

巧妙删除带子目录的目录Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/644/2021\\_2022\\_\\_E5\\_B7\\_A7\\_E5\\_A6\\_99\\_E5\\_88\\_A0\\_E9\\_c103\\_644654.htm](https://www.100test.com/kao_ti2020/644/2021_2022__E5_B7_A7_E5_A6_99_E5_88_A0_E9_c103_644654.htm) 如现在在操作系统中有个目录文件compierebook，在其下面有两个子目录文件，分别为chap01与chap02。现在用户需要删除目录文件compierebook。此时该如何操作呢?如果用户直接使用rmdir compierebook命令，那么系统会提示“目录非空，不能够删除”的错误信息。拿到用户只有先利用命令删除各个子目录，然后再来删除这个 compierebook目录文件吗?其实不用。下面笔者就给大家介绍一个删除带子目录的目录文件。即同时删除所有的子目录文件与当前的目录文件。

一、文件创建与删除的控制管理。在Unix操作系统中，普通文件与目录都是文件。为此在操作系统中无论是普通文件还是目录文件都对应于一个文件名和inode数。当用户创建一个目录时，在这个文件的上级目录文件里会添加关于创建目录的文件名和inode信息。同理，当删除一个目录文件时，也会同时删除其上级目录文件中关于这个要删除目录的信息。所以在删除一个目录时，系统会查看这个目录文件，看看其是否有下级子目录的信息。如果有的话，就会拒绝删除，并提醒用户“目录非空不能够删除”。如果没有下一级子目录信息的话，那么系统就会删除用户指定的目录。所以说，在Unix系统中删除某个目录时，必须一级级的删除。现删除子目录，然后再删除上一级目录。

二、在rmdir中带多个参数。在大部分情况下，我们在使用rmdir命令删除目录的时候，都只是带一个目录文件名参数，如rmdir compierebook。其实查看这个命令的帮

助，可以发现这个命令可以带多个文件名参数，即同时删除多个目录文件。如用户可以使用 `rmdir 目录文件1 目录文件2 目录文件3`(中间利用空格分割)等来同时删除多个目录文件。如果需要注意的是，其在删除目录的时候，有先后顺序。即先删除目录文件1、再删除目录文件2、最后删除目录文件3。所以说，虽然在同一个命令中带入了三个参数，但是其在删除的时候不是一起删除的，而是根据参数的顺序来依次删除文件。这是否可以给我们一点启示呢?既然其实一个个依次删除的，那我们就可以将子目录文件参数放在前、将上级目录文件参数放在后，是否可以一次性删除带子目录的目录文件呢?如上例所示，如果笔者使用 `rmdir chap01 chap02 compierebook`会有什么结果呢?执行这个命令后，大家会惊奇的发现，这个命令可以执行成功。然后再去查看相关的目录，会发现这个目录文件(包括其下级的两个子目录文件)都已经删除了。所以说，我们在删除带子目录的的目录文件时，不需要再分次去执行 `rmdir`命令。而只需要在这个命令中依次填上多个目录文件名参数，然后执行一次即可。在执行这个命令的时候，系统先删除子目录 `chap01`，然后会在其上级目录文件 `compierebook`中将关于 `chap01`目录的文件名和 `inode`信息删除。这个步骤完成后，再来删除第一个文件 `chap02`。其也是进行这两步操作，在删除自身目录的同时更新上级目录文件的相关信息。然后再删除第三个文件。在删除第三个文件的时候，由于在目录文件中已经没有了子目录文件的信息，所以此时这第三个文件就可以正常删除了。在执行这个命令的时候，主要注意三个细节问题。一是这个命令可以带多个目录文件参数，不同的参数之间利用空格隔开。注意这里

不是利用分号或者冒号等符号分割，只需要使用空格即可。

二是这个命令中的各个参数文件是由先后顺序的。如果目录文件的参数位置不对，很可能删除作业不会成功，甚至会出现一些莫名其妙的结果。对于这一点，笔者在后面的内容中还会具体举例子说明。第三在删除某个目录文件时，用户必须位于这个目录之外的位置。如果这个命令中带了多个目录文件名参数时，要在这些指定的目录之外。否则的话系统会提示用户无法删除这些目录。这个比较容易理解，因为在Windows操作系统中也有类似的控制。笔者最好强调一点，当删除某个目录文件时其主要会进行两步操作，分别为删除自身目录与更新上级目录中的信息。当一个命令中带有多个目录文件参数时，这两步操作也是同时完成的。而不是说先全部删除所指定的目录文件，然后再更新其上级目录文件中的信息。理解这个内容，对于真正的理解rmdir命令删除多个目录文件会有很大的帮助。

