**CS469 – To-Do List Project Setup Guide**

This document provides step-by-step instructions for setting up the distributed To-Do List project using Docker, MySQL, and TLS-secured servers. It includes Docker setup, certificate generation, server deployment, and client testing using the updated ssl\_client which now connects with host/port and reads responses until END.

### Dockerfile and docker-compose.yml

**Dockerfile.server**

FROM ubuntu:22.04  
RUN apt-get update && apt-get install -y build-essential libssl-dev libmysqlclient-dev  
WORKDIR /app  
COPY . /app  
RUN gcc -Wall -pthread -o ssl\_server ssl\_server.c -lssl -lcrypto -lmysqlclient  
EXPOSE 8443 8444  
CMD ["./ssl\_server"]

**docker-compose.yml**

version: "3.8"  
  
services:  
 mysql:  
 image: mysql:8.0  
 container\_name: todo\_mysql  
 restart: unless-stopped  
 environment:  
 MYSQL\_ROOT\_PASSWORD: rootpass  
 MYSQL\_DATABASE: todo\_db  
 MYSQL\_USER: dbuser  
 MYSQL\_PASSWORD: dbpass  
 ports:  
 - "3306:3306"  
 volumes:  
 - mysql\_data:/var/lib/mysql  
  
 ssl\_server1:  
 build:  
 context: ../  
 dockerfile: Dockerfile.server  
 container\_name: ssl\_server1  
 restart: unless-stopped  
 depends\_on:  
 - mysql  
 ports:  
 - "8443:8443"  
 environment:  
 SERVER\_PORT: 8443  
 DB\_HOST: mysql  
 DB\_USER: dbuser  
 DB\_PASS: dbpass  
 DB\_NAME: todo\_db  
  
 ssl\_server2:  
 build:  
 context: ../  
 dockerfile: Dockerfile.server  
 container\_name: ssl\_server2  
 restart: unless-stopped  
 depends\_on:  
 - mysql  
 ports:  
 - "8444:8444"  
 environment:  
 SERVER\_PORT: 8444  
 DB\_HOST: mysql  
 DB\_USER: dbuser  
 DB\_PASS: dbpass  
 DB\_NAME: todo\_db  
  
volumes:  
 mysql\_data:

### Step 1 – Clean Docker

Stop and remove all old containers and networks.

sudo docker stop $(sudo docker ps -aq)  
sudo docker rm $(sudo docker ps -aq)  
sudo docker network rm cs469\_net || true

### Step 2 – Create Docker Network

sudo docker network create cs469\_net

### Step 3 – Start MySQL

sudo docker run -d \  
 --name todo\_mysql \  
 --network cs469\_net \  
 -e MYSQL\_ROOT\_PASSWORD=rootpass \  
 -e MYSQL\_DATABASE=todo\_db \  
 -e MYSQL\_USER=dbuser \  
 -e MYSQL\_PASSWORD=dbpass \  
 mysql:8

### Step 4 – Create the To-Do Table

sudo docker exec -it todo\_mysql mysql -u dbuser -p

USE todo\_db;  
CREATE TABLE todo (  
 id INT AUTO\_INCREMENT PRIMARY KEY,  
 task VARCHAR(255) NOT NULL  
);  
EXIT;

### Step 5 – Generate TLS Certificates

Use localhost as CN for server certificate to match client.

rm -f ca.key ca.crt server.key server.csr server.crt  
  
# Generate CA  
openssl genrsa -out ca.key 2048  
openssl req -x509 -new -nodes -key ca.key -sha256 -days 365 -out ca.crt -subj "/CN=CS469-CA"  
  
# Generate Server Cert (CN=localhost)  
openssl genrsa -out server.key 2048  
openssl req -new -key server.key -out server.csr -subj "/CN=localhost"  
openssl x509 -req -in server.csr -CA ca.crt -CAkey ca.key -CAcreateserial -out server.crt -days 365 -sha256

Note: The new client uses --ca ca.crt and connects to localhost and port directly, so a separate client certificate is not required.

### Step 6 – Build ssl\_server Docker Image

sudo docker build -t ssl\_server\_image .

