

MongoDB Replica Sets

An Introduction via a Sample Session

Note : please be sure you have v1.6.0 or higher.

Also check that your client drivers are replica set ready
(old drivers generally work but with lesser functionality).

First, we create a set with a single member. (You can also set up an entire set all at once.)

```
sf1$ mongod --rest --replSet myset
```

setup

First, we create a set with a single member. This is not necessarily typical; see the docs on how to initiate an entire set all at once.

```
sf1$ mongod --rest --replSet myset
```




our hostname

setup

First, we create a set with a single member. This is not necessarily typical; see the docs on how to initiate an entire set all at once.

```
sf1$ mongod --rest --replSet myset
```



Allow use of the built-in web administrative UI at port 28017 (port number + 1000).

First, we create a set with a single member. This is not necessarily typical; see the docs on how to initiate an entire set all at once.

```
sf1$ mongod --rest --replSet myset
```




--replSet *setname*

Our logical setname is “myset”.

We can now check our status

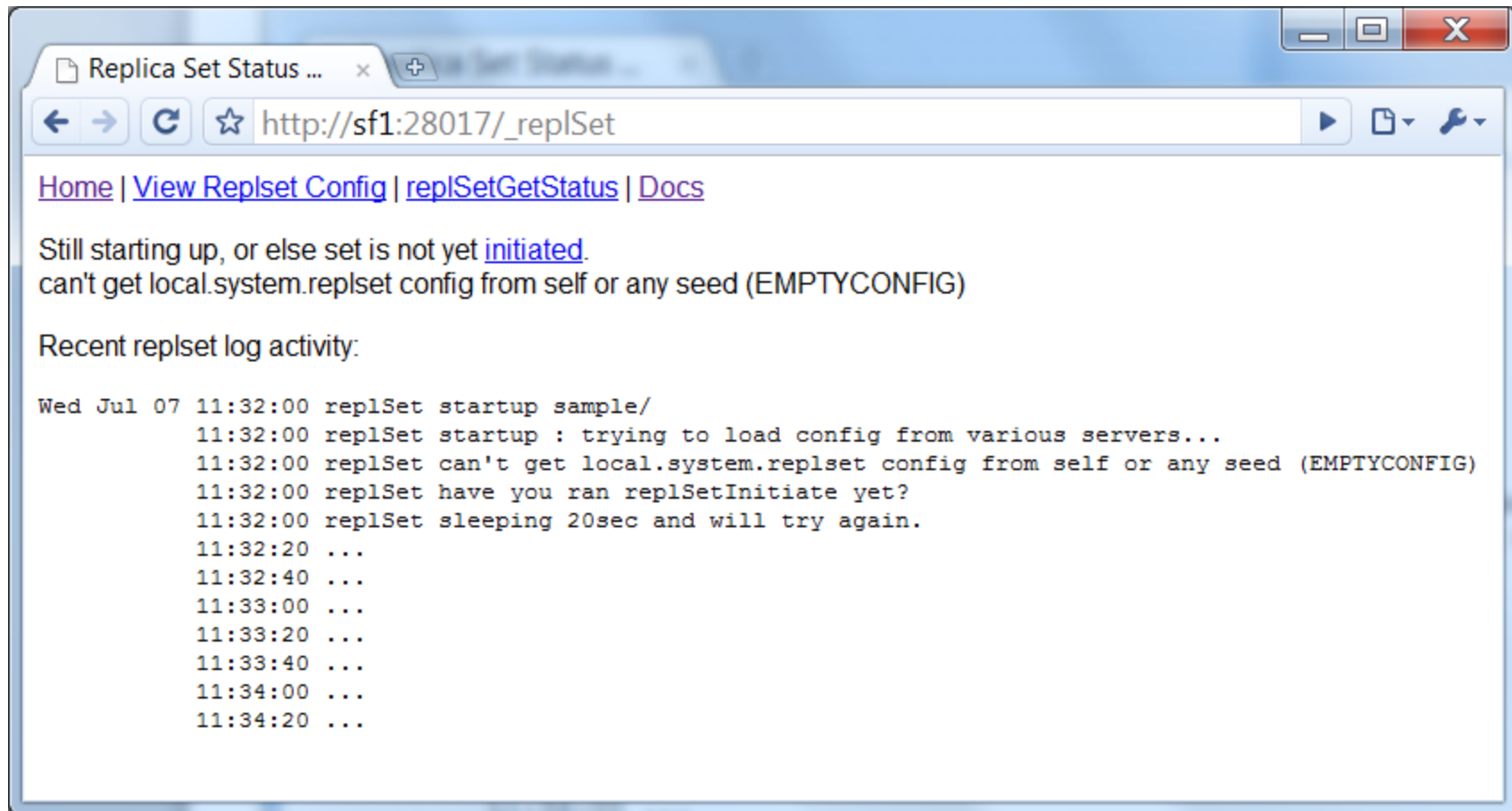
```
sf1$ mongo
> // rs.help()
> rs.status()
{
  "startupStatus" : 3,
  "errmsg" : "can't get local.system.replset (EMPTYCONFIG)",
  "ok" : 0
}
```



We have not yet
initiated the set.

setup

Another way to check status - from a web browser.



setup

Let's initiate the set

```
sf1$ mongo
> rs.initiate()
{
  "info2" : "no configuration explicitly specified -- making one",
  "info"  : "Config now saved locally.  Should come online in about a minute.",
  "ok"    : 1
}
>
```

The initiating member (sf1 here) may have existing data. Other members should start empty.

Checking our status -- this time after successful initiation of the set

```
> rs.status()
{
  "set" : "sample",
  "myState" : 1,
  "members" : [
    {
      "name" : "mb17:27017",
      "self" : true
    }
  ],
  "ok" : 1
}

> db.isMaster()
{
  "ismaster" : true,
  "secondary" : false,
  "hosts" : [
    "sf1"
  ],
  "ok" : 1
}
```

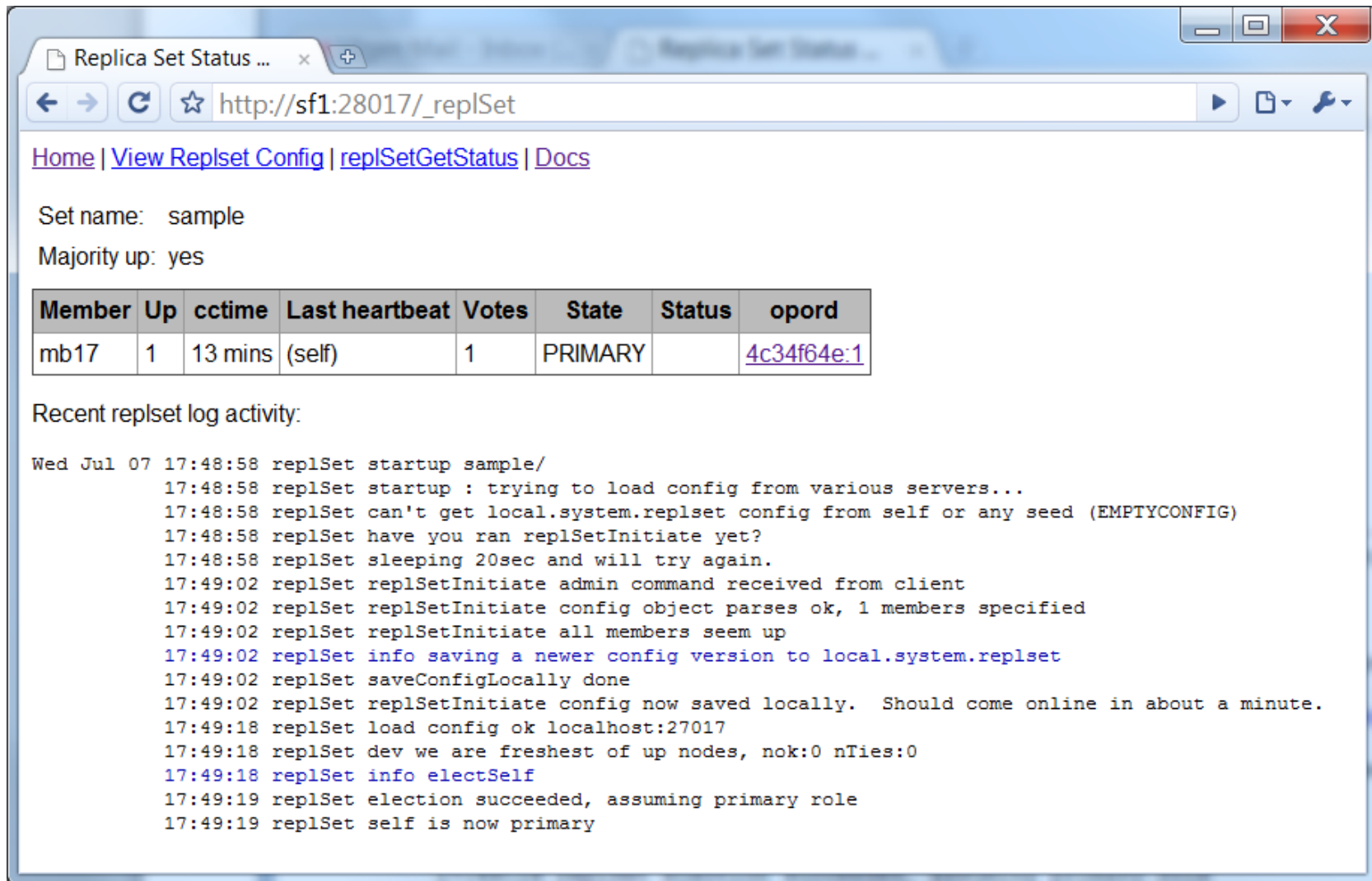
Checking our status -- this time after successful initiation of the set

```
> rs.status()
{
  "set" : "sample",
  "myState" : 1,
  "members" : [
    {
      "name" : "mb17:27017",
      "self" : true
    }
  ],
  "ok" : 1
}

> db.isMaster()
{
  "ismaster" : true,
  "secondary" : false,
  "hosts" : [
    "sf1"
  ],
  "ok" : 1
}
```

