

PID1

PID 1 (Process ID 1) առաջին պրոցեսն է, որն աշխատում է Linux համակարգում միջուկի բեռնում (kernel booting) ավարտելուց հետո: Docker կոնտեյներների գործարկման համատեքստում կոնտեյներին ներսում գտնվող PID 1 գործընթացը պատասխանատու է այդ կոնտեյների ներսում մնացած բոլոր գործընթացների կառավարման համար:

Docker կոնտեյների հետ աշխատելիս կարևոր նկատառումներից մեկն այն է, որ կոնտեյների ներսում PID1 պրոցեսը ճիշտ և լիարժեք կարգավորվի: Սա կարևոր է, քանի որ եթե PID 1 պրոցեսը ձախողվի, ամբողջ կոնտեյները կխափանվի:

Project overview

- **How Docker and docker compose work.**

Docker-ը պլատֆորմ է, որը թույլ է տալիս ստեղծել, փաթեթավորել և տեղակայել (deploy) ծրագրերը (կամ ծրագրային հավելվածները) կոնտեյներներում:

Կոնտեյներները թեթև և ինքնուրույն գործարկվող փաթեթներ են, որոնք պարունակում են այն ամենը (ծրագրային կոդ, system tools, գրադարաններ և այլ կարգավորումներ), ինչ անհրաժեշտ է տվյալ ծրագիրը գործարկելու և աշխատացնելու համար: Docker-ն ապահովում է օգտագործման համար հեշտ ինտերֆեյս՝ ցանկացած համակարգում կոնտեյներ ստեղծելու, գործարկելու և կառավարելու համար:

Docker-ը օգտագործում է client-server ճարտարապետություն: Docker-client-ը շփվում է Docker-daemon-ի հետ, որը պատասխանատու է կոնտեյներների կառուցման (building), գործարկման (running) և կառավարման (managing) համար: **Docker-image**-ները օգտագործվում են կոնտեյներներ ստեղծելու համար: Image-ը read-only template է, որը պարունակում է ծրագրի (հավելվածի) կոդը և դրա բոլոր կախվածությունները (dependencies):

Docker compose-ը գործիք է, որը թույլ է տալիս սահմանել և գործարկել multi-container Docker ծրագրեր (հավելվածներ): Compose-ն օգտագործում է .yaml (YAML - «YAML Ain't Markup Language»), ֆայլ՝ ծրագրի (հավելվածի) համար պահանջվող service-ները, network-երը և volume-ները սահմանելու համար:

Յուրաքանչյուր service սահմանվում է որպես կոնտեյներ և կարող է կարգավորվել միջավայրի (.env) փոփոխականներով, port-երով, volume-ներով և այլ option-ներով:

Երբ **docker-compose.yml** ֆայլը run է արվում, compose-ն ստեղծում է ցանց՝ network (որպեսզի կոնտեյներները իրար հետ և արտաքին այլ ցանցերի հետ հաղորդակցվեն), տեղադրում է պահանջվող volume-ները և start է անում սահմանված service-ները, որպես կոնտեյներներ:

- **The difference between a Docker image used with docker compose and without docker compose.**

Docker compose **օգտագործող և չօգտագործող** Docker image-ների **հիմնական տարբերությունն** այն է, որ Docker compose-ով շատ ավելի հեշտ է կազմել և կառավարել մեկից ավելի կոնտեյներներ YAML (.yaml) ֆայլի միջոցով, մինչդեռ առանց դրա պահանջվում է ավելի շատ ձեռքի աշխատանք՝ կարգավորումների կատարում Docker CLI (command line interface) միջոցով:

- **The benefit of Docker compared to VMs.**

- ✓ Քիչ ռեսուրսատարություն (fewer resource utilization):
- ✓ Տեղափոխման ճկունություն (portability):
- ✓ Արագ գործարկում (faster startup time):
- ✓ Ավելի լավ մեկուսացում (better isolation):
- ✓ Կառավարման պարզություն (easier management):

