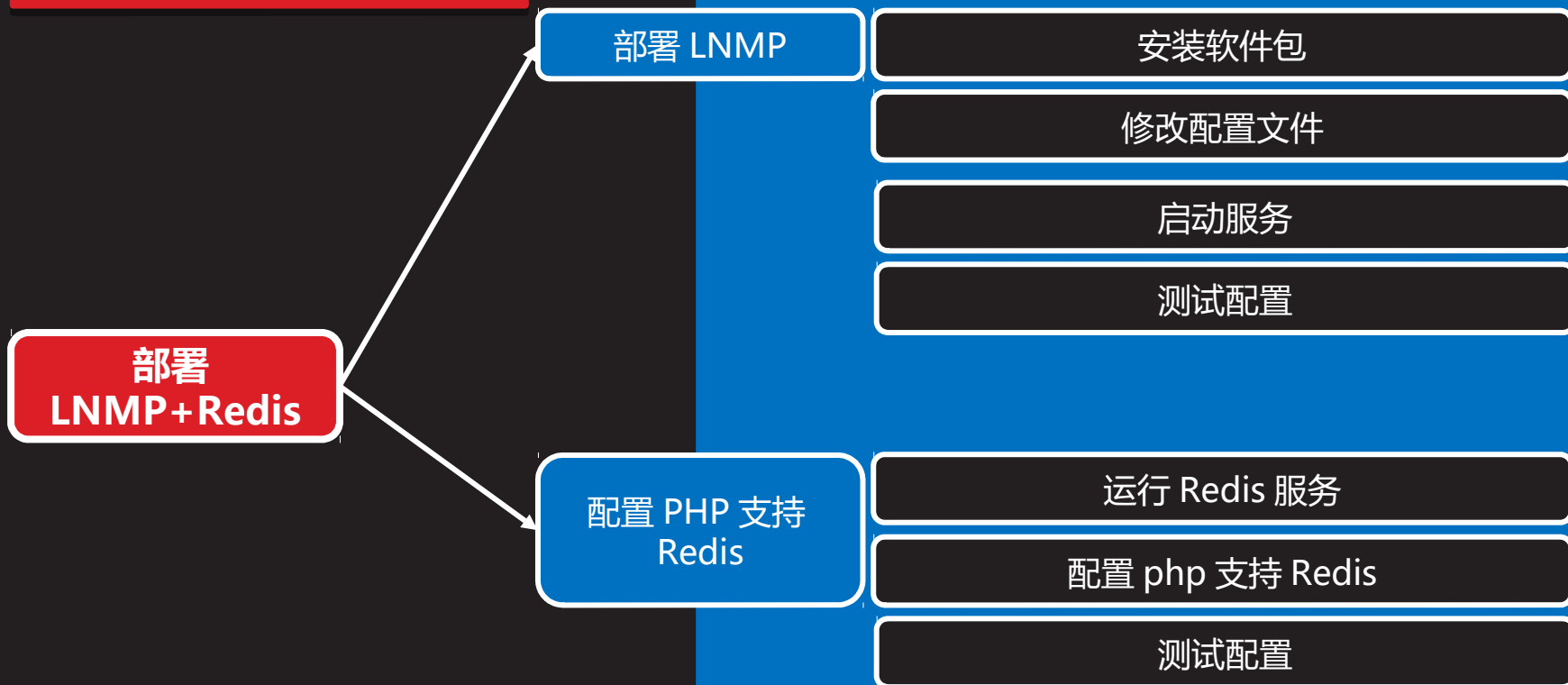
## 案例 2：修改 Redis 服务运行参数

具体要求如下：

- 端口号 6351
- IP 地址 192.168.4.51
- 连接密码 123456
- 客户端连接服务



# 部署 LNMP+Redis



# 部署 LNMP

---

# 安装软件包

- 安装源码 nginx 和 PHP

```
]# yum -y install gcc gcc-c++ pcre-devel zlib-devel  
]# tar -zxf nginx-1.12.2.tar.gz  
]# cd nginx-1.12.2  
]# ./configure --prefix=/usr/local/nginx  
]# make  
]# make install
```

```
[root@bogon ~]# yum -y install php-common  
[root@bogon ~]# rpm -ivh php-fpm-5.4.16-42.el7.x86_64.rpm
```