sf1 is primary ("master"), which makes sense, as it is the only member in the set so far.

Checking status from the web (/_replSet)



Replica Set Status ... x

← → ↻ ☆ http://sf1:28017/_replSet

[Home](#) | [View Replset Config](#) | [replSetGetStatus](#) | [Docs](#)

Set name: sample
Majority up: yes

Member	Up	cctime	Last heartbeat	Votes	State	Status	opord
mb17	1	13 mins	(self)	1	PRIMARY		4c34f64e:1

Recent replset log activity:

```
Wed Jul 07 17:48:58 replSet startup sample/  
17:48:58 replSet startup : trying to load config from various servers...  
17:48:58 replSet can't get local.system.replset config from self or any seed (EMPTYCONFIG)  
17:48:58 replSet have you ran replSetInitiate yet?  
17:48:58 replSet sleeping 20sec and will try again.  
17:49:02 replSet replSetInitiate admin command received from client  
17:49:02 replSet replSetInitiate config object parses ok, 1 members specified  
17:49:02 replSet replSetInitiate all members seem up  
17:49:02 replSet info saving a newer config version to local.system.replset  
17:49:02 replSet saveConfigLocally done  
17:49:02 replSet replSetInitiate config now saved locally. Should come online in about a minute.  
17:49:18 replSet load config ok localhost:27017  
17:49:18 replSet dev we are freshest of up nodes, nok:0 nTies:0  
17:49:18 replSet info electSelf  
17:49:19 replSet election succeeded, assuming primary role  
17:49:19 replSet self is now primary
```

setup

Ready to use now. Some sample operations.

```
sf1$ mongo
> db
test
> db.foo.insert({x:1})
> db.foo.insert({x:3})
> db.foo.update({x:3},{ $set:{y:'abc'}})
> db.foo.findOne()
{ "_id" : ObjectId("4c34f9d36612000000003f30"), "x" : 1 }
>
```

