Libguestfs workshop - 部分小工具的部分用法

我们知道，libguestfs 不仅提供了 c binding 和 python binding 便于人们开发自己的虚拟化应用，类似 libvirt，它也提供若干交互式和非并互式的工具给用户直接使用，以解决用户在虚拟化操作中产生的某些诉求或问题。
这里，我们只是聊聊 libguestfs 自带的非交互式工具中的一部分，不涉及它提供的 guestfish 交互式环境，也不涉及它的 API 用法。在使用这些工具操作虚拟机镜像时，最好不要针对 online 的虚拟机或有并发访问。

例 1：virt-df 可查看虚拟机磁盘的空间情况：

```
suse:~/projects/libguestfs$ ./run virt-df -d sles12sp3 -h
Filesystem                Size       Used  Available  Use%
sles12sp3:/dev/sda2         13G       7.1G       4.9G   57%
sles12sp3:/dev/sdb1        7.7G       1.0G       6.3G   14%
sles12sp3:/dev/sdc1         10G       2.0G       7.0G   21%
```

例 2：virt-filesystems 可列出虚拟机或磁盘镜像中的文件系统，分区，块设备，LVM 等信息

```
suse:~/projects/libguestfs$ ./run virt-filesystems -d sles12sp3 --long --human-readable --uuid
Name       Type       VFS   Label Size Parent UUID
/dev/sda2 filesystem ext4  -     13G  -      3dc1363d-b3d9-42f1-a2ff-f70eee8b00a0
/dev/sdb1 filesystem ext4  -     8.0G -      aba670a1-c9ed-48db-be81-67936eadf5b4
/dev/sdc1 filesystem btrfs -     10G  -      aa65ae0b-d443-4226-bf4f-3cd1abd198f3
```

例 3： virt-ls 查看 linux 和 windows 虚拟机磁盘中的内容：

```
suse:~/projects/libguestfs$ virsh domblklist sles12sp3
Target     Source
------------------------------------------------
vda        /opt/vms/sles12-sp3/disk0.qcow2
vdb        /opt/vms/sles12-sp3/disk1.qcow2
vdc        /opt/vms/sles12-sp3/disk2.qcow2

suse:~/projects/libguestfs$ ./run virt-ls -d sles12sp3 /home
suse

suse:~/projects/libguestfs$ virsh domblklist win10
Target     Source
------------------------------------------------
hda        /opt/vms/win10/disk0.qcow2

suse:~/projects/libguestfs$ ./run virt-ls -d win10 /Users
All Users
Default
Default User
Lin
Public
desktop.ini
```

例 4：当虚拟机中含多虚拟磁盘，用户想查看的磁盘是非'根'磁盘，没有记录在'根'分区的/etc/fstab 中，且磁盘中只有数据，没有 os, 比如用户想查看 sdb1 的内容：

