

✓ Introduction

These steps demonstrate the setup of a distributed CockroachDB cluster across three virtual machines (VM1, VM2, VM3) using a manual static IP configuration in a host-only network environment.

The process includes downloading and installing CockroachDB, configuring static IP addresses for LAN communication, initiating the primary node (VM1), and having secondary nodes join the cluster.

The cluster is run in insecure mode for development and testing purposes. After the cluster is built, SQL schemas and data can be inserted through the primary node (VM1), simulating a realistic backend for a distributed online marketplace system.

This report documents each step taken from environment setup to database interaction and verification.

CockroachDB Cluster Setup on 3 VMs (Manual IP Configuration)

✓ Step 1: Install CockroachDB (On Each Node)

```
# Download and extract CockroachDB
curl https://binaries.cockroachdb.com/cockroach-v21.2.7.linux-amd64.tgz
| tar -xzv

# Move the binary to your $PATH
sudo mv cockroach-v21.2.7.linux-amd64/cockroach /usr/local/bin/

# Cleanup
rm -rf cockroach-v21.2.7.linux-amd64*
```

Verify installation:

```
cockroach version
```

Note: If the internet is not working:

```
sudo dhclient
```

Create data directory (Choose any location you want) :

```
sudo mkdir -p /cockroach/cockroach-data
sudo chown $USER:$USER /cockroach -R
```

✓ **Step 2: Configure Networking (Static IPs)**

On all VMs, configure a host-only adapter with a static IP.

Node 1 (192.168.100.10)

```
sudo nmcli connection modify "Wired connection 1" \
  ipv4.addresses 192.168.100.10/24 \
  ipv4.gateway 192.168.100.1 \
  ipv4.dns "8.8.8.8 1.1.1.1" \
  ipv4.method manual
```

```
sudo nmcli connection down "Wired connection 1"
sudo nmcli connection up "Wired connection 1"
ip a
```

Repeat for:

- **Node 2:** 192.168.100.11
- **Node 3:** 192.168.100.12

Test connectivity:

```
ping 192.168.100.10
ping 192.168.100.11
ping 192.168.100.12
```

✓ **Step 3: Start CockroachDB Cluster**

Node 1 (Initial Node / Bootstrap)

```
cockroach start \
  --insecure \
  --listen-addr=192.168.100.10:26257 \
  --http-addr=192.168.100.10:8080 \
  --join=192.168.100.10:26257 \
  --store=/cockroach/cockroach-data
```

Cluster Initialization (Only ONCE on Node 1):

```
cockroach init --insecure --host=192.168.100.10
```

Important: Do NOT run `cockroach init` on other nodes after Node 1. Initialization should be done once only.

Node 2 and Node 3 (Join the Cluster)

```
cockroach start \  
  --insecure \  
  --listen-addr=192.168.100.11:26257 \  
  --http-addr=192.168.100.11:8080 \  
  --join=192.168.100.10:26257 \  
  --store=/cockroach/cockroach-data  
cockroach start \  
  --insecure \  
  --listen-addr=192.168.100.12:26257 \  
  --http-addr=192.168.100.12:8080 \  
  --join=192.168.100.10:26257 \  
  --store=/cockroach/cockroach-data
```

✓ Step 4: Web UI Monitoring

Access CockroachDB Admin UI:

`http://192.168.100.10:8080`

✓ Step 5: SQL Access & Data Operations

Create the data sql files:

```
nano create_tables.sql
```

Then write the database schema description

Example SQL Workflow:

```
CREATE DATABASE testdb;  
USE testdb;  
CREATE TABLE users (id INT PRIMARY KEY, name STRING);  
INSERT INTO users (id, name) VALUES (1, 'name1');  
SELECT * FROM users;
```

```
nano insert_data.sql
```

```
INSERT INTO users (id, name) VALUES (3, 'Name3');
```

Then write the database data

example:

```
USE testdb;
```

```
INSERT INTO users (id, name) VALUES (1, 'Name1');
```

```
INSERT INTO users (id, name) VALUES (2, 'Name2');
```

Connect to CockroachDB SQL:

```
cockroach sql --insecure --host=192.168.100.10:26257
```

Run SQL Scripts:

```
cockroach sql --insecure --host=192.168.100.10 < create_tables.sql
```

```
cockroach sql --insecure --host=192.168.100.10 < insert_data.sql
```

Test the db from any node:

Example from node 2:

```
cockroach sql --insecure --host=192.168.100.11
```

Then:

```
USE testdb;
```

```
SELECT * FROM users;
```