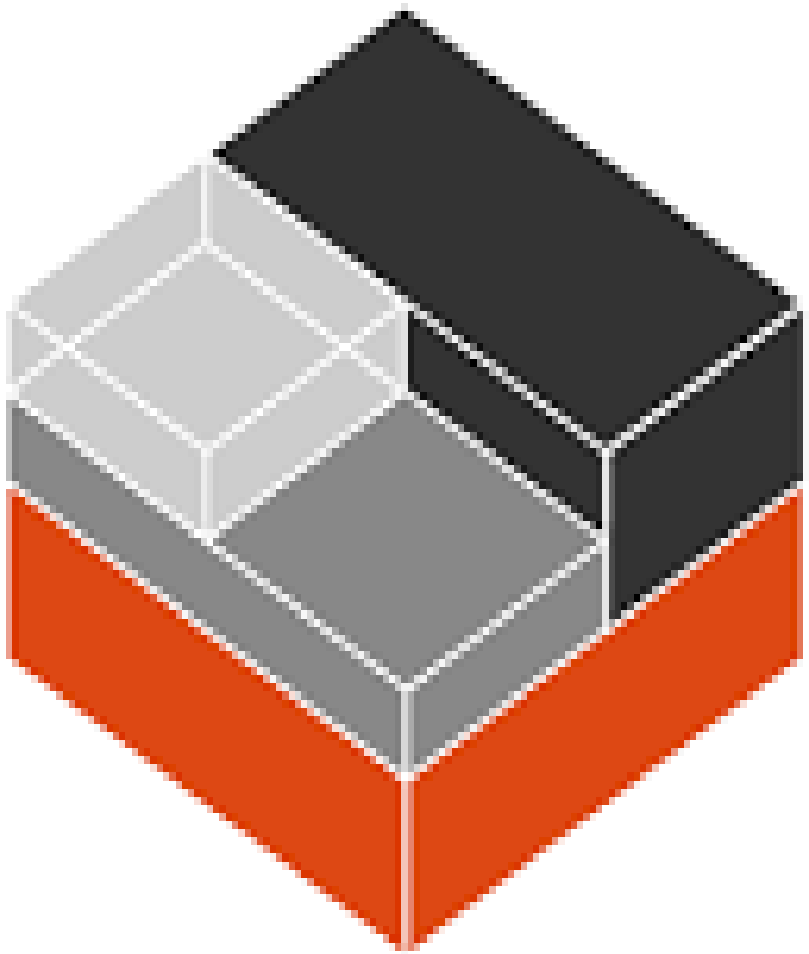


LXC

Linux Containers – Следующее поколение виртуальных машин для облака

Определения

- **Linux Containers (LXC → LinuX Containers)**
  - Легковесные виртуальные машины
  - Реализовано средствами современного ядра Linux
  - VMs без гипервизора

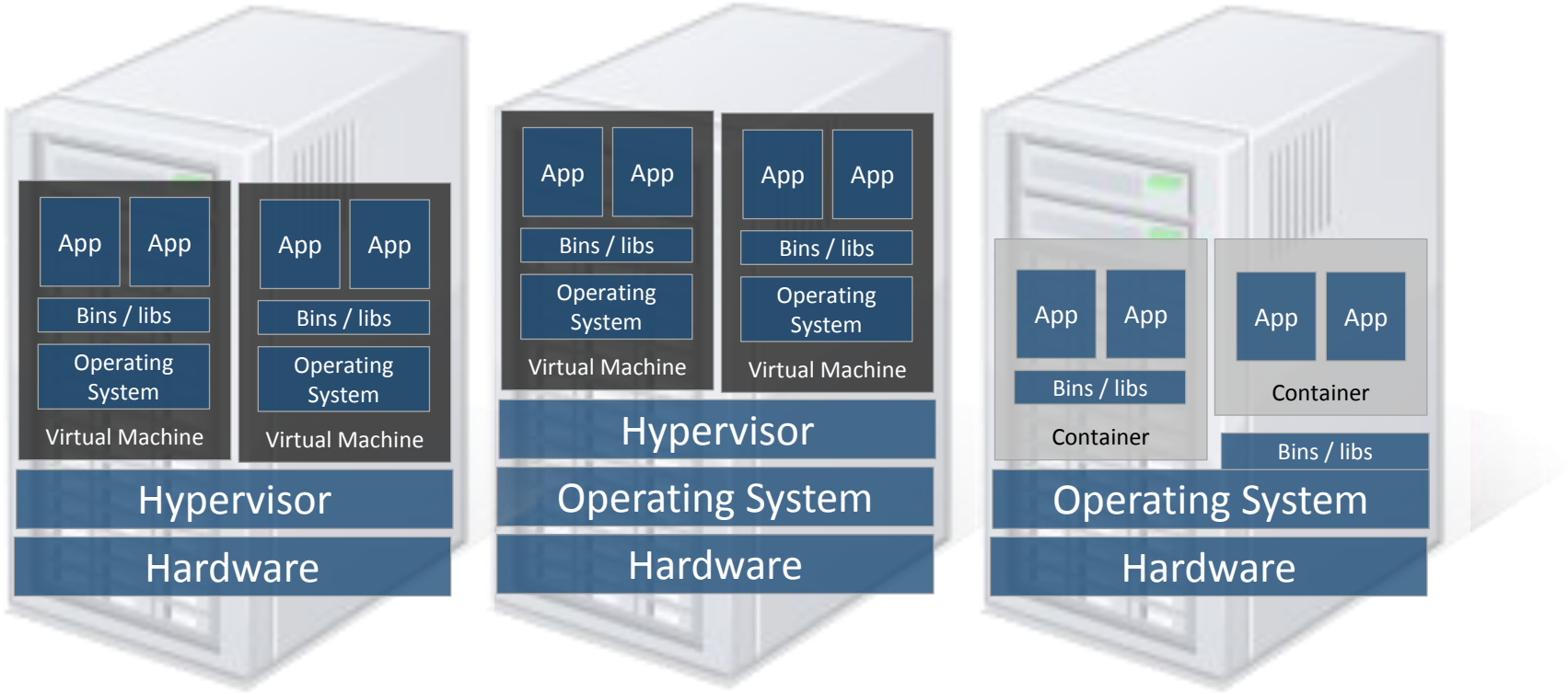
- ***Контейнер для***
  - (Linux) Операционной системы
  - Одного или нескольких приложений

- **LXC как технология ≠ LXC “tools”**

Гипервизоры против Linux контейнеров

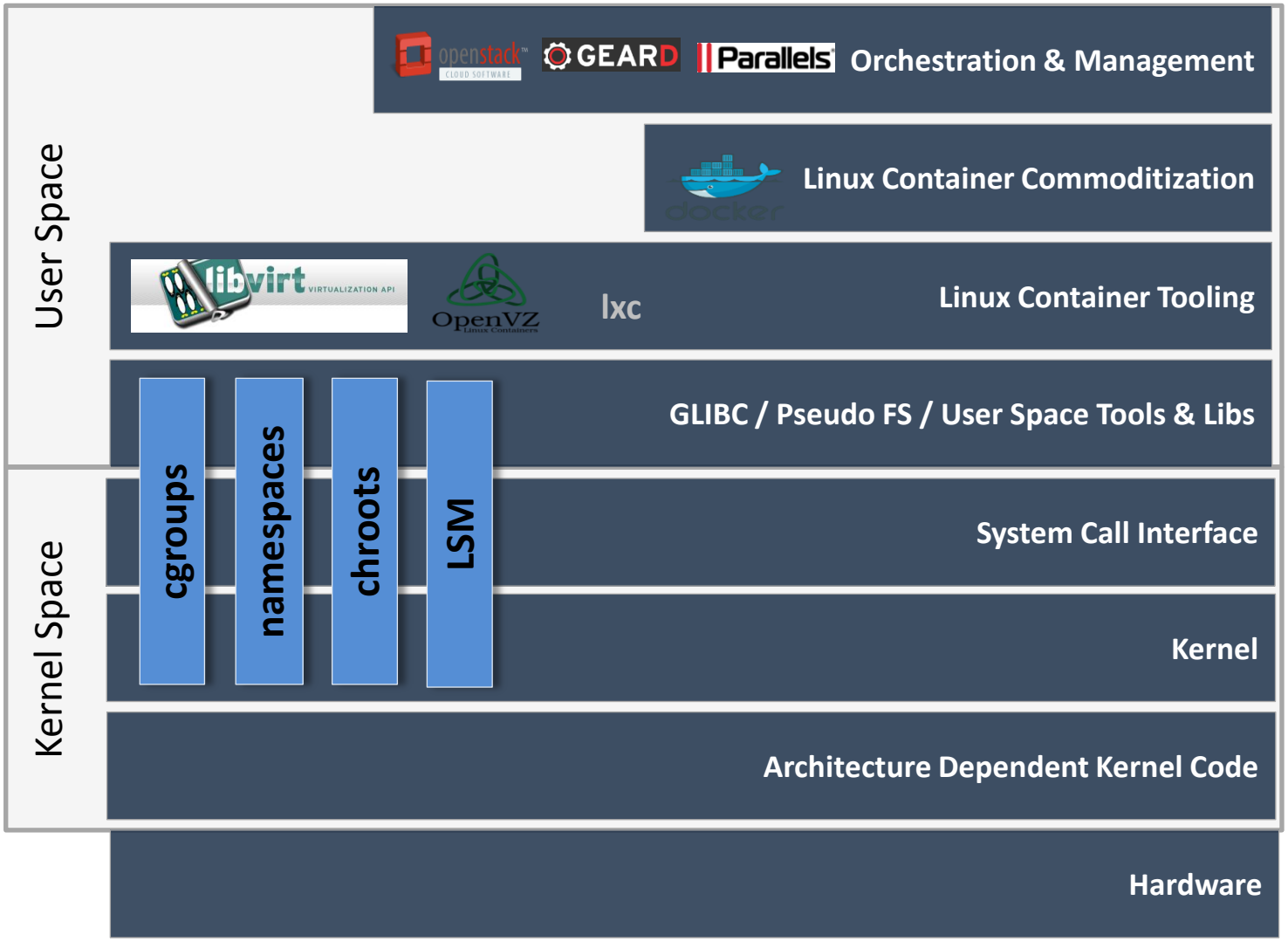
Контейнеры разделяют ресурсы хостовой операционной системы, следовательно Они легковесные, но имеют одинаковую версию ядра.

Контейнеры изолированные, но разделяют ядро и требуемые модули хостовой ОС



Type 1 Hypervisor **Type 2 Hypervisor** **Linux Containers**

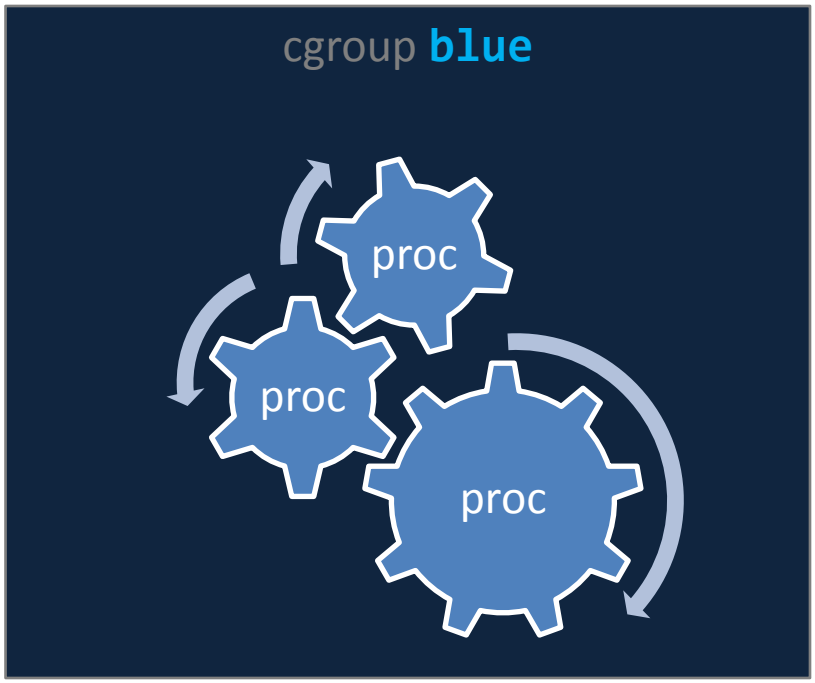
Стек технологий LXC



Группы управления Linux(cgroups)

- **Проблема**
  - Как можно контролировать и анализировать выполнение задач(процессов)?

- **Решение→ группы управления (cgroups)**
  - Доступ к устройствам
  - Ограничения ресурсов
  - Приоритеты
  - Привязка
  - Управление
  - Вложенности



Подсистема Linux cgroup

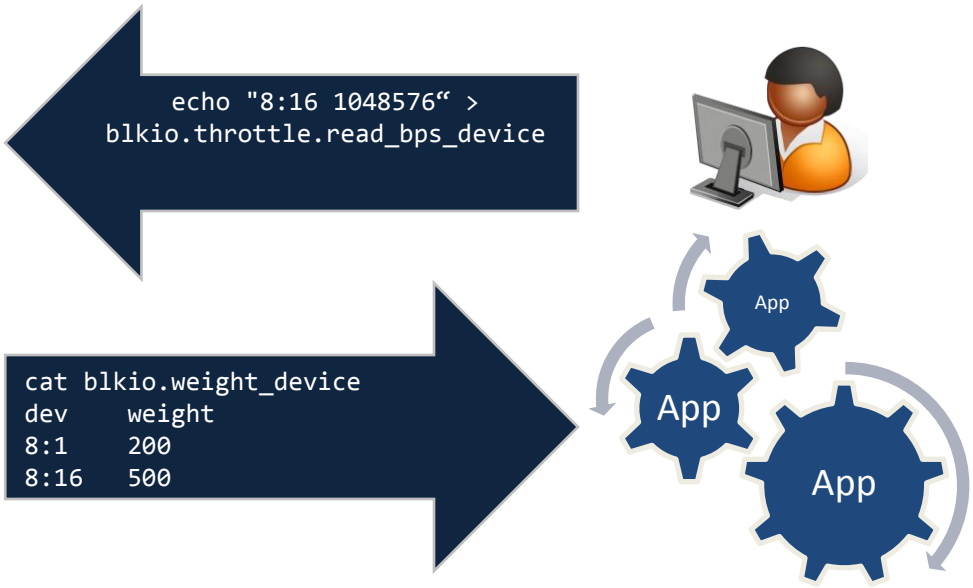
Подсистема	Контролируемые параметры
blkio	- Ограничение доступа к устройствам ввода/вывода. Группа на устройство. - Ограничения для блочных устройств как на количество IO операций, так и на количество IO операций за секунду.
cpu	- Лимитирует процессорное время(микросекунды или секунды) для группы процессов. - Лимит на время на одном CPU. - Получение процессорного времени пропорционально приоритету.
cpuset	- Доступ к группе CPUs. - Возможность мигрировать между областями ОЗУ. - Блокировки доступа к памяти, приоритеты и тд.
devices	- Определяет к каким устройствам есть доступ.
freezer	- Приостанавливает/возобновляет работы группы процессов.
memory	- Лимит памяти для группы(в байтах). - Подкачка, контроль памяти и тд.
hugetlb	- Лимит использования hugeTLB страниц. - Контроль использования hugeTLB группой.
net_cls	- Разделение пакетов по id сети. - Приоритет для пакетов группы.
net_prio	- Приоритет на сетевую активность на интерфейс.

Интерфейс псевдофайловой системы Linux cgroups

- **Псевдо фс Linux - интерфейс для cgroups**
  - **Директория на каждую подсистему cgroup**
  - **Чтение / запись в файл в директории своей группы**

/sys/fs/cgroup/my-lxc

```
|-- blkio
|   |-- blkio.io_merged
|   |-- blkio.io_queued
|   |-- blkio.io_service_bytes
|   |-- blkio.io_serviced
|   |-- blkio.io_service_time
|   |-- blkio.io_wait_time
|   |-- blkio.reset_stats
|   |-- blkio.sectors
|   |-- blkio.throttle.io_service_bytes
|   |-- blkio.throttle.io_serviced
|   |-- blkio.throttle.read_bps_device
|   |-- blkio.throttle.read_iops_device
|   |-- blkio.throttle.write_bps_device
|   |-- blkio.throttle.write_iops_device
|   |-- blkio.time
|   |-- blkio.weight
|   |-- blkio.weight_device
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- cpu
|   |-- ...
|-- ...
`-- perf_event
```



Структура ФС Linux cgroups

```
/sys/fs/cgroup/

|-- blkio
|   |-- blkio.io_merged
|   |-- blkio.io_queued
|   |-- blkio.io_service_bytes
|   |-- blkio.io_serviced
|   |-- blkio.io_service_time
|   |-- blkio.io_wait_time
|   |-- blkio.reset_stats
|   |-- blkio.sectors
|   |-- blkio.throttle.io_service_bytes
|   |-- blkio.throttle.io_serviced
|   |-- blkio.throttle.read_bps_device
|   |-- blkio.throttle.read_iops_device
|   |-- blkio.throttle.write_bps_device
|   |-- blkio.throttle.write_iops_device
|   |-- blkio.time
|   |-- blkio.weight
|   |-- blkio.weight_device
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- lxc
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- cpu
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- cpu.cfs_period_us
|   |-- cpu.cfs_quota_us
|   |-- cpu.rt_period_us
|   |-- cpu.rt_runtime_us
|   |-- cpu.shares
|   |-- cpu.stat
|   |-- lxc
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- cpuacct
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- cpuacct.stat
|   |-- cpuacct.usage
|   |-- cpuacct.usage_percpu
|   |-- lxc
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- cpuset
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- cpuset.cpu_exclusive
|   |-- cpuset.cpus
|   |-- cpuset.mem_exclusive
|   |-- cpuset.mem_hardwall
|   |-- cpuset.memory_migrate
|   |-- cpuset.memory_pressure
|   |-- cpuset.memory_pressure_enabled
|   |-- cpuset.memory_spread_page
|   |-- cpuset.memory_spread_slab
|   |-- cpuset.mems
|   |-- cpuset.sched_load_balance
|   |-- cpuset.sched_relax_domain_level
|   |-- lxc
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- devices
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- devices.allow
|   |-- devices.deny
|   |-- devices.list
|   |-- lxc
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- freezer
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- lxc
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- hugetlb
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- hugetlb.2MB.failcnt
|   |-- hugetlb.2MB.limit_in_bytes
|   |-- hugetlb.2MB.max_usage_in_bytes
|   |-- hugetlb.2MB.usage_in_bytes
|   |-- lxc
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
|-- memory
|   |-- cgroup.clone_children
|   |-- cgroup.event_control
|   |-- cgroup.procs
|   |-- lxc
|   |-- memory.failcnt
|   |-- memory.force_empty
|   |-- memory.limit_in_bytes
|   |-- memory.max_usage_in_bytes
|   |-- memory.memsw.failcnt
|   |-- memory.memsw.limit_in_bytes
|   |-- memory.memsw.max_usage_in_bytes
|   |-- memory.memsw.usage_in_bytes
|   |-- memory.move_charge_at_immigrate
|   |-- memory.numa_stat
|   |-- memory.oom_control
|   |-- memory.soft_limit_in_bytes
|   |-- memory.stat
|   |-- memory.swappiness
|   |-- memory.usage_in_bytes
|   |-- memory.use_hierarchy
|   |-- notify_on_release
|   |-- release_agent
|   `-- tasks
```

cgroup subsystem mounts

- Root of cgroup subsystem for a given cgroup controller
- All cgroup instances are children of root
- All processes initially in `tasks`

cgroup instance

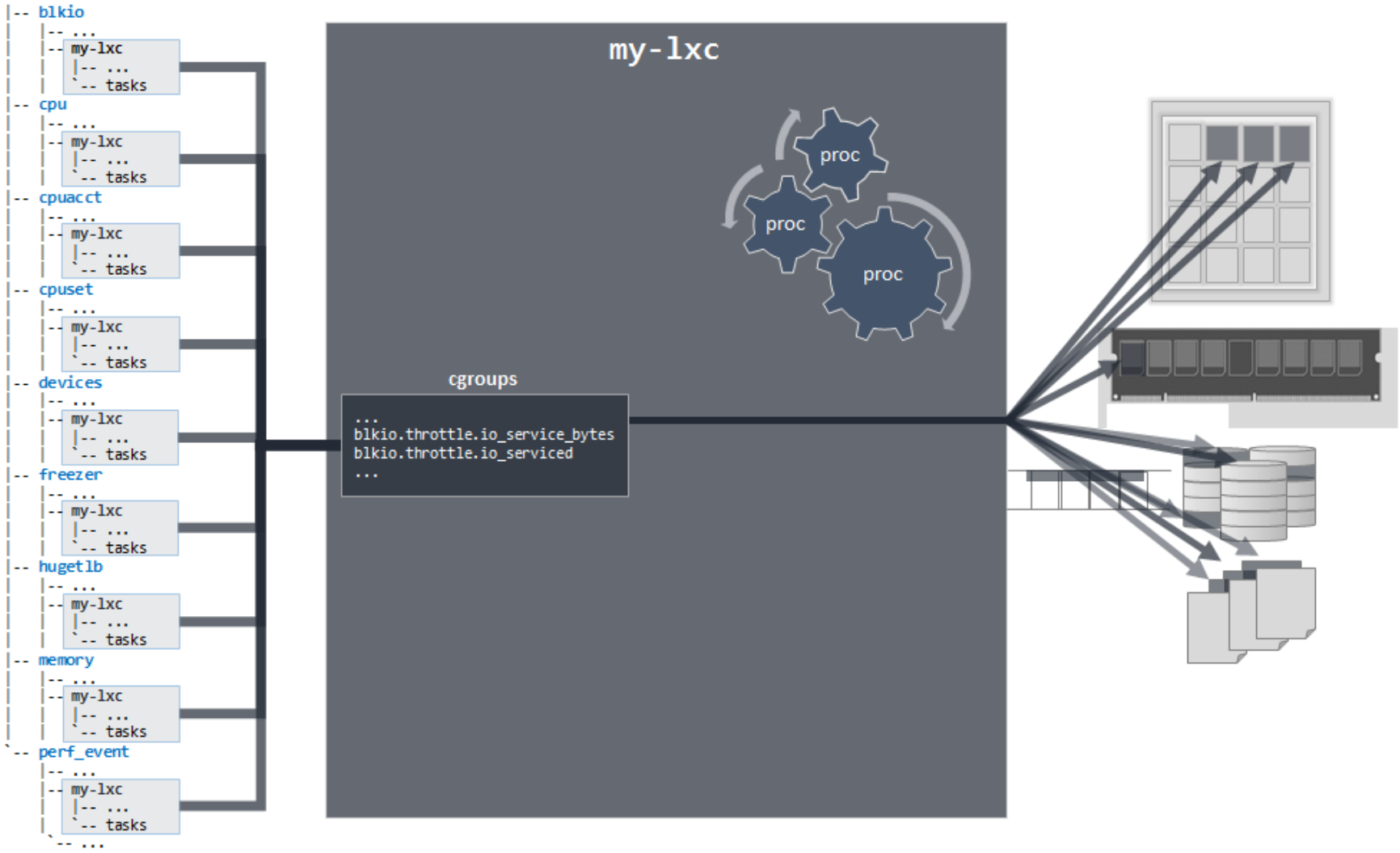
- Individual cgroup instance
- Contains a full set of pseudo files for the subsystem
- Hierarchical controls influenced by parent
- Individual processes added to `tasks` pseudo file