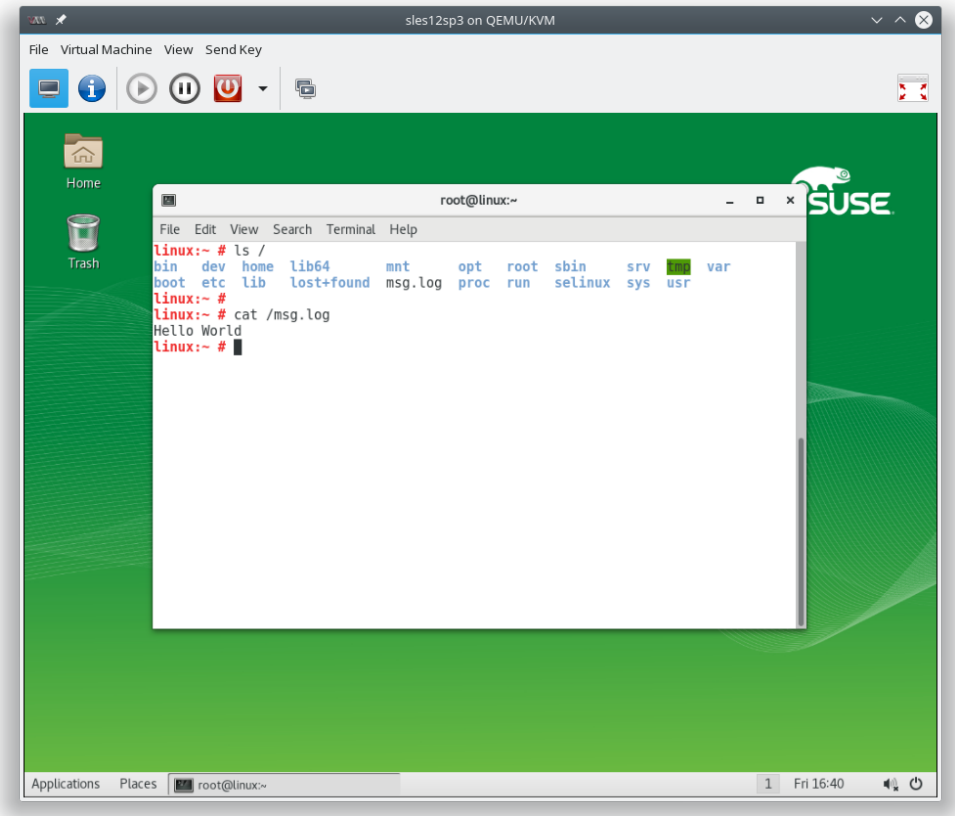
```
suse:~/projects/libguestfs$ virsh domblklist sles12sp3
Target     Source
------------------------------------------------
vda        /opt/vms/sles12-sp3/disk0.qcow2
