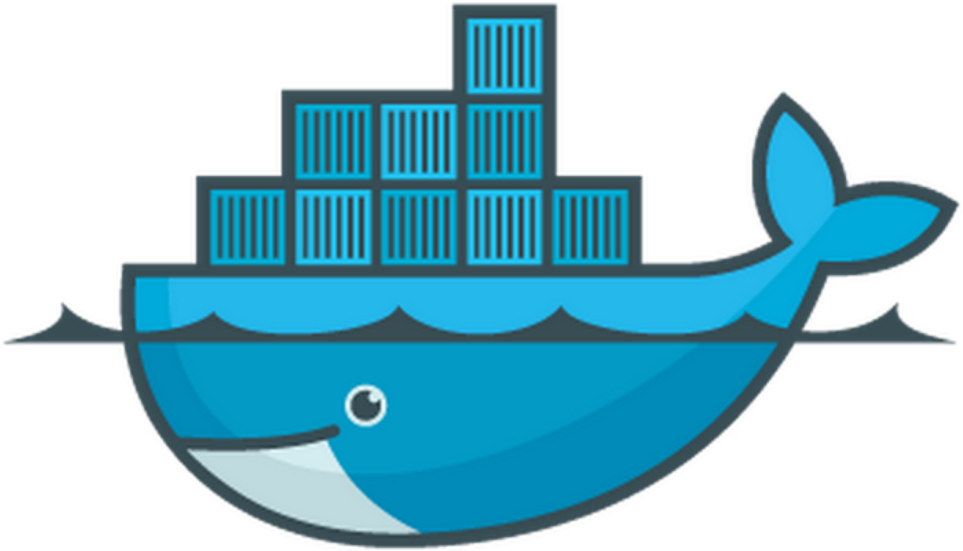


Docker Internals

Rohit Jnagal

jnagal@

Containers @ Google

Containers at scale.

Resource Isolation.