User space management

- Cgroup interface is the pseudo FS itself
- Read counter files to obtain metrics (`cat`, `head`, etc.)
- Write values into control files

```
`-- perf_event
    |-- cgroup.clone_children
    |-- cgroup.event_control
    |-- cgroup.procs
    |-- lxc
    |-- notify_on_release
    |-- release_agent
    `-- tasks
```

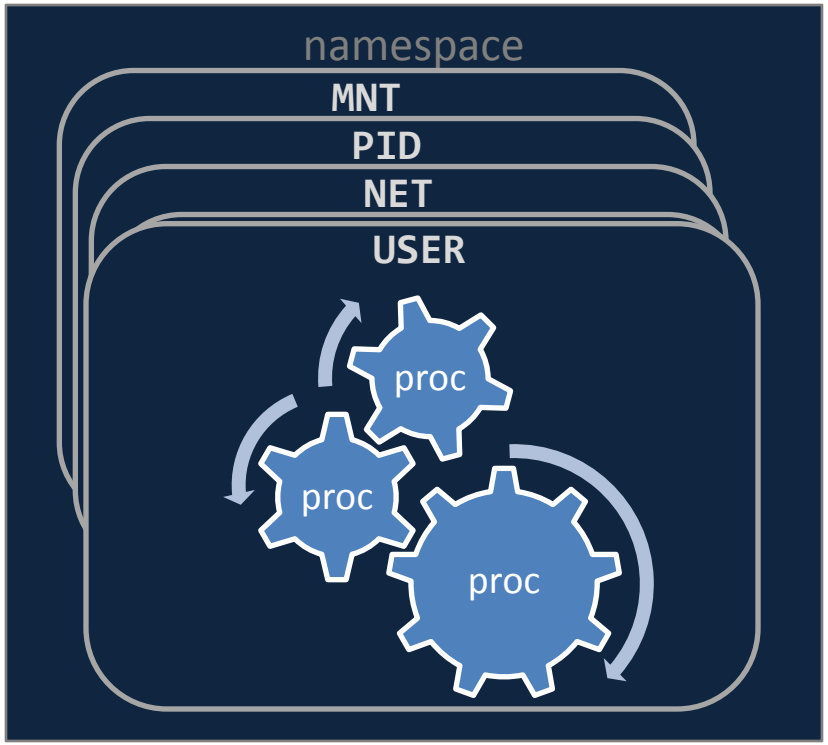
Как строится контейер?



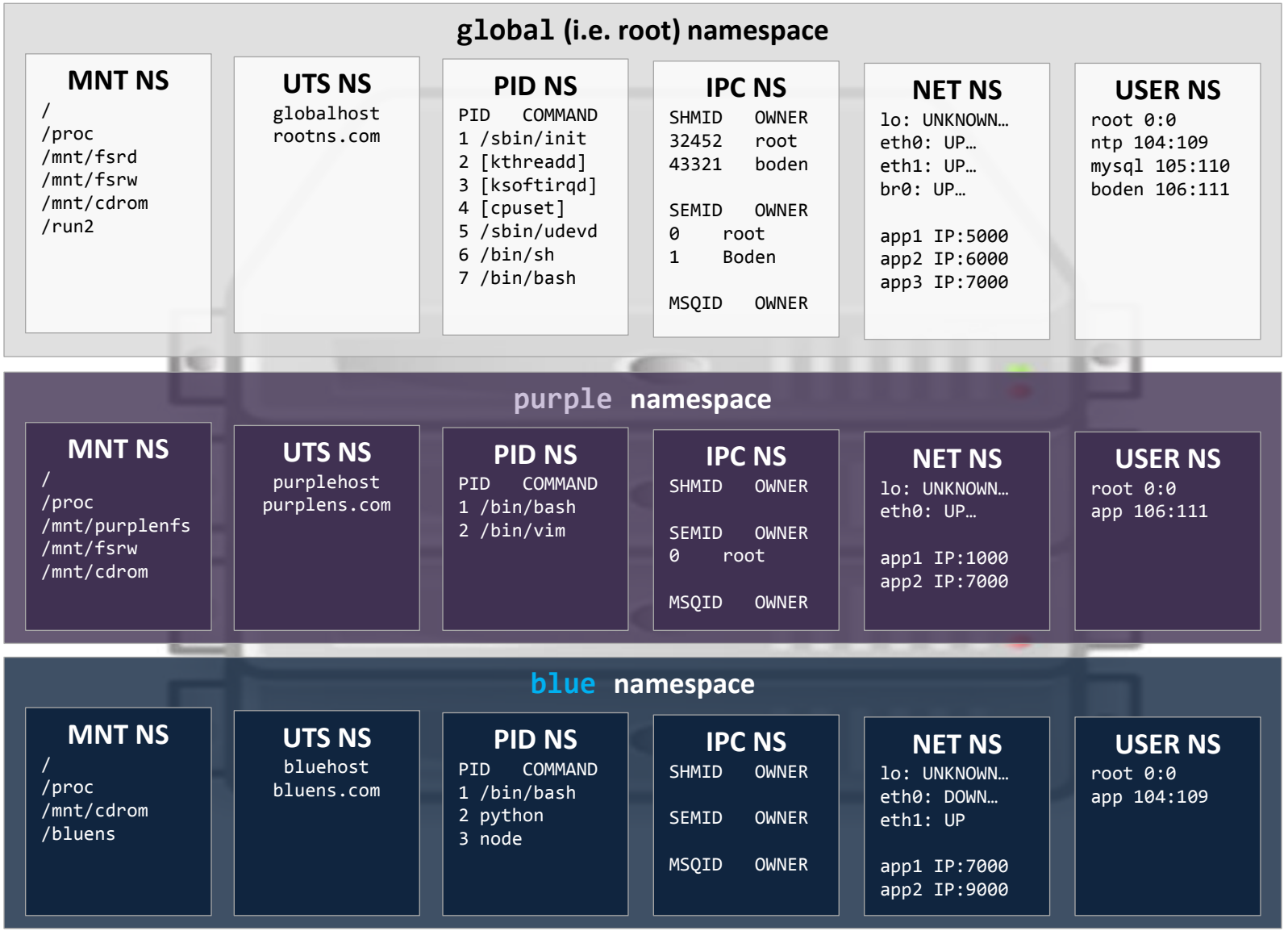
Пространства имен

- **Проблема**
  - **Как предоставить изолированное представление глобальных ресурсов задачам(процессам)**

- **Решение→ пространства имен**
  - **MNT;** точки монтирования, файловые системы и тд.
  - **PID;** процессы
  - **NET;** маршрутизация и тд..
  - **IPC;** System V IPC
  - **UTS;** хостовое и доменное имя
  - **USER;** UID и GID



