# Come installare Docker Engine

## prerequisiti

Se avete docker desktop installato, è necessario rimuoverlo completamente!

Questo però comporterà anche la cancellazione dei vari container, quindi, se necessario, **fatevi un backup dei DB!**

**Per il corretto funzionamento, eseguite tutti gli script di questo documento da un Powershell con i permessi da amministratore. Lanciare gli script con POWERSHELL e non con POWERSHELL ISE!**

### Rimuovere Docker Desktop

Andiamo a disinstallare Docker Desktop dai settings di windows:

1. Apriamo “App e Funzionalità” dalle impostazioni di windows
2. Cerchiamo Docker Desktop nella lista
3. Lanciamo la disinstallazione

Una volta finito la disinstallazione, è necessario lanciare uno script powershell per assicurarsi di rimuovere tutti i file che potrebbero essere rimasti.

Eseguiamo quindi il file “1\_UninstallDockerDesktop.ps1”

**n.b: Lo script potrebbe mostrare tante scritte rosse, non c’è da preoccuparsi, vuol dire che non ha trovato alcune cartelle ed è cosa buona e giusta**

### Installazione Docker Engine

Per installare Docker Engine, basterà eseguire il file Powershell “2\_InstallUpdateDocker.ps1”

### Aggiornare Docker Engine

Eseguire lo script “2\_InstallUpdateDocker.ps1”, controllerà lui se ci sono nuove versioni e se si vi chiederà se vorrete installarle.

### Post-Installazione (Facoltativo)

Per la natura di Docker, normalmente bisogna usarlo con i premessi di amministratore.

Per il nostro lavoro però non è sempre necessario, anzi con la certificazione ISO in arrivo, dovremmo poter usare docker senza i permessi di amministratore.

Docker utilizza un protocollo per comunicare tra i container che si chiama npipe. Per natura di windows, questi npipe sono utilizzabili solamente ed esclusivamente con permessi di amministratore.

Quindi come fare?

Ci sono due strade:

1. Cambiare cosa usa docker per comunicare (esporre npipe su TCP)
2. Dare i permessi al nostro utente di comunicare con Docker

Quale scegliere? Tutte e due sono valide come soluzioni, cambia poco.

#### Esporre docker engine su TCP

Eseguire lo script “3\_ExposeDockerTCP.ps1”

Questo script, andrà a modificare un file “deamon.json” che è un file di configurazione di docker. Qua gli diremo di usare tcp://127.0.0.1:2375 al posto di npipe per comunicare.

Andrà poi a modificare la variabile d’ambiente “DOCKER\_HOST” e gli dirà di usare TCP.

#### Assegnare i permessi all’utente

Eseguire lo script “3\_GrantUserPermission.ps1” vi chiederà di inserire il vostro nome utente con il dominio quindi, ad esempio: “EOS\sprimo”.

Una volta finito, il vostro utente sarà dentro ad un gruppo “docker-users” e chiunque sarà li dentro avrà la possibilità di usare docker senza essere admin!

Una volta scelta una di queste due strade, chiudete tutte le finestre di Powershell, poi apritene un’altra **senza i permessi di amministratore,** e provate a eseguire il comando “docker ps -a” e vi dovrà dare una lista dei vostri container!

### Utilizzo di Docker Engine

Docker engine parte all’avvio del PC. Non avendo più una interfaccia grafica per avviare, fermare, rimuovere i containers, avrete 3 possibilità:

1. Utilizzare l’estensione di Docker di VSCode: <https://marketplace.visualstudio.com/items?itemName=ms-azuretools.vscode-docker>
2. Utilizzare Portainer: <https://docs.portainer.io/start/install-ce/server/docker/wcs>
3. Usare docker da commandline (il più veloce di tutti)