vdb        /opt/vms/sles12-sp3/disk1.qcow2
vdc        /opt/vms/sles12-sp3/disk2.qcow2
```

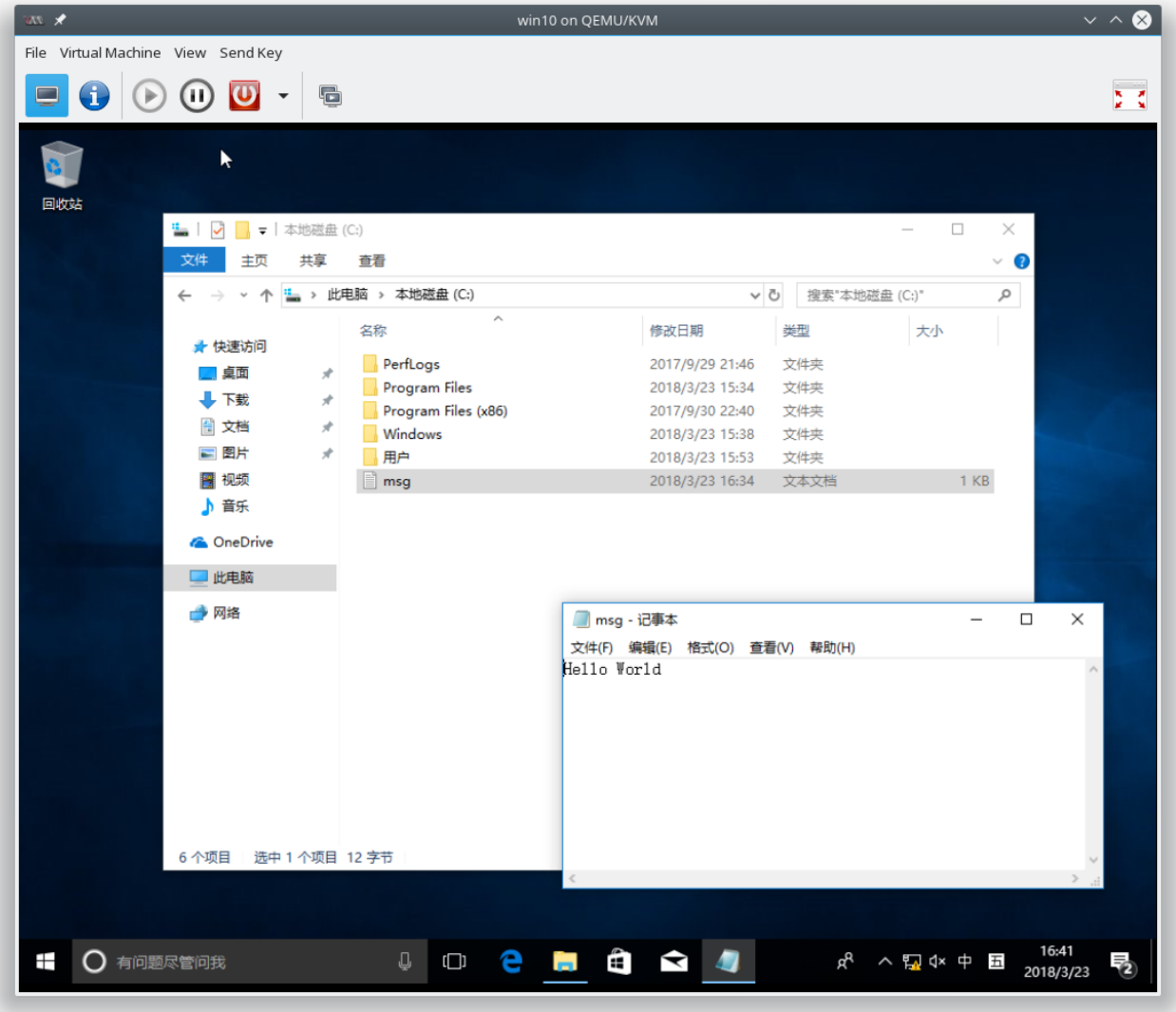
suse:~/projects/libguestfs$ ./run **virt-ls** -d sles12sp3 -m /dev/sdb1:/:acl,user_xattr /

注意，变更 mountpoint 和 fstype 我感觉好像有 bugs, 导致这两个选项是不可用的, 我还没有看细节。
也就是说，不可以变更 mountpoint,也不能指定文件系统类型，后果就是 btrfs 格式的非'根'虚拟磁盘，无法被 virt-ls.
比如本例中，就无法查看/dev/sdc1 的内容。

例 5：**virt-copy-in**, **virt-copy-out**, **virt-cat**

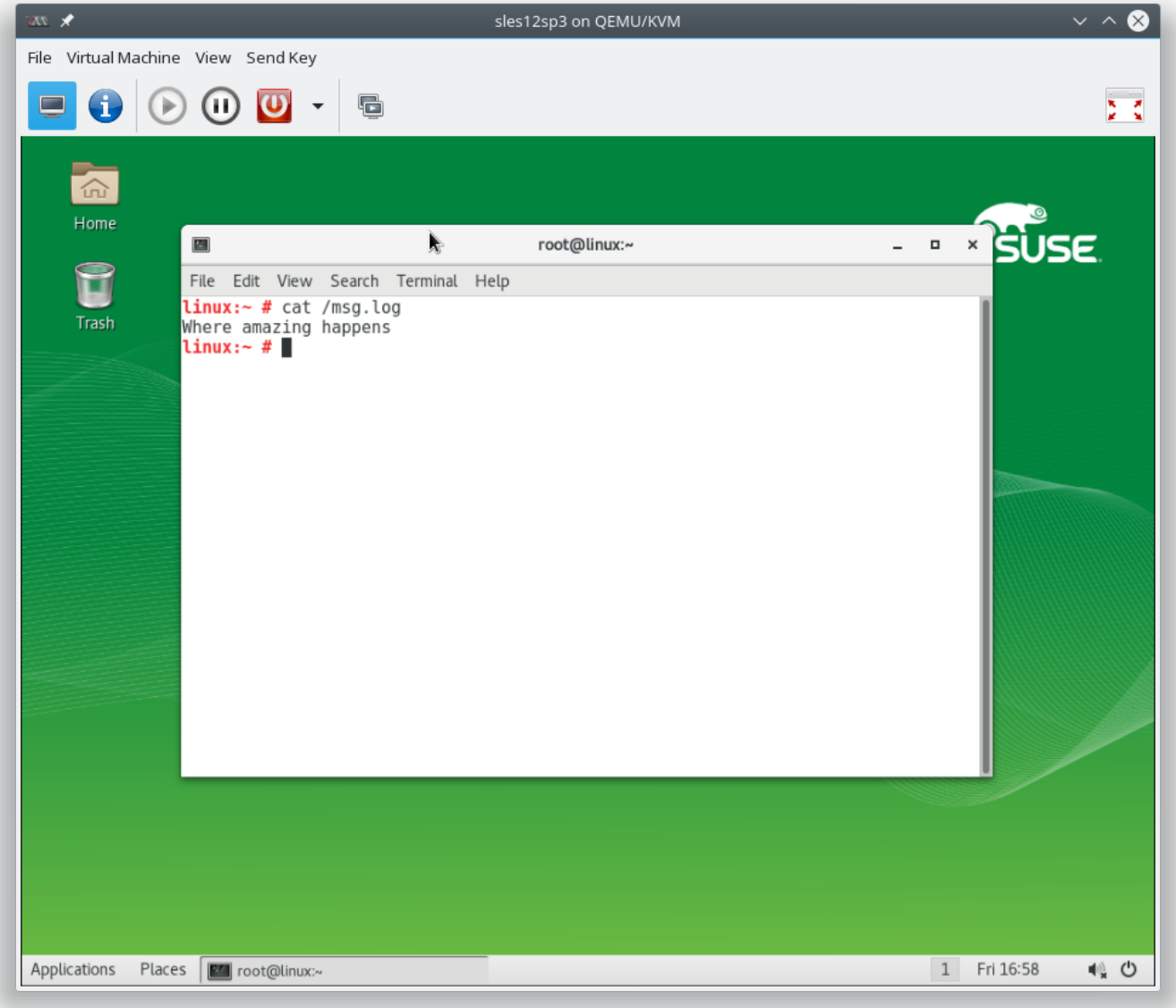
```
suse:~/projects/libguestfs$ echo 'Hello World' > /tmp/msg.log
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ sudo ./run virt-copy-in -d sles12sp3 /tmp/msg.log /
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ sudo ./run virt-copy-in -d win10 /tmp/msg.log /
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ rm /tmp/msg.log
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ sudo ./run virt-cat -d sles12sp3 /msg.log
Hello World
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ sudo ./run virt-cat -d win10 /msg.log
Hello World
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ sudo ./run virt-copy-out -d sles12sp3 /msg.log /tmp/
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ cat /tmp/msg.log
Hello World
suse:~/projects/libguestfs$ sudo rm /tmp/msg.log
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ sudo ./run virt-copy-out -d win10 /msg.log /tmp/
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ cat /tmp/msg.log
Hello World
suse:~/projects/libguestfs$ sudo rm /tmp/msg.log
suse:~/projects/libguestfs$
```


sles12sp3 on QEMU/KVM
File Virtual Machine View Send Key
Home
Trash
root@linux:~
File Edit View Search Terminal Help
linux:~ # ls /
bin dev home lib64 mnt opt root sbin srv tmp var
boot etc lib lost+found msg.log proc run selinux sys usr
linux:~ #
linux:~ # cat /msg.log
Hello World
linux:~ #
SUSE
Applications Places root@linux:~
1 Fri 16:40


win10 on QEMU/KVM
File Virtual Machine View Send Key
回收站
本地磁盘 (C:)
文件 主页 共享 查看
此电脑 › 本地磁盘 (C:)
搜索"本地磁盘 (C:)"
快速访问
桌面
下载
文档
图片
视频
音乐
OneDrive
此电脑
网络
名称 修改日期 类型 大小
PerfLogs 2017/9/29 21:46 文件夹
Program Files 2018/3/23 15:34 文件夹
Program Files (x86) 2017/9/30 22:40 文件夹
Windows 2018/3/23 15:38 文件夹
用户 2018/3/23 15:53 文件夹
msg 2018/3/23 16:34 文本文档 1 KB
6 个项目 选中 1 个项目 12 字节
msg - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
Hello World
有问题尽管问我
16:41
2018/3/23

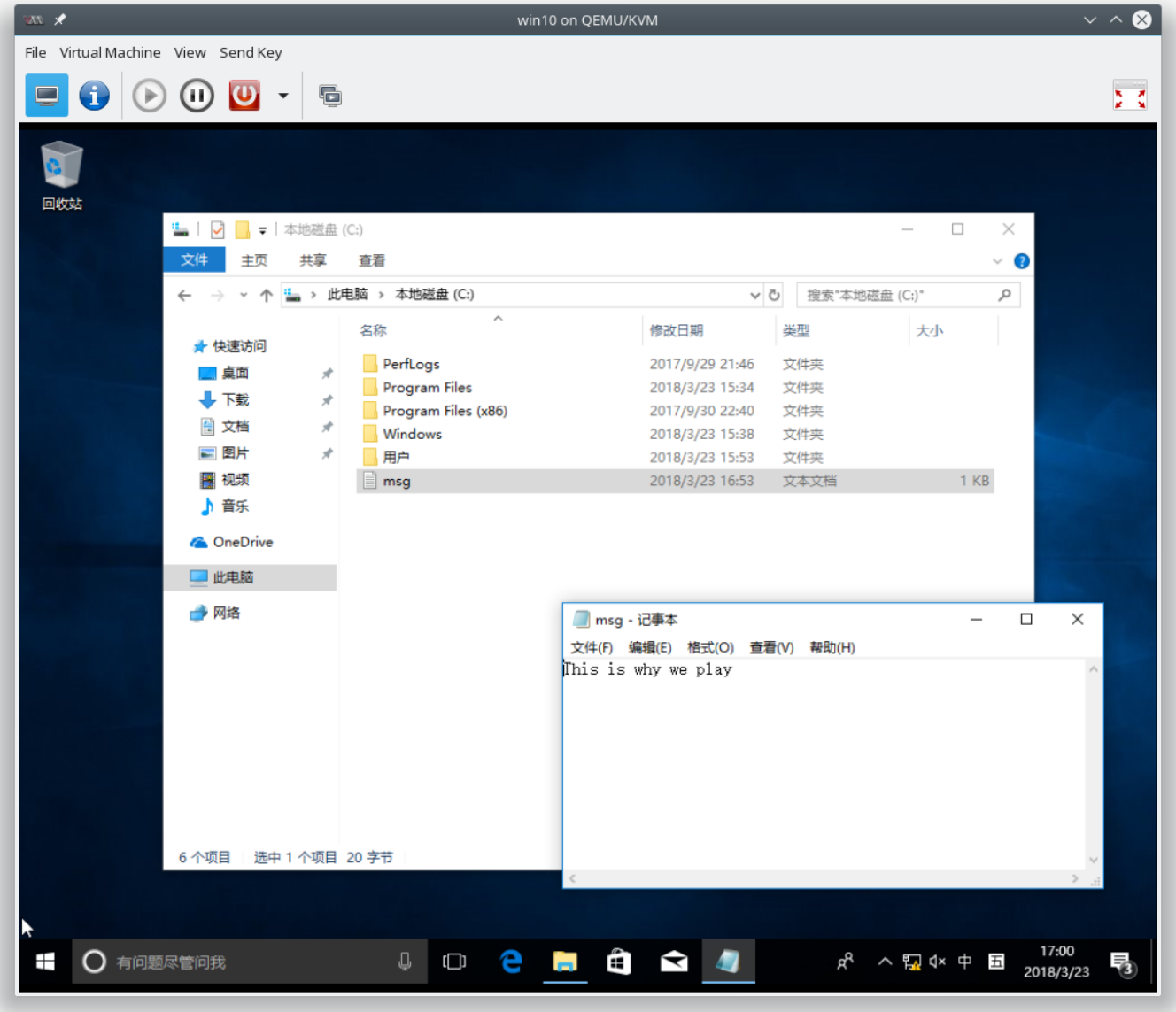
例 6：virt-edit 去编辑 linux 虚拟机和 windows 虚拟机中的文本文件:
suse:~/projects/libguestfs$ ./run **virt-edit** -d sles12sp3 /msg.log
在 host 端弹出的 vi 编辑器中，变更 sles12sp3 虚拟机中的文件/msg.log 的内容为:
Where amazing happens

然后启动虚拟机，去验证下:



suse:~/projects/libguestfs$ ./run **virt-edit** -d win10 /msg.log
在 host 端弹出的 vi 编辑器中，变更 win10 虚拟机中的文件/msg.log 的内容为:
This is why we
play

然后启动虚拟机，去验证下:



例 7：virt-inspector 检查虚拟机镜像信息

```
suse:~/projects/libguestfs$ sudo ./run virt-inspector sles12sp3_raw
<?xml version="1.0"?>
<operatingsystems>
  <operatingsystem>
    <root>/dev/sda5</root>
    <name>linux</name>
    <arch>x86_64</arch>
    <distro>sles</distro>
    <product_name>SUSE Linux Enterprise Server 12 SP3</product_name>
    <major_version>12</major_version>
    <minor_version>3</minor_version>
    <package_format>rpm</package_format>
    <package_management>zypper</package_management>
    <hostname>guest03</hostname>
    <osinfo>sles12sp3</osinfo>
    <mountpoints>
      <mountpoint dev="/dev/sda5">/</mountpoint>
      <mountpoint dev="/dev/sda6">/srv</mountpoint>
      <mountpoint dev="/dev/sda2">/home</mountpoint>
    </mountpoints>
    <filesystems>
      <filesystem dev="/dev/sda1">
        <type>swap</type>
        <uuid>04a1b57a-c970-48d7-95f4-c0f6fac48acd</uuid>
      </filesystem>
      <filesystem dev="/dev/sda2">
        <type>ext4</type>
        <uuid>983d6474-ea88-4991-a1e4-d9b92d9788a2</uuid>
      </filesystem>
      <filesystem dev="/dev/sda5">
        <type>ext4</type>
        <uuid>fc380e15-8bb8-4a8f-bf3b-857620de3314</uuid>
      </filesystem>
      <filesystem dev="/dev/sda6">
        <type>ext4</type>
        <uuid>d41e79bb-0644-4cc0-bd50-04dadf7d6c39</uuid>
      </filesystem>
    </filesystems>
    <applications/>
  </operatingsystem>
</operatingsystems>
suse:~/projects/libguestfs$
```

例 8：virt-resize 虚拟机磁盘扩容

在准备新镜像的阶段，
如果老镜像是 raw 格式,那么生成新镜像的方式可以用 qemu-img 或 truncate，后者使用起来更直观,比如共计想增加 6GB 的空间:
truncate -r olddisk newdisk
truncate -s +6G newdisk

如果老镜像是 qcow2 格式，那么自然就不能使用'+nG'这种方式了。

注意:
* virt-resize 不可以用于逻辑分区
* 扩容前，一定要先确认新镜像所在的 host 磁盘分区上有足够的空间
* 虚拟机镜像中若使用了 lvm，则 virt-resize 语法略有不同，自查

假设用户想给虚拟机’sles12sp3’增加合计 6GB 的空间, 步骤请参考下面的截图:

```
suse:~/projects/libguestfs$ sudo virsh domblklist sles12sp3
Target     Source
------------------------------------------------
vda        /opt/vms/sles12sp3/disk0.qcow2
```

老镜像的基本信息是：

```
suse:~/projects/libguestfs$ qemu-img info /opt/vms/sles12sp3/disk0.qcow2
image: /opt/vms/sles12sp3/disk0.qcow2
file format: qcow2
virtual size: 15G (16106127360 bytes)
disk size: 3.4G
cluster_size: 65536
Format specific information:
    compat: 1.1
    lazy refcounts: false
    refcount bits: 16
    corrupt: false
suse:~/projects/libguestfs$
```

生成空的新镜像：

```
suse:~/projects/libguestfs$ sudo qemu-img create -f qcow2 /opt/vms/sles12sp3/new0.qcow2 21G
Formatting '/opt/vms/sles12sp3/new0.qcow2', fmt=qcow2 size=22548578304 cluster_size=65536 lazy_refcounts=off refcount_bits=16
suse:~/projects/libguestfs$
```