Пространства имен Linux - концепция



Linux пространства имен & cgroups: Доступность

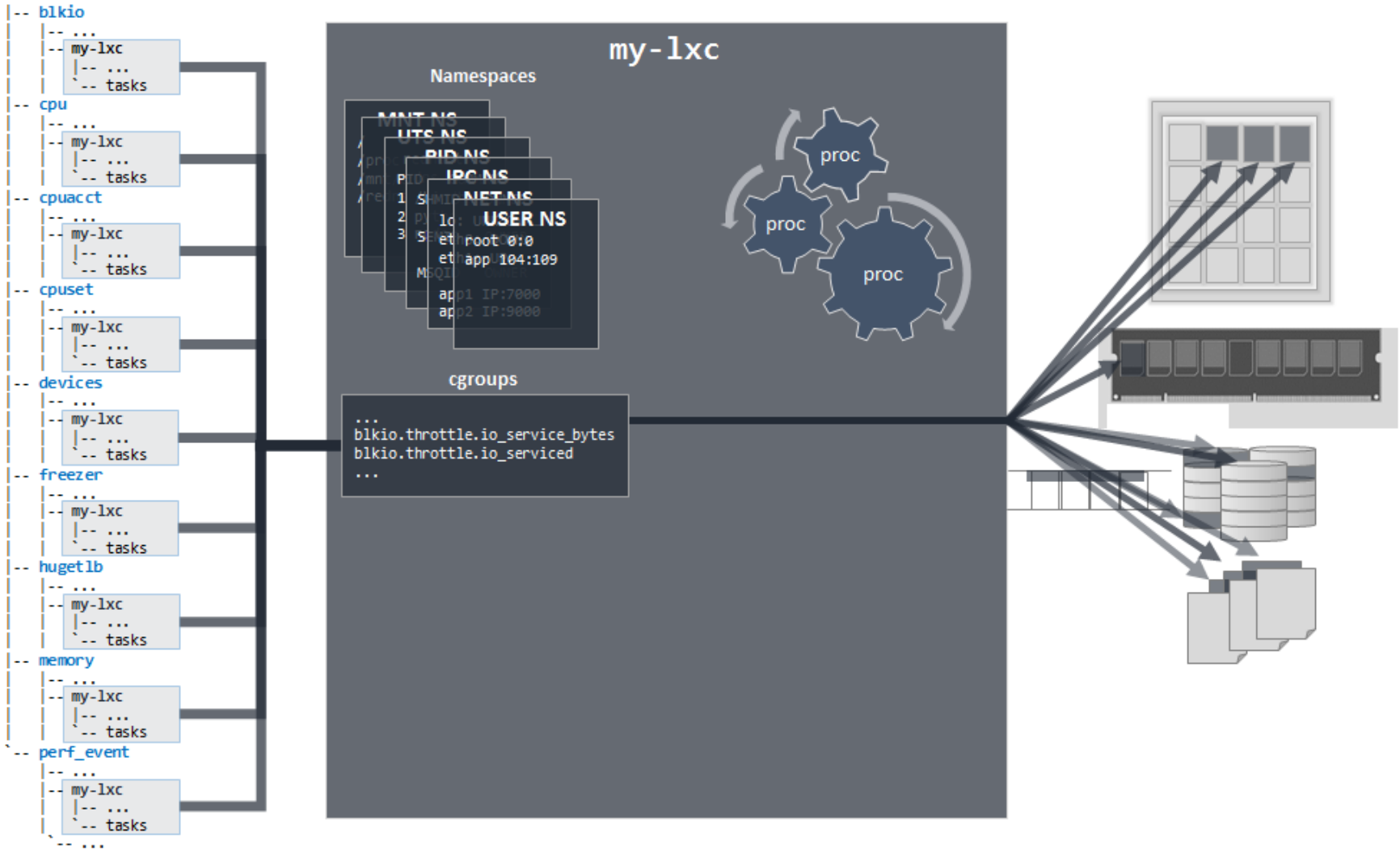
	Kernel 2.4.19	Kernel 2.6.19	Kernel 2.6.24	Kernel 2.6.29	Kernel 3.3	Kernel 3.8	Kernel 3.10+
namespaces							
Mount	✓	✓	✓	✓	✓	✓	✓
UTS		✓	✓	✓	✓	✓	✓
IPC		✓	✓	✓	✓	✓	✓
PID			✓	✓	✓	✓	✓
Network				✓	✓	✓	✓
User						✓	✓
cgroups							
Initial support			✓	✓	✓	✓	✓
net_prio					✓	✓	✓
net_cls					✓	✓	✓
Blkio async I/O							✓

Заметка: Пользовательские пространства имен поддерживаются в ядре 3.8+, но дистрибутивы не пподдерживают:

- **Отражение LXC UID/GID между контейнером и хостом**
- **Non-root LXC создание**

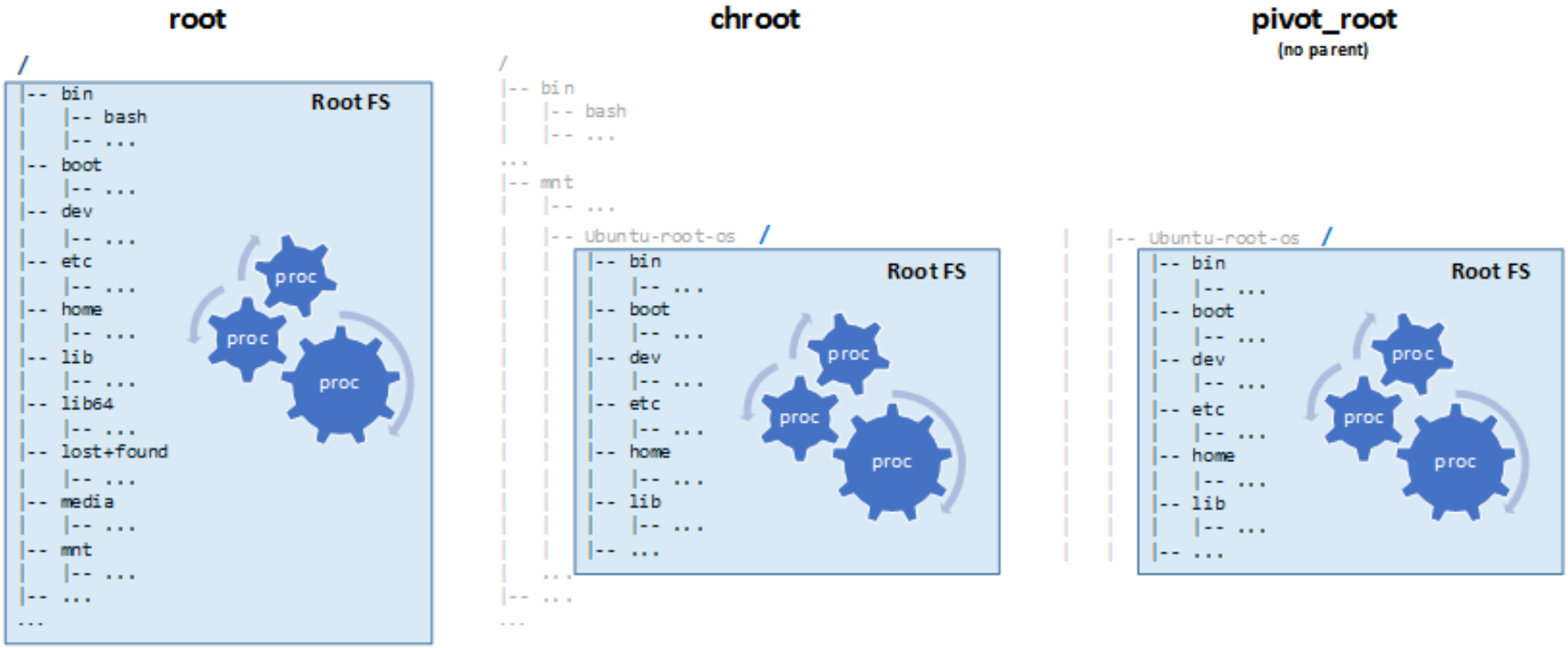
Distribution	Kernel	Notes
Ubuntu 12.04.02 (LTS)	3.5.0-23.35	
Ubuntu 13.10	3.11	User namespace disabled
RHEL 6.4	2.6.32	
RHEL 7 (2014)	3.7	
Debian 7	3.2	User namespace enabled but not fully supported
Gentoo 20122112	3.6.8	
Fedora 18	3.6.10	
Fedora 19	3.10.x	User namespace enabled but not fully supported
Linux Mint 13 (LTS)	3.2.14	
Mageia 3 (STS)	3.8.13	

Как строится контейнер?

