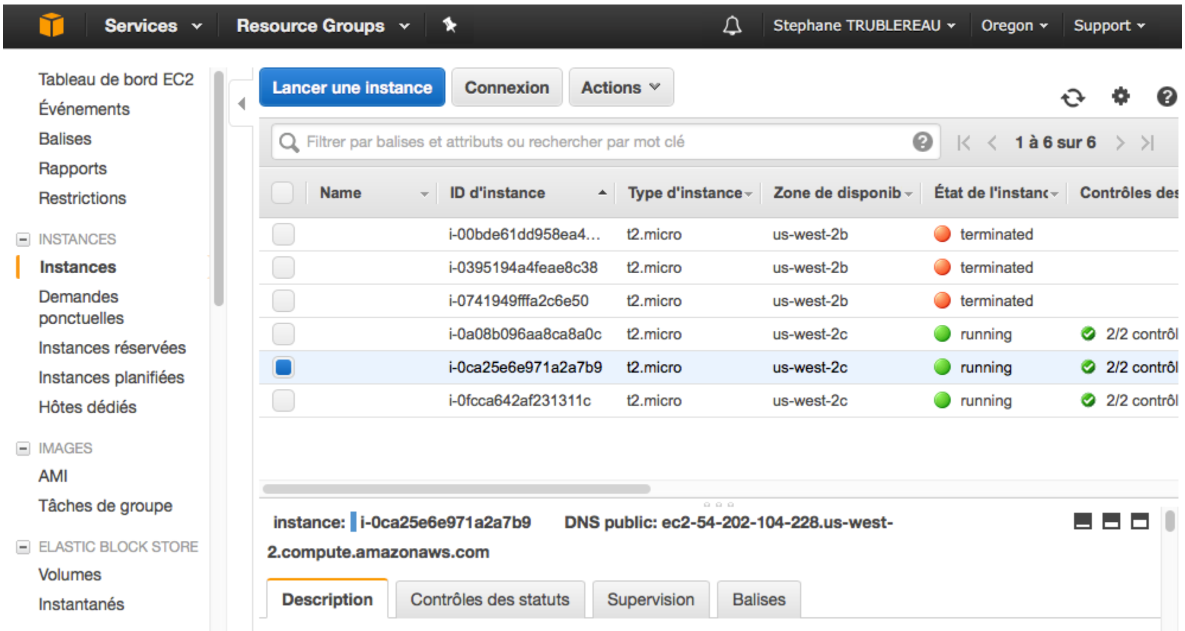
AWS-NOSQL

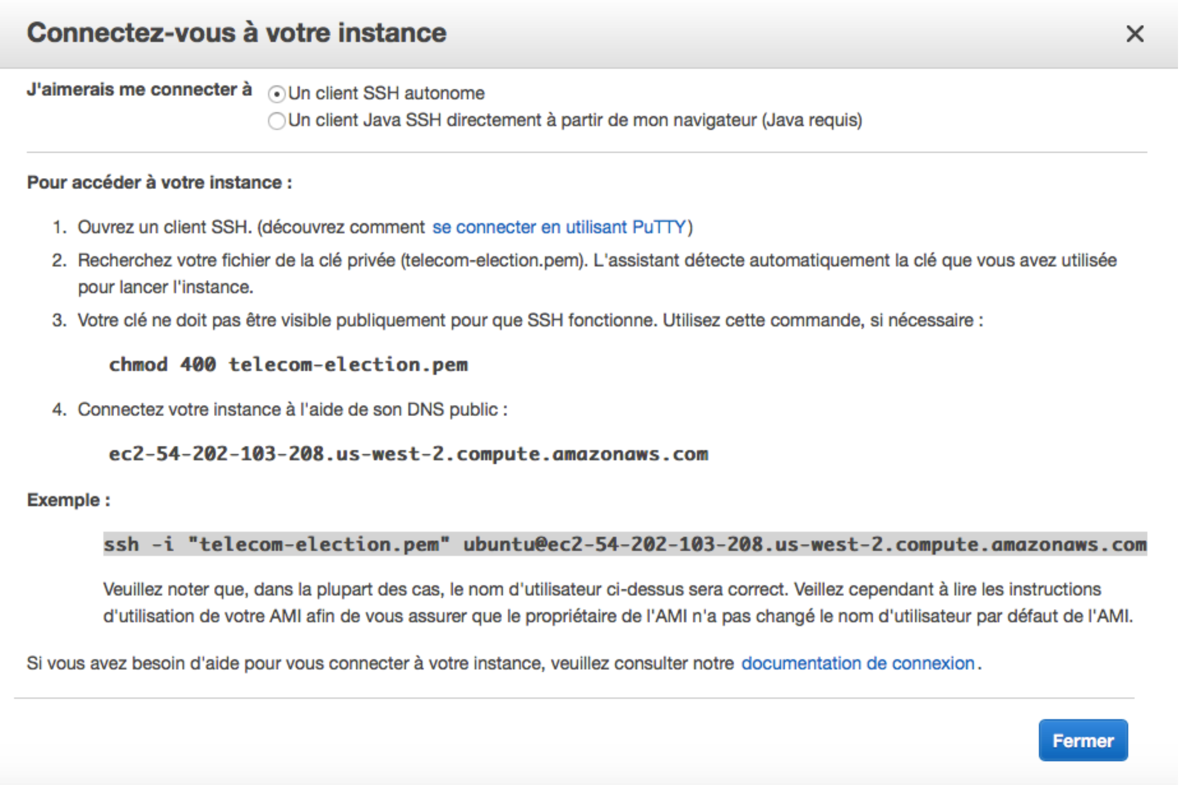
Se mettre sous EC2



	Name	ID d'instance	Type d'instance	Zone de disponib	État de l'instanc	Contrôles des
□		i-00bde61dd958ea4…	t2.micro	us-west-2b	● terminated	
□		i-0395194a4feae8c38	t2.micro	us-west-2b	● terminated	
□		i-0741949fffa2c6e50	t2.micro	us-west-2b	● terminated	
□		i-0a08b096aa8ca8a0c	t2.micro	us-west-2c	● running	✓ 2/2 contrôl
■		i-0ca25e6e971a2a7b9	t2.micro	us-west-2c	● running	✓ 2/2 contrôl
□		i-0fcca642af231311c	t2.micro	us-west-2c	● running	✓ 2/2 contrôl

instance: i-0ca25e6e971a2a7b9 DNS public: ec2-54-202-104-228.us-west-2.compute.amazonaws.com

aller sur connexion d'une instance



Connectez-vous à votre instance ✕

J'aimerais me connecter à ● Un client SSH autonome
○ Un client Java SSH directement à partir de mon navigateur (Java requis)

Pour accéder à votre instance :

1. Ouvrez un client SSH. (découvrez comment se connecter en utilisant PuTTY)
2. Recherchez votre fichier de la clé privée (telecom-election.pem). L'assistant détecte automatiquement la clé que vous avez utilisée pour lancer l'instance.
3. Votre clé ne doit pas être visible publiquement pour que SSH fonctionne. Utilisez cette commande, si nécessaire :

```
chmod 400 telecom-election.pem
```

4. Connectez votre instance à l'aide de son DNS public :

```
ec2-54-202-103-208.us-west-2.compute.amazonaws.com
```

Exemple :

```
ssh -i "telecom-election.pem" ubuntu@ec2-54-202-103-208.us-west-2.compute.amazonaws.com
```