Basic concepts

Docker volume-ը մեխանիզմ է, որը թույլ է տալիս մշտական պահպանել կոնտեյնների կողմից ստեղծված և օգտագործվող տվյալները: Volume-ները պահպանվում են կոնտեյնների ֆայլային համակարգից դուրս, ինչը նշանակում է, որ դրանք կարող են օգտագործվել կոնտեյններների միջև տվյալների փոխանակման համար, որոնք աշխատում են միևնույն հոստի կամ կլաստերի տարբեր հոստերի վրա: Նրանք նաև անկախ են կոնտեյնների կյանքի ցիկից, և նրանցում պահվող տվյալները պահպանվում են կոնտեյնների ջնջվելուց կամ restart լինելուց հետո:

NGINX-ը բաց կոդով վեբ սերվերի ծրագիր է (open-source web server software), որն օգտագործվում է ինտերնետում ստատիկ և դինամիկ բովանդակություն (content) սպասարկելու համար: Այն ստեղծվել է 2004 թվականին Իգոր Սիսուևի կողմից և հայտնի է իր բարձր արդյունավետությամբ, մասշտաբայնությամբ և հուսալիությամբ: NGINX-ը հաճախ օգտագործվում է որպես հակադարձ պրոքսի սերվեր (reverse proxy server), այսինքն՝ այն հանդես է գալիս որպես միջնորդ client-ների և server-ների միջև:

MariaDB-ն բաց կոդով (open-source) կապակցված տվյալների բազայի կառավարման համակարգ է (RDBMS - relational database management system), որը MySQL-ի թեթև անալոգն է և ունի արդյունավետ կատարողականություն, մասշտաբայնություն և անվտանգության առանձնահատկություններ: Այն լայնորեն օգտագործվում է բազմաթիվ կազմակերպությունների կողմից և լիցենզավորված է GNU General Public License-ի ներքո:

WordPress-ը բովանդակության կառավարման համակարգ է (content management system), որը թույլ է տալիս user-ներին ստեղծել և կառավարել իրենց կայքերը: Այն ապահովում է օգտագործողի համար հարմար ինտերֆեյս, customizable theme-եր և plugin-ներ և մի շարք առանձնահատկություններ, որոնք թույլ են տալիս օգտվողներին հեշտությամբ հրապարակել և կառավարել content՝ առանց առաջադեմ տեխնիկական հմտությունների անհրաժեշտության:

Redis-ը հիշողության մեջ open-source տվյալների «key : value»-ի տիպի շտեմարան է , որը սովորաբար օգտագործվում է քեշավորման համար՝ ապահովելով տվյալների արագ հասանելիություն: Այն տվյալները պահում է հիշողության մեջ և թույլ է տալիս շատ արագ կատարել գրել/կարդալու գործողություններ (ի տարբերություն ավելի «դանդաղ» կրիչների, օրինակ՝ դիսկերի):

FTP server-ը (File Transfer Protocol) ստանդարտ ցանցային պրոտոկոլ է, որն օգտագործվում է client-ի և server-ի միջև ցանցի միջոցով ֆայլեր փոխանակելու համար: FTP-ն աշխատում է server-ի վրա և հնարավորություն է տալիս client-ներին միանալ server-ին և ֆայլեր փոխանանցել կամ ստանալ այնտեղից:

Adminer-ը վեբի վրա հիմնված (web-based) տվյալների բազայի կառավարման գործիք է, որը թույլ է տալիս օգտատերերին հարմար ինտերֆեյսի միջոցով հեշտությամբ կառավարել և փոփոխել իրենց տվյալների բազաները:

Portainer-ը կոնտեյններների կառավարման open-source պլատֆորմ է, որն ապահովում է վեբի վրա հիմնված (web-based) գրաֆիկական ինտերֆեյս (GUI)՝ Docker միջավայրերը կառավարելու համար: Այն թույլ է տալիս օգտվողներին հեշտությամբ տեղակայել, կառավարել և վերահսկել իրենց Docker կոնտեյններները, image-ները, volume-ները և ցանցերը կենտրոնացված ինտերֆեյսից: Այն պարզեցնում է կոնտեյնների կառավարումը բոլոր մակարդակների փորձառության օգտատերերի համար՝ սկսած սկսնակներից մինչև առաջադեմ օգտվողներ: