

Infraestruturas de Sistemas Distribuídos

Mestrado em Engenharia Informática e de Computadores

Aspetos práticos sobre Zookeeper

Tabela de Conteúdos

1	Introdução.....	3
2	Sobre o quadro tecnológico Apache Zookeeper.....	3
2.1	Instalação em sistema operativo Linux (Ubuntu)	3
3	Conclusões	7
4	Referências.....	7

1 Introdução

Neste documento são sumarizados aspetos de implementação do recurso computacional Apache Zookeeper que pode ser utilizado enquanto elemento de infraestrutura de sistemas distribuídos. Sugere-se que o conceito de infraestrutura seja relativizado na medida em que na estruturação de um sistema informático enquanto elementos ISystem/CES/Service, o que designamos por elementos de infraestrutura são de facto artefactos tecnológicos que suportam outros artefactos; elementos Service.

Segue-se uma apresentação sumária da configuração de uma instalação Apache Zookeeper seja no modo de instância única (*standalone*) seja através de um Ensemble no que é designado por modo quórum.

2 Sobre o quadro tecnológico Apache Zookeeper

Como definido na página do projeto Apache Zookeeper, o que passemos a designar por elemento zookeeper, é um serviço de gestão de elementos de dados estruturados numa árvore de nós (*Znodes*), e.g., registo de referências para serviços no quadro OSGi Remote Services, a avaliar em prática posterior. O serviço é disponibilizado com a possibilidade de configuração de réplicas no garante de uma disponibilidade acrescida no acesso aos nós. As funcionalidades implementadas, i.e., os serviços disponibilizados pelo elemento zookeeper podem ser usados por outros elementos distribuídos, em execução *em espaços de endereçamento diferentes, eventualmente em diferentes instâncias de um sistema operativo ou contentor, enquanto nós interligados por rede (IP)*.

“is a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services. All of these kinds of services are used in some form or another by distributed applications. ...” [\[link\]](#).

2.1 Instalação em sistema operativo Linux (Ubuntu)

A instalação do zookeeper no sistema operativo Ubuntu, na máquina virtual *iservervm.ova* (servidor *iserver* com user/password → *is/is*) fornecida ([link](#)) pode ser realizada seguindo os passos que se enumeram de seguida.

- Partindo da imagem fornecida, pré configurada, depois de criar uma réplica (clone na base de referência), pode alterar o nome do servidor com o comando `hostnamectl` com a alteração do novo nome no ficheiro `/etc/hosts`

```
si@iserver0:~/Desktop$ sudo hostnamectl set-hostname iserver0
si@iserver0:~/Desktop$ sudo vi /etc/hosts
127.0.0.1          localhost
192.168.56.101     iserver0
...
```

Sugere-se que instancie cinco máquinas virtuais, *iservervm0*, *iservervm1*, *iservervm2*, *iservervm3*, *iservervm4*, *iservervm5*, em que cada uma executa o servidor Ubuntu com o nome, respetivamente ***iserver0***, ***iserver1***, ***iserver2***. De notar que, a partir do sistema operativo Host, Windows se a infraestrutura de virtualização VirtualBox estiver instalada diretamente sobre Windows, a

comunicação com os sistemas operativos guest iserver0, iserver1, iserver2 deverá ser verificada se acessível pelo nome dos servidores. Se executado o comando ping (opção -4 para mostrar o endereço IPV4) num terminal/console (linha de comandos), o nome do servidor não for resolvido, deverá usar diretamente o endereço IP.

```
C:\Users\aosorio>ping iserver0 -4
Pinging iserver0.local [192.168.56.107] with 32 bytes of data:
Reply from 192.168.56.107: bytes=32 time<1ms TTL=64
...
Ping statistics for 192.168.56.107:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Em cada um dos servidores deverá instalar o sistema zookeeper seguindo os passos que se detalham de seguida. De acordo com o Filesystem Hierarchy Standard ([FHS](#)) recursos partilhados devem ser instalados nas diretorias /usr ou /opt. Na sugestão de procedimentos de instalação do zookeeper em link é adotada a diretoria /opt. Adotamos os procedimentos de instalação da comunidade DigitalOcean em [link](#).

Configurações ao nível do sistema operativo Ubuntu.

- Criação do utilizador zk

```
sudo useradd zk -m
```
- Configura o interpretador de comandos, bash por omissão

```
sudo usermod --shell /bin/bash zk
```
- Define palavra chave para o utilizador zk (sugere-se zk)

```
sudo passwd zk
```
- O utilizador zk é associado ao grupo zk de modo a que possa executar comandos privilegiados

```
sudo usermod -aG sudo zk
```
- Configuração de acessos SSH

```
sudo vi /etc/ssh/sshd_config
PermitRootLogin no
DenyUsers zk
```
- Após o que deverá reiniciar o servidor ssh. Caso não esteja instalado, deverá proceder à sua instalação.

```
sudo systemctl restart sshd
```
- Instalação do servidor ssh, caso não esteja ainda instalado.

```
sudo apt-get update
sudo apt-get upgrade
sudo apt install openssh-server
```
- Criação de uma diretoria para dados zookeeper e atribuição de direitos de acesso ao grupo utilizador zk.

```
sudo mkdir -p /data/zookeeper
sudo chown zk:zk /data/zookeeper
```

Caso particular de uma máquina virtual em Provider Cloud

Se usar uma máquina virtual Ubuntu num fornecedor de serviços de cloud, e.g. a Amazon Web Services ([AWS](#)) com uma máquina sem custos associados, como é o caso da *Amazon Machine Image* (AMI), como se mostra na Figura 1, terá que instalar o ambiente Java (JRE) para poder executar o sistema Zookeeper.

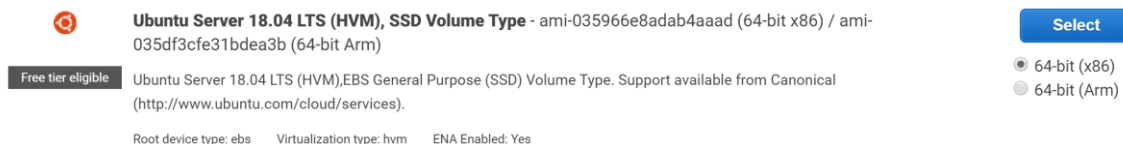


Figura 1 – Exemplo de imagem sem custos, no fornecedor de Cloud Amazon

- Instalação na máquina virtual Java aberta Open Java versão JRE, deverá executar os comandos:

```
sudo add-apt-repository ppa:openjdk-r/ppa
sudo apt-get update
sudo apt-get install openjdk-8-jre
```

Instalação do Zookeeper.

- Obtenção do ficheiro de instalação e extração para a diretoria corrente, /opt. É ainda criada uma ligação lógica com a designação zookeeper e “ownership” ao utilizador zk.

```
cd /opt
sudo wget https://downloads.apache.org/zookeeper/zookeeper-3.8.0/apache-
zookeeper-3.8.0-bin.tar.gz
sudo tar -xvf apache-zookeeper-3.8.0-bin.tar.gz
sudo ln -s apache-zookeeper-3.8.0-bin zookeeper
sudo chown -R zk:zk apache-zookeeper-3.8.0-bin
sudo chown -h zk:zk zookeeper
```

Configuração do Zookeeper no modo standalone.

- A configuração do zookeeper baseia-se num conjunto de ficheiros localizados em /opt/zookeeper/conf. A diretoria contém um exemplo de ficheiro de configuração, zoo_sample.cfg que pode ser usado como base para a definição do ficheiro de configuração zoo.cfg. Começa por se autenticar como utilizador zk, password zk. De seguida deve editar o ficheiro de configuração zoo.cfg e introduzir os valores de tickTime, dataDir, clientPort e maxClientCnxns.

```
su zk
Password: <zk>
vi /opt/zookeeper/conf/zoo.cfg
tickTime=2000
dataDir=/data/zookeeper
```

```
clientPort=2181
maxClientCnxns=60
```

Configuração do Zookeeper no modo quorum.

- No modo quórum o ficheiro de configuração declara os servidores que constituem o Ensemble.

```
vi /opt/zookeeper/conf/zoo.cfg
tickTime=2000
dataDir=/data/zookeeper
clientPort=2181
maxClientCnxns=60
initLimit=5
syncLimit=2
server.1=iserver0:2888:3888
server.2=iserver1:2888:3888
server.3=iserver2:2888:3888
server.4=iserver3:2888:3888
server.5=iserver4:2888:3888
```

- No modo quórum tem ainda que colocar na diretoria dataDir, no caso, em /data/zookeeper, um ficheiro com a designação myid cujo conteúdo é a identificação do servidor. Por exemplo, para o servidor 1 deverá escrever 1 no ficheiro myid, e.g. usando o comando echo do Linux.

```
echo 1 > /data/zookeeper/myid
cat /data/zookeeper/myid
1
```

Problemas de comunicação quando é usado o nome do servidor no próprio nó.

Recorda-se a necessidade de verificação de resolução do nome de servidor sem o que deverá usar endereços IP para referir os nós do ensemble. No caso da máquina virtual fornecida iservervm, se executar o comando ping com o nome do servidor, é retornado o endereço de loopback, localhost.

- Ping sobre o próprio, iserver2, obtém-se o endereço 127.0.1.1.

```
zk@iserver2:/opt/zookeeper$ ping iserver2
PING iserver2 (127.0.1.1) 56(84) bytes of data.
64 bytes from iserver2 (127.0.1.1): icmp_seq=1 ttl=64 time=0.037 ms
...
```

Execução do servidor (serviços) Zookeeper.

- O servidor é iniciado invocando o script (Shell) zkServer.sh com o argumento start.

```
cd /opt/zookeeper
bin/zkServer.sh start
...
Starting zookeeper ... STARTED
zk@iserver0:/opt/zookeeper$
```

- Para terminar o servidor basta passar como o argumento stop.

```
bin/zkServer.sh stop
```

Acesso através de cliente Zookeeper em Linux.

- Pode verificar se a instalação está operacional, invocando o cliente incluído no projeto Apache Zookeeper.

```
bin/zkCli.sh -server 127.0.0.1:2181
...
WatchedEvent state:SyncConnected type:None path:null
[zk: 127.0.0.1:2181(CONNECTED) 0]
```

- Através deste cliente simples pode “navegar” na estrutura de znodes, entretanto criados por comando ou através de acesso programático, e.g., por um `ces`, enquanto elemento de um sistema informático.

```
[zk: 127.0.0.1:2181(CONNECTED) 0] ls /
[zookeeper]
[zk: 127.0.0.1:2181(CONNECTED) 1]
```

Acesso através de cliente Zookeeper em Windows.

- Cliente em Windows.

```
%ZK_HOME%\bin\zkCli.cmd -server "iserver0, iserver1, iserver2, iserver3, iserver4"
```

3 Conclusões

<a desenvolver>

4 Referências