

## Contenidors

### Definició

Podem definir un "contenedor" software com el conjunt de llibreries, configuracions i executables que es troben "empaquetats" d'una determinada forma que permet posar en marxa una aplicació determinada (l'aplicació "continguda") de la mateixa manera en qualsevol sistema operatiu, sigui quin sigui aquest.

Els contenidors són vistos per part de la màquina que els executa (la màquina "amfitriona") com simples processos, però la gràcia està en que els contenidors "no veuen" cap altre procés existent a la màquina amfitriona: es pensen que només estan executant-se ells i res més. A més és possible limitar també la configuració de xarxa d'un contenidor, o els dispositius de blocs accessibles per ell, o la RAM disponible per ell, etc de manera que l'entorn reconegut per un contenidor podrà ser molt més limitat (i separat) que el que veuria un procés estàndard executat directament a la màquina amfitriona.

En tot cas, l'explicació anterior és molt genèrica perquè és necessari distingir dos grans tipus de contenidors, la funcionalitat i utilitat dels quals és diferent: els contenidors "d'aplicació" i els contenidors "de sistema".

### Contenedor d'aplicació vs contenidor de sistema

Un **contenedor d'aplicació** és un contenidor que té com a finalitat executar una sola aplicació dins d'un entorn aïllat portàtil. Tecnologies que permeten implementar aquest tipus de contenidors són, per exemple, Docker (<https://www.docker.com>) o Podman (<https://podman.io>). La seva utilitat principal és desplegar aplicacions individuals de manera fàcil i ràpida en entorns heterogenis i no controlats. És el tipus de contenidor que als desenvolupadors d'aplicacions els interessa més perquè els permet no haver-se de preocupar per si el seu programa funcionarà en tal o qual sistema, ja que el propi contenidor (com diu el seu nom) incorpora tot el necessari (i no més) per executar-se allà on sigui. Així doncs, ens podem trobar amb un contenidor que executi Apache, un altre que executi MySQL, un altre que executi Bash, etc.

**NOTA:** Tal com veurem, els contenidors tenen limitat l'ús que poden fer dels recursos del sistema (CPU, RAM, etc). Per tant, hi ha la garantia de que un contenidor no els monopolitzarà. D'altra banda, els contenidors ofereixen un entorn "estanc", afavorint l'aïllament d'aplicacions i, per tant, la seguretat de tot el sistema.

Un **contenedor de sistema** és un contenidor que és capaç d'executar múltiples aplicacions en paral·lel gràcies a contenir, en lloc d'una sola aplicació, una gran part d'un sistema operatiu complet al seu interior (de fet, tot excepte el kernel). Tecnologies que permeten implementar aquest tipus de contenidors són, per exemple, LXD (<https://linuxcontainers.org>) o *systemd-nspawn*. La seva utilitat principal és servir com una alternativa més lleugera a l'ús de màquines virtuals (que sí que incorporen tot el SO complet, incloent un kernel), ja que ofereixen una despesa de recursos (CPU, RAM,...) molt més baixa i això fa que un mateix sistema amfitrió pugui executar molt més contenidors de sistemes que no pas màquines virtuals.

**NOTA:** Generalment, es podrà escollir si executar Systemd (com a PID 1) dins d'un contenidor de sistema o no. Si s'escull que sí, el comportament del contenidor serà molt similar al d'un sistema operatiu complet (però arrencat amb el kernel de la màquina amfitriona). Això vol dir que es podran administrar internament múltiples serveis (cadascun configurat amb el seu propi arxíu "unit"), gestionar els diversos logs generats, fer ús del canal Dbus, etc, etc