# The neo4j.conf file

The main source of configuration settings

## JVM-specific configuration settings

[**dbms.memory.heap.initial\_size**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.memory.heap.initial_size)

Sets the initial heap size for the JVM. By default, the JVM heap size is calculated based on the available system resources.

[**dbms.memory.heap.max\_size**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.memory.heap.max_size)

Sets the maximum size of the heap for the JVM. By default, the maximum JVM heap size is calculated based on the available system resources.

[**dbms.jvm.additional**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.jvm.additional)

Sets additional options for the JVM. The options are set as a string and can vary depending on JVM implementation.

## List currently active settings

CALL dbms.listConfig()

YIELD name, value

WHERE name STARTS WITH 'dbms.default'

RETURN name, value

ORDER BY name;

# File locations

### Default file locations

|  |  |
| --- | --- |
| /usr/bin | Running script & built-in tools, such as, cypher-shell & neo4j-admin. |
| /etc/neo4j/neo4j.conf | The Neo4j configuration settings |
| /var/lib/neo4j/data | All data-related content, such as, databases, transactions, cluster |
| /var/lib/neo4j/import | All CSV files that the command LOAD CSV uses as sources to import data |
| /usr/share/neo4j/labs | Contains APOC Core |
| /usr/share/neo4j/lib | All Neo4j dependencies |
| /var/log/neo4j/ | The Neo4j log files  journalctl --unit=neo4j |
| /var/lib/neo4j/metrics | The Neo4j built-in metrics for monitoring database |
| /var/lib/neo4j/plugins | Custom code that extends Neo4j |
| /var/lib/neo4j/run | The processes IDs. |

sudo tree /etc/neo4j

sudo tree /var/lib/neo4j

sudo tree /usr/share/neo4j

sudo tree /var/log/neo4j

### Customize your file locations

The file locations can also be customized by using environment variables and options.

The locations of <neo4j-home> and conf can be configured using environment variables:

|  |  |
| --- | --- |
| **Location** | **Environment variable** |
| <neo4j-home> | NEO4J\_HOME |
| conf | NEO4J\_CONF |

**Example:**

export $NEO4J\_HOME=/Users/florin/Downloads/neo4j-community-3.5.3

export NEO4J\_CONF=$NEO4J\_HOME/conf

**Rest of the locations can be configured in the conf/neo4j.conf file**

#dbms.directories.data=data

#dbms.directories.plugins=plugins

#dbms.directories.logs=logs

#dbms.directories.lib=lib

#dbms.directories.run=run

#dbms.directories.metrics=metrics

# Ports

| **Name** | **Default port** | **Related configuration setting** |
| --- | --- | --- |
| Backup | 6362 | [**dbms.backup.listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.backup.listen_address) |
| HTTP | 7474 | [**dbms.connector.http.listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.connector.http.listen_address) |
| HTTPS | 7473 | [**dbms.connector.https.listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.connector.https.listen_address) |
| Bolt | 7687 | [**dbms.connector.bolt.listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.connector.bolt.listen_address) |
| Causal Cluster discovery management | 5000 | [**causal\_clustering.discovery\_listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_causal_clustering.discovery_listen_address) |
| Causal Cluster transaction | 6000 | [**causal\_clustering.transaction\_listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_causal_clustering.transaction_listen_address) |
| Causal Cluster RAFT | 7000 | [**causal\_clustering.raft\_listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_causal_clustering.raft_listen_address) |
| Causal Cluster routing connector | 7688 | [**dbms.routing.listen\_address**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.routing.listen_address) |
| Graphite monitoring | 2003 | [**metrics.graphite.server**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_metrics.graphite.server) |
| Prometheus monitoring | 2004 | [**metrics.prometheus.endpoint**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_metrics.prometheus.endpoint) |

# Password and user recovery

sudo systemctl stop neo4j

vim /etc/neo4j/neo4j.conf

**#Set:**

#dbms.default\_listen\_address=<You Configuration>

dbms.security.auth\_enabled=false

sudo systemctl start neo4j

cypher-shell -d system

neo4j@system> ALTER USER neo4j SET PASSWORD 'Azure@123';

neo4j@system> :exit

sudo systemctl stop neo4j

sudo vim /etc/neo4j/neo4j.conf

#Set

dbms.security.auth\_enabled=true

dbms.default\_listen\_address=<You Configuration>

sudo systemctl start neo4j

### Recover an unassigned admin role

If you have no user assigned to the admin role, you can grant an admin role to an existing user (assuming your existing user is named neo4j):

sudo systemctl stop neo4j

sudo vim /etc/neo4j/neo4j.conf

#Set

dbms.security.auth\_enabled=false

#dbms.default\_listen\_address=<You Configuration>

sudo systemctl start neo4j

# Note: Wait for 10 Seconds

cypher-shell -d system

neo4j@system> GRANT ROLE admin TO neo4j;

