

山东大学 计算机科学与技术 学院

云计算技术 课程实验报告

学号：202200130048	姓名：陈静雯	班级：6																		
实验题目：虚拟化技术练习一 VMware																				
实验学时：2	实验日期：2025. 3. 19																			
<p>实验目的：通过本实验，学生将学会搭建和熟悉 VMware 虚拟化技术，了解其基本概念和操作流程，提升对虚拟化技术的理解和应用能力。</p> <p>具体包括：了解 VMware 虚拟化环境的配置和部署，并配置网络连接，使其能够与本地计算机互相通信；在虚拟机中创建一个共享文件夹，使得本地计算机能够访问该共享文件夹；实现快照的创建、恢复以及虚拟机的克隆，并撰写实验报告。</p>																				
<p>硬件环境：</p> <p>计算机一台</p>																				
<p>软件环境：</p> <p>Windows、Linux、MacOS</p>																				
<p>实验步骤与内容：</p> <p>1. 了解 VMware 虚拟化环境的配置和部署，并配置网络连接，使其能够与本地计算机互相通信</p> <p>（1）在虚拟机设置中，选择网络适配器类型（如桥接、NAT、仅主机等）</p> <p>NAT 网络模式：宿主机可以看做一个路由器，虚拟机通过宿主机的网络来访问 Internet</p> <div><div><div>硬件选项</div><table><thead><tr><th>设备</th><th>摘要</th></tr></thead><tbody><tr><td>内存</td><td>4 GB</td></tr><tr><td>处理器</td><td>4</td></tr><tr><td>硬盘 (SCSI)</td><td>20 GB</td></tr><tr><td>CD/DVD (SATA)</td><td>正在使用文件 D:\ubuntu\ubu...</td></tr><tr><td>网络适配器</td><td>NAT</td></tr><tr><td>USB 控制器</td><td>存在</td></tr><tr><td>声卡</td><td>自动检测</td></tr><tr><td>显示</td><td>自动检测</td></tr></tbody></table></div><div><div>设备状态</div><div><input type="checkbox"/> 已连接(C) <input checked="" type="checkbox"/> 启动时连接(O)</div><div>网络连接<div><input type="radio"/> 桥接模式(B): 直接连接物理网络 <input type="checkbox"/> 复制物理网络连接状态(P) <input checked="" type="radio"/> NAT 模式(N): 用于共享主机的 IP 地址 <input type="radio"/> 仅主机模式(H): 与主机共享的专用网络 <input type="radio"/> 自定义(U): 特定虚拟网络 VMnet1 (仅主机模式) <input type="radio"/> LAN 区段(L): <div></div></div><div>LAN 区段(S)...高级(V)...</div></div></div></div>			设备	摘要	内存	4 GB	处理器	4	硬盘 (SCSI)	20 GB	CD/DVD (SATA)	正在使用文件 D:\ubuntu\ubu...	网络适配器	NAT	USB 控制器	存在	声卡	自动检测	显示	自动检测
设备	摘要																			
内存	4 GB																			
处理器	4																			
硬盘 (SCSI)	20 GB																			
CD/DVD (SATA)	正在使用文件 D:\ubuntu\ubu...																			
网络适配器	NAT																			
USB 控制器	存在																			
声卡	自动检测																			
显示	自动检测																			

## (2) 查看虚拟机 ip 地址

```
orange@orange-VMware-Virtual-Platform:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.156.128 netmask 255.255.255.0 broadcast 192.168.156.255
    inet6 fe80::5183:9c02:45ad:27e prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:ae:20:1d txqueuelen 1000 (Ethernet)
    RX packets 112 bytes 14782 (14.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 89 bytes 10215 (10.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 123 bytes 11567 (11.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 123 bytes 11567 (11.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

## (3) 虚拟机 ping 主机