using

Write operations are logged to local.oplog.rs

Replication oplog

Server : mb17
ns : local.oplog.rs

ts	h	op	ns	rest
Wed Jul 07 17:49:02 2010:1	0	"n"	""	o: { msg: "initiating set" }
Wed Jul 07 18:04:04 2010:1	f84971400000011	"i"	"test.foo"	o: { _id: ObjectId('4c34f9d36612000000003f30'), x: 1.0 }
Wed Jul 07 18:04:05 2010:1	df2da21000093f4	"i"	"test.foo"	o: { _id: ObjectId('4c34f9d56612000000003f31'), x: 3.0 }
Wed Jul 07 18:04:31 2010:1	672425f2050713ad	"u"	"test.foo"	o2: { _id: ObjectId('4c34f9d56612000000003f31') } o: { \$set: { y: "abc" } }

Log length in time: 0.258056 hours

using

Let's now add a second member sf2 to the set. (Obviously a replica set is not very useful with only one member!)

```
sf2$ # first, start a mongod instance on sf2  
sf2$ mongod --rest --replSet myset
```

reconf

Now let's tell the existing set (which was only sf1 before) about sf2's membership.

```
$ mongo --host sf1

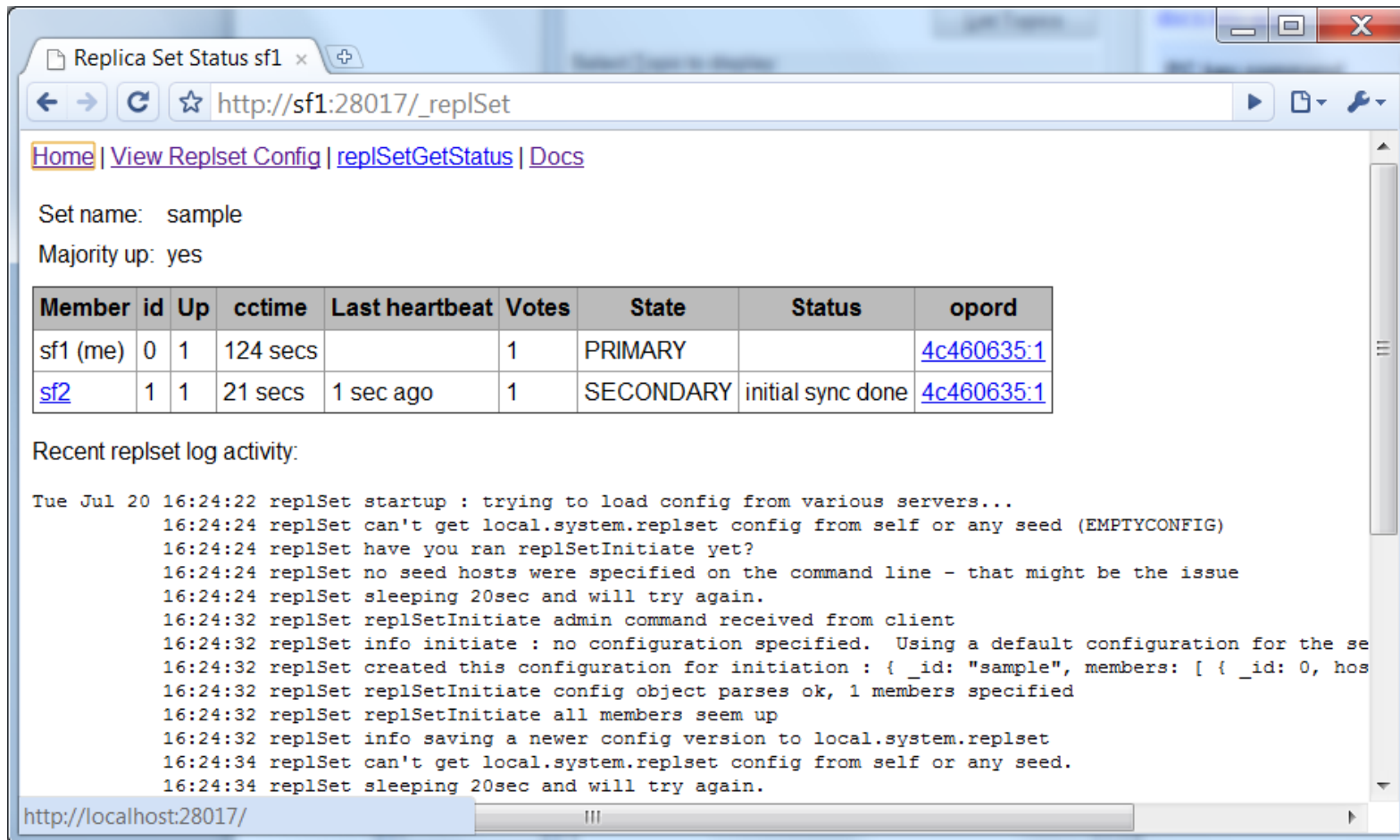
> rs.add("sf2");
{ "ok" : 1 }

> rs.conf()
{
  "_id" : "myset",
  "version" : 2,
  "members" : [
    {
      "_id" : 0,
      "host" : "sf1"
    },
    {
      "_id" : 1,
      "host" : "sf2"
    }
  ]
}

>
```

