

# UCS Hyperflex Zookeeper Service - Spiegazione e verifica

## Sommario

[Introduzione](#)

[Controlla stato del servizio Esposizione](#)

[Servizio Query Zookeeper](#)

[File di log di Zookeeper in una configurazione dinamica](#)

[File di registro di Zookeeper da bundle di supporto \(storfs\)](#)

## Introduzione

Questo documento descrive ZooKeeper, che è essenzialmente un servizio centralizzato per sistemi distribuiti a un archivio valori chiave gerarchico. Viene utilizzato per fornire un servizio di configurazione distribuita, un servizio di sincronizzazione e un registro di denominazione per sistemi distribuiti di grandi dimensioni. L'architettura di ZooKeeper supporta l'alta disponibilità tramite servizi ridondanti. I clienti possono quindi chiedere a un altro capo ZooKeeper se il primo non risponde. I nodi ZooKeeper memorizzano i propri dati in uno spazio dei nomi gerarchico, in modo simile a un file system o a una struttura dati. I client possono leggere e scrivere nei nodi e in questo modo disporre di un servizio di configurazione condiviso. ZooKeeper può essere visto come un sistema di trasmissione atomico attraverso il quale gli aggiornamenti sono totalmente ordinati.

ZooKeeper offre queste caratteristiche principali:

- Sistema affidabile: il sistema è molto affidabile perché continua a funzionare anche in caso di guasto di un nodo.
- Architettura semplice - L'architettura di ZooKeeper è abbastanza semplice; utilizza uno spazio dei nomi gerarchico condiviso che facilita il coordinamento dei processi.
- Elaborazione rapida - ZooKeeper è particolarmente veloce per i carichi di lavoro dominanti in lettura.
- Scalabile - Le prestazioni di ZooKeeper possono essere migliorate con l'aggiunta di nodi.

In HX, esiste questa implementazione specifica:

- Il servizio chiamato **exitor** gestisce l'avvio/lo spegnimento dello zookeeper.
- I processi all'interno del cluster HX sono client verso Zookeeper e comunicano tramite la porta tcp **2181** ex storfs, stmgr e così via.
- I sistemi con più di cinque nodi avranno alcuni nodi in modalità standalone. I sistemi con cinque nodi o meno non devono mai avere un nodo in modalità standalone.
- Numero minimo di nodi necessari per il quorum =  $N/2 + 1$ .

Ad esempio, per un cluster a tre nodi -  $N/2=1,5$  Arrotondato a  $1 + 1 = 2$  (è possibile tollerare un solo errore di nodo)

Ad esempio, per un cluster a cinque nodi -  $N/2=2,5$  Arrotondato a  $2 + 1 = 3$  (sono tollerati solo gli errori di due nodi)

Poiché per un cluster ZK vengono eseguiti solo cinque nodi, è possibile tollerare un massimo di due errori di nodo per qualsiasi numero di nodi nel cluster. Ciò è vero per i nodi convergenti.

## Controlla stato del servizio Esposizione

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# service exhibitor status  
exhibitor start/running, process 4905
```

```
root@help:/var/log/springpath# ps -aux | grep -i exhibitor  
root 12519 0.0 0.2 4690592 198892 ? Ssl May19 7:19 exhibitor -cp exhibitor.jar:/etc/exhibitor/ -  
Xmx256M -XX:+HeapDumpOnOutOfMemoryError -  
XX:HeapDumpPath=/var/log/exhibitor_heap_dump_2019_05_19_22:19:48.hprof -  
Dlog4j.configuration=file:///etc/exhibitor/log4j.properties -  
Dspringpath.zkdownscript=/usr/share/springpath/storfs-misc/zkMonitor.sh -  
Djava.security.egd=file:/dev/./urandom -jar exhibitor.jar --hostname 10.197.252.100 -c file --  
fsconfigdir /etc/exhibitor --port 8180 --listenaddress 10.197.252.100  
root@help:/var/log/springpath# pidof exhibitor  
12519
```

## Servizio Query Zookeeper

Zookeeper dispone di una sintassi di comando di quattro lettere che consente di eseguire query su stato, elenchi di connessioni, numero di znode e così via.

Controllare lo stato di zookeeper sul nodo locale - (ruok ==> Continuare?. imok=>Sono OK.

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# echo ruok|nc localhost 2181  
imok
```

Controllate se lo zookeeper è un puntale o un puntale.