lmctfy

cAdvisor

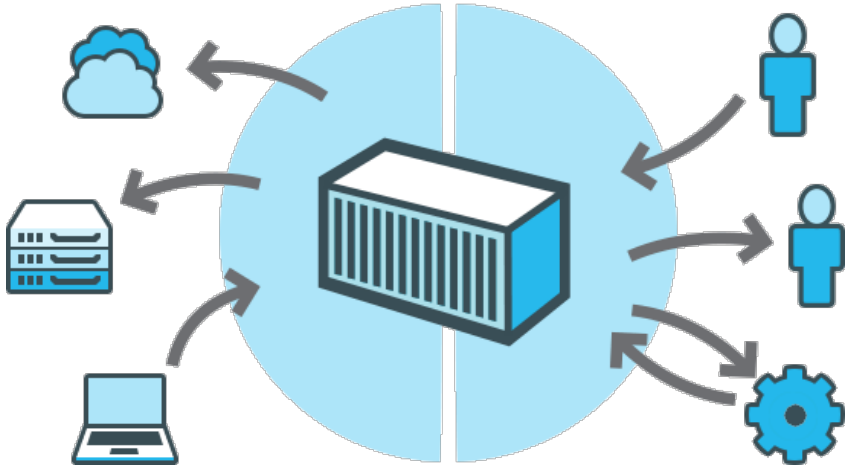
libcontainer

Kubernetes

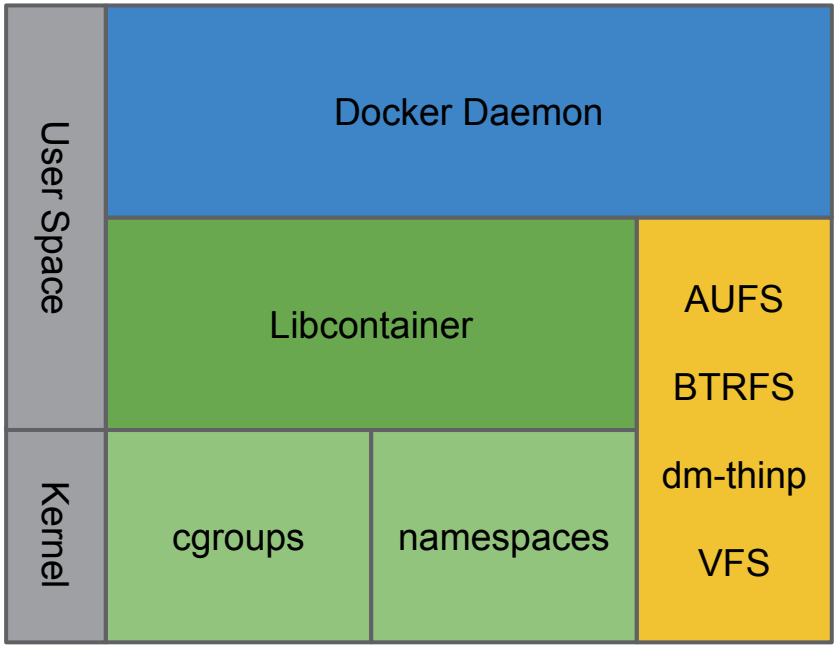
Docker 101

Build Once, Configure Once.

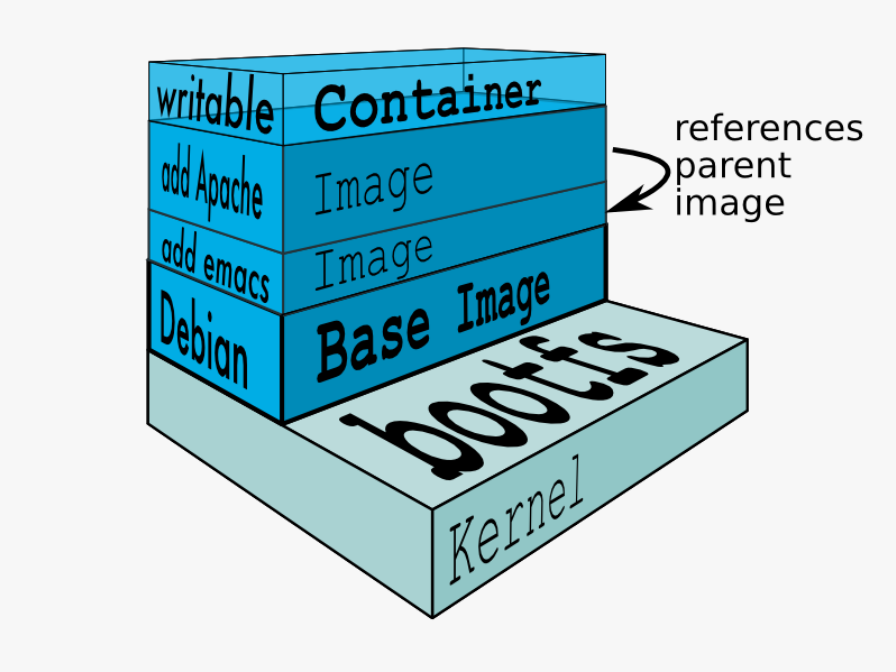
Deploy Everything*
Everywhere*
Reliably & Consistently
Efficiently
Cheaply

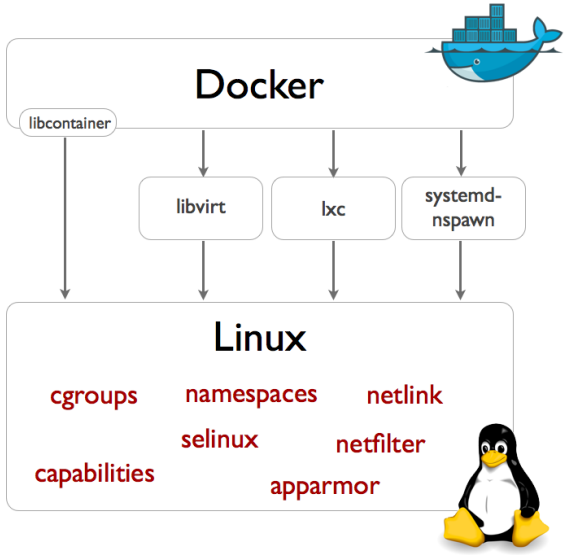


Docker Components



Docker Components





Docker Grounds up: Filesystem

File-system Isolation:

- Building a rootfs dir and chroot into it.
- With mount namespace, use pivot-root.

