

Remina KASM

- [Setting Up Remmina with Docker](#)

Setting Up Remmina with Docker

Are you looking for a convenient way to manage remote desktop connections? Dockerizing Remmina can provide a seamless solution. In this guide, we'll walk you through the process of setting up Remmina within a Docker container, ensuring ease of deployment and configuration.

Overview

Remmina is a popular open-source remote desktop client application that allows users to access remote computers over various protocols such as RDP, VNC, SSH, and more. By containerizing Remmina with Docker, you can isolate its environment, dependencies, and configuration, making it portable and easy to manage across different systems.

Prerequisites

Before getting started, ensure that you have Docker installed on your system. You can install Docker by following the official installation guide for your operating system: [Docker Installation Guide](#).

Docker Configuration

Below is a Docker Compose configuration for setting up Remmina within a Docker container:

```
services:
  remmina:
    image: lscr.io/linuxserver/remmina:latest
    container_name: remmina
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Etc/UTC
      - CUSTOM_USER=
```

```
- PASSWORD=  
volumes:  
  - /path/to/config:/config  
ports:  
  - 3000:3000  
  - 3001:3001  
shm_size: "2gb"  
restart: unless-stopped
```

Explanation

- **image:** Specifies the Docker image to use. Here, we're using the latest version of the Remmina image from the LinuxServer repository.
- **container_name:** Sets the name for the Docker container.
- **environment:** Defines environment variables for the container, including user ID (PUID), group ID (PGID), timezone (TZ), custom user, and password.
- **volumes:** Maps a local directory to the container's `/config` directory, allowing persistent storage for Remmina's configuration files.
- **ports:** Exposes ports `3000` and `3001` for Remmina's web interface.
- **shm_size:** Sets the shared memory size for the container.
- **restart:** Specifies the restart policy for the container.

Usage

1. **Create a Docker Compose file:** Copy the provided Docker Compose configuration into a file named `docker-compose.yml`.
2. **Adjust configuration:** Modify the environment variables as needed, especially the `CUSTOM_USER` and `PASSWORD` variables to suit your preferences.
3. **Set up volumes:** Replace `/path/to/config` with the directory path on your host machine where you want to store Remmina's configuration files.
4. **Run Remmina:** Open a terminal, navigate to the directory containing the `docker-compose.yml` file, and run the following command:

```
docker-compose up -d
```

This command will download the Remmina Docker image (if not already available) and start the container in detached mode.

5. **Access Remmina:** Once the container is running, you can access Remmina's web interface by navigating to `http://localhost:3000` in your web browser.

Conclusion

By Dockerizing Remmina, you can simplify its deployment and management while maintaining flexibility and portability. With this setup, you can easily access remote desktops using Remmina's intuitive interface, all within a Docker container.