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# echo srvr | nc localhost 2181  
Zookeeper version: 3.4.6--1, built on 06/16/2015 22:50 GMT  
Latency min/avg/max: 0/0/101  
Received: 213128515  
Sent: 213164119  
Connections: 6  
Outstanding: 0  
Zxid: 0xa000301d0  
Mode: leader  
Node count: 17090
```

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# echo stat | nc localhost 2181  
Zookeeper version: 3.4.6--1, built on 06/16/2015 22:50 GMT  
Clients:  
/192.168.5.161:56128[1] (queued=0,recved=169146196,sent=169162634)  
/192.168.5.161:38614[1] (queued=0,recved=186015,sent=186017)  
/192.168.5.164:44412[1] (queued=0,recved=184398,sent=184399)  
/192.168.5.164:44447[1] (queued=0,recved=561168,sent=563034)  
/127.0.0.1:60060[0] (queued=0,recved=1,sent=0)  
/192.168.5.161:58754[1] (queued=0,recved=39233,sent=39261)
```

```
Latency min/avg/max: 0/0/101  
Received: 213109927  
Sent: 213145531  
Connections: 6  
Outstanding: 0
```

```
Zxid: 0xa000301d0
Mode: leader
Node count: 17090
```

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# echo mntr | nc localhost 2181
zk_version      3.4.6--1, built on 06/16/2015 22:50 GMT
zk_avg_latency  0
zk_max_latency  101
zk_min_latency  0
zk_packets_received  213148668
zk_packets_sent    213184272
zk_num_alive_connections  6
zk_outstanding_requests  0
zk_server_state   leader
zk_znode_count    17090
zk_watch_count    4305
zk_ephemerals_count  20
zk_approximate_data_size  1831768
zk_open_file_descriptor_count  43
zk_max_file_descriptor_count  4096
zk_followers      3
zk_synced_followers  3
zk_pending_syncs  0
```

Controllare la configurazione di Zookeeper:

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# echo conf | nc localhost 2181
clientPort=2181
dataDir=/var/zookeeper/version-2
dataLogDir=/var/zookeeper/version-2
tickTime=3000
maxClientCnxns=60
minSessionTimeout=6000
maxSessionTimeout=60000
serverId=3
initLimit=10
syncLimit=3
electionAlg=3
electionPort=3888
quorumPort=2888
peerType=0
```

## File di log di Zookeeper in una configurazione dinamica

In caso di problemi nei servizi di Zookeeper, questi file di log consentono di trovare le tracce:

- `/var/log/zookeeper/zookeeper*` - Gestisce i log archiviati, utili parole chiave di ricerca WARN, ERROR, Goodbye, Leader e così via.
- `/var/log/springpath/zk-*`
- `/var/log/springpath/exhibitor.log`

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# grep -i leader /var/log/zookeeper/zookeeper.log*
```

```
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,088 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Leader@60] - TCP NoDelay set to: true
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,099 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Leader@358] - LEADING - LEADER ELECTION TOOK - 354
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,120 [myid:3] - INFO [LearnerHandler-
/192.168.5.164:36487:LearnerHandler@522] - Received NEWLEADER-ACK message from 0
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,120 [myid:3] - INFO [LearnerHandler-
/192.168.5.163:43451:LearnerHandler@522] - Received NEWLEADER-ACK message from 1
```