# 修改配置文件

- 修改 nginx 服务主配置文件

```
[root@bogon ~]# vim /usr/local/nginx/conf/nginx.conf
location ~ \.php$ {
    root      html;
    fastcgi_pass 127.0.0.1:9000;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME $document_root/$fastcgi_script_name;
    include    fastcgi_params;
}
:wq
```



# 启动服务

- 启动 nginx 服务

```
[root@bogon nginx-1.12.2]# /usr/local/nginx/sbin/nginx  
[root@bogon nginx-1.12.2]# netstat -utnlp | grep :80
```

- 启动 php-fpm 服务

```
[root@bogon lnmp]# systemctl start php-fpm  
[root@bogon lnmp]# netstat -utnlp | grep :9000
```



# 测试配置

- 测试文件

```
[root@bogon lnmp]# vim /usr/local/nginx/html/test.php
<?php
echo "hello world!!!";
?>
:wq
[root@bogon lnmp]#
```

- 访问 nginx 服务

```
[root@bogon lnmp]# elinks --dump http://localhost/test.php
hello world!!!
[root@bogon lnmp]#
```





# 配置 PHP 支持 Redis

---

# 运行 Redis 服务

- 装包并启动服务

```
# tar -xzf redis-4.0.8.tar.gz
# cd redis-4.0.8
#make
#make install
#./utils/install_server.sh // 初始化
# /etc/init.d/redis_6379 start
```



# 配置 php 支持 Redis

- 安装 php 扩展

```
#yum -y install autoconf
#yum -y install automake
#rpm -ivh php-devel-5.4.16-42.el7.x86_64.rpm
# tar -zxf php-redis-2.2.4.tar.gz
# cd phpredis-2.2.4/
[root@bogon phpredis-2.2.4]# /usr/bin/phpize
Configuring for:
PHP Api Version:      20100412
Zend Module Api No:   20100525
Zend Extension Api No: 220100525
# ./configure --with-php-config=/usr/bin/php-config
[root@bogon phpredis-2.2.4]# make
[root@bogon phpredis-2.2.4]# make install
[root@bogon ~]# vim /etc/php.ini
extension_dir = "./"
extension = "redis.so"
```



# 测试配置

- 查看是否支持模块

```
[root@bogon ~]# php -m | grep -i redis  
redis
```

- 编写测试文件

```
[root@bogon bin]# cat  
/usr/local/nginx/html/redis.php  
<?php  
$redis = new redis();  
$redis->connect('127.0.0.1',6379);  
$redis->set('redistest','666666');  
echo $redis->get('redistest');  
?>
```

- 访问 nginx 服务

```
[root@bogon bin]# elinks --dump http://localhost/redis.php  
666666
```



## 案例 3：部署 LNMP+Redis

具体要求如下：

- 在主机 192.168.4.52 上部署 LNMP 环境
- 配置 192.168.4.52 主机，把数据存储到本机的 redis 服务。



# 总结和答疑

---

总结和答疑

服务无法关闭

问题现象

原因分析

# 服务无法关闭

---

# 问题现象

- 故障错误信息

```
[root@svr5 ~]# service redis_6379 stop
```

```
Waiting for Redis to shutdown ...
```

```
(error) NOAUTH Authentication required.
```





# 原因分析

- 分析故障信息
  - 设置 redis 主从后，连接主服务器需要密码
  - 而 redis 启动脚本中未设置密码
- 分析故障原因
  - redis 启动脚本通过执行 `redis-cli -p 6379 shutdown` 关闭 redis，因此没有密码是无法关闭服务的
  - 修改脚本 `redis-cli -p 6379 -a 密码 shutdown`

