# server端部署

以linux系统为例，安装aerospike server，详细细节可以参考官网的linux系统安装指导，详见https://www.aerospike.com/docs/operations/install/linux/el6/index.html。

## 下载

wget -O aerospike.tgz '<https://www.aerospike.com/download/server/latest/artifact/el6>'

## 解压安装

解压刚下载好的压缩包，使用如下命令：

tar -xvf aerospike.tgz

执行该命令后会获得aerospike-server-community-<version>-el6的文件夹。

进入上述文件夹，执行其中的安装命令便会安装aerospike server&tools，命令如下：

cd aerospike-server-community-<version>-el6/

sudo ./asinstall

等待安装执行完成会在多个目录下生成相应内容：

Server启动脚本 /etc/init.d/aerospike

日志 /var/log/aerospike/aerospike.log

安装目录 /opt/aerospike

## 启动

执行如下命令便可完成server端启动

service aerospike start

## 验证

方法一：

使用如下命令验证server端的启动情况，

service aerospike status

如正常启动的话可到类似如下的结果：

asd (pid 12867) is running...

方法二：

查看日志信息验证，执行如下命令：

grep cake /var/log/aerospike/aerospike.log

正常启动情况下会得到类似如下的结果：

Jun 22 2014 03:35:33 GMT: INFO (as): (as.c::376) service ready: soon there will be cake!

如server启动异常也可以跟上上述日志进行问题定位，aerospike官网提供了一下常见问题的处理方式，可以进行参考，网址如下：

<https://www.aerospike.com/docs/operations/troubleshoot/startup/>。

至此，aerospike的一个单节点server已经搭建完成。现在server使用的配置文件是安装时生成的默认配置文件，接下来讲解如何通过修改配置文件来完成一个aerospike集群环境的搭建。

# Aerospike集群环境搭建

## 1.概述

Aerospike集群环境搭建官网上有详细的添加节点、删除节点的指导，网址如下：<https://www.aerospike.com/docs/operations/manage/cluster_mng/index.html。>

Aerospike集群只要通过修改配置文件中network->heartbeat的配置项实现新节点的自动添加到已有集群，而不需要原有集群的重启。新节点若想能够自动添加到现有集群需要满足以下条件：

* 集群中所有节点（包括新增节点）必须拥有相同数目的namespace；
* 在节点之间配置文件中namespace部分中的多个namespace的顺序是一致的；
* 对于只存内存和永久保存的namespace来说，都需要保证每个节点分配给每个namespace的资源（内存/磁盘空间）是一致的；
* 若namespace是永久保存类型的，那么分配给这个namespace的所有设备必须是相同容量的。

Aerospike中关于heartbeat有两种方式：mesh（tcp）和multicast（udp），详情见官网解释，https://www.aerospike.com/docs/operations/configure/network/heartbeat/index.html。接下来分别介绍集群环境搭建。

注：在向已有集群中添加节点时，最好一个节点一个节点的单独添加，在确保前一个已经成功加入到集群后在添加后一个。

## Mesh

现在三台机器10.10.35.19, 10.10.35.20, 10.10.35.21，首先将19 20搭建成一个2台机器的小集群，然后再将21加入到这个集群。

首先修改19,20这两台机器的配置文件，如下：

10.10.35.19的配置文件修改使用命令是

vi /etc/aerospike/aerospike.conf

# Aerospike database configuration file for deployments using mesh heartbeats.

service {

user root

group root

paxos-single-replica-limit 1 # Number of nodes where the replica count is automatically reduced to 1.

pidfile /var/run/aerospike/asd.pid

service-threads 4

transaction-queues 4

transaction-threads-per-queue 4

proto-fd-max 15000

node-id-interface eth0 #需要是使用ip对应的网卡

}

logging {

# Log file must be an absolute path.

file /var/log/aerospike/aerospike.log {

context any info

}

}

network {

service {

address any

access-address 10.10.35.19

port 3000

}

heartbeat {

mode mesh

address 10.10.35.19

port 3002 # Heartbeat port for this node.