三、若目录文件名位置放置不当会导致什么后果? 如果在使用这个rmdir命令的时候，这个目录文件名参数的位置放置不对，会有什么后果呢?如笔者现在是按这个格式书写这个命令：`rmdir compierebook chap01 chap02`。在执行了这个命令之后，我们再查看系统目录会发现compierebook这个目录文件没有被删除。但是其下面的两个子目录文件 `chao01`与`chap02`已经被删除了。最后系统会提示用户“目录非空不能够删除”。这到底是怎么回事呢? 在解释这个原因之前，笔者再给大家举一个例子。如笔者现在使用命令`rmdir chap01 compierebook chap02`。将第一个目录文件参数与第二个目录文件参数位置对调一下。然后再执行这个命令，会出现什么结果呢?其实这个命令的执行结果与上面

这个命令的执行结果一样。compierebook目录文件仍然存在，但是其与两个子文件已经被删除了。不过最后系统仍然会提示一个“目录非空、删除失败”的错误信息。Unix操作系统到底是在变什么模式呢？其实，我们只要将这个命令分解成三个命令即可。如可以将`rmdir compierebook chap01 chap02`命令分解为`rmdir compierebook`、`rmdir chap01`与`rmdir chap02`。在执行第一个命令的时候，由于此时其下面的子目录文件还没有被删除，故第一个命令执行失败。然后再执行第二个命令，由于在目录文件`chao01`下面没有子目录文件了，故可以正常删除。同理，第三个命令也正确执行了。从以上的分析中至少可以看出三点。一是某个目录删除失败并不会影响其他的目录的删除操作。二是这个命令并不会循环操作。即在删除第一个目录文件时，如果失败了，则会直接跳过。在其他目录删除完成之后，不会再回过头来尝试再次删除上次删除失败的目录文件。由于不会循环操作，为此这个目录文件参数位置的放置就直接影响到这个命令的执行结果。第三是目录删除成功不会有错误警告信息。但是如果目录删除失败了，则会提醒错误信息。如果有两个目录文件没有被成功删除，那么系统就会提醒两个错误信息。并且会提示，那个目录文件没有被成功删除。所以这个命令执行完成后，如果有错误信息的话，那么仔细查看这个错误提示就非常有必要。特别是对比错误信息的条数与提供的目录文件参数的数量，可以判断有多少目录文件被成功删除了。如利用脚本程序来实现这个功能的话，就可以利用`cat`命令来统计错误信息出现的次数，然后自动与参数的个数进行比较，并得出有多少目录文件被成功删除的结论。四、利用`mkdir`来建立目录树。

了解了这个rmdir目录文件删除操作的原理之后，那么再来理解mkdir命令来建立目录树就会比较轻松了。建立目录树就是一次性建立多个目录，并且是带子目录的目录。也就是说，现在要一次性建立三个目录computerbook、chap01、chap02。其中后面两个目录为第一个目录的子目录。现在这三个目录都不存在。一个个的运行mkdir命令虽然可以最终实现既定的结果，但是显然操作起来比较麻烦。如果要建立目录比较多的话，那要执行能够多个命令了。其实根据上面的rmdir命令的工作原理，大家也可以想的到，通过mkdir命令加多个参数的形式来一次性建立目录树。如要实现上面的需求，可以使用命令mkdir computerbook chap01 chap02命令即可。注意参数的顺序非常的重要。如果参数的位置前后颠倒的话，那么这个命令就可能执行不成功。会一个目录文件都没创建或者只是创建了部分的目录。所以在使用这个命令的时候，需要注意参数的顺序。同时要清楚一点，带多个参数的情况下，其建立各个目录的动作是相对独立的。了解这些内容后，那么利用mkdir命令来一次性创建目录树就会变得简单许多。

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](http://www.100test.com)