Veuillez noter que, dans la plupart des cas, le nom d'utilisateur ci-dessus sera correct. Veillez cependant à lire les instructions d'utilisation de votre AMI afin de vous assurer que le propriétaire de l'AMI n'a pas changé le nom d'utilisateur par défaut de l'AMI.

Si vous avez besoin d'aide pour vous connecter à votre instance, veuillez consulter notre documentation de connexion.

Option : A faire une fois
si vous avez mis la clé sous .ssh commande de type (nom du fichier elecom-election.pem)

Sous terminal poste local :

exemple :

```
 dhcpwifi-22-89:~ stephanetrublereau$ ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-202-104-228.us-west-2.compute.amazonaws.com
The authenticity of host 'ec2-54-202-104-228.us-west-2.compute.amazonaws.com (54.202.104.228)' can't be established.
ECDSA key fingerprint is SHA256:Vaz1yT23tx5hqLOn7nfexsVtSmkR6tx7XyhCCuKK9Eo.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-54-202-104-228.us-west-2.compute.amazonaws.com,54.202.104.228' (ECDSA) to the list of known hosts.

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
5 package(s) needed for security, out of 9 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-14-140 ~]$
```

```
ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-244-43-231.us-west-2.compute.amazonaws.com
ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-244-44-107.us-west-2.compute.amazonaws.com
ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-149-76-149.us-west-2.compute.amazonaws.com
```

```
ssh -i "telecom-election.pem" ec2-user@ec2-35-160-27-42.us-west-2.compute.amazonaws.com
```