# List one or more other nodes, one ip-address & port per line:

mesh-seed-address-port 10.10.35.19 3002

mesh-seed-address-port 10.10.35.20 3002

#mesh-seed-address-port 10.10.35.21 3002

interval 250

timeout 10

}

fabric {

port 3001

}

info {

port 3003

}

}

namespace test {

# Data in memory without persistance namespace

replication-factor 2

memory-size 4G

default-ttl 30d # 30 days, use 0 to never expire/evict.

storage-engine memory

}

namespace bar {

memory-size 4G # Maximum memory allocation for primary and secondary indexes.

# Warning - legacy data in defined raw partition devices will be erased.

# These partitions must not be mounted by the file system.

single-bin true # Required true by data-in-index.

data-in-index true # Enables in index integer store.

storage-engine device { # Configure the storage-engine to use persistence

# Add raw device(s). Maximum size is 2 TiB

file /opt/aerospike/bar.dat # Location of data file on server.

# device /dev/sdb

# device /dev/sdc

# The 2 lines below optimize for SSD.

#scheduler-mode noop

#write-block-size 128K # adjust block size to make it efficient for SSDs.

filesize 1G # Max size of each file in GiB. Maximum size is 2TiB

data-in-memory true # Required true by data-in-index.

}

}

10.10.35.20机器上的配置文件：

vi /etc/aerospike/aerospike.conf

# Aerospike database configuration file for deployments using mesh heartbeats.

service {

user root

group root

paxos-single-replica-limit 1 # Number of nodes where the replica count is automatically reduced to 1.

pidfile /var/run/aerospike/asd.pid

service-threads 4

transaction-queues 4

transaction-threads-per-queue 4

proto-fd-max 15000

node-id-interface eth0 #需要是使用ip对应的网卡

}

logging {

# Log file must be an absolute path.

file /var/log/aerospike/aerospike.log {

context any info

}

}

network {

service {

address any

access-address 10.10.35.20

port 3000

}

heartbeat {

mode mesh

address 10.10.35.20

port 3002 # Heartbeat port for this node.

# List one or more other nodes, one ip-address & port per line:

mesh-seed-address-port 10.10.35.19 3002

mesh-seed-address-port 10.10.35.20 3002

#mesh-seed-address-port 10.10.35.21 3002

interval 250

timeout 10

}

fabric {

port 3001

}

info {

port 3003

}

}

namespace test {

# Data in memory without persistance namespace

replication-factor 2

memory-size 4G

default-ttl 30d # 30 days, use 0 to never expire/evict.

storage-engine memory

}

namespace bar {

memory-size 4G # Maximum memory allocation for primary and secondary indexes.

# Warning - legacy data in defined raw partition devices will be erased.

# These partitions must not be mounted by the file system.

single-bin true # Required true by data-in-index.

data-in-index true # Enables in index integer store.

storage-engine device { # Configure the storage-engine to use persistence

# Add raw device(s). Maximum size is 2 TiB

file /opt/aerospike/bar.dat # Location of data file on server.

# device /dev/sdb

# device /dev/sdc

# The 2 lines below optimize for SSD.

#scheduler-mode noop

#write-block-size 128K # adjust block size to make it efficient for SSDs.

filesize 1G # Max size of each file in GiB. Maximum size is 2TiB

data-in-memory true # Required true by data-in-index.

}

}

上面配置文件修改的重点内容已经有红色标出。按照上述配置修改完aerospike的配置文件后，分别在19、20两台机器上启动aerospike，执行命令是 service aerospike start。

如果两台机器都以正常启动，在日志中可以查到类似如下内容就表明aerospike已经在以集群方式运行。

grep 'CLUSTER-SIZE' /var/log/aerospike/aerospike.log

Jan 28 2017 01:00:03 GMT: INFO (info): (ticker.c:169) NODE-ID bb9af1f8d270008 CLUSTER-SIZE 2