neo4j@system> :exit

sudo systemctl stop neo4j

sudo vim /etc/neo4j/neo4j.conf

dbms.security.auth\_enabled=true

dbms.default\_listen\_address=<You Configuration>

sudo systemctl start neo4j

### Recover the admin role

If you have removed the admin role from your system entirely, you can recreate the role with its original capabilities

sudo systemctl stop neo4j

sudo vim /etc/neo4j/neo4j.conf

#Set

#dbms.default\_listen\_address=<You Configuration>

dbms.security.auth\_enabled=false

sudo systemctl start neo4j

#Create a custom admin role using a client such as Cypher Shell, or the Neo4j Browser

cypher-shell -d system

neo4j@system> CREATE ROLE admin;

neo4j@system> GRANT ALL DBMS PRIVILEGES ON DBMS TO admin;

neo4j@system> GRANT TRANSACTION MANAGEMENT ON DATABASE \* TO admin;

neo4j@system> GRANT START ON DATABASE \* TO admin;

neo4j@system> GRANT STOP ON DATABASE \* TO admin;

neo4j@system> GRANT MATCH {\*} ON GRAPH \* TO admin;

neo4j@system> GRANT WRITE ON GRAPH \* TO admin;

neo4j@system> GRANT ALL ON DATABASE \* TO admin;

neo4j@system> GRANT ROLE admin TO neo4j;

neo4j@system> :exit

sudo systemctl stop neo4j

sudo vim /etc/neo4j/neo4j.conf

#Set

dbms.security.auth\_enabled=true

dbms.default\_listen\_address=<You Configuration>

sudo systemctl start neo4j

# Configure Neo4j connectors

1. Available connectors

| ***Default connectors and their ports*** | | |
| --- | --- | --- |
| **Connector name** | **Protocol** | **Default port number** |
| dbms.connector.bolt | Bolt | 7687 |
| dbms.connector.http | HTTP | 7474 |
| dbms.connector.https | HTTPS | 7473 |

1. Configuration options

The connectors are configured by settings on the format dbms.connector.<connector-name>.<setting-suffix>>

**Example 1. Specify listen\_address for the Bolt connector**

To listen for Bolt connections on all network interfaces (0.0.0.0) and on port 7000, set the listen\_address for the Bolt connector:

dbms.connector.bolt.listen\_address=0.0.0.0:7000

**Defaults for addresses**

Possible to specify defaults for the configuration options with listen\_address and advertised\_address suffixes

Setting a default value will apply to all the connectors, unless specifically configured for a certain connector.

dbms.default\_listen\_address

dbms.connector.bolt.listen\_address=0.0.0.0:7000

This is equivalent to

dbms.default\_listen\_address=0.0.0.0

dbms.connector.bolt.listen\_address=:7000

# Configure dynamic settings

Neo4j Enterprise Edition supports changing some configuration settings at runtime, without restarting the service.

**Note:**

Changes to the configuration at runtime are not persisted

To avoid losing changes when restarting Neo4j make sure to update neo4j.conf as well.

**Example: Discover dynamic settings**

CALL dbms.listConfig()

YIELD name, dynamic, value

WHERE dynamic

RETURN name, dynamic, value

ORDER BY name

**Update dynamic settings**

An administrator is able to change some configuration settings at runtime, without restarting the service.

CALL dbms.setConfigValue('dbms.logs.query.enabled', 'info')

# Transaction logs

The transaction logs record all write operations in the database

Important configuration settings for transaction logging:

| **Transaction log configuration** | **Default value** | **Description** |
| --- | --- | --- |
| [**dbms.directories.transaction.logs.root**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.directories.transaction.logs.root) | transactions | Root location where Neo4j will store transaction logs for configured databases. |
| [**dbms.tx\_log.rotation.retention\_policy**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.tx_log.rotation.retention_policy) | 7 days | Make Neo4j keep the logical transaction logs for being able to backup the database. Can be used for specifying the threshold to prune logical logs after. |
| [**dbms.tx\_log.rotation.size**](https://neo4j.com/docs/operations-manual/current/reference/configuration-settings/#config_dbms.tx_log.rotation.size) | 250M | Specifies at which file size the logical log will auto-rotate. Minimum accepted value is 128K (128 KiB). |

CALL dbms.listConfig()

YIELD name, value

WHERE name="dbms.directories.transaction.logs.root"

RETURN name, value

ORDER BY name

CALL dbms.listConfig()

YIELD name, value

WHERE name="dbms.tx\_log.rotation.retention\_policy"

RETURN name, value

ORDER BY name

CALL dbms.listConfig()

YIELD name, value

WHERE name="dbms.tx\_log.rotation.size"

RETURN name, value

ORDER BY name