### Step 7 – Run Two SSL Servers

sudo docker run -d --name ssl\_server1 --network cs469\_net -p 8443:8443 ssl\_server\_image ./ssl\_server 8443  
sudo docker run -d --name ssl\_server2 --network cs469\_net -p 8444:8444 ssl\_server\_image ./ssl\_server 8444

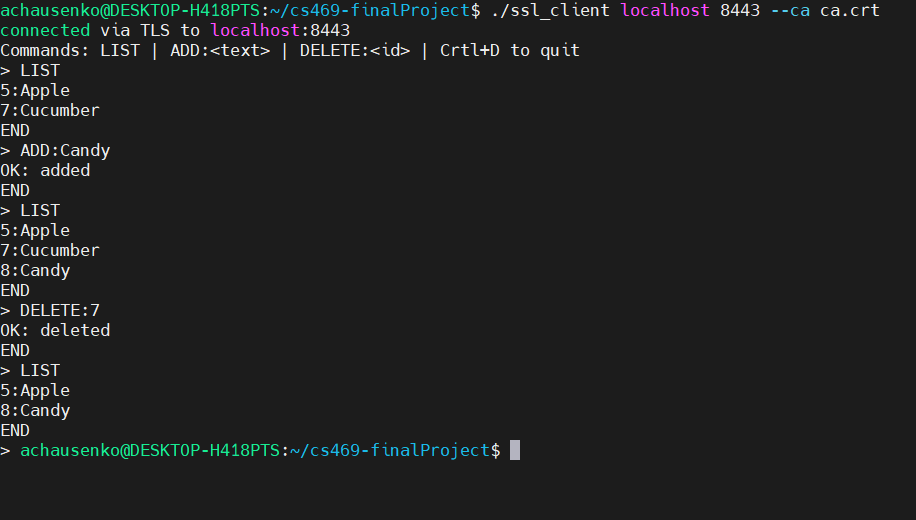
### Step 8 – Test with ssl\_client

Use the new client syntax with host and port and CA certificate.

./ssl\_client localhost 8443 --ca ca.crt

Commands inside client REPL:

> LIST  
> ADD:Apples  
> DELETE:2

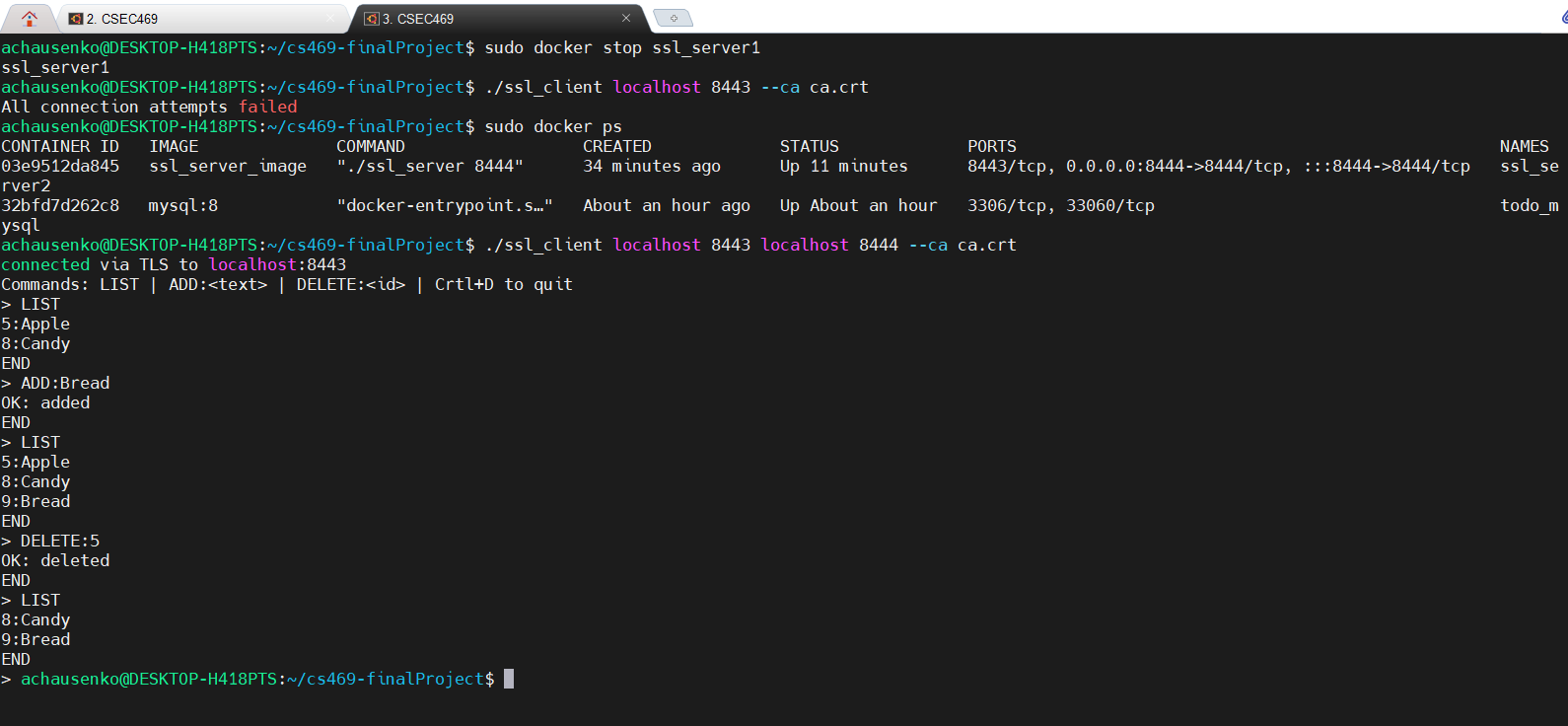


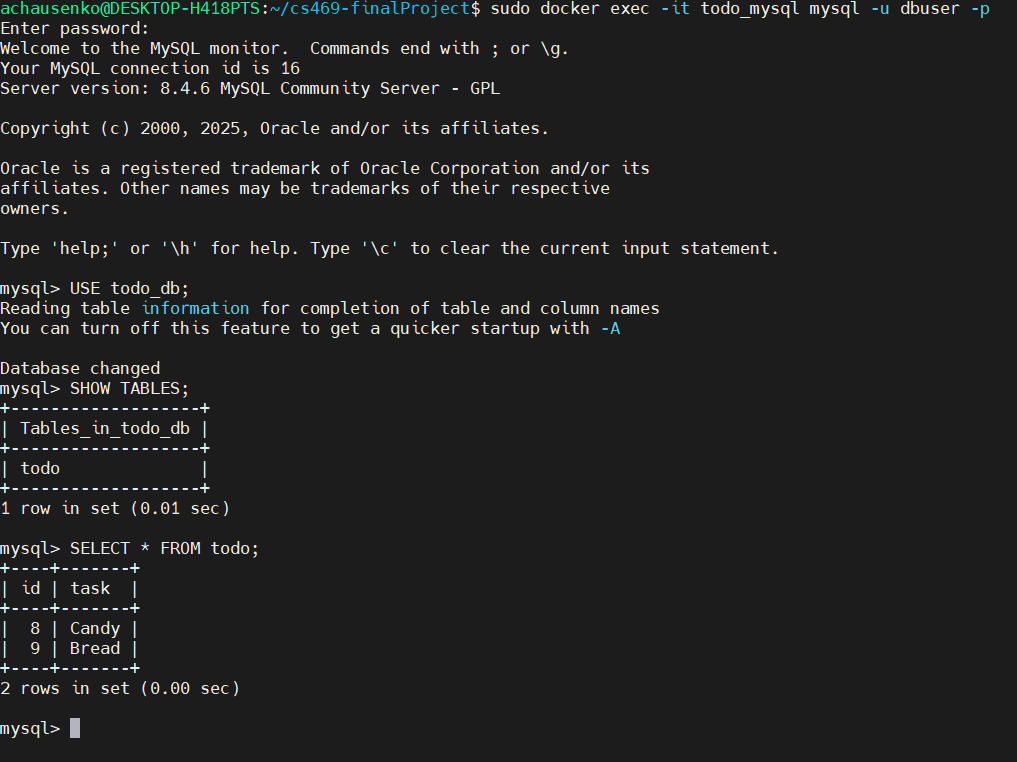
### Step 9 – Failover Test

Stop the first server and confirm client connects to second.

sudo docker stop ssl\_server1  
sudo docker ps  
./ssl\_client localhost 8443 localhost 8444 --ca ca.crt

Client will try localhost:8443, fail, and connect to localhost:8444 automatically.

  
  
MySQL verification:



Or we can use docker:

sudo docker exec -it todo\_mysql mysql -u dbuser -pdbpass -e "USE todo\_db; SELECT \* FROM todo;"