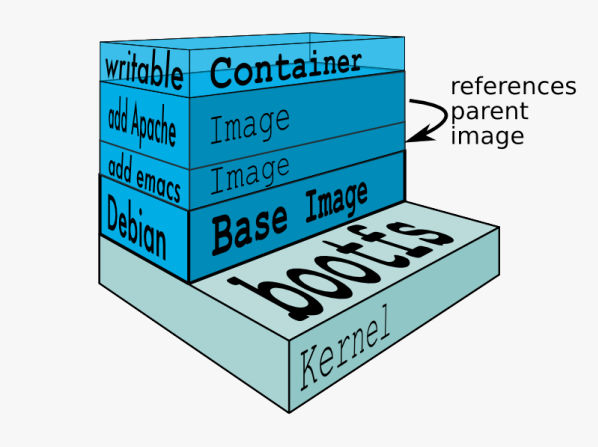
Features:

- Layering, CoW, Caching, Diffing

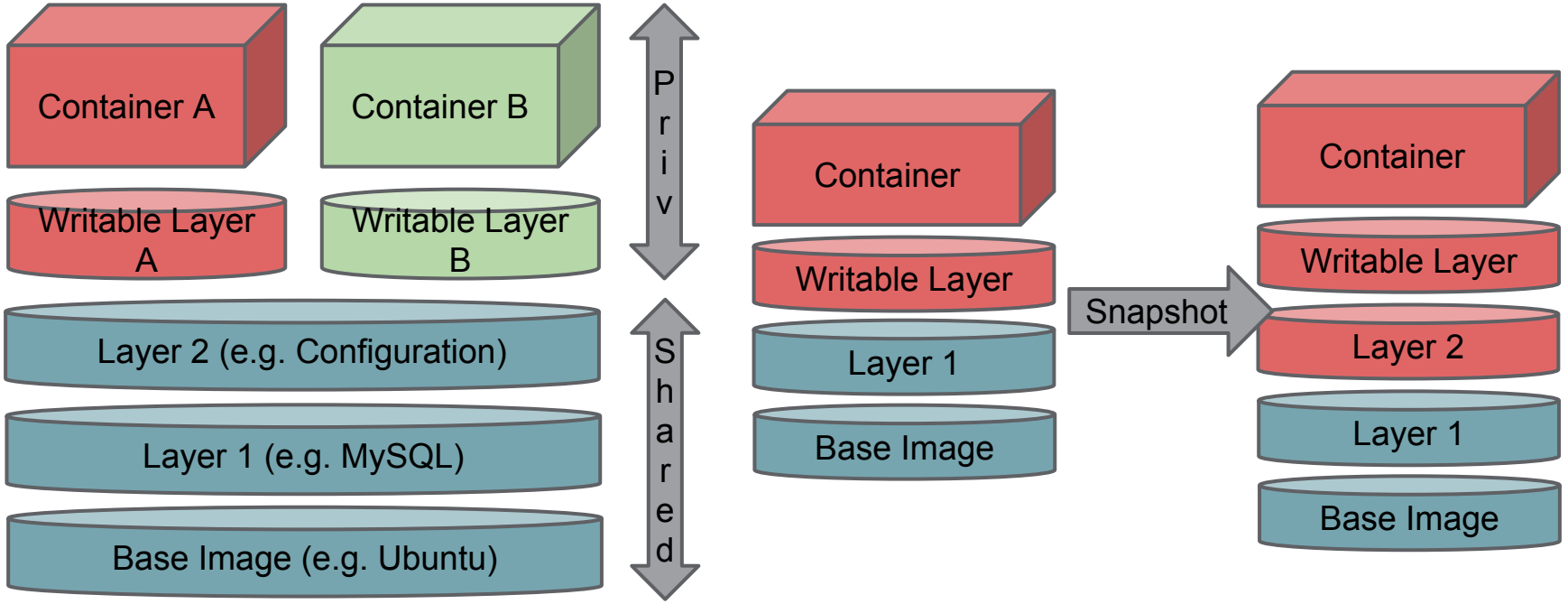
Solutions:

- UnionFS, Snapshotting FS, VFS

AUFS in action



Filesystem


Container A
Container B
Writable Layer A
Writable Layer B
Priv
Layer 2 (e.g. Configuration)
Layer 1 (e.g. MySQL)
Base Image (e.g. Ubuntu)
Shared
Container
Writable Layer
Layer 1
Base Image
Snapshot
Container
Writable Layer
Layer 2
Layer 1
Base Image

Docker Grounds up: Filesystem

	Union Filesystems	Snapshotting Filesystems	Copy-on-write block devices
Provisioning	Superfast Supercheap	Fast Cheap	Fast Cheap
Changing small files	Superfast Supercheap	Fast Cheap	Fast Costly
Changing large files	Slow (first time) Inefficient (copy-up!)	Fast Cheap	Fast Cheap
Diffing	Superfast	Superfast	Slow
Memory usage	Efficient	Efficient	Inefficient (at high densities)
Drawbacks	Random quirks AUFS not mainline !AUFS more quirks	ZFS not mainline BTRFS not as nice	Higher disk usage Great performance (except diffing)
Bottom line	**Ideal for PAAS and high density things**	**This is the Future (probably)**	**Dodge Ram 3500**

From: Jérôme Petazzoni

Docker Grounds up: Namespaces

- Process trees.
- Mounts.
- Network.
- User accounts.
- Hostnames.
- Inter-process communication.

```
pid_t pid = clone(..., flags, ...)

    CLONE_NEWUTS     hostname, domainname
    CLONE_NEWIPC     IPC objects
    CLONE_NEWPID     Process IDs
    CLONE_NEWNET     Network configuration
    CLONE_NEWNS      File system mounts
    CLONE_NEWUSER    User and Group IDs

setns(int fd, int nstype)
    CLONE_NEWIPC
    CLONE_NEWNET
    CLONE_NEWUTS

Also: unshare(flags)
```

Docker Grounds up: Resource Isolation

Cgroups : Isolation and accounting

- cpu
- memory
- block i/o
- devices
- network
- numa
- freezer

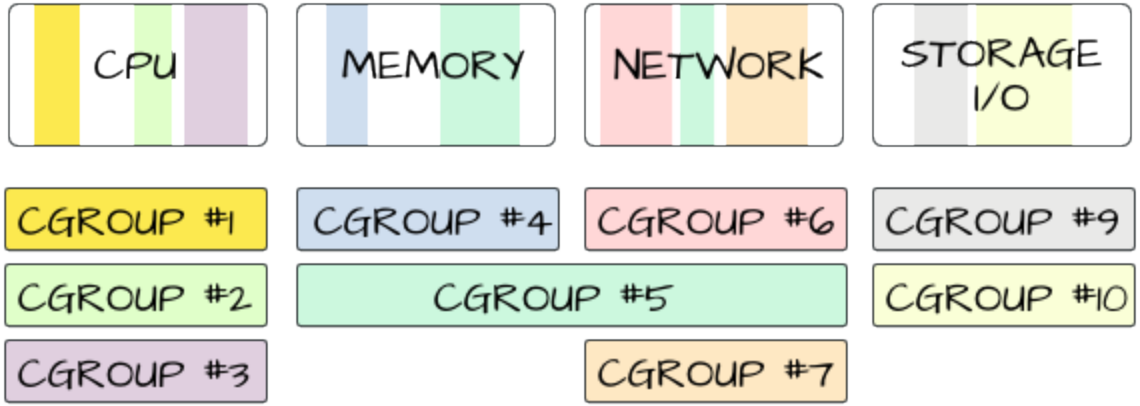


image credit: mairin

Docker Grounds up: Add Security

Security Layers

- Linux Capabilities.
- User namespaces: Unprivileged users.
- nosuid & ro mounts.
- Seccomp-bpf
- GRSEC and PAX
- Device cgroups
- Access Control: SELinux & AppArmor
- Future: Namespace aware sys/proc

image credit: Leo Reynolds

Questions

Thanks,

jnagal@google
@jnagal