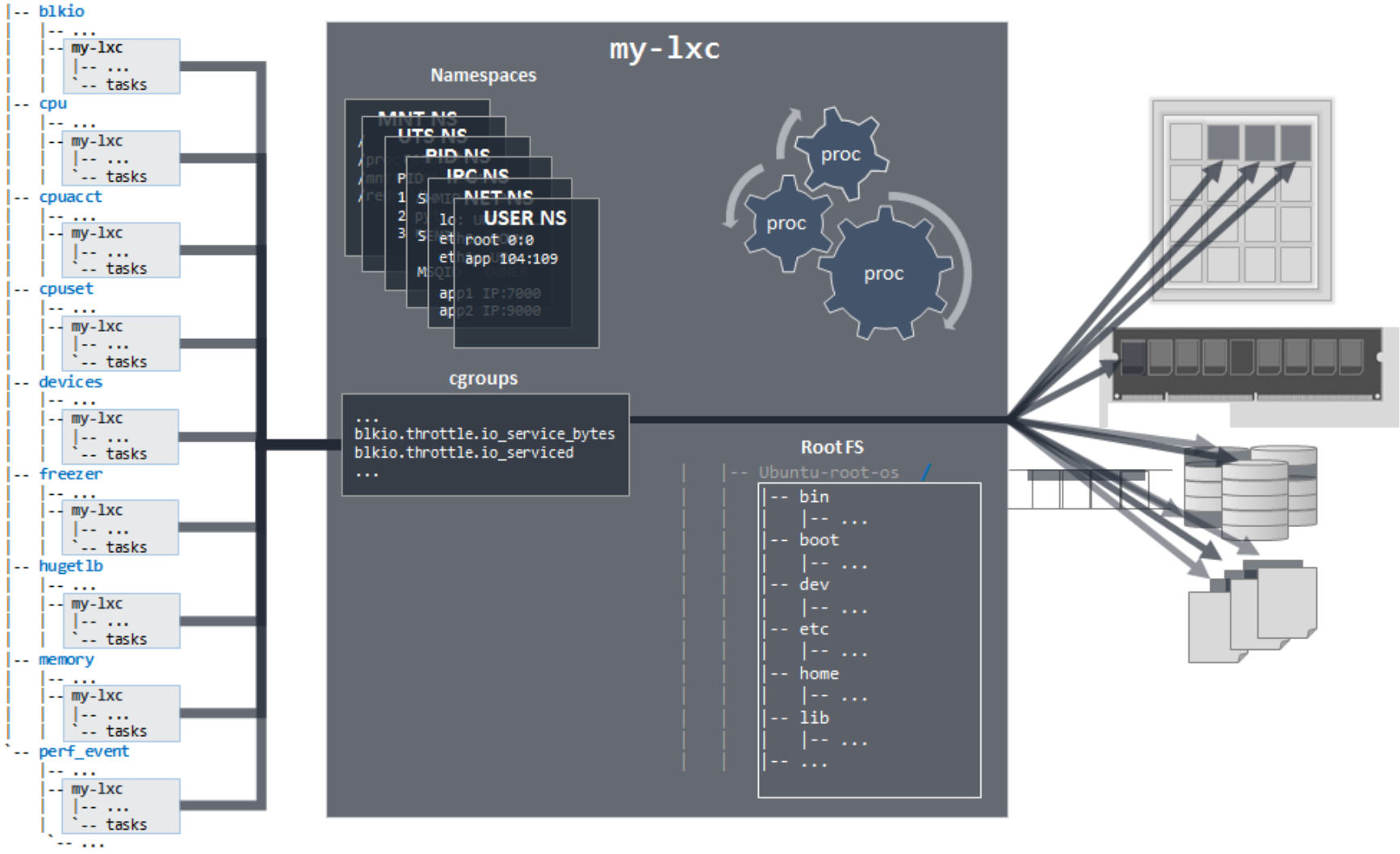


Linux chroot & pivot_root

- Использование `pivot_root` перекрывает использование chroot
- `Точка привязки pivot_root становится новой ФС`

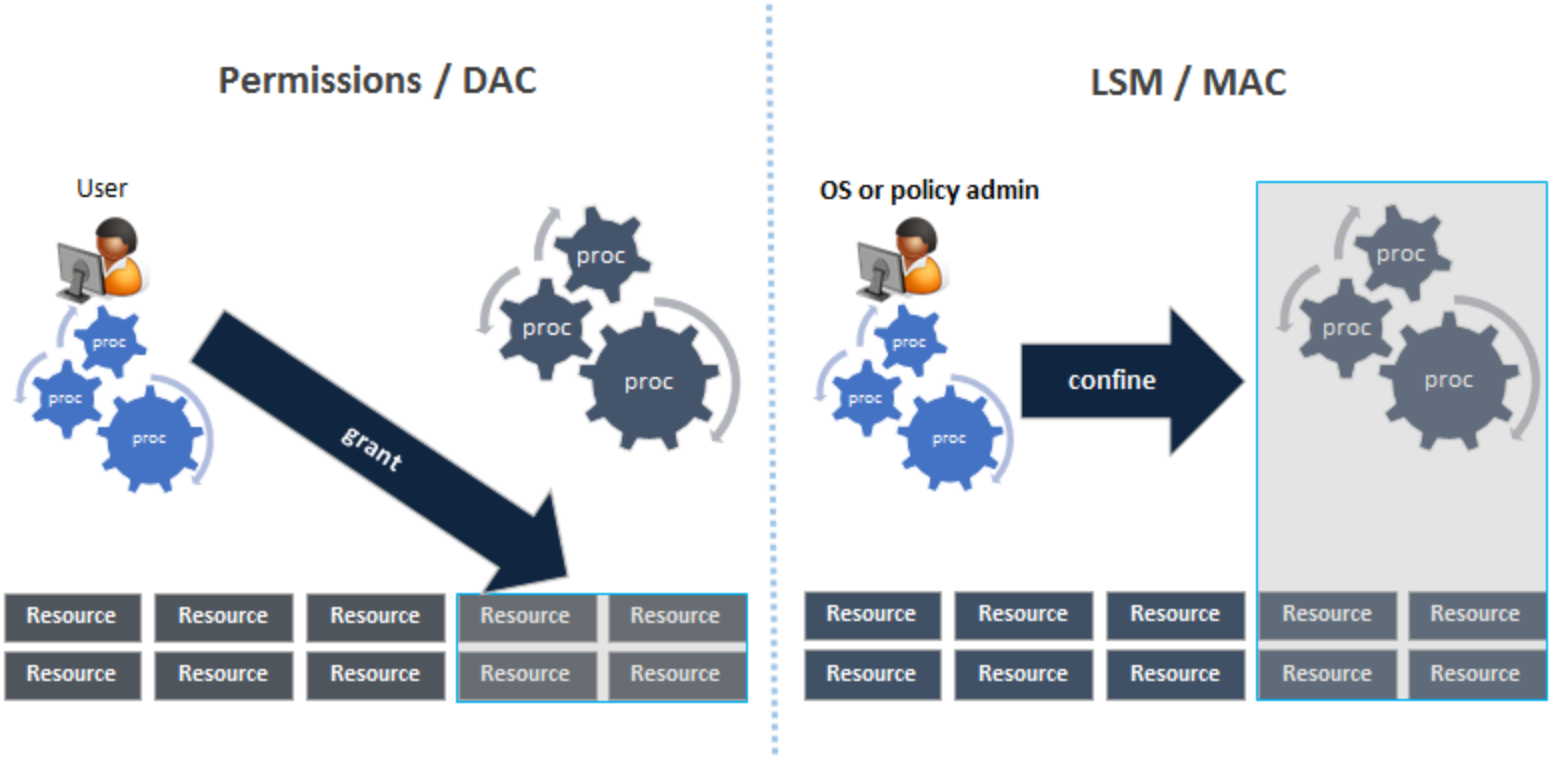


Как строится контейнер?