我们可以通过 virt-df、virt-filesystems 与 virt-inspector 先分析下要扩容的虚拟机中的分区、挂载与容量使用情况：

```
suse:~/projects/libguestfs$ sudo ./run virt-df -d sles12sp3 --human-readable
Filesystem                                Size       Used  Available  Use%
sles12sp3:/dev/sda2                       5.0G       818M       3.7G   16%
sles12sp3:/dev/sda5                       7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@            7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/boot/grub2/i386-pc
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/boot/grub2/x86_64-efi
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/opt        7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/srv        7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/tmp        7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/usr/local
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots/2/snapshot
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots/3/snapshot
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots/4/snapshot
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots/5/snapshot
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots/6/snapshot
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots/7/snapshot
                                          7.0G       1.6G       5.2G   24%
sles12sp3:btrfsvol:/dev/sda5/@/.snapshots/8/snapshot
                                          7.0G       1.6G       5.2G   24%
sles12sp3:/dev/sda6                       2.0G        85M       1.7G    5%
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ sudo ./run virt-inspector -d sles12sp3 | \
> xpath -e /operatingsystems/operatingsystem[1]/mountpoints
Found 1 nodes in stdin:
-- NODE --
<mountpoints>
      <mountpoint dev="/dev/sda5">/</mountpoint>
      <mountpoint dev="btrfsvol:/dev/sda5/@/opt">/opt</mountpoint>
      <mountpoint dev="btrfsvol:/dev/sda5/@/srv">/srv</mountpoint>
      <mountpoint dev="btrfsvol:/dev/sda5/@/tmp">/tmp</mountpoint>
      <mountpoint dev="/dev/sda6">/var</mountpoint>
      <mountpoint dev="/dev/sda2">/home</mountpoint>
      <mountpoint dev="btrfsvol:/dev/sda5/@/usr/local">/usr/local</mountpoint>
      <mountpoint dev="btrfsvol:/dev/sda5/@/.snapshots">/.snapshots</mountpoint>
      <mountpoint dev="btrfsvol:/dev/sda5/@/boot/grub2/i386-pc">/boot/grub2/i386-pc</mountpoint>
      <mountpoint dev="btrfsvol:/dev/sda5/@/boot/grub2/x86_64-efi">/boot/grub2/x86_64-efi</mountpoint>
    </mountpoints>
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ sudo ./run virt-filesystems -d sles12sp3 --long --all --human-readable
Name                                    Type       VFS     Label MBR Size Parent
/dev/sda1                               filesystem swap    -     -   1.0G -
/dev/sda2                               filesystem btrfs   -     -   5.0G -
/dev/sda3                               filesystem unknown -     -   1.0K -
/dev/sda5                               filesystem btrfs   -     -   7.0G -
btrfsvol:/dev/sda5/@                    filesystem btrfs   -     -   -    -
```