```
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,120 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Leader@943] - Have quorum of supporters, sids: [ 0,1,3
]; starting up and setting last processed zxid: 0x100000000
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,272 [myid:3] - INFO
[WorkerReceiver[myid=3]:FastLeaderElection@597] - Notification: 1 (message format version), 3
(n.leader), 0x0 (n.zxid), 0x1 (n.round), LOOKING (n.state), 2 (n.sid), 0x0 (n.peerEpoch) LEADING
(my state)
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,291 [myid:3] - INFO [LearnerHandler-
/192.168.5.162:48778:LearnerHandler@486] - Sending snapshot last zxid of peer is 0x0 zxid of
leader is 0x100000000sent zxid of db as 0x100000000
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:59:26,298 [myid:3] - INFO [LearnerHandler-
/192.168.5.162:48778:LearnerHandler@522] - Received NEWLEADER-ACK message from 2
```

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# grep -i warn
```

```
/var/log/zookeeper/zookeeper.log*
```

```
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:46:30,354 [myid:] - WARN
[main:QuorumPeerMain@113] - Either no config or no quorum defined in config, running in
standalone mode
/var/log/zookeeper/zookeeper.log.7:2016-10-14 22:52:55,238 [myid:] - WARN
[main:QuorumPeerMain@113] - Either no config or no quorum defined in config, running in
standalone mode
```

```
root@SpringpathControllerMSH7NHXRFL:/var/log/zookeeper# grep -i goodbye
```

```
/var/log/zookeeper/zookeeper.log*
```

```
/var/log/zookeeper/zookeeper.log.1:2017-01-23 03:55:50,429 [myid:3] - WARN [LearnerHandler-
/192.168.5.163:44118:LearnerHandler@646] - ***** GOODBYE /192.168.5.163:44118 *****
/var/log/zookeeper/zookeeper.log.1:2017-01-24 23:30:14,956 [myid:3] - WARN [LearnerHandler-
/192.168.5.164:44720:LearnerHandler@646] - ***** GOODBYE /192.168.5.164:44720 *****
/var/log/zookeeper/zookeeper.log.3:2016-12-01 23:45:22,510 [myid:3] - WARN [LearnerHandler-
/192.168.5.164:44051:LearnerHandler@646] - ***** GOODBYE /192.168.5.164:44051 *****
/var/log/zookeeper/zookeeper.log.3:2016-12-08 00:36:37,752 [myid:3] - WARN [LearnerHandler-
/192.168.5.162:46577:LearnerHandler@646] - ***** GOODBYE /192.168.5.162:46577 *****
/var/log/zookeeper/zookeeper.log.4:2016-11-22 23:45:30,957 [myid:3] - WARN [LearnerHandler-
/192.168.5.163:49016:LearnerHandler@646] - ***** GOODBYE /192.168.5.163:49016 *****
/var/log/zookeeper/zookeeper.log.4:2016-11-23 00:03:59,397 [myid:3] - WARN [LearnerHandler-
/192.168.5.164:45952:LearnerHandler@646] - ***** GOODBYE /192.168.5.164:45952 *****
/var/log/zookeeper/zookeeper.log.4:2016-12-01 22:51:00,538 [myid:3] - WARN [LearnerHandler-
/192.168.5.163:45284:LearnerHandler@646] - ***** GOODBYE /192.168.5.163:45284 *****
/var/log/zookeeper/zookeeper.log.5:2016-11-10 23:39:47,477 [myid:3] - WARN [LearnerHandler-
/192.168.5.163:43576:LearnerHandler@646] - ***** GOODBYE /192.168.5.163:43576 *****
/var/log/zookeeper/zookeeper.log.5:2016-11-11 00:49:39,782 [myid:3] - WARN [LearnerHandler-
/192.168.5.164:35219:LearnerHandler@646] - ***** GOODBYE /192.168.5.164:35219 *****
```

## Alcuni log di esempio - Registrazione di Zookeeper

```
2017-01-22 23:47:29,427 [myid:3] - INFO [Thread-2:QuorumCnxManager$Listener@504] - My election
bind port: /192.168.5.161:3888
2017-01-22 23:47:29,435 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:QuorumPeer@714] - LOOKING
2017-01-22 23:47:29,438 [myid:3] - INFO
[QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:FastLeaderElection@815] - New election. My id = 3,
proposed zxid=0x9000a6b4d
2017-01-22 23:47:29,443 [myid:3] - INFO [WorkerReceiver[myid=3]:FastLeaderElection@597] -
Notification: 1 (message format version), 2 (n.leader), 0x800055ea0 (n.zxid), 0x1 (n.round),
FOLLOWING (n.state), 0 (n.sid), 0x9 (n.peerEpoch) LOOKING (my state)
2017-01-22 23:47:29,444 [myid:3] - INFO [WorkerReceiver[myid=3]:FastLeaderElection@597] -
Notification: 1 (message format version), 2 (n.leader), 0x800055ea0 (n.zxid), 0x1 (n.round),
FOLLOWING (n.state), 1 (n.sid), 0x9 (n.peerEpoch) LOOKING (my state)
2017-01-22 23:47:29,444 [myid:3] - INFO [WorkerReceiver[myid=3]:FastLeaderElection@597] -
Notification: 1 (message format version), 3 (n.leader), 0x9000a6b4d (n.zxid), 0x1 (n.round),
LOOKING (n.state), 3 (n.sid), 0x9 (n.peerEpoch) LOOKING (my state)
```

