

# Atelier 3 : Démarrage du cluster avec Kraft.

## Créer les fichiers serverN.properties

server2.properties

### Etape 1

`node.id = 2` # 1, 2 ou 3

Chaque instance doit avoir un `node.id` différent : 1, 2, 3 (ou 0,1,2,3)

C'est l'identité du nœud dans le cluster KRaft.

### Etape 2 : La liste des 3 contrôleurs

On doit passer de :

`controller.quorum.voters=1@localhost:9093`

à la liste complète des 3 noeuds contrôleurs :

`controller.quorum.voters=1@localhost:9093,2@localhost:9095,3@localhost:9097`

Ceci est valable pour les 3 server.properties

### Etape 3 : listeners + advertised.listeners (ports uniques par brokers)

On met en place des couples (brokerPort,controllerPort) différent à chaque instance :

- Broker1 : 9092 / 9093
- Broker2 : 9094 / 9095
- Broker3 : 9096 / 9097

`listeners=PLAINTEXT://:9092,CONTROLLER://:9093`  
`advertised.listeners=PLAINTEXT://localhost:9092`

`listeners=PLAINTEXT://:9094,CONTROLLER://:9095`  
`advertised.listeners=PLAINTEXT://localhost:9094`

`listeners=PLAINTEXT://:9096,CONTROLLER://:9097`  
`advertised.listeners=PLAINTEXT://localhost:9096`

Si tes 3 brokers sont sur 3 machines différentes, tu peux garder 9092 partout côté broker et 9093 partout côté controller, mais avec des hosts différents (ex: kafka1, kafka2, kafka3).

#### Etape 4 : log.dirs (répertoire unique par instance)

log.dirs=C:/Kafka/data/broker\${[node.id](#)}

ou

log.dirs=/tmp/kraft-combined-logs-[{node.id}](#)

/tmp/kraft-logs-1

/tmp/kraft-logs-2

/tmp/kraft-logs-3

## Générer un cluster-id

Depuis le répertoire racine kafka - On génère un identifiant unique au cluster

```
KAFKA_CLUSTER_ID="$(bin/kafka-storage.sh random-uuid)"
```

```
echo $KAFKA_CLUSTER_ID
```

Pour windows 🖱

#### Lancez powershell

```
cd $env:KAFKA_HOME\bin\windows
```

On génère un identifiant de cluster :

```
$clusterId = .\kafka-storage.bat random-uuid
```

Notez la valeur (ex. 689ba0c6-5d9c-4b9c-8a2e-47db0707c234).

Pour la session courante : `$env:CLUSTER_ID=$clusterId`

## Formater le stockage (une fois par broker)

```
bin/kafka-storage.sh format -t $KAFKA_CLUSTER_ID -c config/kraft/server.properties
```

ou

```
.\kafka-storage.bat format --config C:\Kafka\config-kraft\server-1.properties `
```

```
--cluster-id %CLUSTER_ID% `
```

```
--ignore-formatted
```

Refaire avec **server-2.properties**, **server-3.properties** (changez le numéro).

# Démarrer les brokers

Ouvrez **trois** consoles ; dans chacune :

## # Broker 1

```
.\kafka-server-start.bat C:\Kafka\config-kraft\server-1.properties
```

**Astuce** : ajoutez **start /B** devant la commande pour lancer en arrière-plan :

```
start /B .\kafka-server-start.bat C:\Kafka\config-kraft\server-1.properties
```

## Puis broker 2 (et 3) :

```
start /B .\kafka-server-start.bat C:\Kafka\config-kraft\server-2.properties
```

```
start /B .\kafka-server-start.bat C:\Kafka\config-kraft\server-3.properties
```

# Premier topic & test rapide

## # Créer un topic

```
.\kafka-topics.bat --create --topic test --partitions 3 --replication-factor 3 `
--bootstrap-server localhost:9092
```

## # publish

```
echo hello kraft | .\kafka-console-producer.bat --topic test --bootstrap-server
localhost:9092
```

## # subscribe (autre fenêtre)

```
.\kafka-console-consumer.bat --topic test --from-beginning --bootstrap-server
localhost:10092
```

Il est parfois nécessaire de supprimer les logs existants :

```
rm -rf /tmp/kraft-combined-logs
```