Virtualizációs technológiák történeti áttekintése

Hagyományos (hipervizor-alapú) virtualizáció története

A hardver-szintű virtualizáció nem újkeletű technológia – gyökerei egészen az 1960-as évekig nyúlnak vissza.

- **1960-as évek:** Az IBM mainframe gépeken megjelent az első virtualizációs megközelítés, például az IBM CP-40 rendszer. Ezek célja az volt, hogy több felhasználó párhuzamosan használhassa ugyanazt a fizikai gépet külön „virtuális gépeken”.
- **1990-es évek vége – 2000-es évek eleje:** A számítógépek teljesítménye lehetővé tette, hogy x86 architektúrán is hatékony virtuális gépeket futtassunk. Megjelentek a kereskedelmi hipervizorok, mint a **VMware Workstation** (1999), majd **VMware ESX**, később a **Microsoft Hyper-V**, **Xen**, **KVM**, **VirtualBox** stb.
- **2000-es évek közepétől:** A virtualizáció központi szerepet kapott az adatközpontokban, szerver konszolidációra, magas rendelkezésre állás elérésére és erőforrás-optimalizálásra.

A hipervizor-alapú megoldások a fizikai hardver fölött helyezkednek el, és lehetővé teszik, hogy több teljes értékű operációs rendszer párhuzamosan fusson, egymástól izoláltan.

Konténeralapú virtualizáció története (Docker és társai)

A konténerizáció gondolata szintén régebbi, de népszerűvé csak a 2010-es években vált.

- **1979:** Az első előfutár a Unix chroot parancs volt, amely képes volt korlátozni egy folyamat fájlrendszerbeli láthatóságát.
- **2000-es évek eleje:** A Linux konténer technológiák fejlődni kezdtek, például:
  - **FreeBSD jails (2000)** – izolált környezeteket biztosított
  - **OpenVZ (2005)** – kernel-szintű virtualizáció Linuxon
  - **LXC (Linux Containers, 2008)** – konténer technológia a Linux kernelre építve
- **2013:** Megjelenik a **Docker**, amely az LXC-re épült, és egyszerű, felhasználóbarát CLI-t, API-t, valamint könnyen hordozható konténerképeket kínált.
- **2015-től:** A konténeres technológiák robbanásszerű fejlődése, megjelenik:
  - **Kubernetes** – konténerek automatizált menedzselésére
  - **containerd**, **CRI-O**, **Podman** – alternatív konténerfuttatók
  - **Docker Compose**, **Docker Swarm** – összetettebb rendszerek kezelése

A konténeres virtualizáció így a mikroszolgáltatás-alapú architektúrák, DevOps, CI/CD és a felhőalapú alkalmazások motorja lett.

Összegzés

- A **hagyományos virtualizáció** erősen izolált, teljes rendszerekre épít, és hosszú múltra tekint vissza, főként szervervirtualizációra használták.
- A **konténeres virtualizáció** a modern alkalmazásfejlesztés és -futtatás eszköze, gyors, könnyű és ideális a mikroszolgáltatások korában.

	Hagyományos virtualizáció	**Konténer alapú virtualizáció**
Virtualizáció típusa	Hardver-szintű virtualizáció (hipervizor segítségével)	Operációs rendszer szintű virtualizáció
Futtatott egységek	Teljes virtuális gépek (VM-ek), külön operációs rendszerrel	Konténerek, közös OS kernel használatával
Erőforrásigény	Magas – minden VM saját OS-t futtat	Alacsony – csak a szükséges alkalmazási környezetet tartalmazza
Indítási idő	Lassabb – másodpercek vagy percek	Gyors – néhány másodperc alatt indítható
Izoláció mértéke	Erős izoláció, külön kernel és OS	Közepes izoláció, közös kernel
Rugalmasság és hordozhatóság	Korlátozott – nagyobb méret, OS-függő	Nagyon jó – platformfüggetlen konténerképek, könnyű deploy
Erőforrás-megosztás	A hipervizor szabályozza a VM-ek között	Az operációs rendszer szabályozza a konténerek között
Biztonság	Magasabb – erősebb határok a VM-ek között	Kisebb, de könnyebb frissítés és patch-elés
Példák	VMware, VirtualBox, Microsoft Hyper-V, KVM	Docker, Podman, containerd, LXC

Érdekes video: https://www.youtube.com/watch?v=4RUiVAlJE2w

From:
https://edu.iit.uni-miskolc.hu/ - **Institute of Information Science - University of Miskolc**

Permanent link:
https://edu.iit.uni-miskolc.hu/tanszek:oktatas:informacios_rendszerek_integralasa:virtualizacio

Last update: **2025/03/28 13:18**