Installation de mongoDB

```
pwd
/home/ubuntu/
```

Se mettre sous le repertoire

```
cd /etc
```

Suivre :

https://docs.mongodb.com/v3.0/tutorial/install-mongodb-on-amazon/

```
sudo vi /etc/yum.repos.d/mongodb-org-3.0.repo
```

Sous vi éditer et ajouter les lignes suivantes

```
i

[mongodb-org-3.0]
name=MongoDB Repository
baseurl=https://repo.mongodb.org/yum/amazon/2013.03/mongodb-org/3.0/x86_64/
gpgcheck=0
enabled=1

:wq
```

```
sudo yum install -y mongodb-org
```

exemple :

```
[ec2-user@ip-172-31-14-140 etc]$ sudo yum install -y mongodb-org
Modules complémentaires chargés : priorities, update-motd, upgrade-helper
amzn-main/latest                                      l 2.1 kB     00:00
amzn-updates/latest                                    l 2.3 kB     00:00
mongodb-org-3.0                                        l 2.5 kB     00:00
mongodb-org-3.0/primary_db                               l  83 kB     00:00
Résolution des dépendances
--> Lancement de la transaction de test
---> Le paquet mongodb-org.x86_64 0:3.0.14-1.amzn1 sera installé
--> Traitement de la dépendance : mongodb-org-tools = 3.0.14 pour le paquet :
mongodb-org-3.0.14-1.amzn1.x86_64
--> Traitement de la dépendance : mongodb-org-shell = 3.0.14 pour le paquet :
mongodb-org-3.0.14-1.amzn1.x86_64
--> Traitement de la dépendance : mongodb-org-server = 3.0.14 pour le paquet :
mongodb-org-3.0.14-1.amzn1.x86_64
--> Traitement de la dépendance : mongodb-org-mongos = 3.0.14 pour le paquet :
mongodb-org-3.0.14-1.amzn1.x86_64
--> Lancement de la transaction de test
---> Le paquet mongodb-org-mongos.x86_64 0:3.0.14-1.amzn1 sera installé
---> Le paquet mongodb-org-server.x86_64 0:3.0.14-1.amzn1 sera installé
---> Le paquet mongodb-org-shell.x86_64 0:3.0.14-1.amzn1 sera installé
---> Le paquet mongodb-org-tools.x86_64 0:3.0.14-1.amzn1 sera installé
--> Résolution des dépendances terminée

Dépendances résolues

==============================================================
====================
 Package              Architecture
                          Version           Dépôt              Taille
==============================================================
====================
Installation :
 mongodb-org            x86_64      3.0.14-1.amzn1      mongodb-org-3.0      5.8 k
Installation pour dépendances :
 mongodb-org-mongos      x86_64      3.0.14-1.amzn1      mongodb-org-3.0      4.1 M
 mongodb-org-server      x86_64      3.0.14-1.amzn1      mongodb-org-3.0      8.6 M
 mongodb-org-shell       x86_64      3.0.14-1.amzn1      mongodb-org-3.0      4.4 M
 mongodb-org-tools       x86_64      3.0.14-1.amzn1      mongodb-org-3.0       28 M

Résumé de la transaction
================================================================================
====================
Installation   1 Paquet (+4 Paquets en dépendance)

Taille totale des téléchargements : 45 M
Taille d'installation : 118 M
Downloading packages:
(1/5): mongodb-org-3.0.14-1.amzn1.x86_64.rpm                l 5.8 kB     00:00
(2/5): mongodb-org-mongos-3.0.14-1.amzn1.x86_64.rpm          l 4.1 MB     00:00
(3/5): mongodb-org-server-3.0.14-1.amzn1.x86_64.rpm          l 8.6 MB     00:00
(4/5): mongodb-org-shell-3.0.14-1.amzn1.x86_64.rpm           l 4.4 MB     00:00
(5/5): mongodb-org-tools-3.0.14-1.amzn1.x86_64.rpm           l  28 MB     00:00
--------------------------------------------------------------------------------
Total                                        17 MB/s l  45 MB  00:02
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installation : mongodb-org-mongos-3.0.14-1.amzn1.x86_64                    1/5
  Installation : mongodb-org-tools-3.0.14-1.amzn1.x86_64                     2/5
  Installation : mongodb-org-shell-3.0.14-1.amzn1.x86_64                     3/5
  Installation : mongodb-org-server-3.0.14-1.amzn1.x86_64                    4/5
  Installation : mongodb-org-3.0.14-1.amzn1.x86_64                           5/5
  Vérification : mongodb-org-server-3.0.14-1.amzn1.x86_64                    1/5
  Vérification : mongodb-org-3.0.14-1.amzn1.x86_64                           2/5
  Vérification : mongodb-org-shell-3.0.14-1.amzn1.x86_64                     3/5
  Vérification : mongodb-org-tools-3.0.14-1.amzn1.x86_64                     4/5
  Vérification : mongodb-org-mongos-3.0.14-1.amzn1.x86_64                    5/5

Installé :
  mongodb-org.x86_64 0:3.0.14-1.amzn1

Dépendances installées :
  mongodb-org-mongos.x86_64 0:3.0.14-1.amzn1
  mongodb-org-server.x86_64 0:3.0.14-1.amzn1
  mongodb-org-shell.x86_64 0:3.0.14-1.amzn1
  mongodb-org-tools.x86_64 0:3.0.14-1.amzn1

Terminé !
[ec2-user@ip-172-31-14-140 etc]$
```

```
sudo vi mongod.conf
```