reconf

Web status page is a bit more interesting now.



Replica Set Status sf1 x

http://sf1:28017/_replSet

[Home](#) | [View Replset Config](#) | [replSetGetStatus](#) | [Docs](#)

Set name: sample
Majority up: yes

Member	id	Up	cctime	Last heartbeat	Votes	State	Status	opord
sf1 (me)	0	1	124 secs		1	PRIMARY		4c460635:1
sf2	1	1	21 secs	1 sec ago	1	SECONDARY	initial sync done	4c460635:1

Recent replset log activity:

```
Tue Jul 20 16:24:22 replSet startup : trying to load config from various servers...
16:24:24 replSet can't get local.system.replset config from self or any seed (EMPTYCONFIG)
16:24:24 replSet have you ran replSetInitiate yet?
16:24:24 replSet no seed hosts were specified on the command line - that might be the issue
16:24:24 replSet sleeping 20sec and will try again.
16:24:32 replSet replSetInitiate admin command received from client
16:24:32 replSet info initiate : no configuration specified. Using a default configuration for the se
16:24:32 replSet created this configuration for initiation : { _id: "sample", members: [ { _id: 0, hos
16:24:32 replSet replSetInitiate config object parses ok, 1 members specified
16:24:32 replSet replSetInitiate all members seem up
16:24:32 replSet info saving a newer config version to local.system.replset
16:24:34 replSet can't get local.system.replset config from self or any seed.
16:24:34 replSet sleeping 20sec and will try again.
```

http://localhost:28017/

reconf

Web status page is a bit more interesting now.

Even number of votes is not helpful for forming proper majorities. So let's add an *arbiter*.

Replica Set Status sf1

http://sf1:28017/_replSet

[Home](#) | [View Replset Config](#) | [replSetGetStatus](#) | [Docs](#)

Set name: sample
Majority up: yes

Member	id	Up	cctime	Last heart beat	Votes	State	Status	opord
sf1 (me)	0	1	124 secs		1	PRIMARY		4c460635:1
sf2	1	1	21 secs	1 sec ago	1	SECONDARY	initial sync done	4c460635:1

Recent replset log activity:

```
Tue Jul 20 16:24:22 replSet startup : trying to load config from various servers...
16:24:24 replSet can't get local.system.replset config from self or any seed (EMPTYCONFIG)
16:24:24 replSet have you ran replSetInitiate yet?
16:24:24 replSet no seed hosts were specified on the command line - that might be the issue
16:24:24 replSet sleeping 20sec and will try again.
16:24:32 replSet replSetInitiate admin command received from client
16:24:32 replSet info initiate : no configuration specified. Using a default configuration for the se
16:24:32 replSet created this configuration for initiation : { _id: "sample", members: [ { _id: 0, hos
16:24:32 replSet replSetInitiate config object parses ok, 1 members specified
16:24:32 replSet replSetInitiate all members seem up
16:24:32 replSet info saving a newer config version to local.system.replset
16:24:34 replSet can't get local.system.replset config from self or any seed.
16:24:34 replSet sleeping 20sec and will try again.
```

http://localhost:28017/

arbiter

An arbiter is a server which monitors the other members and breaks ties in elections. We'll make sf3 our arbiter.