```
orange@orange-VMware-Virtual-Platform:~$ ping 192.168.254.245
PING 192.168.254.245 (192.168.254.245) 56(84) bytes of data:
64 bytes from 192.168.254.245: icmp_seq=1 ttl=128 time=2.00 ms
64 bytes from 192.168.254.245: icmp_seq=2 ttl=128 time=2.76 ms
64 bytes from 192.168.254.245: icmp_seq=3 ttl=128 time=1.57 ms
64 bytes from 192.168.254.245: icmp_seq=4 ttl=128 time=1.53 ms
64 bytes from 192.168.254.245: icmp_seq=5 ttl=128 time=2.56 ms
64 bytes from 192.168.254.245: icmp_seq=6 ttl=128 time=2.11 ms
64 bytes from 192.168.254.245: icmp_seq=7 ttl=128 time=1.60 ms
64 bytes from 192.168.254.245: icmp_seq=8 ttl=128 time=4.05 ms
64 bytes from 192.168.254.245: icmp_seq=9 ttl=128 time=2.81 ms
64 bytes from 192.168.254.245: icmp_seq=10 ttl=128 time=2.61 ms
64 bytes from 192.168.254.245: icmp_seq=11 ttl=128 time=2.89 ms
64 bytes from 192.168.254.245: icmp_seq=12 ttl=128 time=1.81 ms
64 bytes from 192.168.254.245: icmp_seq=13 ttl=128 time=3.58 ms
```

## (4) 主机 ping 虚拟机

```
C:\Users\c' j' w>ping 192.168.156.128

正在 Ping 192.168.156.128 具有 32 字节的数据:
来自 192.168.156.128 的回复: 字节=32 时间=1ms TTL=64
来自 192.168.156.128 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.156.128 的回复: 字节=32 时间=1ms TTL=64
来自 192.168.156.128 的回复: 字节=32 时间<1ms TTL=64

192.168.156.128 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms, 最长 = 1ms, 平均 = 0ms
```

## 2. 在虚拟机中创建一个共享文件夹，使得本地计算机能够访问该共享文件夹

### (1) 在虚拟机设置中添加共享文件夹



### (2) 在终端中设置虚拟机

```
quiet@quiet-linux:~$ vmware-hgfsclient
czsystem
```

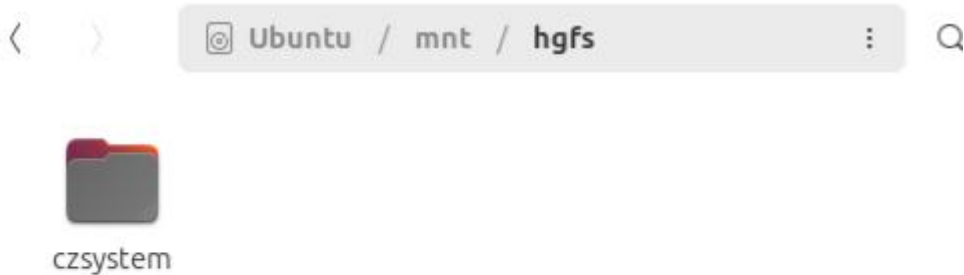
### (3) 在挂载目录里创建要挂载到的文件夹

```
quiet@quiet-linux:~$ sudo mkdir /mnt/hgfs
[sudo] quiet 的密码:
```

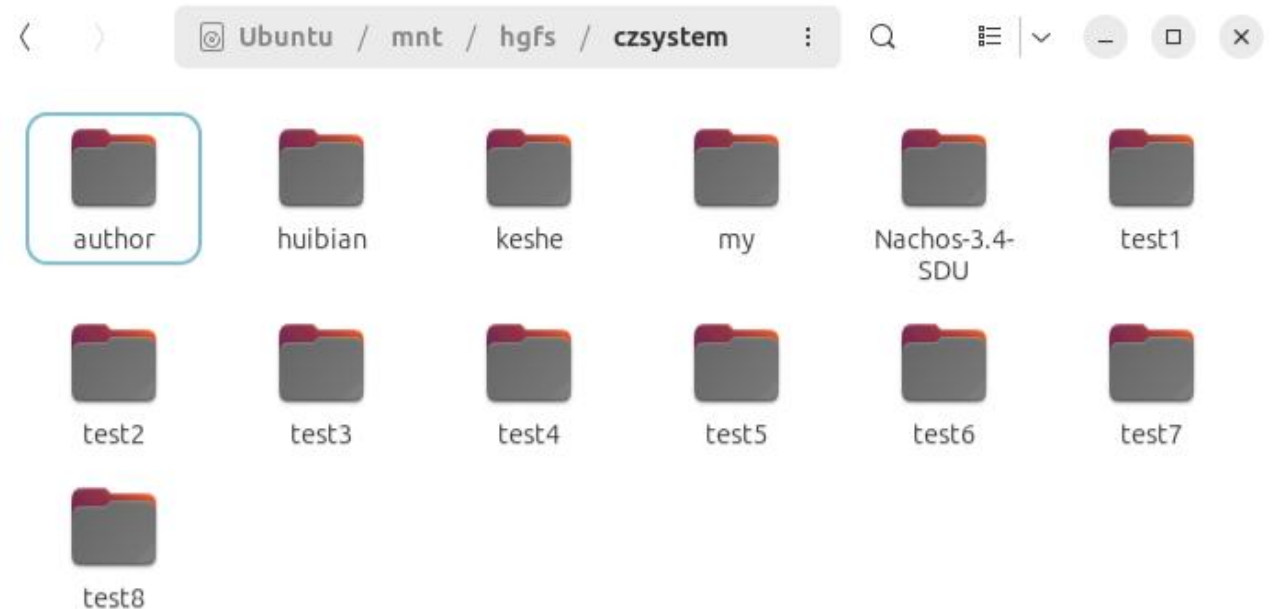
### (4) 挂载共享目录