modifier la ligne contenant l’IP 127.0.0.1 par 0.0.0.0

```
:wq
```

creation d’un dossier data/db à la racine

```
sudo mkdir -p /data/db
```

```
sudo service mongod restart
```

ajouter terminal

```
mongo
```

créer une base de données

```
use election
```

```
db.createUser({ user:'strublereau', pwd: 'azerty', roles:[{role: 'readWrite', db:'election'}]})
```

```
> db.createUser({
... user: 'strublereau',
... pwd: 'azerty',
... roles: [{role: 'readWrite', db: 'election'}]})

Successfully added user: {
        "user" : "strublereau",
        "roles" : [
                {
                        "role" : "readWrite",
                        "db" : "election"
                }
        ]
}
# creer aun enregistrement
db.test.insert({name :
db.test.find()
```

Se mettre sur terminal poste local : (si mongo est installe en local

```
mongo -u strublereau -p azerty 54.202.224.44/election
```

Replicat des trois instances

Essayer de reprendre la procedure

1- se connecter sur 3 terminaux sur les trois serveurs :

Un exemple :

```
dhcpwifi-22-89:~ stephanetrublereau$ ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-214-213-143.us-west-2.compute.amazonaws.com
Last login: Fri Jan  6 15:12:02 2017 from dhcpwifi-22-89.enst.fr

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
```

2- sur chaque terminal : Modifier /etc/mongod.conf pour indiquer le path data

```
[ec2-user@ip-172-31-12-45 /]$ sudo vi /etc/mongod.conf
```

dbpath=/data *(au lieu de var/lib/mongo)*

3- sur les trois terminaux

mongo

(si message Please make at least 3379MB available in /data/db/journal or use --smallfiles)

sudo mongod --replSet rs0 --smallfiles dans notre cas sur le deuxième **(**ec2-54-201-210-236)

4 ouvrir un quatrième terminal et reconnection sur le master :

```
===============
dhcpwifi-22-108:~ stephanetrublereau$ source .bashrc
dhcpwifi-22-108:~ stephanetrublereau$ aws1
Last login: Tue Jan 10 09:50:25 2017 from dhcpwifi-22-108.enst.fr

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
[ec2-user@ip-172-31-20-9 ~]$ mongo

MongoDB shell version: 3.0.14
connecting to: test
Server has startup warnings:
2017-01-10T10:20:04.212+0000 I CONTROL  [initandlisten] ** WARNING: You are running this process as the root user, which is not recommended.
2017-01-10T10:20:04.212+0000 I CONTROL  [initandlisten]
2017-01-10T10:20:04.212+0000 I CONTROL  [initandlisten]
2017-01-10T10:20:04.212+0000 I CONTROL  [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/defrag is 'always'.
2017-01-10T10:20:04.212+0000 I CONTROL  [initandlisten] **        We suggest setting it to 'never'
2017-01-10T10:20:04.212+0000 I CONTROL  [initandlisten]
> use election
```

```
switched to db election
> db.getUser("strublereau")   # user nest pas creep sur le cluster
2017-01-10T10:21:56.548+0000 E QUERY    Error: not master
    at Error (<anonymous>)
    at DB.getUser (src/mongo/shell/db.js:1321:15)
    at (shell):1:4 at src/mongo/shell/db.js:1321
> rs.initiate()
{
        "info2" : "no configuration explicitly specified -- making one",
        "me" : "ip-172-31-20-9:27017",
        "ok" : 1
}
rs0:OTHER> rs.conf()
{
        "_id" : "rs0",
        "version" : 1,
        "members" : [
                {
                        "_id" : 0,
                        "host" : "ip-172-31-20-9:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                        "slaveDelay" : 0,
                        "votes" : 1
                }
        ],
        "settings" : {
                "chainingAllowed" : true,
                "heartbeatTimeoutSecs" : 10,
                "getLastErrorModes" : {

                },
                "getLastErrorDefaults" : {
                        "w" : 1,
                        "wtimeout" : 0
                }
        }
}
```

