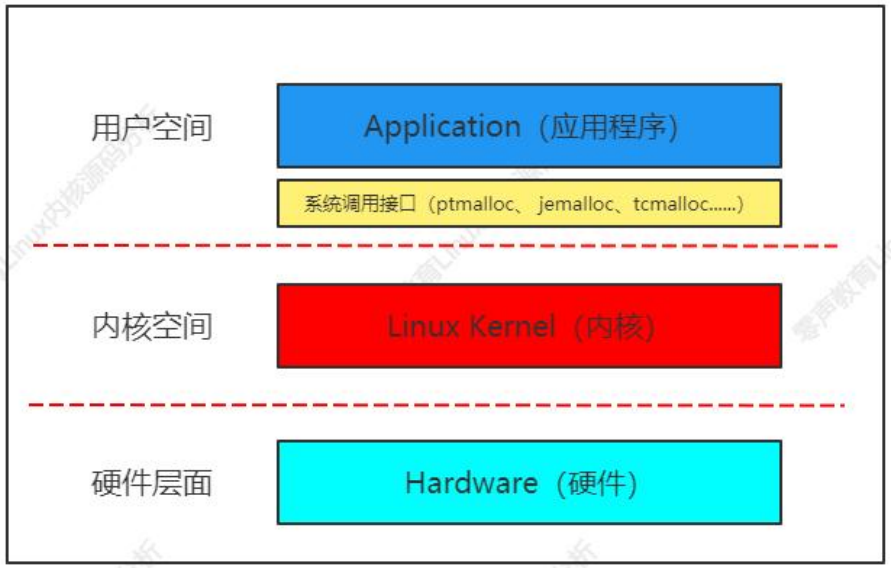
第0006讲 1Linux内核《内存布局和堆管理》



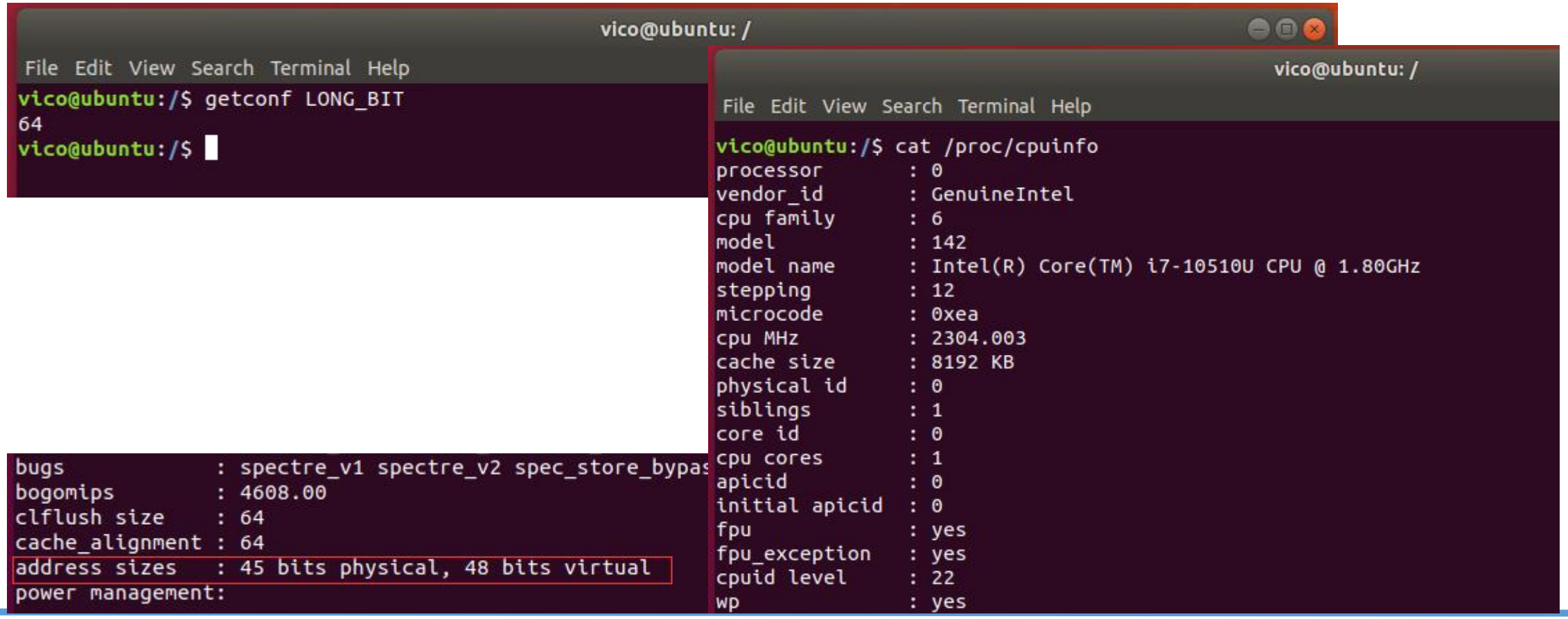
零声学院讲师：Vico老师

一、Linux内核内存布局

二、堆管理

一、Linux内核内存布局

64位Linux一般使用48位来表示虚拟地址空间，45位表示物理地址。通过命令：cat /proc/cpuinfo。查看Linux内核位数和proc文件系统输出系统软硬件信息如下：



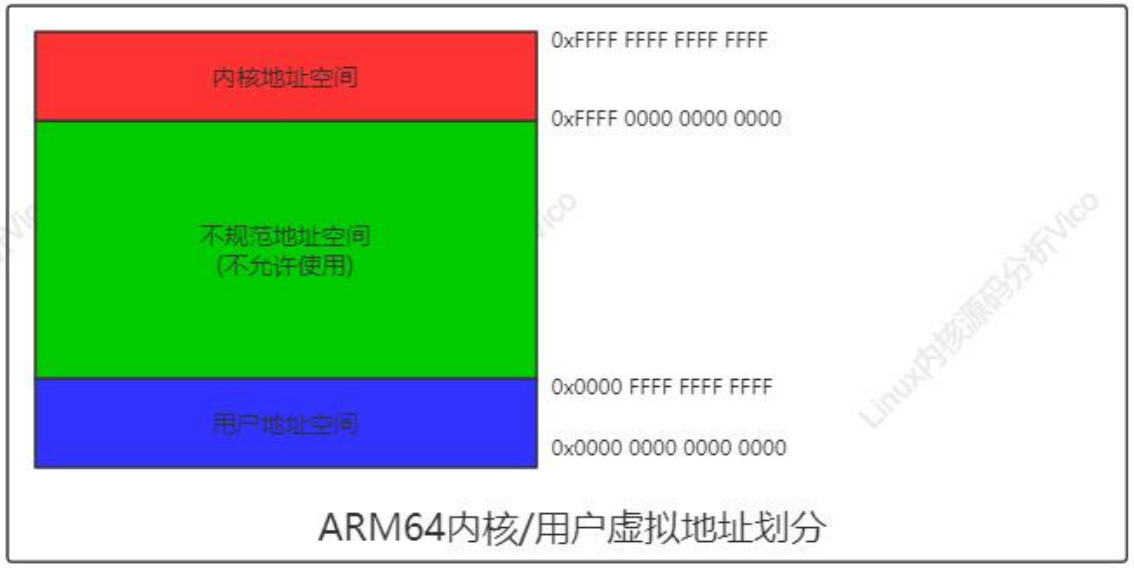
x86_64架构体系内核分布情况

```
File  Edit  View  Search  Terminal  Help
vico@ubuntu:/$ cat /proc/meminfo
MemTotal:       12165652 kB
MemFree:         9174508 kB
MemAvailable:   10371032 kB
Buffers:          163888 kB
Cached:          1178056 kB
SwapCached:            0 kB
Active:          1807440 kB
Inactive:         498900 kB
Active(anon):     956956 kB
Inactive(anon):    14936 kB
Active(file):     850484 kB
Inactive(file):   483964 kB
Unevictable:          16 kB
Mlocked:              16 kB
SwapTotal:       2097148 kB
SwapFree:        2097148 kB
Dirty:               320 kB
Writeback:             0 kB
AnonPages:        964412 kB
Mapped:           263468 kB
Shmem:             16356 kB
KReclaimable:     183968 kB
Slab:             267248 kB
SReclaimable:     183968 kB
SUnreclaim:        83280 kB
KernelStack:       12128 kB
PageTables:        42020 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:     8179972 kB
Committed_AS:    4746628 kB
VmallocTotal:   34359738367 kB
VmallocUsed:       27780 kB
VmallocChunk:          0 kB
Percpu:            49152 kB
HardwareCorrupted:     0 kB
AnonHugePages:         0 kB
ShmemHugePages:        0 kB
ShmemPmdMapped:        0 kB
FileHugePages:         0 kB
FilePmdMapped:         0 kB
CmaTotal:              0 kB
CmaFree:               0 kB
HugePages_Total:       0
HugePages_Free:        0
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       2048 kB
Hugetlb:               0 kB
DirectMap4k:      231232 kB
DirectMap2M:     6191104 kB
DirectMap1G:     7340032 kB
vico@ubuntu:/$
```