```
quiet@quiet-linux:~$ sudo /usr/bin/vmhgfs-fuse .host:/ /mnt/hgfs -o allow_other  
-o uid=1000 -o gid=1000 -o umask=022
```

(5) 挂载成功



(6) 本地可以往里添加文件



(7) 设置开机自动挂载

在终端输入 `sudo vim /etc/fstab`，用 `vim` 打开该文件，添加

`.host:/ /mnt/hgfs fuse.vmhgfs-fuse allow_other,uid=1000,gid=1000,umask=022 0 0`



```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>          <dump>  <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/ad4e5452-db06-462d-8e11-23b5686c8efc /  ext4 defaults 0 1
/swap.img          none      swap     sw                0        0
.host:/ /mnt/hgfs fuse.vmhgfs-fuse allow_other,uid=1000,gid=1000,umask=022 0 0

-- INSERT --
```

### 3. 实现快照的创建、恢复以及虚拟机的克隆

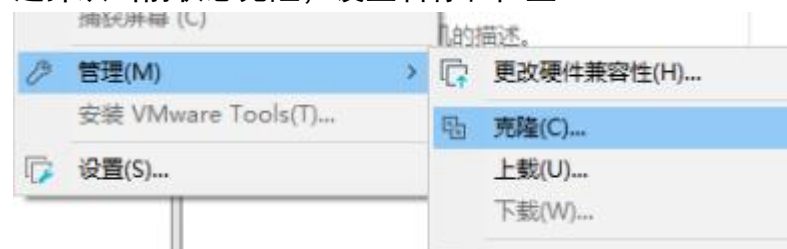
### (1) 拍摄快照

(2) 在虚拟机设置中，快照选项：选择恢复到快照



### (3) 虚拟机克隆

选择从当前状态克隆，设置名称和位置



### 结论分析与体会：

- 1、配置和部署虚拟化环境：在本次实验中，我学会了如何搭建和配置 VMware 虚拟化环境。这包括了安装和设置虚拟机软件，分配计算资源以及配置网络连接等。通过这一步骤，我对虚拟化技术的基本原理有了更深入的理解。
- 2、网络连接配置：在配置网络连接的过程中，我学会了如何使虚拟机与本地计算机互相通信。这一步骤对于实现虚拟化环境的功能性至关重要，因为它使得虚拟机可以像真实计算机一样在网络中进行通信。
- 3、共享文件夹设置：通过在虚拟机中创建共享文件夹，能够轻松地实现本地计算机与虚拟机之间的文件共享。这对于在虚拟化环境中进行文件传输和共享是非常方便的，也提高了工作效率。
- 4、快照的创建和恢复：实验中，我学会了如何创建虚拟机的快照，并能够随时恢复到先前的状态。这在软件开发、测试和系统维护等方面都具有重要意义，可以有效地保护数据和系统的完整性。
- 5、虚拟机克隆：通过实现虚拟机的克隆，我可以快速地部署多个相似的虚拟机实例，节省了大量的时间和资源。这对于进行实验测试、软件开发和构建复杂系统环境都具有重要意义。