

Linux内核中的DeviceMapper机制(下) PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022_Linux_E5_86_85_E6_A0_c103_144267.htm 用户空间部分 Device mapper在用户空间相对简单，主要包括device mapper库和dmsetup工具。Device mapper库就是对ioctl、用户空间创建删除device mapper逻辑设备所需必要操作的封装，dmsetup是一个提供给用户直接可用的创建删除device mapper设备的命令行工具。因为它们的功能和流程相对简单，在本文中对它们的细节就不介绍了，用户空间主要负责如下工作：1、发现每个mapped device相关的target device；2、根据配置信息创建映射表；3、将用户空间构建好的映射表传入内核，让内核构建该mapped device对应的dm_table结构；4、保存当前的映射信息，以便未来重新构建。下面我们主要通过实例来说明dmsetup的使用，同时进一步说明device mapper这种映射机制。用户空间中最主要的工作就是构建并保存映射表，下面给出一些映射表的例子：1) 0 1024 linear /dev/sda 204 1024 512 linear /dev/sdb 766 1536 128 linear /dev/sdc 0 2) 0 2048 striped 2 64 /dev/sda 1024 /dev/sdb 0 3) 0 4711 mirror core 2 64 nosync 2 /dev/sda 2048 /dev/sdb 1024 例子1中将逻辑设备0~1023扇区、1024~1535扇区以及1536~1663三个地址范围分别以线形映射的方式映射到/dev/sda设备第204号扇区、/dev/sdb设备第766号扇区和/dev/sdc设备的第0号扇区开始的区域。例子2中将逻辑设备从0号扇区开始的，长度为2048个扇区的段以条带的方式映射的到/dev/sda设备的第1024号扇区以及/dev/sdb设备的第0号扇区开始的区域。同时告诉内核这个条带类型的target

driver存在2个条带设备与逻辑设备做映射，并且条带的大小是64个扇区，使得驱动可以该值来拆分跨设备的IO请求。例子3中将逻辑设备从0号扇区开始的，长度为4711个扇区的段以镜像的方式映射到/dev/sda设备的第2048个扇区以及/dev/sdb设备的第1024号扇区开始的区域。映射表确定后，创建、删除逻辑设备的操作就相对简单，通过dmsetup如下命令就可以完成相应的操作。 dmsetup create 设备名 映射表文件 /* 根据指定的映射表创建一个逻辑设备 */ dmsetup reload 设备名 映射表文件 /* 为指定设备从磁盘中读取映射文件，重新构建映射关系 */ dmsetup remove 设备名 /* 删除指定的逻辑设备 */ 当用户空间根据映射表下达创建逻辑设备命令后，device mapper在内核中就根据传入的参数和映射关系建立逻辑地址到物理地址的映射关系。根据映射表例子1中的映射关系建立的设备如图4所示，图中的下半部分就抽象地描绘出了按照该映射表在内核中建立的逻辑地址到物理地址的映射关系。 100Test 下载频道开通，各类考试题目直接下载。详细请访问

www.100test.com