# pensez à ajouter une règle au niveau du groupe de sécurité : IP27017
Ajouter print écran

```
rs0:PRIMARY> rs.add("ip-172-31-31-201:27017")
{ "ok" : 1 }
```

```
rs0:PRIMARY> rs.add("ip-172-31-22-186:27017")
{ "ok" : 1 }
rs0:PRIMARY> rs.conf()
{
        "_id" : "rs0",
        "version" : 3,
        "members" : [
                {
                        "_id" : 0,
                        "host" : "ip-172-31-20-9:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                        "slaveDelay" : 0,
                        "votes" : 1
                },
                {
                        "_id" : 1,
                        "host" : "ip-172-31-31-201:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                        "slaveDelay" : 0,
                        "votes" : 1
                },
                {
                        "_id" : 2,
                        "host" : "ip-172-31-22-186:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                        "slaveDelay" : 0,
                        "votes" : 1
                }
        ],
        "settings" : {
                "chainingAllowed" : true,
                "heartbeatTimeoutSecs" : 10,
                "getLastErrorModes" : {

                },
                "getLastErrorDefaults" : {
                        "w" : 1,
                        "wtimeout" : 0
                }
        }
}
rs0:PRIMARY> rs.status()
{
        "set" : "rs0",
        "date" : ISODate("2017-01-10T10:24:37.184Z"),
        "myState" : 1,
        "members" : [
                {
                        "_id" : 0,
                        "name" : "ip-172-31-20-9:27017",
                        "health" : 1,
                        "state" : 1,
                        "stateStr" : "PRIMARY",
                        "uptime" : 273,
                        "optime" : Timestamp(1484043862, 1),
                        "optimeDate" : ISODate("2017-01-10T10:24:22Z"),
                        "electionTime" : Timestamp(1484043749, 2),
                        "electionDate" : ISODate("2017-01-10T10:22:29Z"),
                        "configVersion" : 3,
                        "self" : true
                },
                {
                        "_id" : 1,
                        "name" : "ip-172-31-31-201:27017",
                        "health" : 1,
                        "state" : 2,
                        "stateStr" : "SECONDARY",
                        "uptime" : 43,
                        "optime" : Timestamp(1484043862, 1),
                        "optimeDate" : ISODate("2017-01-10T10:24:22Z"),
                        "lastHeartbeat" : ISODate("2017-01-10T10:24:36.127Z"),
                        "lastHeartbeatRecv" : ISODate("2017-01-10T10:24:35.843Z"),
                        "pingMs" : 0,
                        "syncingTo" : "ip-172-31-20-9:27017",
                        "configVersion" : 3
                },
                {
                        "_id" : 2,
                        "name" : "ip-172-31-22-186:27017",
                        "health" : 1,
                        "state" : 2,
                        "stateStr" : "SECONDARY",
                        "uptime" : 15,
                        "optime" : Timestamp(1484043862, 1),
                        "optimeDate" : ISODate("2017-01-10T10:24:22Z"),
                        "lastHeartbeat" : ISODate("2017-01-10T10:24:36.129Z"),
                        "lastHeartbeatRecv" : ISODate("2017-01-10T10:24:36.138Z"),
                        "pingMs" : 0,
                        "syncingTo" : "ip-172-31-20-9:27017",
                        "configVersion" : 3
                }
        ],
        "ok" : 1
}
# Création du user
rs0:PRIMARY> db.createUser({ user:'strublereau', pwd: 'azerty', roles:[{role:
'readWrite', db:'election'}]})
Successfully added user: {
        "user" : "strublereau",
        "roles" : [
                {
                        "role" : "readWrite",
                        "db" : "election"
                }
        ]
}
rs0:PRIMARY> db.test.insert({name:"pierre"})
WriteResult({ "nInserted" : 1 })
rs0:PRIMARY> db.test.find()
{ "_id" : ObjectId("5874b70e9df975d2ac231291"), "name" : "pierre" }
rs0:PRIMARY> cfg=rs.conf()
{
        "_id" : "rs0",
        "version" : 3,
        "members" : [
                {
                        "_id" : 0,
                        "host" : "ip-172-31-20-9:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                        "slaveDelay" : 0,
                        "votes" : 1
                },
                {
```

```
			"_id" : 1,
			"host" : "ip-172-31-31-201:27017",
			"arbiterOnly" : false,
			"buildIndexes" : true,
			"hidden" : false,
			"priority" : 1,
			"tags" : {

			},
			"slaveDelay" : 0,
			"votes" : 1
		},
		{
			"_id" : 2,
			"host" : "ip-172-31-22-186:27017",
			"arbiterOnly" : false,
			"buildIndexes" : true,
			"hidden" : false,
			"priority" : 1,
			"tags" : {

			},
			"slaveDelay" : 0,
			"votes" : 1
		}
	],
	"settings" : {
		"chainingAllowed" : true,
		"heartbeatTimeoutSecs" : 10,
		"getLastErrorModes" : {

		},
		"getLastErrorDefaults" : {
			"w" : 1,
			"wtimeout" : 0
		}
	}
}

rs0:PRIMARY> cfg.members[0].host = "172.31.31.100:27017"
54.187.196.92:27017
rs0:PRIMARY> cfg.members[1].host = "172.31.31.101:27017"
54.201.210.236:27017
rs0:PRIMARY> cfg.members[2].host = "172.31.31.102:27017"
54.218.17.6:27017

172.31.12.45
rs0:PRIMARY> cfg
{
```

```
"_id" : "rs0",
"version" : 3,
"members" : [
        {
                "_id" : 0,
                "host" : "172.31.31.100:27017",
                "arbiterOnly" : false,
                "buildIndexes" : true,
                "hidden" : false,
                "priority" : 1,
                "tags" : {

                },
                "slaveDelay" : 0,
                "votes" : 1
        },
        {
                "_id" : 1,
                "host" : "172.31.31.101:27017",
                "arbiterOnly" : false,
                "buildIndexes" : true,
                "hidden" : false,
                "priority" : 1,
                "tags" : {

                },
                "slaveDelay" : 0,
                "votes" : 1
        },
        {
                "_id" : 2,
                "host" : "172.31.31.102:27017",
                "arbiterOnly" : false,
                "buildIndexes" : true,
                "hidden" : false,
                "priority" : 1,
                "tags" : {

                },
                "slaveDelay" : 0,
                "votes" : 1
        }
],
"settings" : {
        "chainingAllowed" : true,
        "heartbeatTimeoutSecs" : 10,
        "getLastErrorModes" : {

        },
                "getLastErrorDefaults" : {
                        "w" : 1,
                        "wtimeout" : 0
                }
        }
}
rs0:PRIMARY> rs.reconfig(cfg)
{ "ok" : 1 }
rs0:PRIMARY> rs.conf()
{
        "_id" : "rs0",
        "version" : 4,
        "members" : [
                {
                        "_id" : 0,
                        "host" : "172.31.31.100:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                        "slaveDelay" : 0,
                        "votes" : 1
                },
                {
                        "_id" : 1,
                        "host" : "172.31.31.101:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                        "slaveDelay" : 0,
                        "votes" : 1
                },
                {
                        "_id" : 2,
                        "host" : "172.31.31.102:27017",
                        "arbiterOnly" : false,
                        "buildIndexes" : true,
                        "hidden" : false,
                        "priority" : 1,
                        "tags" : {

                        },
                "slaveDelay" : 0,
                "votes" : 1
            }
        ],
        "settings" : {
            "chainingAllowed" : true,
            "heartbeatTimeoutSecs" : 10,
            "getLastErrorModes" : {

            },
            "getLastErrorDefaults" : {
                "w" : 1,
                "wtimeout" : 0
            }
        }
}
```