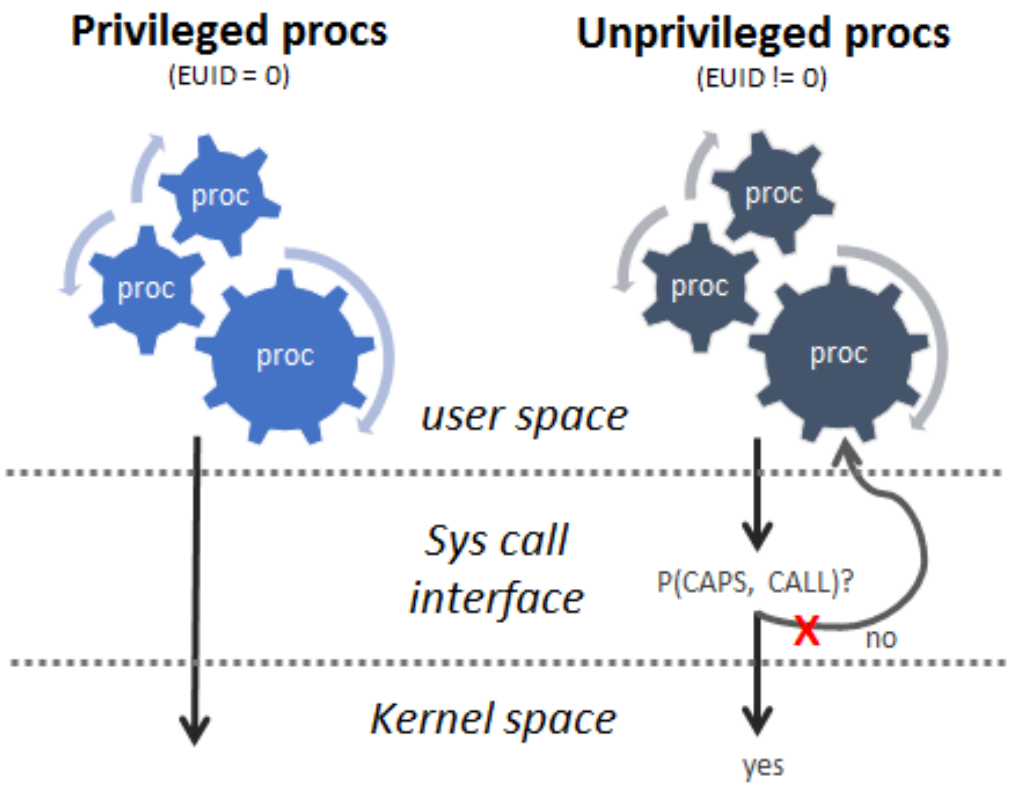
Безопасность системы Linux & MAC

- Linux Security Modules (LSM) – модули ядра, предоставляющие MAC
- MAC vs DAC
  - В MAC, Администратор(пользователь или процесс) предоставляет доступ объекту/инициатору
  - В DAC, Владелец ресурса предоставляет полномочия процессу
- Существует поддержка LSM, включая: AppArmor, SELinux, GRSEC, etc.



Возможности Linux

- **Привилегии процесса определяются системными вызовами**
- **Может быть привязан к LXC процессу(ам)**



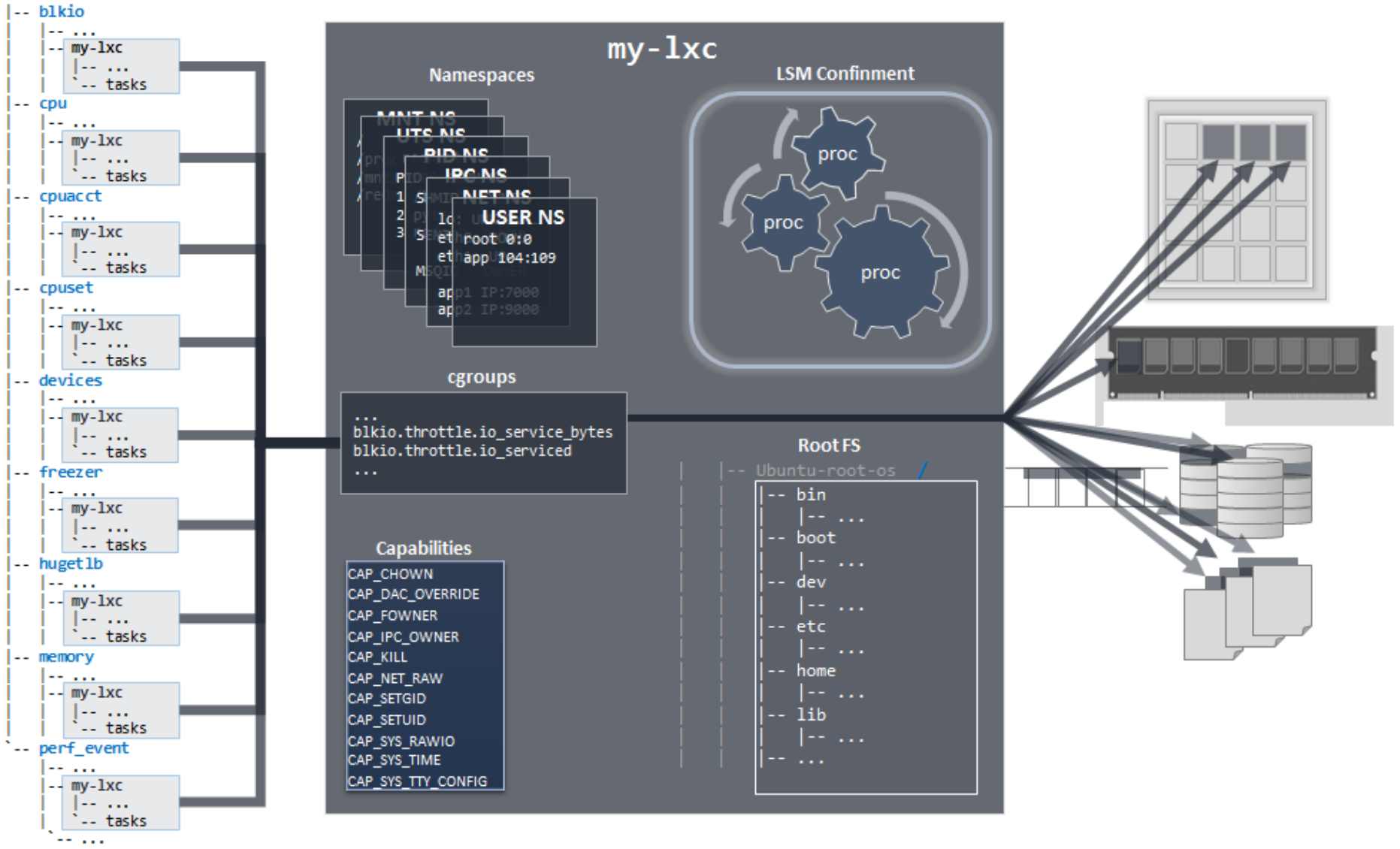
Capabilities

CAP_AUDIT_CONTROL
CAP_AUDIT_WRITE
CAP_CHOWN
CAP_DAC_OVERRIDE
CAP_DAC_READ_SEARCH
CAP_FOWNER
CAP_FSETID
CAP_IPC_LOCK
CAP_IPC_OWNER
CAP_KILL
CAP_LEASE
CAP_LINUX_IMMUTABLE
CAP_MKNOD
CAP_NET_ADMIN
CAP_NET_BIND_SERVICE
CAP_NET_BROADCAST
CAP_NET_RAW
CAP_SETGID
CAP_SETPCAP
CAP_SETUID
CAP_SYS_ADMIN
CAP_SYS_BOOT
CAP_SYS_CHROOT
CAP_SYS_MODULE
CAP_SYS_NICE
CAP_SYS_PACCT
CAP_SYS_PTRACE
CAP_SYS_RAWIO
CAP_SYS_RESOURCE
CAP_SYS_TIME
CAP_SYS_TTY_CONFIG

Другие меры безопасности

- Сокращение общего доступа к ФС с использованием привязок монтирования
- Linux seccomp
- Ограничить системные вызовы
- Обновлять ядро Linux
- Пользовательские пространства имен в 3.8+ ядрах
- Запуск контейнеров в качестве пользователя без полномочий root
- Отображение UID / GID в контейнере

Как строится контейнер?