2017-01-22 23:47:29,444 [myid:3] - INFO [WorkerReceiver[myid=3]:FastLeaderElection@597] - Notification: 1 (message format version), 2 (n.leader), 0x800055ea0 (n.zxid), 0x1 (n.round), FOLLOWING (n.state), 1 (n.sid), 0x9 (n.peerEpoch) LOOKING (my state)  
2017-01-22 23:47:29,445 [myid:3] - INFO [WorkerReceiver[myid=3]:FastLeaderElection@597] - Notification: 1 (message format version), 2 (n.leader), 0x800055ea0 (n.zxid), 0x1 (n.round), LEADING (n.state), 2 (n.sid), 0x9 (n.peerEpoch) LOOKING (my state)  
2017-01-22 23:47:29,445 [myid:3] - INFO [WorkerReceiver[myid=3]:FastLeaderElection@597] - Notification: 1 (message format version), 2 (n.leader), 0x800055ea0 (n.zxid), 0x1 (n.round), FOLLOWING (n.state), 0 (n.sid), 0x9 (n.peerEpoch) LOOKING (my state)  
2017-01-22 23:47:29,446 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:QuorumPeer@784] - FOLLOWING  
2017-01-22 23:47:29,449 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Learner@86] - TCP NoDelay set to: true  
2017-01-22 23:47:29,449 [myid:3] - INFO [WorkerReceiver[myid=3]:FastLeaderElection@597] - Notification: 1 (message format version), 2 (n.leader), 0x800055ea0 (n.zxid), 0x1 (n.round), LEADING (n.state), 2 (n.sid), 0x9 (n.peerEpoch) FOLLOWING (my state)  
2017-01-22 23:47:29,660 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:zookeeper.version=3.4.6--1, built on 06/16/2015 22:50 GMT  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:host.name=SpringpathControllerMSH7NHXRFL  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:java.version=1.7.0\_79  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:java.vendor=Oracle Corporation  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:java.home=/usr/lib/jvm/java-7-openjdk-amd64/jre  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:java.class.path=/usr/share/zookeeper/bin/./build/classes:/usr/share/zookeeper/bin/./build/lib/\*.jar:/usr/share/zookeeper/bin/./lib/slf4j-log4j12-1.6.1.jar:/usr/share/zookeeper/bin/./lib/slf4j-api-1.6.1.jar:/usr/share/zookeeper/bin/./lib/netty-3.7.0.Final.jar:/usr/share/zookeeper/bin/./lib/log4j-1.2.16.jar:/usr/share/zookeeper/bin/./lib/jline-0.9.94.jar:/usr/share/zookeeper/bin/./zookeeper-3.4.6.jar:/usr/share/zookeeper/bin/./src/java/lib/\*.jar:/usr/share/zookeeper/bin/./conf:  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:java.library.path=/usr/java/packages/lib/amd64:/usr/lib/x86\_64-linux-gnu/jni:/lib/x86\_64-linux-gnu:/usr/lib/x86\_64-linux-gnu:/usr/lib/jni:/lib:/usr/lib  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:java.io.tmpdir=/tmp  
2017-01-22 23:47:29,661 [myid:3] - INFO [QuorumPeer[myid=3]/0:0:0:0:0:0:0:0:2181:Environment@100] - Server environment:java.compiler=