6- On arrête le primaire (CTRL + C) sur le terminal du replicat et on le redémarre

```
rs0:PRIMARY> rs.status()
2017-01-10T10:41:38.161+0000 I NETWORK  DBClientCursor::init call() failed
2017-01-10T10:41:38.162+0000 E QUERY    Error: error doing query: failed
   at DBQuery._exec (src/mongo/shell/query.js:83:36)
   at DBQuery.hasNext (src/mongo/shell/query.js:240:10)
   at DBCollection.findOne (src/mongo/shell/collection.js:187:19)
   at DB.runCommand (src/mongo/shell/db.js:58:41)
   at DB.adminCommand (src/mongo/shell/db.js:66:41)
   at Function.rs.status (src/mongo/shell/utils.js:937:37)
   at (shell):1:4 at src/mongo/shell/query.js:83
2017-01-10T10:41:38.163+0000 I NETWORK  trying reconnect to 127.0.0.1:27017
(127.0.0.1) failed
2017-01-10T10:41:38.163+0000 I NETWORK  reconnect 127.0.0.1:27017
(127.0.0.1) ok
rs0:SECONDARY> rs.status()
{
        "set" : "rs0",
        "date" : ISODate("2017-01-10T10:41:42.297Z"),
        "myState" : 2,
        "members" : [
            {
                "_id" : 0,
                "name" : "172.31.31.100:27017",
                "health" : 1,
                "state" : 2,
                "stateStr" : "SECONDARY",
                "uptime" : 15,
                "optime" : Timestamp(1484044535, 1),
                "optimeDate" : ISODate("2017-01-10T10:35:35Z"),
                "configVersion" : 4,
                "self" : true
```

```
		},
		{
			"_id" : 1,
			"name" : "172.31.31.101:27017",
			"health" : 1,
			"state" : 2,
			"stateStr" : "SECONDARY",
			"uptime" : 14,
			"optime" : Timestamp(1484044535, 1),
			"optimeDate" : ISODate("2017-01-10T10:35:35Z"),
			"lastHeartbeat" : ISODate("2017-01-10T10:41:41.765Z"),
			"lastHeartbeatRecv" : ISODate("2017-01-10T10:41:40.475Z"),
			"pingMs" : 1,
			"configVersion" : 4
		},
		{
			"_id" : 2,
			"name" : "172.31.31.102:27017",
			"health" : 1,
			"state" : 1,
			"stateStr" : "PRIMARY",
			"uptime" : 14,
			"optime" : Timestamp(1484044535, 1),
			"optimeDate" : ISODate("2017-01-10T10:35:35Z"),
			"lastHeartbeat" : ISODate("2017-01-10T10:41:41.765Z"),
			"lastHeartbeatRecv" : ISODate("2017-01-10T10:41:40.965Z"),
			"pingMs" : 1,
			"electionTime" : Timestamp(1484044875, 1),
			"electionDate" : ISODate("2017-01-10T10:41:15Z"),
			"configVersion" : 4
		}
	],
	"ok" : 1
}
```

