

Rocket Chat

- [Installing Rocketchat with Docker](#)
- [Nginx Conf File Example](#)
- [Updating Rocket Chat](#)

Installing Rocketchat with Docker

1. Create directory `mkdir rocketchat`
2. `cd rocketchat`
3. `sudo nano compose.yml`

```
4. volumes:
    mongodb_data: { driver: local }

services:
    rocketchat:
        image: registry.rocket.chat/rocketchat/rocket.chat:latest
        restart: always
        labels:
            traefik.enable: "true"
            traefik.http.routers.rocketchat.rule: Host(`${DOMAIN:-}`)
            traefik.http.routers.rocketchat.tls: "true"
            traefik.http.routers.rocketchat.entrypoints: https
            traefik.http.routers.rocketchat.tls.certresolver: le
        environment:
            MONGO_URL: "${MONGO_URL:-\
                mongodb://${MONGODB_ADVERTISED_HOSTNAME:-\
mongodb}:${MONGODB_INITIAL_PRIMARY_PORT_NUMBER:-27017}}/\
                ${MONGODB_DATABASE:-rocketchat}?replicaSet=${MONGODB_REPLICA_SET_NAME:-rs0}"
            MONGO_OPLOG_URL: "${MONGO_OPLOG_URL:-\
                -mongodb://${MONGODB_ADVERTISED_HOSTNAME:-\
mongodb}:${MONGODB_INITIAL_PRIMARY_PORT_NUMBER:-27017}}/\
                local?replicaSet=${MONGODB_REPLICA_SET_NAME:-rs0}"
            ROOT_URL: ${ROOT_URL:-http://localhost:${HOST_PORT:-3000}}
            PORT: ${PORT:-3000}
            DEPLOY_METHOD: docker
            DEPLOY_PLATFORM: ${DEPLOY_PLATFORM:-}
            REG_TOKEN: ${REG_TOKEN:-}
        depends_on:
```

```
- mongodb
expose:
  - ${PORT:-3000}
ports:
  - "${BIND_IP:-0.0.0.0}:${HOST_PORT:-3000}:${PORT:-3000}"
```

mongodb:

```
image: docker.io/bitnami/mongodb:${MONGODB_VERSION:-5.0}
```

```
restart: always
```

volumes:

```
- mongodb_data:/bitnami/mongodb
```

environment:

```
MONGODB_REPLICA_SET_MODE: primary
```

```
MONGODB_REPLICA_SET_NAME: ${MONGODB_REPLICA_SET_NAME:-rs0}
```

```
MONGODB_PORT_NUMBER: ${MONGODB_PORT_NUMBER:-27017}
```

```
MONGODB_INITIAL_PRIMARY_HOST: ${MONGODB_INITIAL_PRIMARY_HOST:-mongodb}
```

```
MONGODB_INITIAL_PRIMARY_PORT_NUMBER: ${MONGODB_INITIAL_PRIMARY_PORT_NUMBER:-27017}
```

```
MONGODB_ADVERTISED_HOSTNAME: ${MONGODB_ADVERTISED_HOSTNAME:-mongodb}
```

```
MONGODB_ENABLE_JOURNAL: ${MONGODB_ENABLE_JOURNAL:-true}
```

```
ALLOW_EMPTY_PASSWORD: ${ALLOW_EMPTY_PASSWORD:-yes}
```

5. CTR + X --> 'y' --> "Enter"

6. `sudo nano .env`

7. `### Rocket.Chat configuration`

```
# Rocket.Chat version
```

```
# see:- https://github.com/RocketChat/Rocket.Chat/releases
```

```
#RELEASE=
```

```
# MongoDB endpoint (include ?replicaSet= parameter)
```

```
#MONGO_URL=
```

```
# MongoDB endpoint to the local database
```

```
#MONGO_OPLOG_URL=
```

```
# IP to bind the process to
```

```
#BIND_IP=
```

```
# URL used to access your Rocket.Chat instance
```

```
#ROOT_URL=
```

```
# Port Rocket.Chat runs on (in-container)
```

```
#PORT=
```

```
# Port on the host to bind to
#HOST_PORT=

### MongoDB configuration
# MongoDB version/image tag
#MONGODB_VERSION=
# See:- https://hub.docker.com/r/bitnami/mongodb

### Traefik config (if enabled)
# Traefik version/image tag
#TRAEFIK_RELEASE=
# Domain for https (change ROOT_URL & BIND_IP accordingly)
#DOMAIN=
# Email for certificate notifications
#LETSencrypt_EMAIL=

#####
```

8. CTR + X --> 'y' --> "Enter"
9. `docker-compose up -d`
10. Go to 127.0.0.1:3000 in a browser

Nginx Conf File Example

```
# Upstreams
upstream backend {
    server 127.0.0.1:4000;
}

# HTTPS Server
server {
    listen 443;
    server_name example.domain.com;

    # You can increase the limit if your need to.
    client_max_body_size 200M;

    error_log /var/log/nginx/rocketchat.access.log;

    location / {
        proxy_pass http://backend;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
        proxy_set_header Host $http_host;

        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto https;
        proxy_set_header X-Nginx-Proxy true;

        proxy_redirect off;
    }
}
```

Updating Rocket Chat

- cd into the directory of yml file

- ```
docker-compose stop rocketchat
docker-compose rm rocketchat
docker-compose up -d rocketchat
```