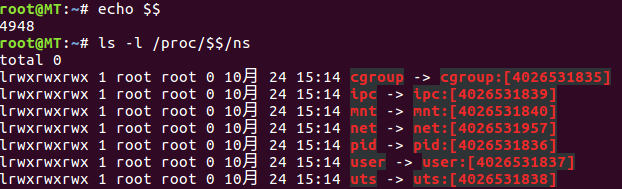
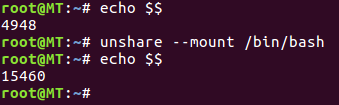
**Linux内核命名空间**

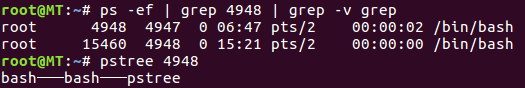
## mnt namespace



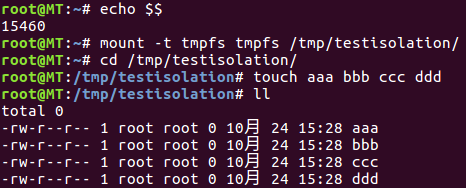
使用unshare隔离mnt namespace



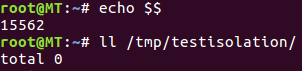
15460和4948两个进程的关系，我们能清楚地看到这是父子关系的两个进程，虽然都是bash



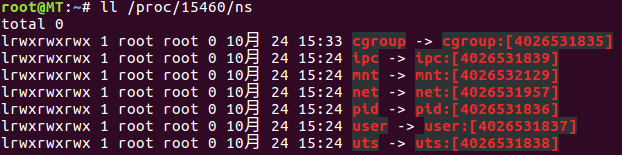
在15460进程下创建文件



在新开启的终端下看不到文件



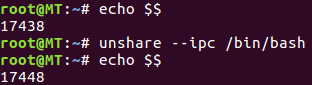
可以看到mnt在不同的命名空间



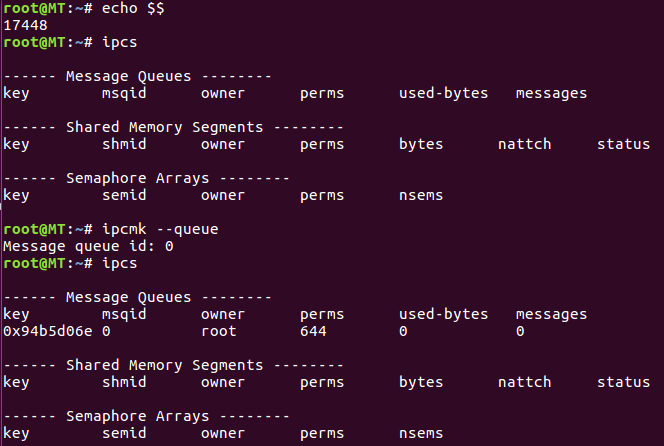
调用unshare()的主要作用就是不启动一个新进程就可以起到隔离的效果，相当于跳出原先的namespace进行操作。这样，你就可以进行一些需要隔离的操作

# ipc namespace

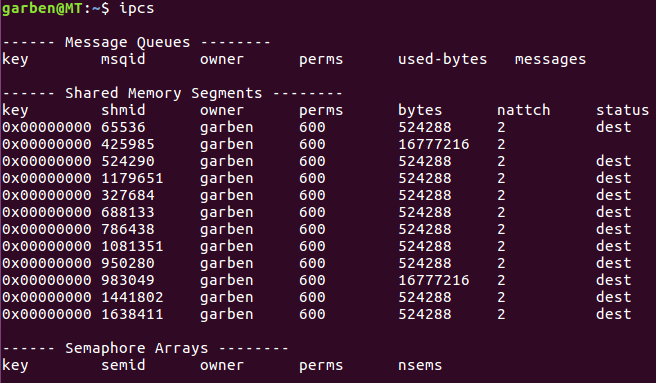
使用unshare隔离ipc namespace



在此命令空间下创建消息队列

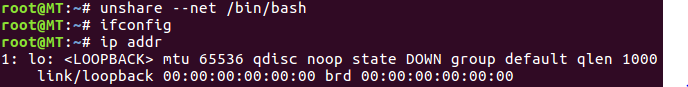


在另一个进程中查找不到刚创建的消息队列，ipc隔离成功

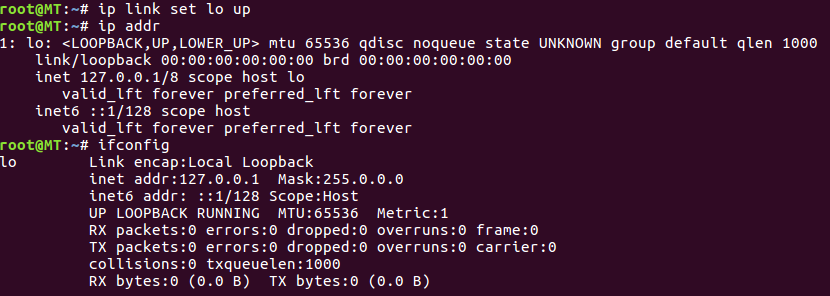


# net namespace

创建一个net的命令空间，发现是一个纯净的网络空间，连环路地址都没有

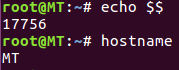


添加环路地址

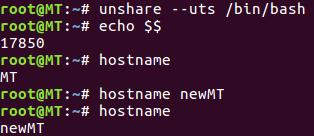


# uts namespace

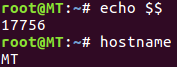
查看本机hostname



在uts命名空间中更改hostname



返回原进程查看hostname，发现并没有改变

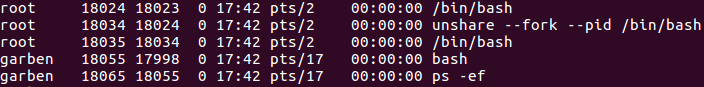


# pid namespace

新fork一个在pid namespace的进程，看到在其命名空间里pid显示为1



然而这个进程的真是pid是18034



# user namespace

创建一个user namespace，发现其用户名已变为nobody