接下来对21这台机器进行配置，同时修改19,20两台机器的配置文件，即将heartbeat中红色标识部分list中添加21这台机器。

10.10.35.21的配置如下：

vi /etc/aerospike/aerospike.conf

# Aerospike database configuration file for deployments using mesh heartbeats.

service {

user root

group root

paxos-single-replica-limit 1 # Number of nodes where the replica count is automatically reduced to 1.

pidfile /var/run/aerospike/asd.pid

service-threads 4

transaction-queues 4

transaction-threads-per-queue 4

proto-fd-max 15000

node-id-interface eth0 #需要是使用ip对应的网卡

}

logging {

# Log file must be an absolute path.

file /var/log/aerospike/aerospike.log {

context any info

}

}

network {

service {

address any

access-address 10.10.35.21

port 3000

}

heartbeat {

mode mesh

address 10.10.35.21

port 3002 # Heartbeat port for this node.

# List one or more other nodes, one ip-address & port per line:

mesh-seed-address-port 10.10.35.19 3002

mesh-seed-address-port 10.10.35.20 3002

mesh-seed-address-port 10.10.35.21 3002

interval 250

timeout 10

}

fabric {

port 3001

}

info {

port 3003

}

}

namespace test {

# Data in memory without persistance namespace

replication-factor 2

memory-size 4G

default-ttl 30d # 30 days, use 0 to never expire/evict.

storage-engine memory

}

namespace bar {

memory-size 4G # Maximum memory allocation for primary and secondary indexes.

# Warning - legacy data in defined raw partition devices will be erased.

# These partitions must not be mounted by the file system.

single-bin true # Required true by data-in-index.

data-in-index true # Enables in index integer store.

storage-engine device { # Configure the storage-engine to use persistence

# Add raw device(s). Maximum size is 2 TiB

file /opt/aerospike/bar.dat # Location of data file on server.

# device /dev/sdb

# device /dev/sdc

# The 2 lines below optimize for SSD.

#scheduler-mode noop

#write-block-size 128K # adjust block size to make it efficient for SSDs.

filesize 1G # Max size of each file in GiB. Maximum size is 2TiB

data-in-memory true # Required true by data-in-index.

}

}

然后在21上启动aerospike，如正常启动的话可以在日志中看到如下内容即表明新的节点已经成功加入到原有集群之中。可以看到集群size已经有2变成了3。

Dec 20 2017 06:40:01 GMT: INFO (info): (ticker.c:164) NODE-ID bb963cdb4005472 CLUSTER-SIZE 3

## Multicast

Multicast与Mesh两个方式建立集群性质是一样的，唯一的区别就是heartbeat的配置。下面给出Multicast方式下，19/20/21三台机器的heartbeat相关配置。

10.10.35.19 heartbeat相关配置：

heartbeat {

mode multicast # Send heartbeats using Multicast

multicast-group 239.1.99.2 # multicast address

port 9918 # multicast port

address 10.10.35.19 # (Optional) (Default any) IP of the NIC to

# use to send out heartbeat and bind

# fabric ports

interval 250

timeout 10

}

10.10.35.20 heartbeat相关配置：

heartbeat {

mode multicast # Send heartbeats using Multicast

multicast-group 239.1.99.2 # multicast address

port 9918 # multicast port

address 10.10.35.20 # (Optional) (Default any) IP of the NIC to

# use to send out heartbeat and bind

# fabric ports

interval 250

timeout 10

}

10.10.35.21 heartbeat相关配置：

heartbeat {

mode multicast # Send heartbeats using Multicast

multicast-group 239.1.99.2 # multicast address

port 9918 # multicast port

address 10.10.35.21 # (Optional) (Default any) IP of the NIC to

# use to send out heartbeat and bind

# fabric ports

interval 250

timeout 10

}

同前一种方式一样，修改还配置文件后在每台机器上重启server，三台机器会自动建立成一个集群。正常情况下日志会出现类似如下内容：

Dec 20 2017 06:57:52 GMT: INFO (info): (ticker.c:164) NODE-ID bb963cdb4005472 CLUSTER-SIZE 3

