Antivirus scanning using ClamAV

Problem statement:

Design and implement a containerized antivirus scanning solution using ClamAV. The goal is to simulate a real-world malware detection scenario locally using Docker Desktop, analyse ClamAV's architecture and scanning behaviour, and then deploy the solution on AWS ECS (Elastic Container Service) for scalable cloud-based antivirus scanning.

Python with fast API and dB: No SQL (MongoDB)

My solution:

Creating an API endpoint that accepts the file to upload and then the file is passed to the ClamAV function that checks it and returns the result whether the file is safe or threat. Now using the MongoDB connection, store the scan result including the file metadata and the result given by the ClamAV. Finally the result stored in DB back as API response.

MongoDB

Fast API end point

Client

ClamAV service

1 upload file

5 responses back to the user

4 store the scan result to the DB

2 Store the file locally & pass to the ClamAV

3 Returns the scan result

Docker file for fastAPI to work

FROM python:3.11-slim

WORKDIR /app

# Install system dependencies

RUN apt-get update && apt-get install -y \

    gcc \

    && rm -rf /var/lib/apt/lists/\*

# Copy requirements first for better caching

COPY requirements.txt .

RUN pip install --no-cache-dir -r requirements.txt

# Copy application code

COPY app/ ./app/

COPY main.py .

# Create uploads directory

RUN mkdir -p /app/uploads

EXPOSE 8000

CMD ["uvicorn", "main:app", "--host", "0.0.0.0", "--port", "8000"]

Docker compose file

services:

  clamav:

    image: clamav/clamav:latest

    container\_name: clamav-daemon

    ports:

      - "3310:3310"

    volumes:

      - clamav-db:/var/lib/clamav

    environment:

      - CLAMAV\_NO\_FRESHCLAMD=false

    healthcheck:

      test: ["CMD", "sh", "-c", "clamdtop || nc -z localhost 3310"]

      interval: 30s

      timeout: 10s

      retries: 5

      start\_period: 300s

  mongodb:

    image: mongo:7-jammy

    container\_name: mongodb

    ports:

      - "27017:27017"

    volumes:

      - mongodb-data:/data/db

    environment:

      - MONGO\_INITDB\_DATABASE=clamav\_scanner

  fastapi:

    build: .

    container\_name: fastapi-app

    ports:

      - "8000:8000"

    volumes:

      - ./uploads:/app/uploads

    environment:

      - MONGODB\_URL=mongodb://mongodb:27017

      - CLAMAV\_HOST=clamav

      - CLAMAV\_PORT=3310

    depends\_on:

      clamav:

        condition: service\_healthy

      mongodb:

        condition: service\_started

    restart: unless-stopped

volumes:

  clamav-db:

  mongodb-data:

API : get /scans

[

{

"scan\_id": "07593025-7184-4b27-bee8-953968413212",

"filename": "clean\_file.txt",

"file\_size": 320,

"status": "completed",

"result": "error",

"threat\_name": "Scan status: ERROR",

"scan\_time": 0.009521484375,

"timestamp": "2025-09-28T17:03:14.040000",

"error\_message": null

},

{

"scan\_id": "55606e25-79a5-496f-a495-df434a1ecbb9",

"filename": "clean\_file.txt",

"file\_size": 320,

"status": "completed",

"result": "error",

"threat\_name": "Scan status: ERROR",

"scan\_time": 0.00727605819702148,

"timestamp": "2025-09-28T17:02:40.433000",

"error\_message": null

},

{

"scan\_id": "8ae577b0-8d7b-419a-b348-4683b4ffb456",

"filename": "clean\_file.txt",

"file\_size": 320,

"status": "completed",

"result": "error",

"threat\_name": "Scan status: ERROR",

"scan\_time": 0.00283646583557129,

"timestamp": "2025-09-28T16:56:29.309000",

"error\_message": null

},

Current status :  
As of now it is not completely working to upload to scan but the end points are developed and exposed to hit.