7- on remet le premier serveur (arrêt du master ancien secondaire …. et redémarrage sur terminal)

```
rs0:SECONDARY> rs.status()
{
	"set" : "rs0",
	"date" : ISODate("2017-01-10T10:42:40.778Z"),
	"myState" : 1,
	"members" : [
		{
			"_id" : 0,
			"name" : "172.31.31.100:27017",
			"health" : 1,
			"state" : 1,
			"stateStr" : "PRIMARY",
			"uptime" : 73,
				"optime" : Timestamp(1484044535, 1),
				"optimeDate" : ISODate("2017-01-10T10:35:35Z"),
				"electionTime" : Timestamp(1484044941, 1),
				"electionDate" : ISODate("2017-01-10T10:42:21Z"),
				"configVersion" : 4,
				"self" : true
			},
			{
				"_id" : 1,
				"name" : "172.31.31.101:27017",
				"health" : 1,
				"state" : 2,
				"stateStr" : "SECONDARY",
				"uptime" : 73,
				"optime" : Timestamp(1484044535, 1),
				"optimeDate" : ISODate("2017-01-10T10:35:35Z"),
				"lastHeartbeat" : ISODate("2017-01-10T10:42:39.799Z"),
				"lastHeartbeatRecv" : ISODate("2017-01-10T10:42:40.509Z"),
				"pingMs" : 0,
				"configVersion" : 4
			},
			{
				"_id" : 2,
				"name" : "172.31.31.102:27017",
				"health" : 1,
				"state" : 2,
				"stateStr" : "SECONDARY",
				"uptime" : 8,
				"optime" : Timestamp(1484044535, 1),
				"optimeDate" : ISODate("2017-01-10T10:35:35Z"),
				"lastHeartbeat" : ISODate("2017-01-10T10:42:39.808Z"),
				"lastHeartbeatRecv" : ISODate("2017-01-10T10:42:39.958Z"),
				"pingMs" : 0,
				"configVersion" : 4
			}
		],
		"ok" : 1
}
```

8- creation d’enregistrement de vérification

```
rs0:PRIMARY> db.test.insert({name:"romain"})
WriteResult({ "nInserted" : 1 })
rs0:PRIMARY> db.test.insert({name:"stephane"})
WriteResult({ "nInserted" : 1 })
rs0:PRIMARY> db.test.insert({name:"daphne"})
WriteResult({ "nInserted" : 1 })
rs0:PRIMARY> db.test.insert({name:"fares"})
WriteResult({ "nInserted" : 1 })
rs0:PRIMARY> db.test.insert({name:"adam"})
WriteResult({ "nInserted" : 1 })
```

```
rs0:PRIMARY>
```

10- sur chaque serveur mongo

```
sudo vi /etc/mongorc.js

i
rs.slaveOk()

esc
:wq
```

exemple :

```
Last login: Mon Jan  9 17:13:41 on ttys002

dhcpwifi-22-108:~ stephanetrublereau$ ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-244-43-231.us-west-2.compute.amazonaws.com
Last login: Mon Jan  9 16:08:59 2017 from dhcpwifi-22-108.enst.fr

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
[[ec2-user@ip-172-31-7-89 ~]$ sudo vi /etc/mongorc.js

i
rs.slaveOk()

esc
:wq

[ec2-user@ip-172-31-7-89 ~]$ ^C
[ec2-user@ip-172-31-7-89 ~]$ exit
déconnexion
Connection to ec2-54-244-43-231.us-west-2.compute.amazonaws.com closed.

dhcpwifi-22-108:~ stephanetrublereau$ ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-244-44-107.us-west-2.compute.amazonaws.com
Last login: Mon Jan  9 16:08:26 2017 from dhcpwifi-22-108.enst.fr

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
5 package(s) needed for security, out of 9 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-14-140 ~]$ sudo vi /etc/mongorc.js
```

```
i
rs.slaveOk()

esc
:wq
```

```
[ec2-user@ip-172-31-14-140 ~]$ exit
déconnexion
Connection to ec2-54-244-44-107.us-west-2.compute.amazonaws.com closed.

dhcpwifi-22-108:~ stephanetrublereau$ ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-149-76-149.us-west-2.compute.amazonaws.com
Last login: Mon Jan  9 16:14:16 2017 from dhcpwifi-22-108.enst.fr

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
5 package(s) needed for security, out of 9 available
Run "sudo yum update" to apply all updates.

[ec2-user@ip-172-31-12-45 ~]$ sudo vi /etc/mongorc.js
```

```
i
rs.slaveOk()

esc
:wq
```

```
[ec2-user@ip-172-31-12-45 ~]$ exit
déconnexion
Connection to ec2-54-149-76-149.us-west-2.compute.amazonaws.com closed.
```

11- Lancer un notebook python

```
import pymongo

#client = pymongo.MongoClient("mongodb://strublereau:azerty@54.187.196.92:27017,54.201.210.236:27017,54.218.17.6:27017/election?replicaSet=rs0") # defaults to port 27017
client = pymongo.MongoClient("mongodb://drussier:password@52.214.165.62/elections") # defaults to port 27017
print(client.election.test.find())
curs = client.election.test.find()
for i in curs :
    print(i)
    print(type(i))
```

ANNEXES :
Redémarrage du cluster

Sous repertoire perso :

```
dhcpwifi-23-55:~ stephanetrublereau$ pwd
/Users/stephanetrublereau
```

Modifier sous votre repertoire le .bash_profile pour qu’il lance la commande de prise en compte du fichier .bashrc :

```
vi .bash_profile
i
source ~/.bashrc

Esc
:wq

vi .bashrc
i
alias aws1=‘ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-187-196-92.us-west-2.compute.amazonaws.com’
alias aws2=‘ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54–201-210-236.us-west-2.compute.amazonaws.com’
alias aws3=‘ssh -i ~/.ssh/telecom-election.pem ec2-user@ec2-54-218-17-7.us-west-2.compute.amazonaws.com’

Esc
:wq
```

Copier telecom-election.pem sur votre repertoire .ssh

6d4N33qThCex

mot de passe jupyter : telecomparistech