关于aerospike的更多配置信息参见官网的config章节：

https://www.aerospike.com/docs/operations/configure/index.html。

# aerospike工具

Aerospike常用的工具主要有四个：asadm、aql、asbackup&asrestore、asinfo。在aerospike的安装目录（/opt/aerospike）下的bin文件夹下能看到这四个工具对应的命令，如下图所示：



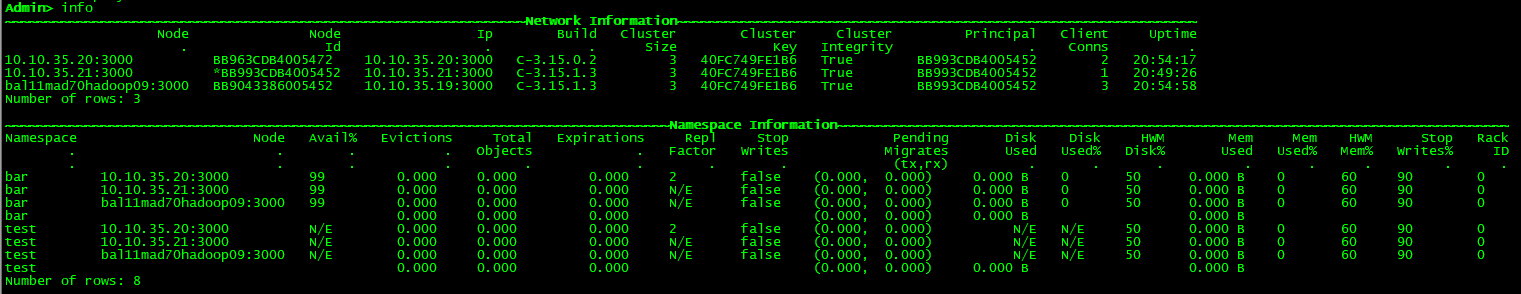
## asadm

Aerospike Asadm是一个主要用于获取当前集群健康状况或是执行动态配置或是动态执行调优命令的一个交互式python工具。

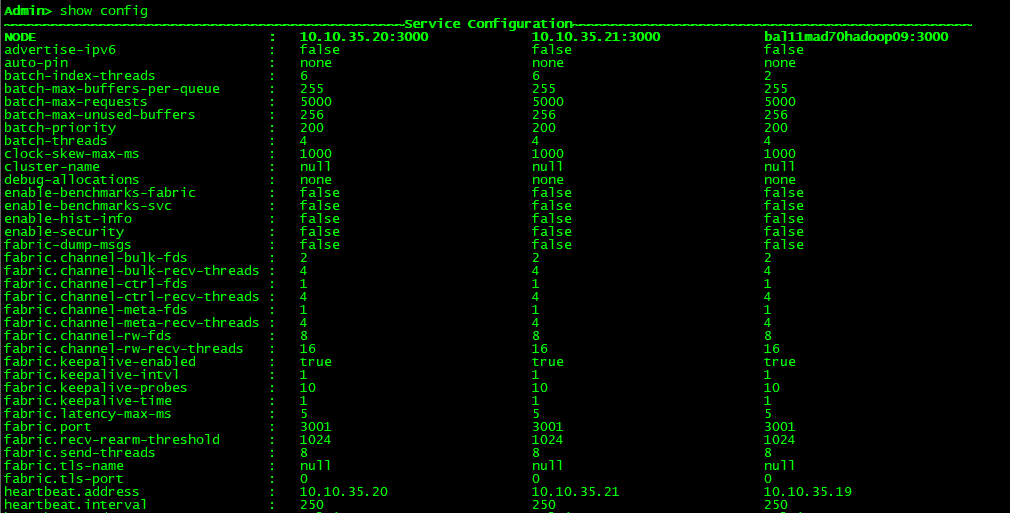
使用asadm --help命令能够显示出相关的帮助信息。直接执行asadm可以进入一个'Admin>'提示符开头的交互式界面，可在该界面内输入相关命令。

