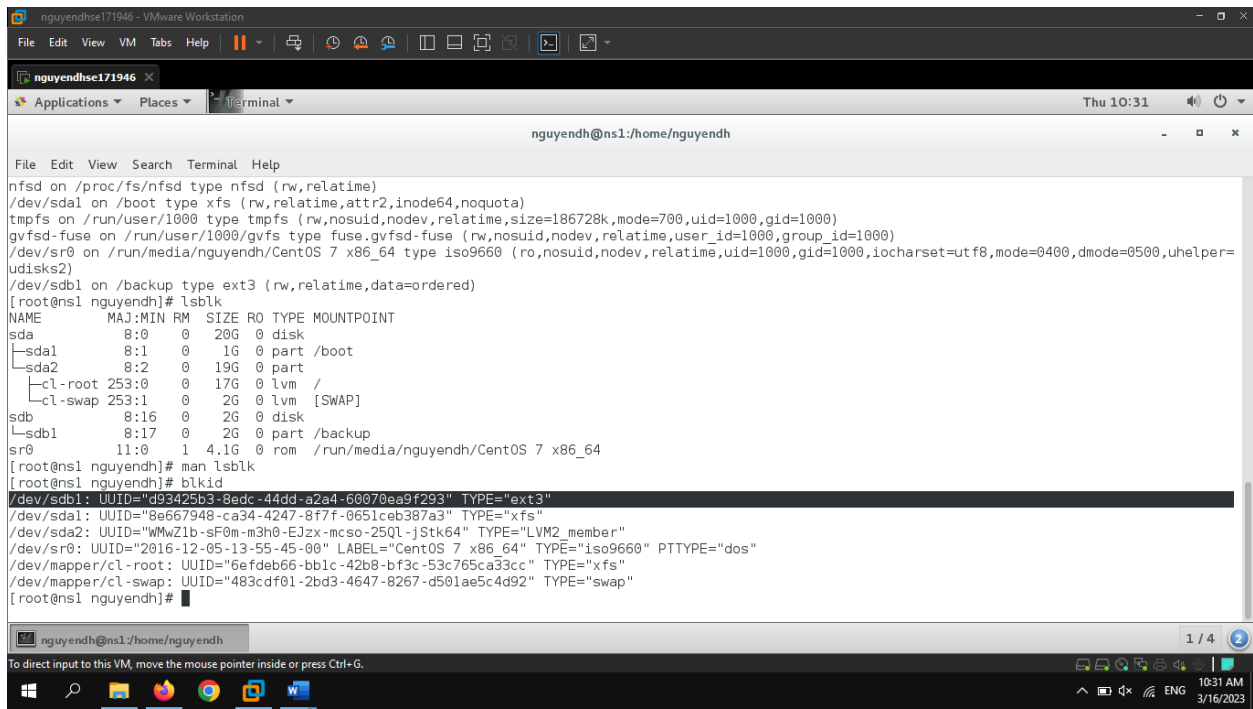


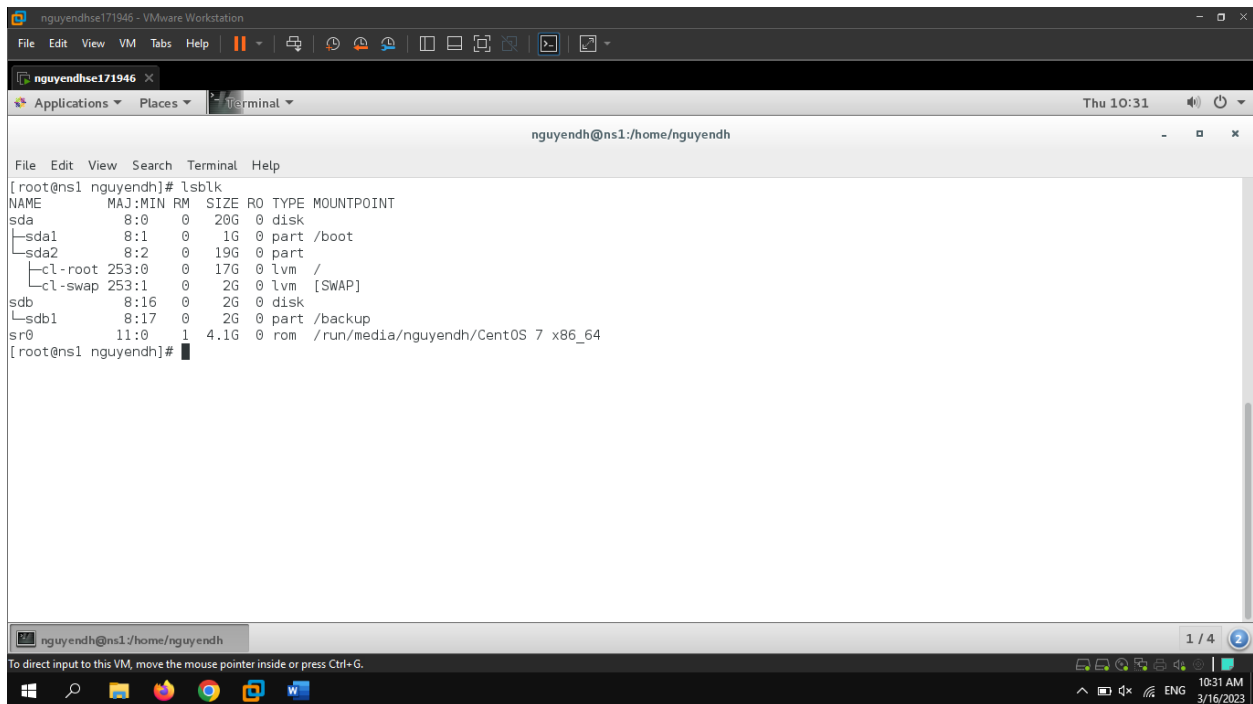
RSYNC BACKUP

Sau khi add xong ổ đĩa, ta sẽ thiết lập ổ đĩa với ext3 bằng câu lệnh **fdisk /dev/sdb** để nhận ổ đĩa, và sau đó sử dụng câu lệnh **mkfs.ext3 /dev/sdb1** để định dạng format cho ổ đĩa.



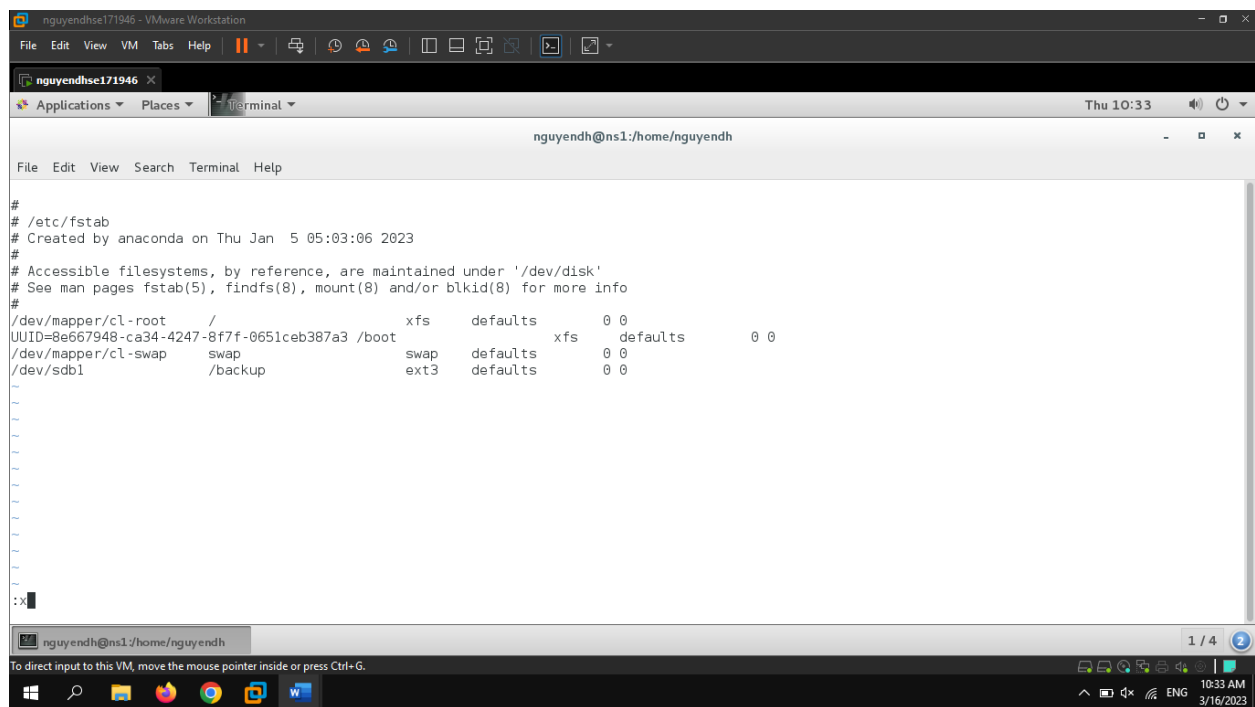
```
nguyendh@ns1: /home/nguyendh
File Edit View Search Terminal Help
nfsd on /proc/fs/nfsd type nfsd (rw,relatime)
/dev/sda1 on /boot type xfs (rw,relatime,attr2,inode64,noquota)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=186728k,mode=700,uid=1000,gid=1000)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
/dev/sr0 on /run/media/nguyendh/CentOS 7 x86_64 type iso9660 (ro,nosuid,nodev,relatime,uid=1000,gid=1000,ioccharset=utf8,mode=0400,dmode=0500,uhe1per=udisks2)
/dev/sdb1 on /backup type ext3 (rw,relatime,data=ordered)
[root@ns1 nguyendh]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda           8:0    0   20G  0 disk
├─sda1        8:1    0    1G  0 part /boot
├─sda2        8:2    0   19G  0 part
│   └─cl-root 253:0   0   17G  0 lvm  /
│       └─cl-swap 253:1   0    2G  0 lvm  [SWAP]
sdb           8:16   0    2G  0 disk
└─sdb1        8:17   0    2G  0 part /backup
sr0          11:0    1  4.1G  0 rom   /run/media/nguyendh/CentOS 7 x86_64
[root@ns1 nguyendh]# man lsblk
[root@ns1 nguyendh]# blkid
/dev/sdb1: UUID="d93425b3-8edc-44dd-a2a4-60070ea9f293" TYPE="ext3"
/dev/sda1: UUID="8e667948-ca34-4247-8f7f-0651ceb387a3" TYPE="xfs"
/dev/sda2: UUID="WMwZ1b-sF0m-m3h0-EJzx-mcso-25Q1-jStk64" TYPE="LVM2_member"
/dev/sr0: UUID="2016-12-05-13-55-45-00" LABEL="CentOS 7 x86_64" TYPE="iso9660" PTTYPE="dos"
/dev/mapper/cl-root: UUID="6efdeb66-bb1c-42b8-bf3c-53c765ca33cc" TYPE="xfs"
/dev/mapper/cl-swap: UUID="483cdf01-2bd3-4647-8267-d501ae5c4d92" TYPE="swap"
[root@ns1 nguyendh]#
```