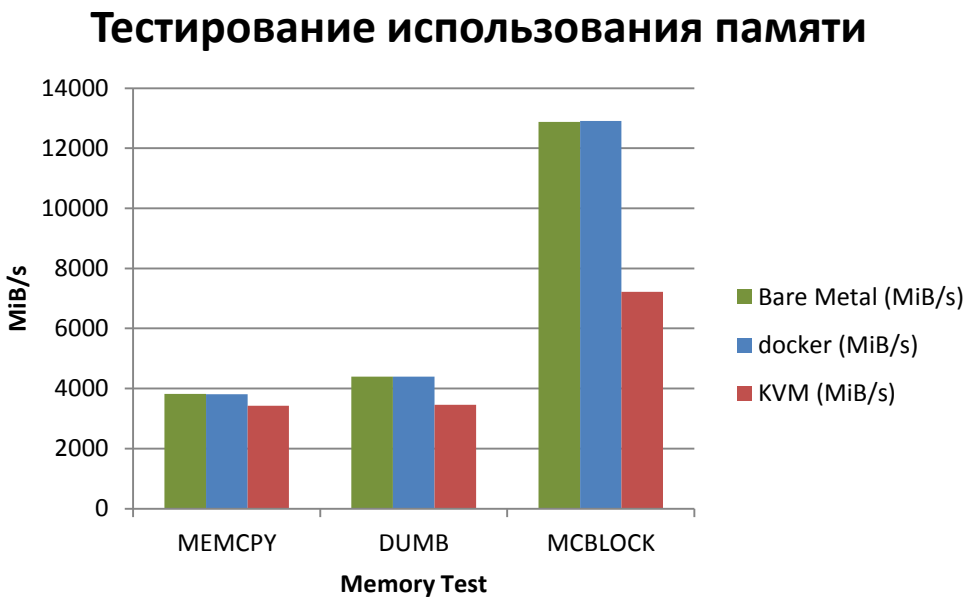


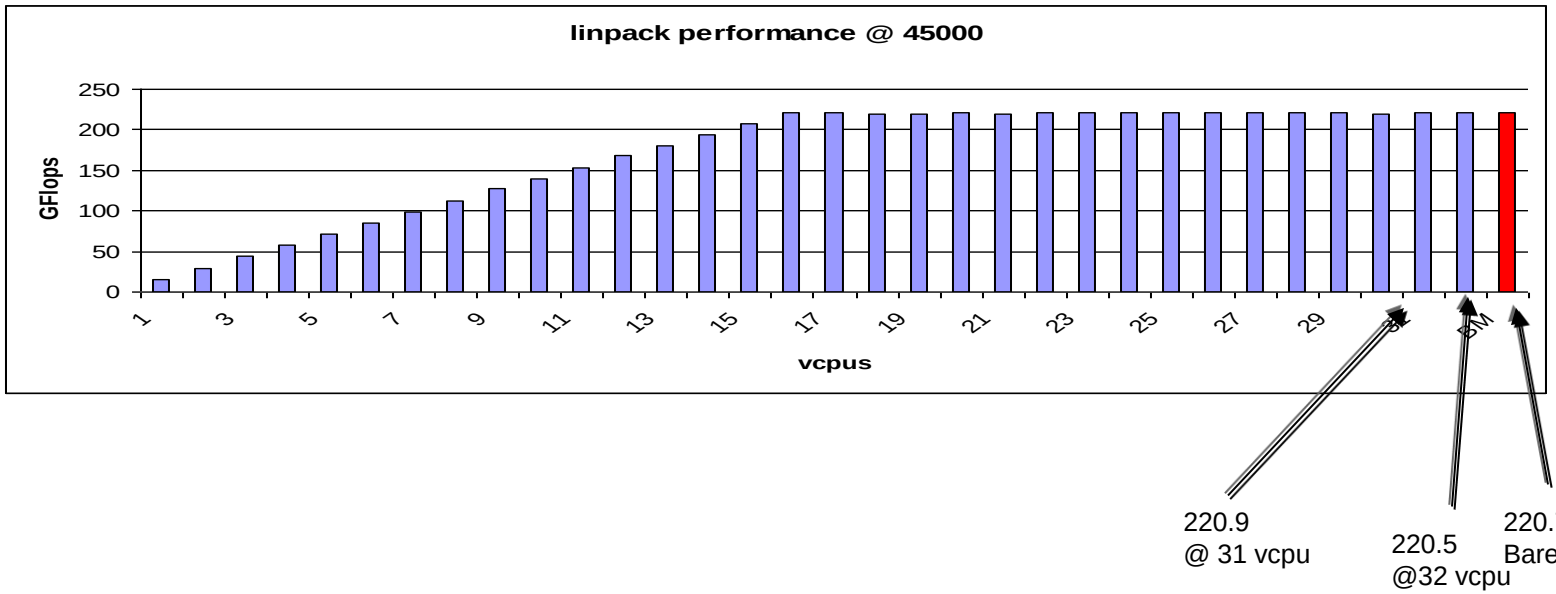
LXC Пропреитарные инструменты

	Virtuozzo	OpenVZ	Linux VServer	Libvirt-lxc	Lxc (tools)	Warden	lmctfy	Docker
Summary	Commerical product using OpenVZ under the hood	Custom Kernel providing well seasoned LXC support	A set of kernel patches providing LXC. Not based on cgroups or namespaces.	Libvirt support for LXC via cgroups and namespaces.	Lib + set of user spaces tools /bindings for LXC.	LXC management tooling used by CF.	Similar to LXC, but provides more intent based focus.	Commoditization of LXC adding support for images, build files, etc.
Part of upstream Kernel?	No	No	Partial	Yes	Yes	Yes	Yes, but additional patches needed for specific features.	Yes
License	Commercial	GNU GPL v2	GNU GPL v2	GNU LGPL	GNU LGPL	Apache v2	Apache v2	Apache v2
APIs / Bindings		- CLI - API	- CLI - C	- CLI - C - Python - Java - C# - PHP	- Python - Lua - GO - CLI			- GO - REST - CLI - Python - Other 3rd party libs
Management plane/ Dashboard	Virtuozzo Parrallels	Virtuozzo Parrallels + others		- OpenStack - Archipel - Virt-Manager	- LXC web panel - Lexy			- OpenStack - Shipyard - Docker UI

Гостевая ОС: как на железном сервере

- **Обычный LXC контейнер дает производительность как и реальный сервер**





LXC Итоги

- **Производительность как и на голом железе**
- **Быстрые вычисления в облаке**
- **Снижение потребления ресурсов (CPU, MEM) на вычислительном узле**

LXC недостатки

Есть несколько недочетов…

- **Отсутствие инструментов и поддержки для корпораций**
- **Живая миграция по-прежнему остается незавершенной**
- **Полная оркестровка между ресурсами (вычисление / хранение / сеть)**
- **Опасения по поводу безопасности**
- **Не известная технология ... пока**
- **Интеграция с существующей виртуализацией и облачным оснащением**
- **Не так много стандартов**
- **Отсутствует набор навыков**
- **И т.п.**

Использованная литература

- http://www.slideshare.net/BodenRussell/realizing-linux-containerslxc
- http://bodenr.blogspot.com/2014/05/kvm-and-docker-lxc-benchmarking-with.html
- https://www.docker.io/
- http://sysbench.sourceforge.net/
- http://dag.wiee.rs/home-made/dstat/
- http://www.openstack.org/
- https://wiki.openstack.org/wiki/Rally
- https://wiki.openstack.org/wiki/Docker
- http://devstack.org/
- http://www.linux-kvm.org/page/Main_Page
- https://github.com/stackforge/nova-docker
- https://github.com/dotcloud/docker-registry
- http://www.netperf.org/netperf/
- http://www.tokutek.com/products/iibench/
- http://www.brendangregg.com/activebenchmarking.html
- http://wiki.openvz.org/Performance