常用交互式命令有：

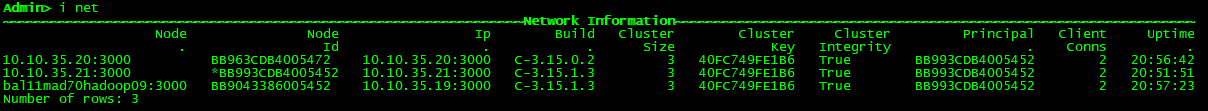
info



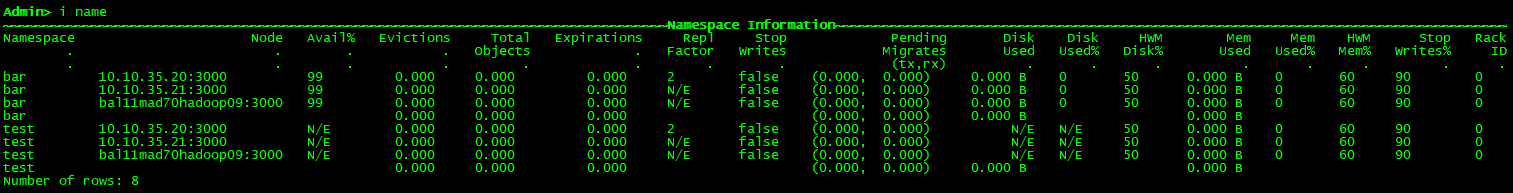
show config



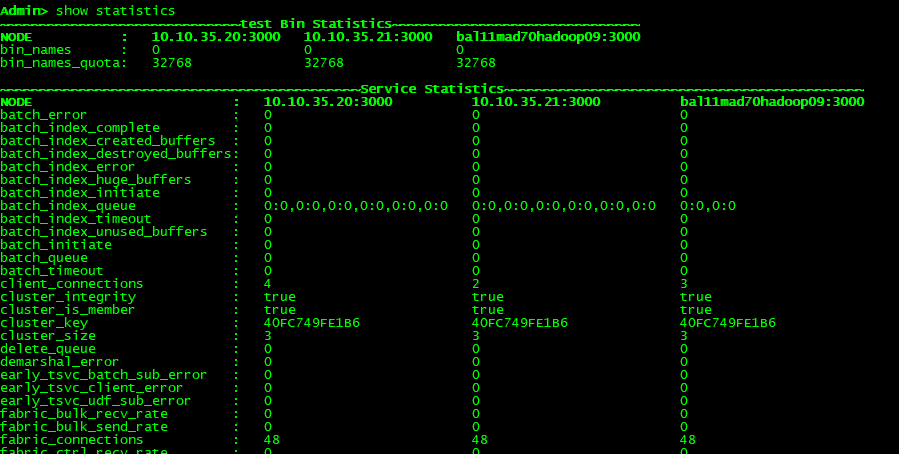
i net(short for info network)



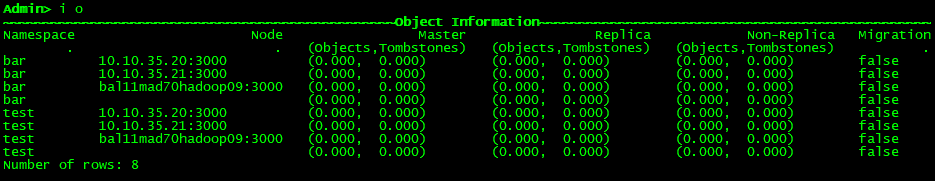
i name



show statistics



i o(short for info object)



info set/ info sindex/ info xdr/info datacenter/

show config/ show config network like heartbeat mesh/show config diff/show config namespace like partition -flip/show distribution time\_to\_live/show latency like writes\_master

watch 5 3 info network

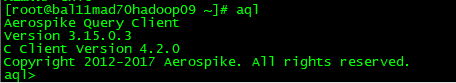
详情见官网asadm页面：<https://www.aerospike.com/docs/tools/asadm/index.html。>