Sau đó sử dụng câu lệnh **mkdir** để tạo ra một mountpoint cho ổ đĩa. Sử dụng câu lệnh **mount -t ext3 /dev/sdb1 /backup**



```
nguyendh@ns1: /home/nguyendh
File Edit View Search Terminal Help
[root@ns1 nguyendh]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda           8:0    0   20G  0 disk
├─sda1        8:1    0    1G  0 part /boot
├─sda2        8:2    0   19G  0 part
│   └─cl-root 253:0   0   17G  0 lvm  /
│       └─cl-swap 253:1   0    2G  0 lvm  [SWAP]
sdb           8:16   0    2G  0 disk
└─sdb1        8:17   0    2G  0 part /backup
sr0          11:0    1  4.1G  0 rom   /run/media/nguyendh/CentOS 7 x86_64
[root@ns1 nguyendh]#
```

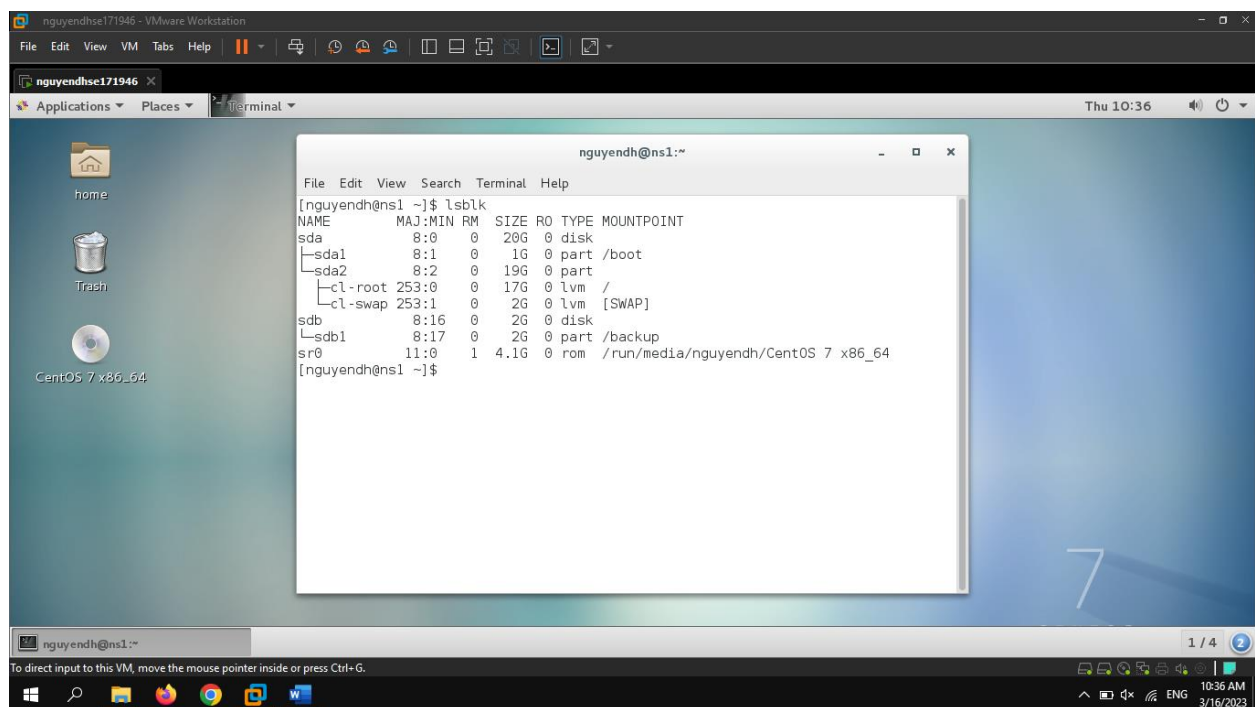
Sau đó ta vào vi /etc/fstab để có thể chỉnh lại setting lúc restart máy sẽ không bị umount.



The screenshot shows a terminal window titled 'nguyendh@ns1:/home/nguyendh'. The terminal displays the contents of the /etc/fstab file, which was created by anaconda on Thu Jan 5 05:03:06 2023. The file lists accessible filesystems and their mount options. The terminal output is as follows:

```
#  
# /etc/fstab  
# Created by anaconda on Thu Jan 5 05:03:06 2023  
# Accessible filesystems, by reference, are maintained under '/dev/disk'  
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info  
#  
/dev/mapper/cl-root / xfs defaults 0 0  
UUID=8e667948-ca34-4247-8f7f-0651ceb387a3 /boot xfs defaults 0 0  
/dev/mapper/cl-swap swap defaults 0 0  
/dev/sdb1 /backup ext3 defaults 0 0
```

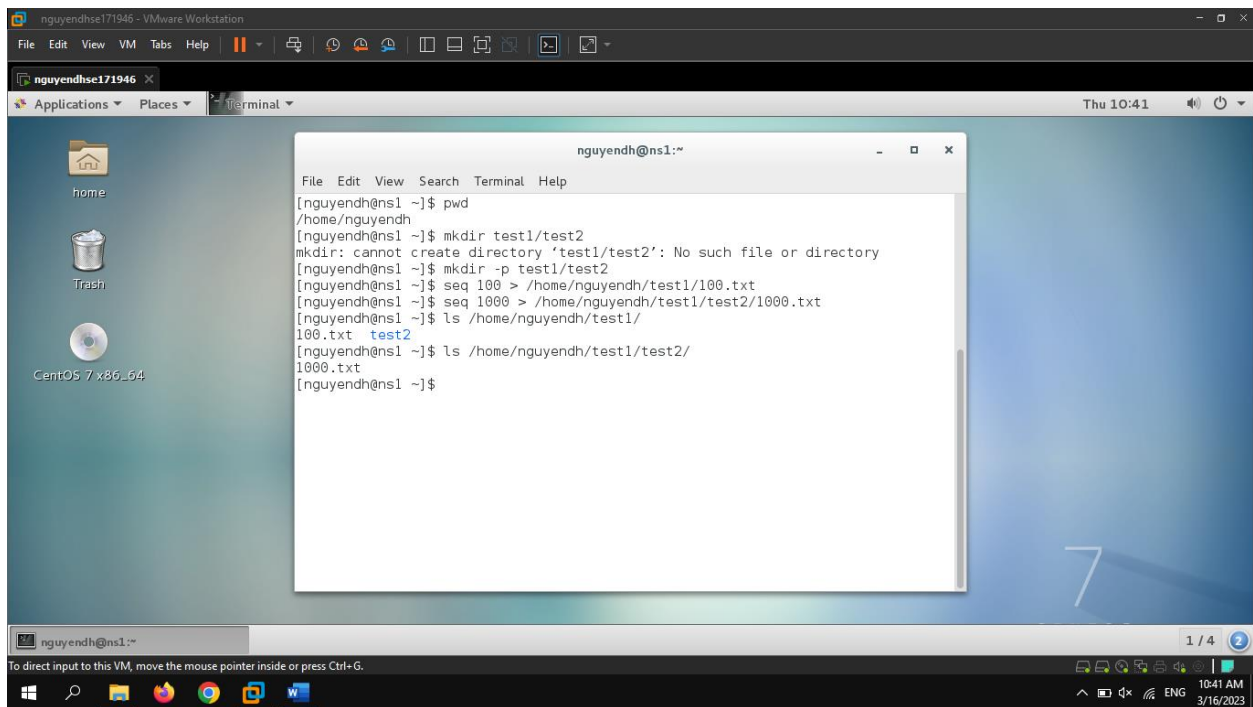
Sau khi init 6 lại thì ta thấy rằng ổ đĩa không bị umount



