Complete Guide: Install Latest Docker & Docker Compose v2 on Ubuntu (EC2 Compatible)

Overview

This document explains how to correctly install the official Docker Engine and Docker Compose v2 plugin. This replaces the old `docker.io` and `docker-compose` packages that come from Ubuntu repositories.

Why this matters: - Modern DevOps pipelines use `docker compose` (v2) - ECS / CI-CD / GitHub Actions depend on Compose v2 syntax - Old compose (v1) causes project naming and networking conflicts

---

Step 0 — Check Existing Installation

```
docker --version
docker-compose --version
```

If you see docker-compose 1.x → you have old version.

Step 1 — Remove Old Docker Packages (Safe)

This will NOT delete images or containers.

```
sudo apt remove docker docker-engine docker.io containerd runc docker-compose -y
```

Clean remaining packages:

```
sudo apt autoremove -y
```

Step 2 — Install Required Dependencies

```
sudo apt update
sudo apt install ca-certificates curl gnupg lsb-release -y
```

Step 3 — Add Docker Official GPG Key

```
sudo mkdir -p /etc/apt/keyrings

curl -fsSL https://download.docker.com/linux/ubuntu/gpg
| sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

Give read permission:

```
sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

Step 4 — Add Official Docker Repository

```
echo
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/
docker.gpg]
https://download.docker.com/linux/ubuntu
$(lsb_release -cs) stable"
| sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

Update package index:

```
sudo apt update
```

Step 5 — Install Docker Engine + Compose Plugin

```
sudo apt install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin -y
```

This installs: - Docker Engine (daemon) - Docker CLI - Buildx - Docker Compose v2 (plugin)

Step 6 — Enable Docker Service

```
sudo systemctl enable docker
sudo systemctl start docker
```

Verify:

```
sudo systemctl status docker
```

Step 7 — Allow Non-Root User (Important for DevOps)

```
sudo usermod -aG docker $USER
newgrp docker
```

Now you can run docker without sudo.

Step 8 — Verify Installation

```
docker --version
docker compose version
```

Expected Output Example:

```
Docker version 26.x
Docker Compose version v2.x
```

Step 9 — Test With Sample Container

```
docker run hello-world
```

If successful → installation correct.

Step 10 — Common Issues & Fixes

Port Already in Use

Check which service is using port 80:

```
sudo ss -tulnp | grep :80
```

Stop nginx/apache if needed:

```
sudo systemctl stop nginx
sudo systemctl disable nginx
```

Old Containers Causing Conflict

```
docker rm -f $(docker ps -aq)
docker network prune -f
```

Key Concept (Interview Question)

Difference between docker-compose and docker compose:

- docker-compose: legacy Python standalone binary (v1)
- docker compose: native Docker CLI plugin (v2) written in Go
- v2 integrates networking, project naming, and buildx support

---

You Are Now Ready For

- Multi-container apps
- Reverse proxy setup
- ECS deployments
- CI/CD pipelines

End of document