ARM64架构采用48位物理寻址方式，最大可寻找256TB的物理地址空间。对于目前应用完全足够，不需要扩展到64位。虚拟地址也同样最大支持48位寻址。Linux内核在大多数体系结构上将地址空间划为：用户空间和内核空间。

用户空间（User space）：0x0000_0000_0000_0000至0x0000_FFFF_FFFF_FFFF。

内核空间（Kernel space）：0xFFFF_0000_0000_0000至0xFFFF_FFFF_FFFF_FFFF。



ARM64内核/用户虚拟地址划分

二、堆管理

堆是进程中主要用于动态分配变量和数据的内存区域，堆的管理对应程序员不是直接可见的。malloc和内核之间的经典接口是brk系统调用，负责扩展/收缩堆。

堆是一个连续的内存区域，在扩展时自下至上增长。其中**mm_struct**结构，包含堆在虚拟地址空间中的起始和当前结束地址(start_brk和brk)。

include > linux > C mm_types.h > kioctx_table

```
378  struct mm_struct {
379      struct {
380          struct vm_area_struct *mmap;        /* list of VMAs */
381          struct rb_root mm_rb;
457          spinlock_t arg_lock; /* protect the below fields */
458          unsigned long start_code, end_code, start_data, end_data;
459          unsigned long start_brk, brk, start_stack;
460          unsigned long arg_start, arg_end, env_start, env_end;
```

1、brk系统调用指定堆在虚拟地址空间中新的结束地址（如果堆将要收缩，当然可以小于当前值）。brk系统调用动态分配，具体Linux内核源码分析如下：

mm > C mmap.c > SYSCALL_DEFINE1(brk, unsigned long, brk)

```
191
192 SYSCALL_DEFINE1(brk, unsigned long, brk)
193 {
194     unsigned long retval;
195     unsigned long newbrk, oldbrk, origbrk;
196     struct mm_struct *mm = current->mm;
197     struct vm_area_struct *next;
198     unsigned long min_brk;
199     bool populate;
200     bool downgraded = false;
```

2、per-CPU计数器，引入它用来加速SMP系统上计数器操作，Linux具体内核源码如下：

include > linux > C percpu_counter.h > ...

```
18  #ifdef CONFIG_SMP
19
20  struct percpu_counter {
21      raw_spinlock_t lock;
22      s64 count;
23  #ifdef CONFIG_HOTPLUG_CPU
24      struct list_head list;  /* All percpu_counters are on a list */
25  #endif
26      s32 __percpu *counters;
27  };
28
```

办学宗旨：一切只为渴望更优秀的你

办学愿景：让技术简单易懂