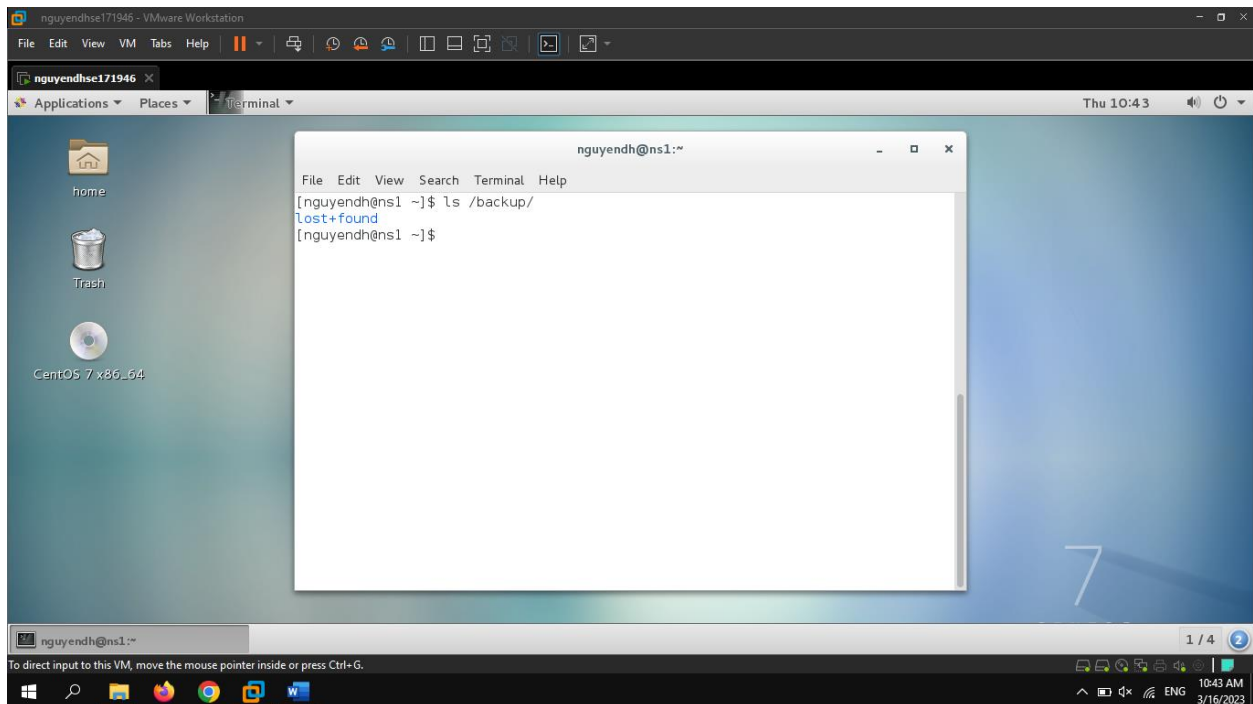
The screenshot shows a desktop environment with a terminal window titled 'nguyendh@ns1:~'. The terminal displays the output of the 'lsblk' command, which lists the block devices and their properties. The terminal output is as follows:

```
[nguyendh@ns1 ~]$ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sda          8:0    0   20G  0 disk  
├─sda1       8:1    0    1G  0 part /boot  
├─sda2       8:2    0    19G  0 part  
├─cl-root    253:0   0    17G  0 lvm /  
└─cl-swap    253:1   0     2G  0 lvm [SWAP]  
sdb          8:16   0     2G  0 disk  
└─sdb1       8:17   0     2G  0 part /backup  
sr0         11:0    1   4.1G  0 rom  /run/media/nguyendh/CentOS 7 x86_64
```

Sau đó ở thư mục của nguyendh, tạo ra thư mục con mkdir -p test1/test2. Sau đó import hai file 100.txt và 1000.txt vào bên trong lần lượt test1 và test2



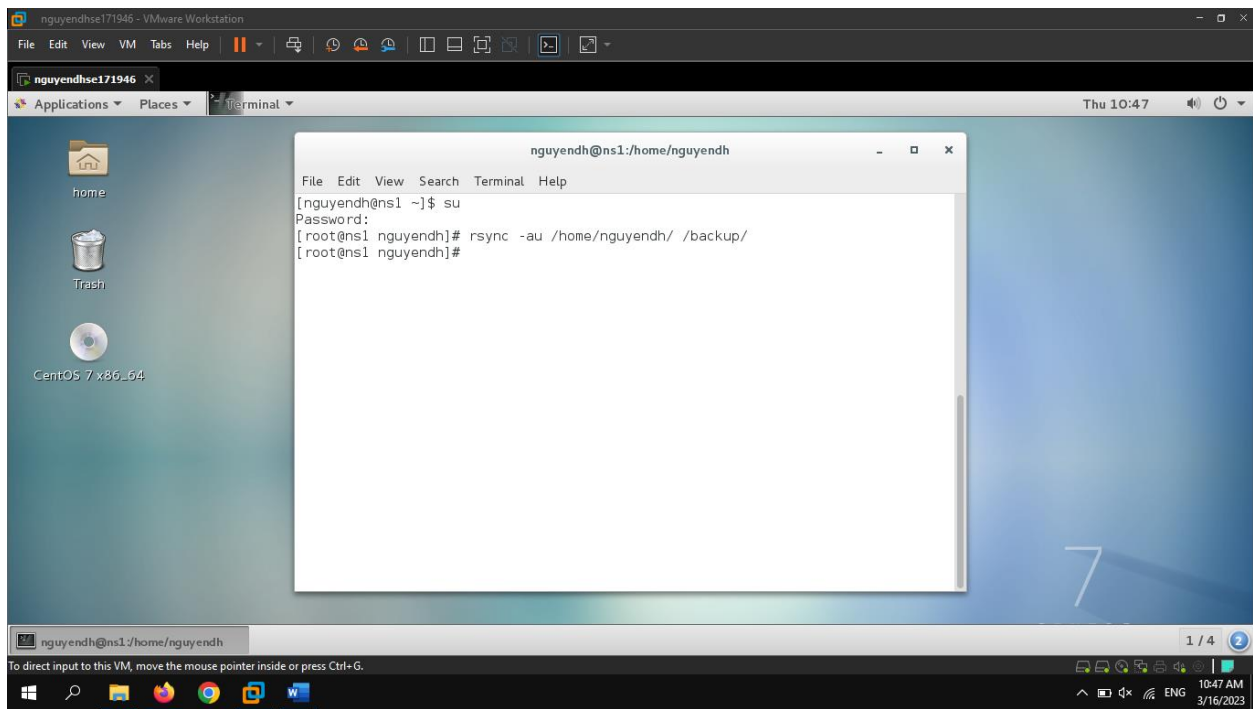
Kiểm tra lại thư mục backup thì bây giờ chưa có gì



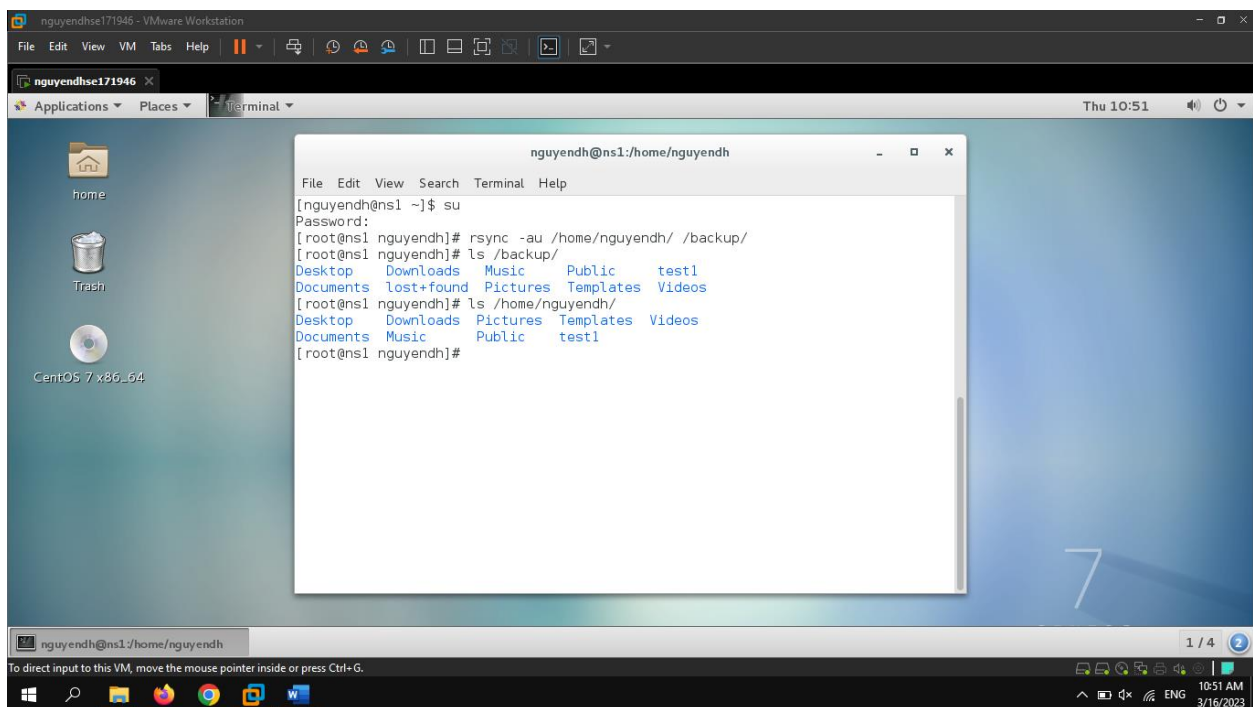
Sau đó sử dụng câu lệnh **rsync -au /home/nguyendh /backup**.

-a : nghĩa là archive. Theo em biết là nó sẽ backup hết

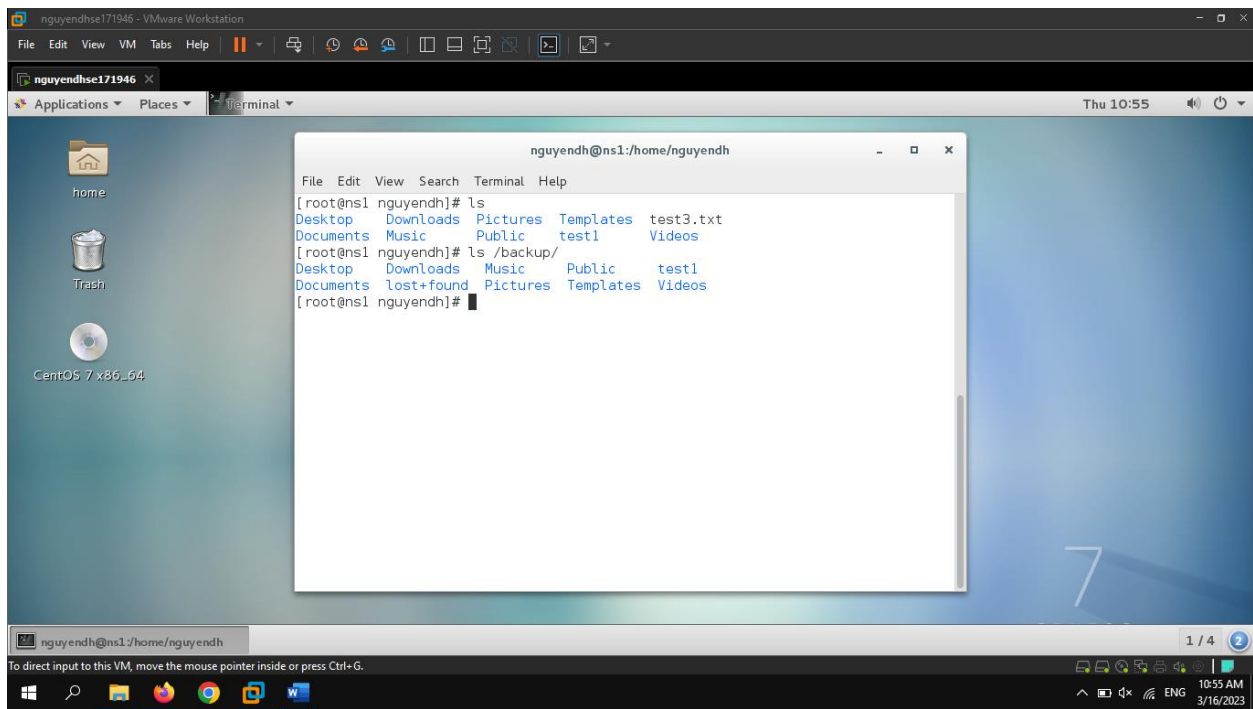
-u : update nghĩa là bỏ qua không truyền các file mà được add thêm vào sau này trong thư mục /home/nguyendh



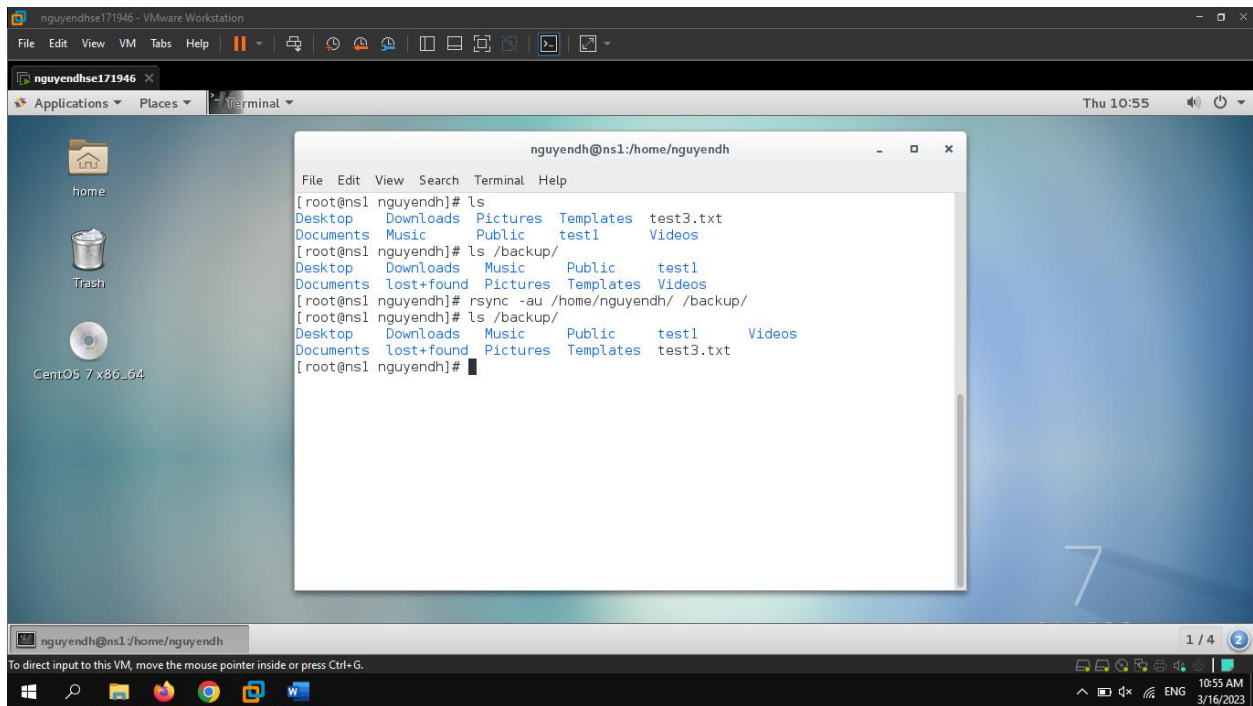
Sau đó ta dễ dàng thấy được là tất cả dữ liệu trong folder /home/nguyendh được chuyển về backup cho thư mục /backup. Sử dụng câu lệnh ls để xem



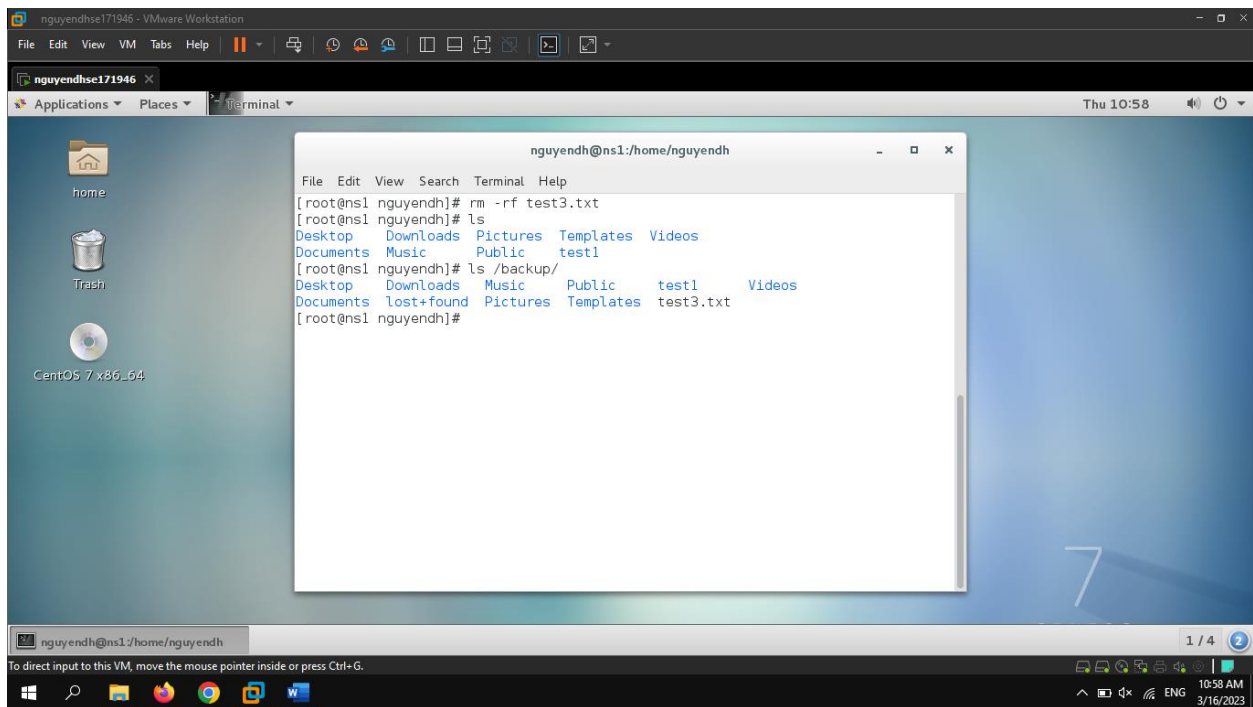
Ví dụ như chúng ta tạo thêm một file thì test3 trong /home/nguyendh thì nếu không backup thì trong folder /backup sẽ không có được file /test3



Thử đem vào backup lại thì ta lại có file text3.txt



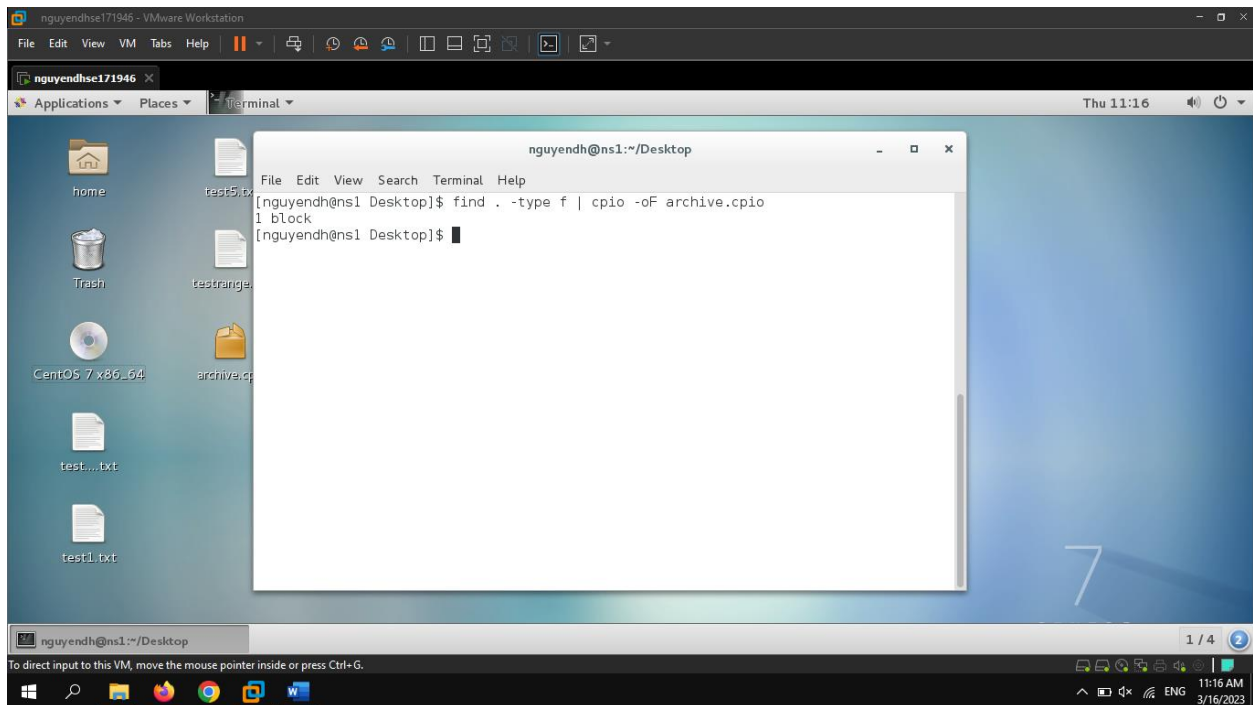
Giả sử như chúng ta muốn xóa file test3.txt tại thư mục nguyendh thì nó cũng sẽ không ảnh hưởng tới file /backup tại vì những gì chúng ta làm sau đó đối với folder nguyendh sẽ không được update trên thư mục backup.



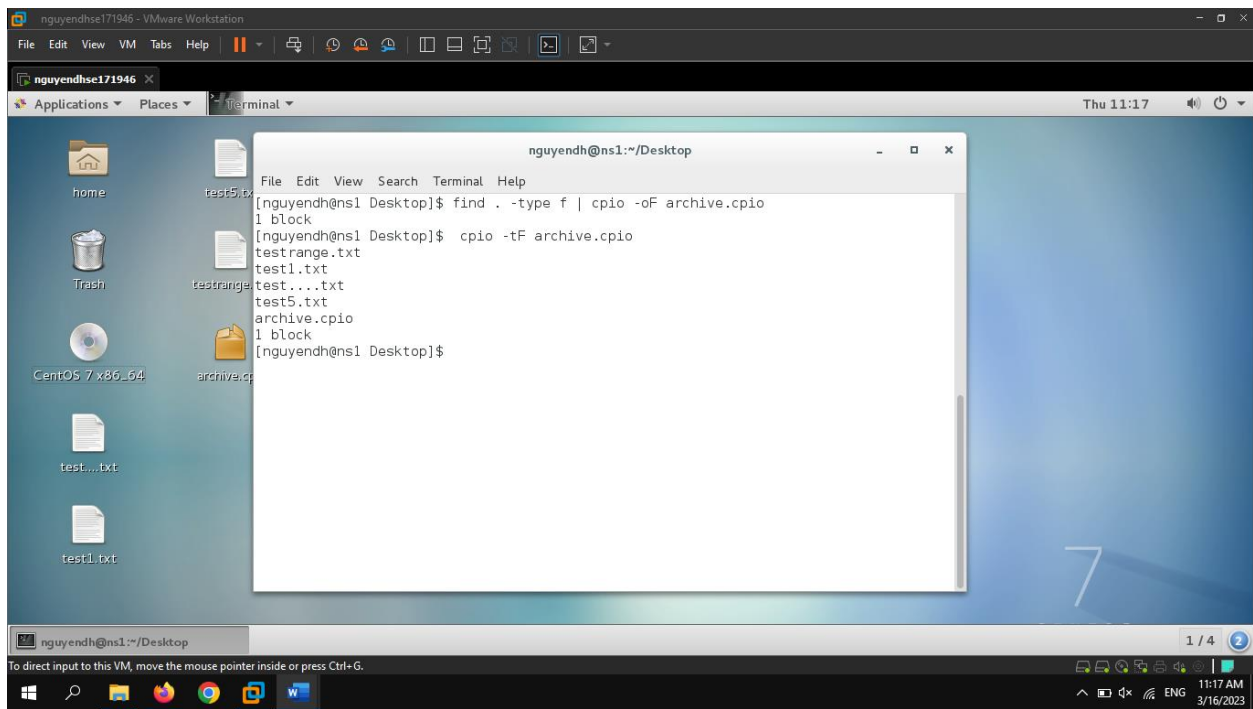
Nếu chúng ta muốn nó xóa luôn trong backup, ta có thể sử dụng lệnh rsync thêm một lần nữa hoặc có thể sử dụng rsync -auv --delete /home/nguyendh /backup

CPIO

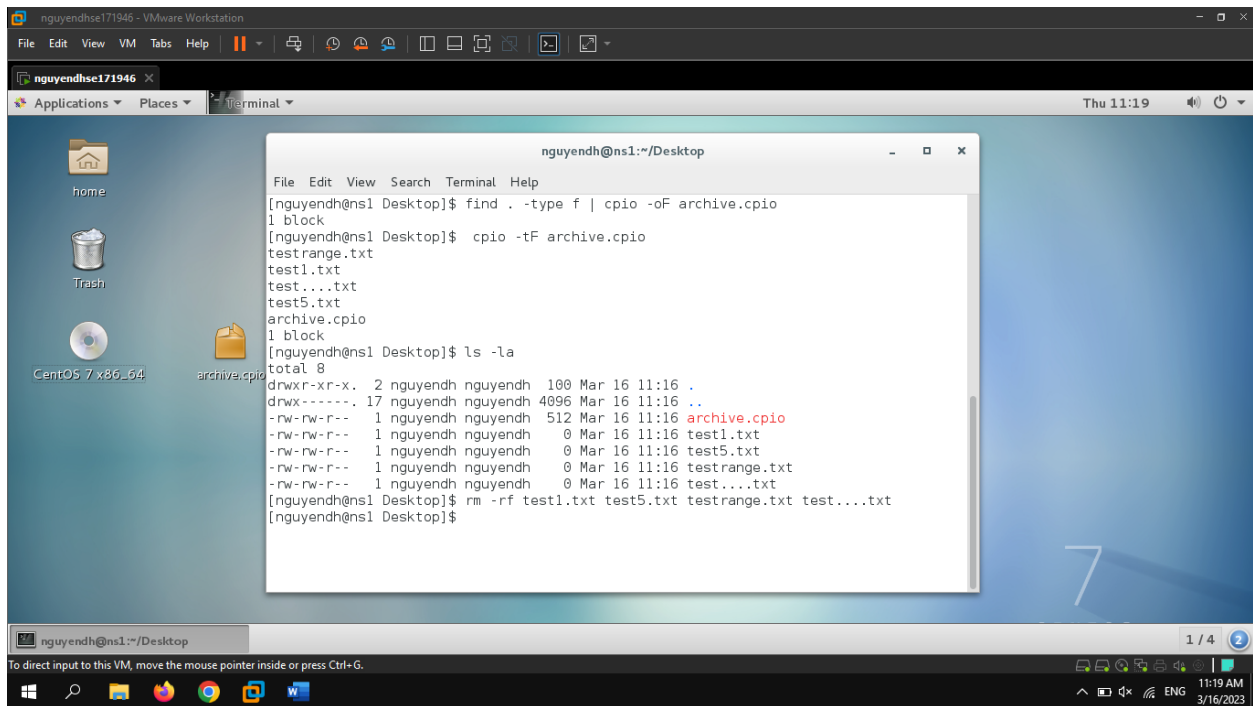
Sử dụng câu lệnh `find . -type f | cpio -oF archive.cpio` để tạo ra file backup cho Desktop



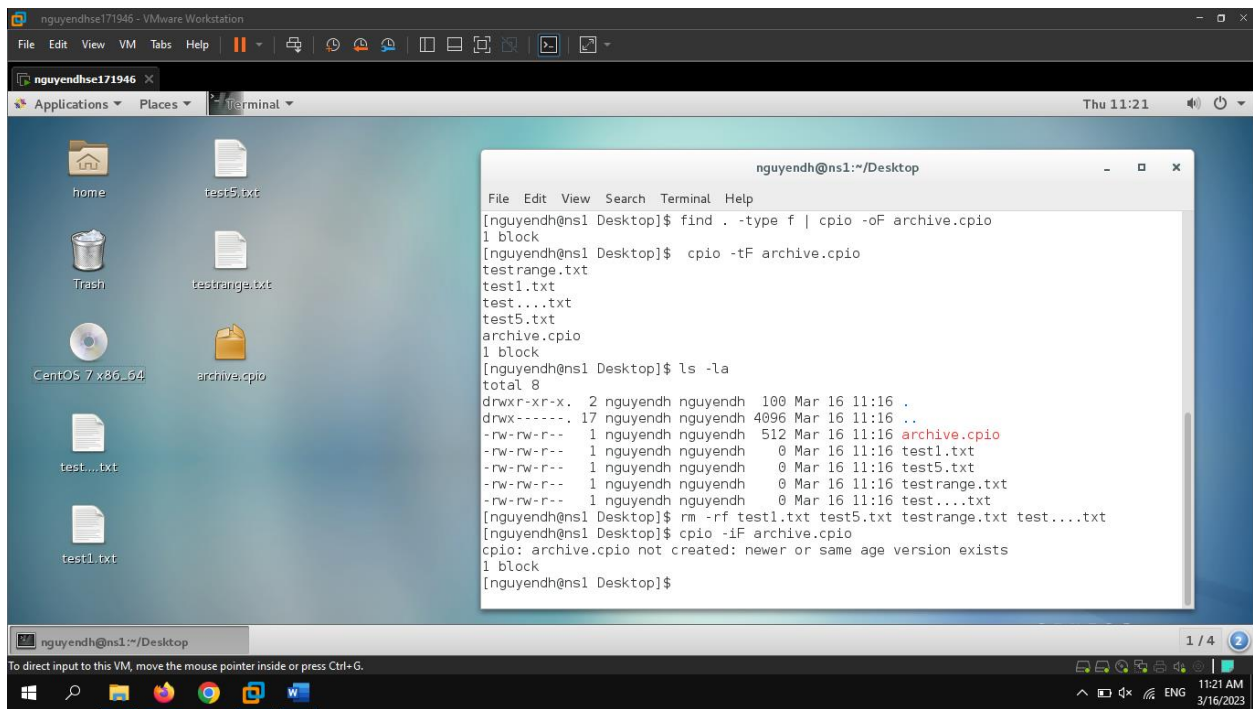
Sau đó sử dụng câu lệnh `cpio -tF archive.cpio` để xem trong archive đã có những gì



Xóa đi các file vừa mới tạo bằng `rm -rf`

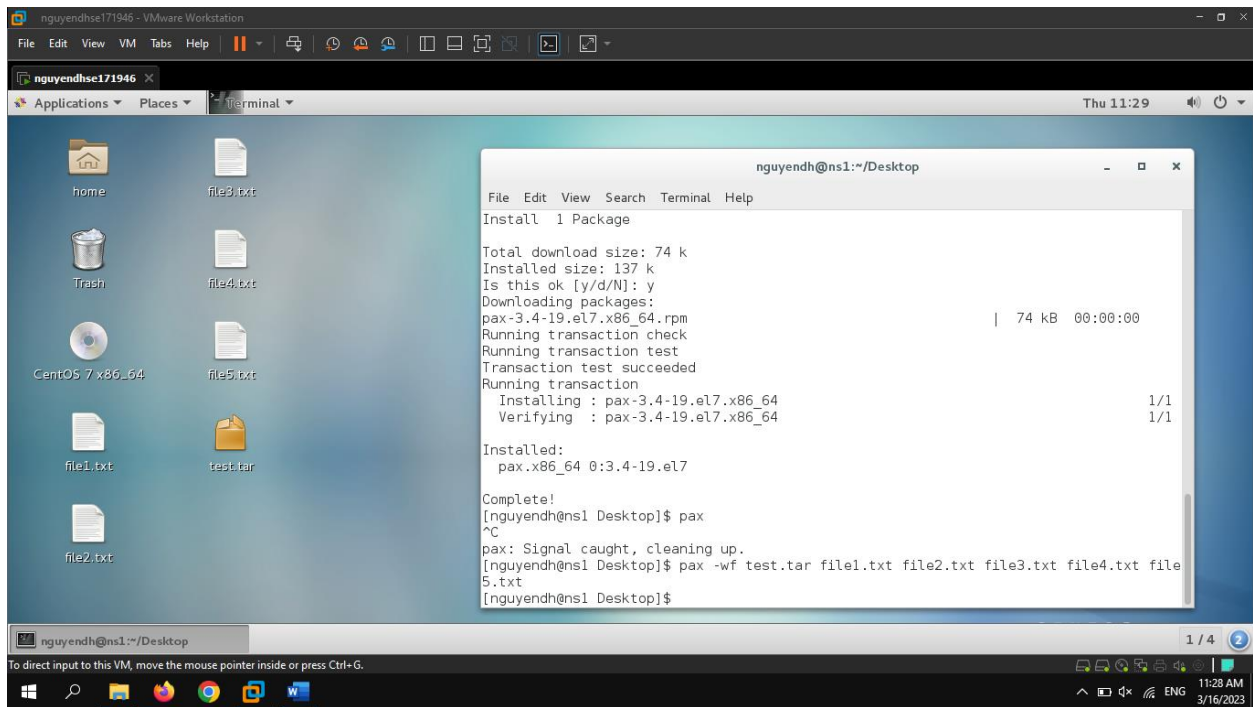


Sau đó chúng ta sử dụng câu lệnh `cpio -iF` để có thể khôi phục lại những gì mà chúng ta đã delete