Arbiters consume very little resources and can run on almost any machine (32 bit machines work well).

```
sf3$ mongod --rest --replSet myset --oplogSize 8
```

arbiter

An arbiter is a server which monitors the other members and breaks ties in elections. We'll make sf3 our arbiter.

Arbiters consume very little resources and can run on almost any machine (32 bit machines work well).

```
sf3$ mongod --rest --replSet myset --oplogSize 8
```

Optional tweak. As we know sf3 is only going to be an arbiter, we make its oplog tiny -- arbiters store a copy of the repl set config in the local database, but that's about it.

arbiter

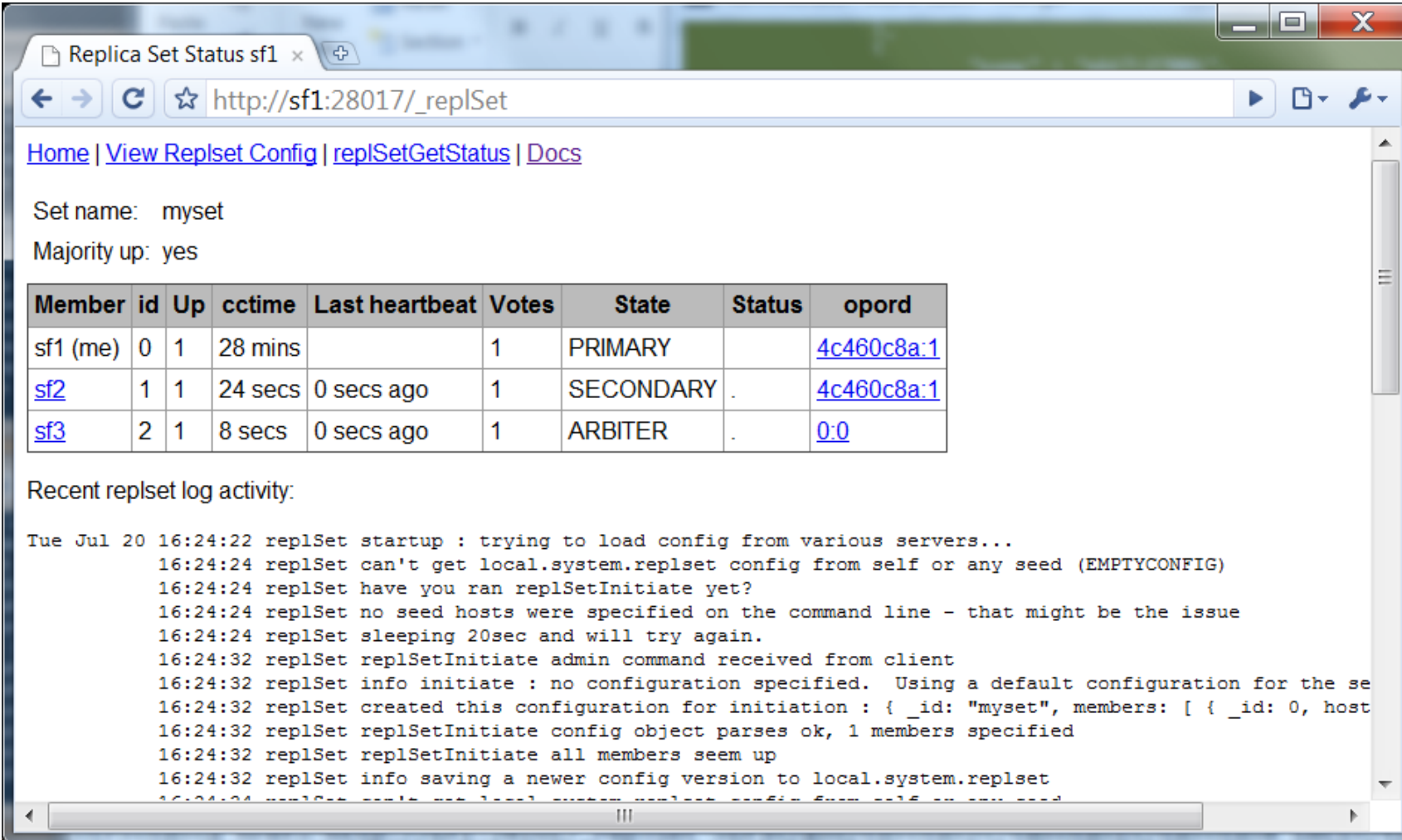
Let's add our new arbiter, node sf3, to the set's configuration.

```
$ mongo --host sf1
> // see current config
> rs.conf()
{
  "_id" : "myset",
  "version" : 2,
  "members" : [
    {
      "_id" : 0,
      "host" : "sf1"
    },
    {
      "_id" : 1,
      "host" : "sf2"
    }
  ]
}

> rs.add( { _id:2, host:"sf3", arbiterOnly:true } )
{ "ok" : 1 }
```