## aql

Aerospike Query Language (aql)是aerospike提供的一个用来管理数据库、UDF、索引的类似sql的命令行接口。Aerospike并不支持SQL作为其管理或是查询的语言，但是通过aql提供了一个类SQL的工具类接口。

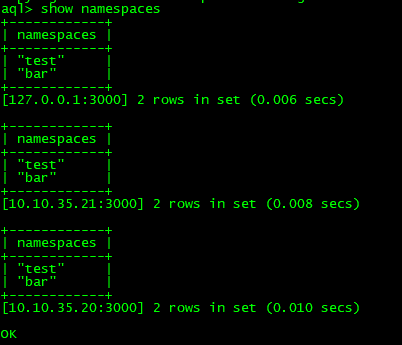
输入aql --help能够显示出有关的帮助信息。输入aql会进入一个’aql>’提示符开始的交互式界面，可在其中输入类似sql语句的查询、管理操作。

如下图所示：

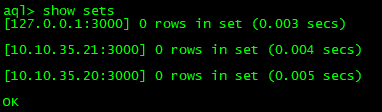


常用命令以及结果如下：

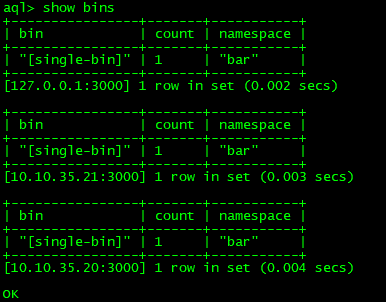
show namespaces



show sets



show bins

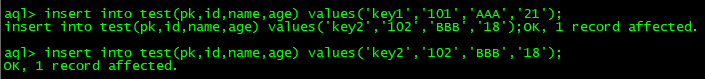


下面是一个简单的插入、查询、删除操作。

插入：

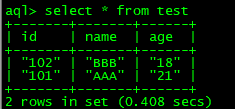
insert into test(pk,id,name,age) values('key1','101','AAA','21');

insert into test(pk,id,name,age) values('key2','102','BBB','18');



查询：

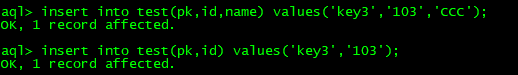
select \* from test

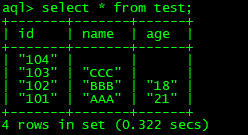


插入缺少某个bin的数据：

insert into test(pk,id,name) values('key3','103','CCC');

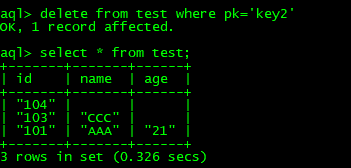
insert into test(pk,id) values('key4','104');





删除：

delete from test where pk='key2'



Aql还有很多语法、功能，包括：数据管理、索引管理、，详情见官方文档的说明：https://www.aerospike.com/docs/tools/aql/index.html。

## asbackup&asrestore

Aerospike提供了工具用于备份和转存集群中的数据。在通常情况下，数据复制（data replication）和交叉数据中心复制（cross data center replication）机制可以保证即使在硬件损坏或是网络断开的情况下数据也不会丢失。但是实践表明创建备份是一种从数据中心事故或是管理节点挂掉等情况中快速恢复的有效方式。

Asbackup：使用aerospike标准的客户端scan接口备份整个namespace或是其中某个指定set的数据。详情见官网：https://www.aerospike.com/docs/tools/backup/asbackup.html。

Asrestore：使用aerospike标准的客户端接口对上面asbackup生成的备份文件进行转存。详情见官网：https://www.aerospike.com/docs/tools/backup/asrestore.html。

## asinfo

Asinfo是aerospike用来提供集群命令、控制功能接口的一个命令行工具，其中包括在集群运行的情况下更改配置的功能。

Asinfo同其他命令去的别在于-u才是打印帮助信息的参数。

当输入asinfo -u时会出现如下的帮助信息：



详情参见官网：https://www.aerospike.com/docs/tools/asinfo/index.html。