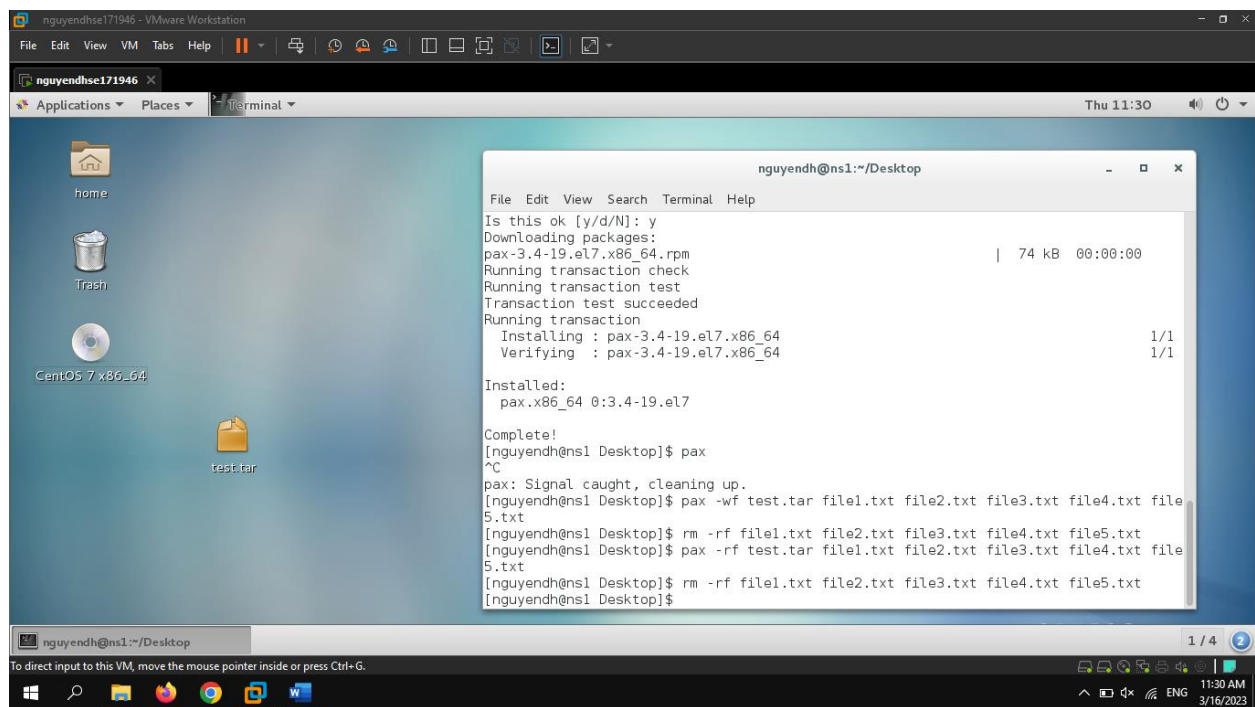


PAX

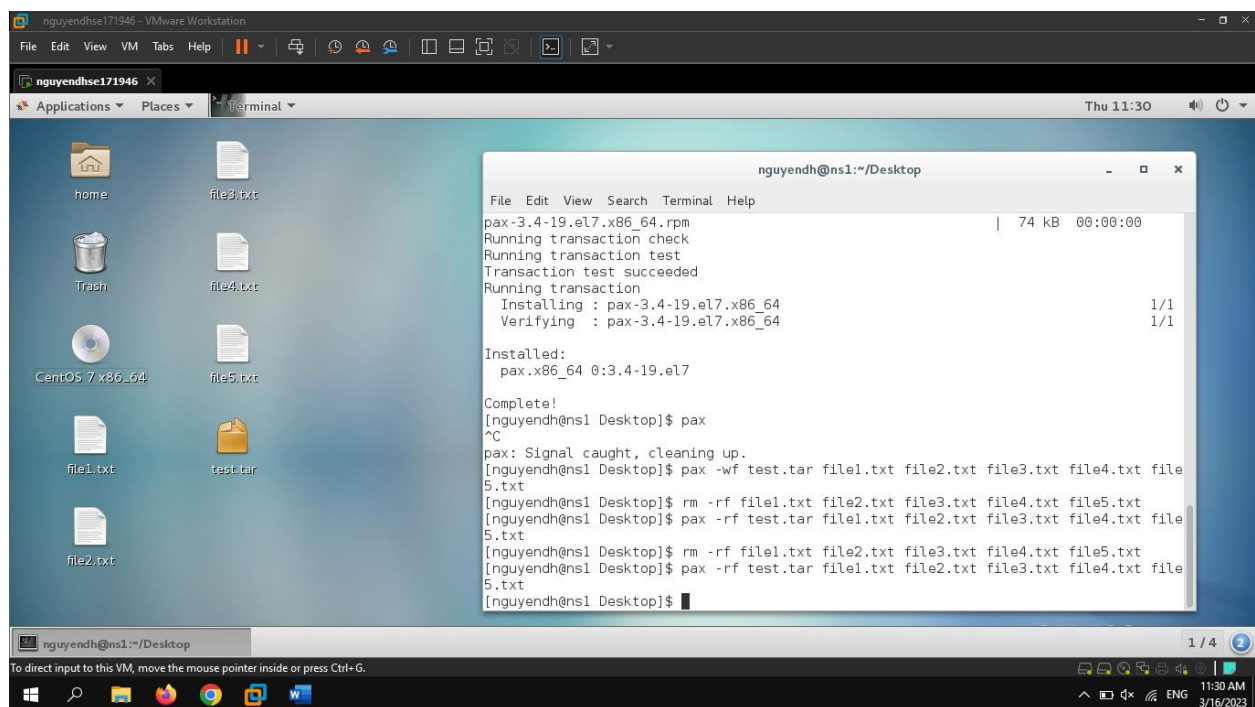
Tạo ra một cái file backup bằng pax thông qua câu lệnh pax -wf



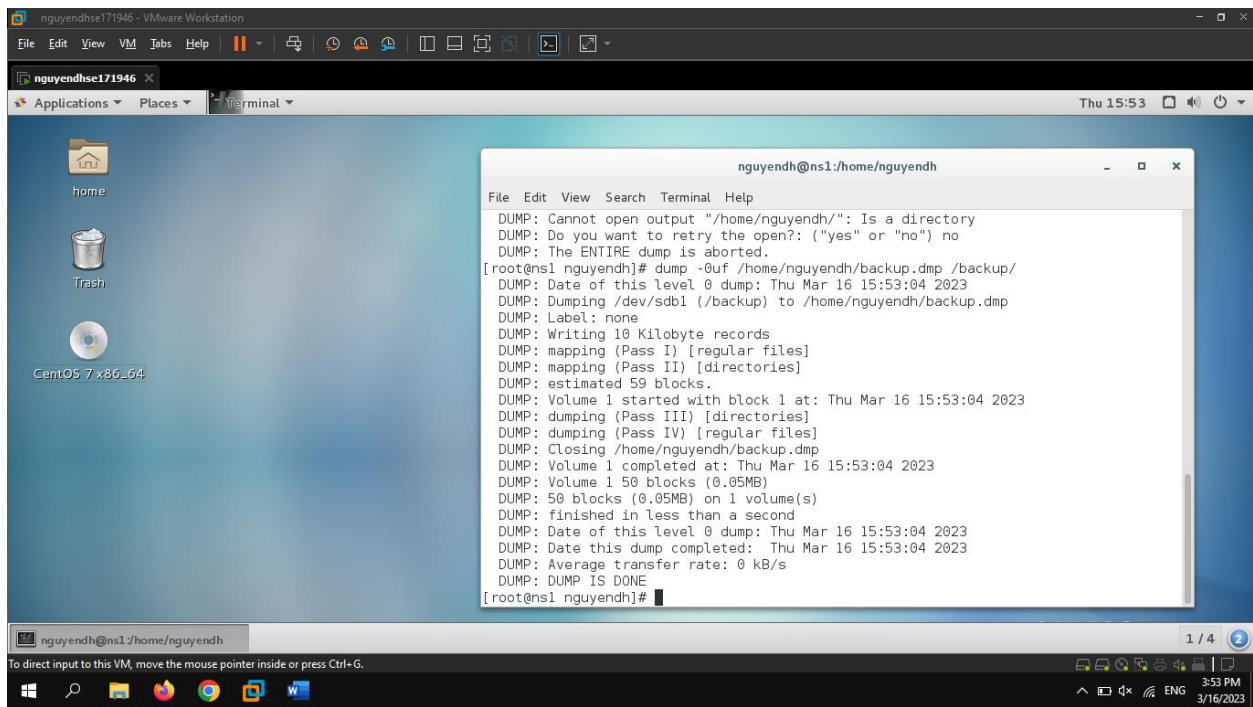
Sử dụng câu lệnh `rm -rf` để xóa hết tất cả các file để test



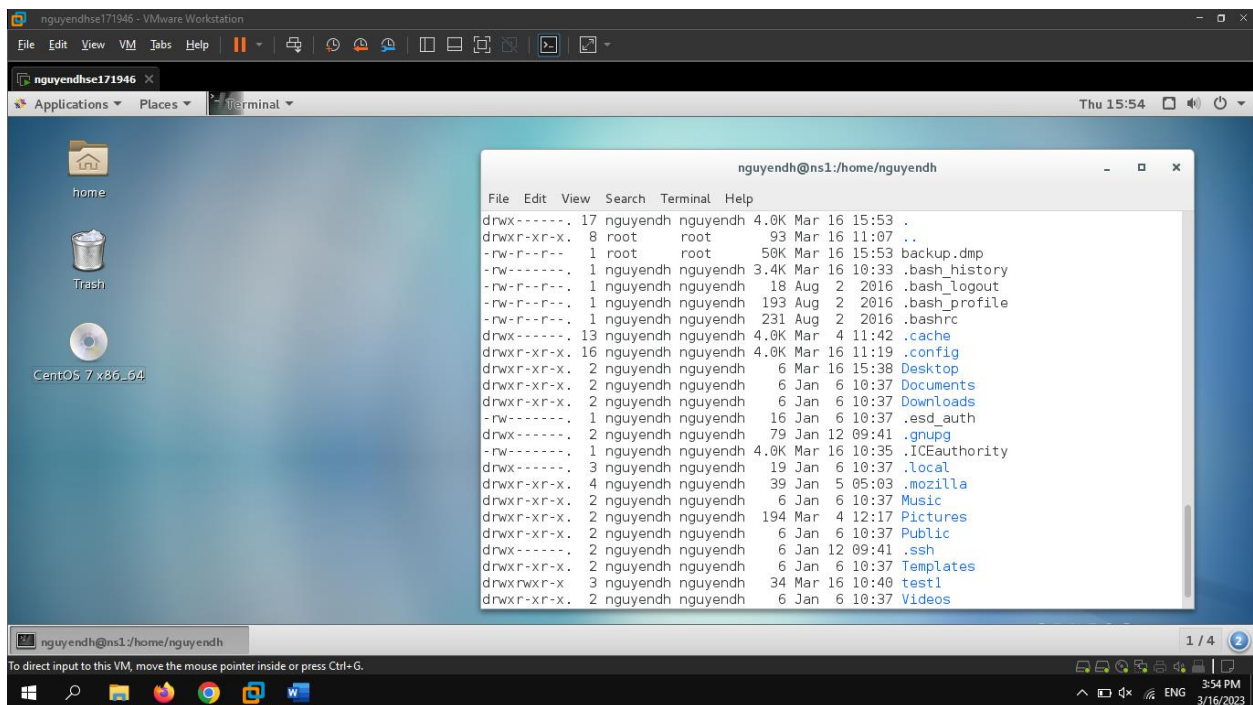
Sau đó sử dụng câu lệnh `pax -rf` để có thể restore lại các file có trong test.tar



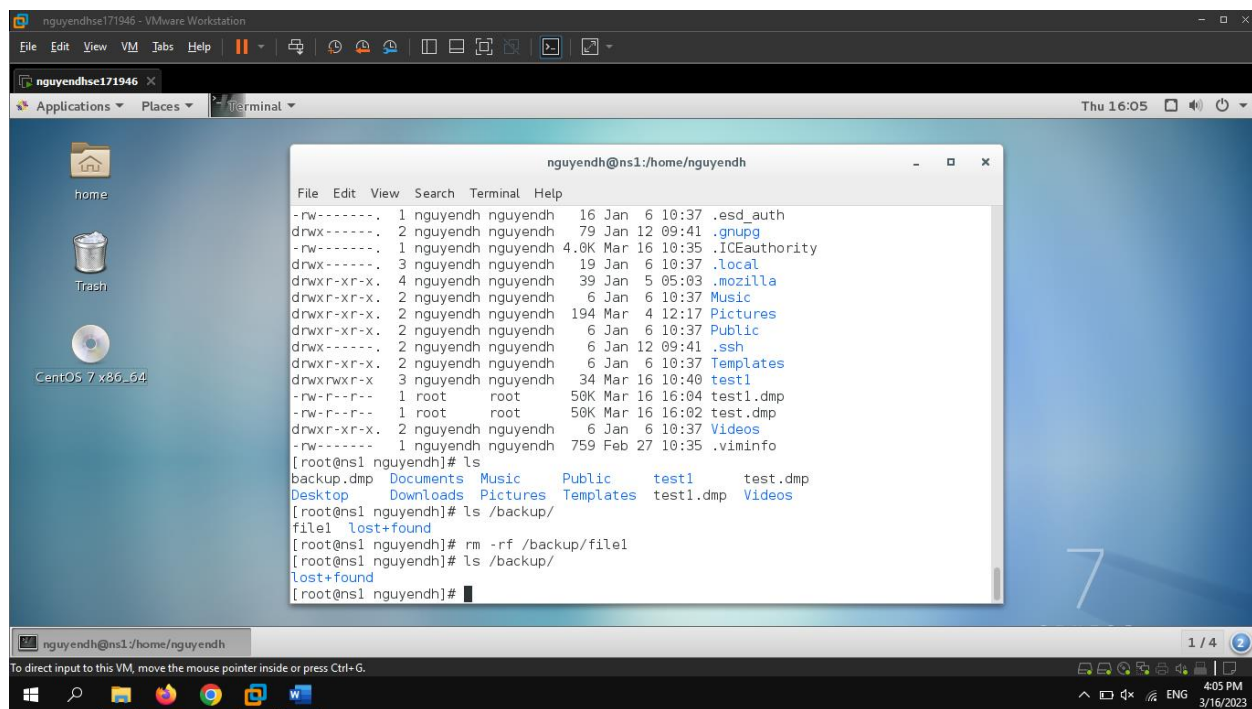
Sử dụng câu lệnh `dump -0uf` để có thể dump file



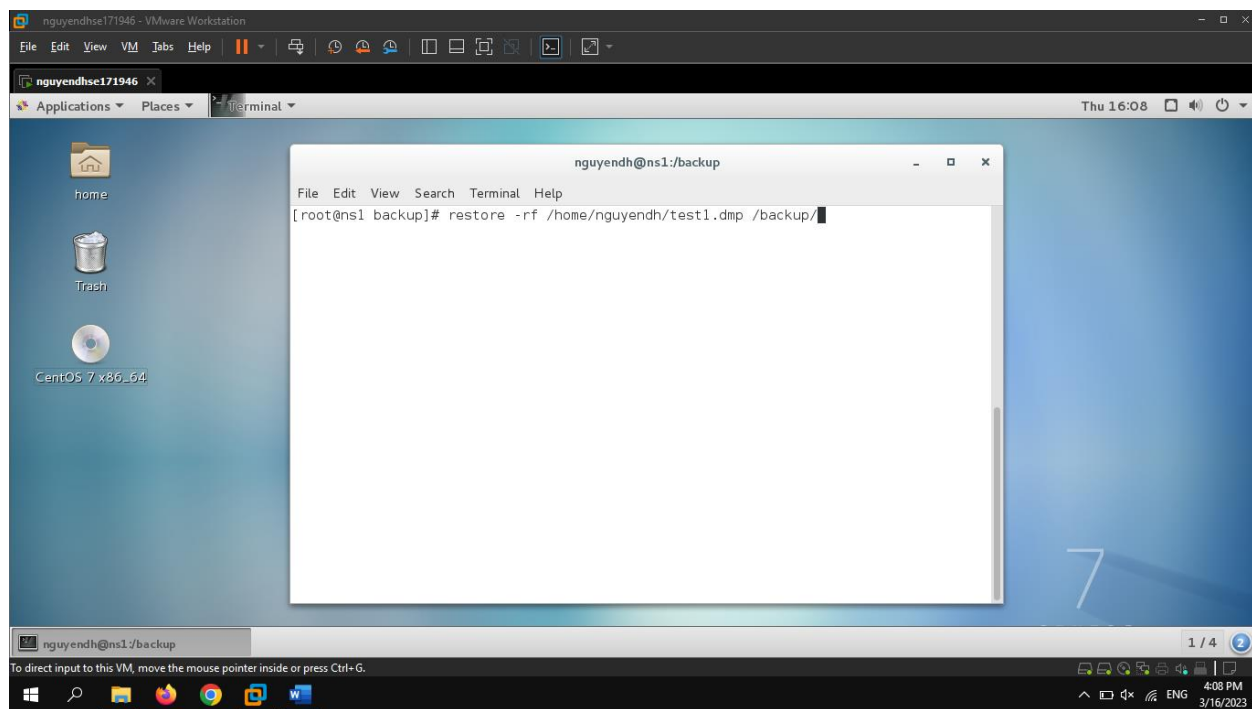
Kiểm tra bằng lệnh `ls -lah` ta có thể thấy được file dump



Theo hình ta sẽ thấy rằng ở đây có file1 trong backup, bây giờ ta sẽ tiến hành xóa nó đi



Sau đó sử dụng câu lệnh **restore** file dump test1.dmp, ta sẽ restore lại được file gốc ban đầu



Sau khi thực hiện câu lệnh, vào backup thì ta thấy file đã trở về ban đầu