arbiter

Let's check our web status page again



The screenshot shows a web browser window with the address bar displaying `http://sf1:28017/_replSet`. The page content includes navigation links, set information, a table of replica set members, and recent log activity.

Set name: myset
Majority up: yes

Member	id	Up	cctime	Last heartbeat	Votes	State	Status	opord
sf1 (me)	0	1	28 mins		1	PRIMARY		4c460c8a:1
sf2	1	1	24 secs	0 secs ago	1	SECONDARY	.	4c460c8a:1
sf3	2	1	8 secs	0 secs ago	1	ARBITER	.	0:0

Recent replset log activity:

```
Tue Jul 20 16:24:22 replSet startup : trying to load config from various servers...
16:24:24 replSet can't get local.system.replset config from self or any seed (EMPTYCONFIG)
16:24:24 replSet have you ran replSetInitiate yet?
16:24:24 replSet no seed hosts were specified on the command line - that might be the issue
16:24:24 replSet sleeping 20sec and will try again.
16:24:32 replSet replSetInitiate admin command received from client
16:24:32 replSet info initiate : no configuration specified. Using a default configuration for the se
16:24:32 replSet created this configuration for initiation : { _id: "myset", members: [ { _id: 0, host
16:24:32 replSet replSetInitiate config object parses ok, 1 members specified
16:24:32 replSet replSetInitiate all members seem up
16:24:32 replSet info saving a newer config version to local.system.replset
16:24:34 replSet can't get local.system.replset config from self or any seed
```

using

If we take sf1 down, the set fails over to sf2.

Home | [View Replset Config](#) | [replSetGetStatus](#) | [Docs](#)

Set name: myset
Majority up: yes

Member	id	Up	cctime	Last heartbeat	Votes	State	Status	opord
sf1	0	0		4.3 mins ago	1		connect/transport error	4c460c8a:1
sf2 (me)	1	1	28 mins		1	PRIMARY	.	4c460c8a:1
sf3	2	1	18 mins	0 secs ago	1	ARBITER	.	0:0

Recent replset log activity:

```
Tue Jul 20 16:24:22 replSet startup : trying to load config from various servers...
16:24:24 replSet can't get local.system.replset config from self or any seed (EMPTYCONFIG)
16:24:24 replSet have you ran replSetInitiate yet?
16:24:24 replSet no seed hosts were specified on the command line - that might be the issue
16:24:24 replSet sleeping 20sec and will try again.
16:24:32 replSet replSetInitiate admin command received from client
16:24:32 replSet info initiate : no configuration specified. Using a default configuration
16:24:32 replSet created this configuration for initiation : { _id: "myset", members: [ { _id: "sf1", host: "sf1:28017", role: "ARBITER", priority: 1, votes: 1, _type: "shard" }, { _id: "sf2", host: "sf2:28017", role: "PRIMARY", priority: 1, votes: 1, _type: "shard" }, { _id: "sf3", host: "sf3:28017", role: "ARBITER", priority: 1, votes: 1, _type: "shard" } ] }
```

using

More Information

- <http://www.mongodb.org/display/DOCS/Replica+Sets>
- Try these in the shell:
 - `rs.help()`
 - `rs.status()`
 - `rs.conf()`
 - `db.isMaster()`

see also