## LEADER ELECTION TOOK

root@SpringpathControllerMSH7NHXRFL:/var/log/springpath# cat zk-debug-storfs.log

2017-01-22 23:47:18,702:5866(0x7fd1f7ef5700):ZOO\_INFO@check\_events@1760: initiated connection to server [192.168.5.163:2181]  
2017-01-22 23:47:18,704:5866(0x7fd1f7ef5700):ZOO\_INFO@check\_events@1807: session establishment

```
complete on server [192.168.5.163:2181], sessionId=0x159165ff6310005, negotiated timeout=17001
2017-01-22 23:47:18,704:5866(0x7fd1f76f4700):ZOO_INFO@process_completions@2170: Calling a
watcher for node s], type = s
2017-01-23 01:50:16,809:5866(0x7fd1f7ef5700):ZOO_ERROR@handle_socket_error_msg@1778: Socket
[192.168.5.163:2181] zk retcode=-4, errno=112(Host is down): failed while receiving a server
response
2017-01-23 01:50:16,818:5866(0x7fd1f76f4700):ZOO_INFO@process_completions@2170: Calling a
watcher for node s], type = s
2017-01-23 01:50:16,818:5866(0x7fd1f7ef5700):ZOO_INFO@check_events@1760: initiated connection to
server [192.168.5.164:2181]
2017-01-23 01:50:16,818:5866(0x7fd1f7ef5700):ZOO_ERROR@handle_socket_error_msg@1778: Socket
[192.168.5.164:2181] zk retcode=-4, errno=112(Host is down): failed while receiving a server
response
2017-01-23 01:50:17,819:5866(0x7fd1f7ef5700):ZOO_ERROR@handle_socket_error_msg@1740: Socket
[192.168.5.162:2181] zk retcode=-4, errno=115(Operation now in progress): poll refused to accept
read/write from the client
```

```
root@help:/var/log/springpath# cat zkEvents.log
```

```
INFO:ZkEvents:Send changes to listeners
INFO:EventDB:Received message{"timestamp": 1559200009008, "description": "Cluster policy
compliance is satisfied", "id": "ClusterPolicyComplianceSatisfiedEvent"}
DEBUG:kazoo.client:Received EVENT: Watch(type=3, state=3,
path=u'/zkEvents/lastModificationTime')
DEBUG:kazoo.client:Sending request(xid=42): GetData(path='/zkEvents/lastModificationTime',
watcher=
```

**Cluster is healthy**

```
root@SpringpathControllerPZTMTRSH7K:/var/log/springpath# tail exhibitor.log
```

```
05-20 05:28:52.223 INFO org.mortbay.log - Started SocketConnector@10.197.252.99:8180
05-20 05:29:20.106 INFO com.netflix.exhibitor.core.activity.ActivityLog - State: down
05-20 05:29:20.106 INFO com.netflix.exhibitor.core.activity.ActivityLog - Attempting to stop
instance
05-20 05:29:20.106 INFO com.netflix.exhibitor.core.activity.ActivityLog - Attempting to
start/restart ZooKeeper
05-20 05:29:20.328 INFO com.netflix.exhibitor.core.activity.ActivityLog - jps didn't find
instance - assuming ZK is not running
05-20 05:29:20.347 INFO com.netflix.exhibitor.core.activity.ActivityLog - Process started via:
/usr/share/zookeeper/bin/zkServer.sh
05-20 05:29:20.353 ERROR com.netflix.exhibitor.core.activity.ActivityLog - ZooKeeper Server:
ZooKeeper JMX enabled by default
05-20 05:29:20.353 ERROR com.netflix.exhibitor.core.activity.ActivityLog - ZooKeeper Server:
Using config: /usr/share/zookeeper/bin/./conf/zoo.cfg
05-20 05:29:21.366 INFO com.netflix.exhibitor.core.activity.ActivityLog - ZooKeeper Server:
Starting zookeeper ... STARTED
05-20 05:29:50.128 INFO com.netflix.exhibitor.core.activity.ActivityLog - State: serving
```

## File di registro di Zookeeper da bundle di supporto (storfs)

In un pacchetto di supporto, questi sono file importanti da esaminare:

zookeeper.log	/var/log/zookeeper
zk-storfs.log	/var/log/springpath
echo_stat_ nc_localhost_2181.out	under cmds_output