综合上面两张截图我们知道，挂载点有 4 个，扩展分区是/dev/sda3：

/dev/sda1 挂载到了 swap 上，　　容量 1GB

/dev/sda5 挂载到了/上，　　容量 7GB,已用 512MB

/dev/sda2 挂载到了/home 上，　　容量 5GB,已用 818MB

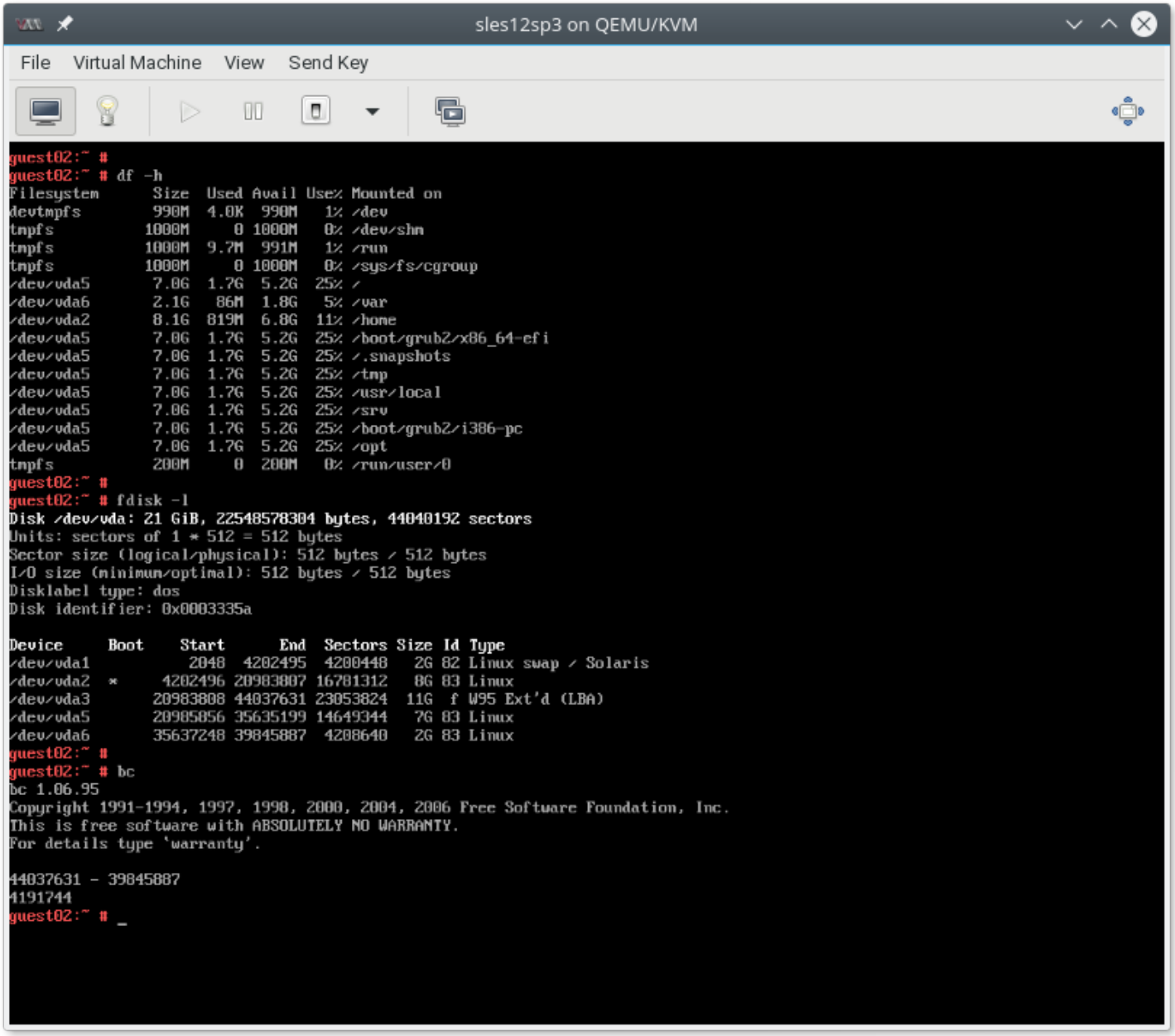
/dev/sda6 挂载到了/var 上，　　容量 5GB,已用 85MB

在新镜像中，我们想让’/home’增加 3GB，’swap’增加 1GB，剩下的容量全给扩展分区做为未分配空间：

```
suse:~/projects/libguestfs$ sudo ./run virt-resize \
> --resize /dev/sda1=+1G \
> --resize /dev/sda2=+3G \
> --expand /dev/sda3 \
> /opt/vms/sles12sp3/disk0.qcow2 \
> /opt/vms/sles12sp3/new0.qcow2
[   0.0] Examining /opt/vms/sles12sp3/disk0.qcow2
**********

Summary of changes:

/dev/sda1: This partition will be resized from 1.0G to 2.0G.  The swap on
/dev/sda1 will be expanded using the ‘mkswap’ method.

/dev/sda2: This partition will be resized from 5.0G to 8.0G.  The
filesystem btrfs on /dev/sda2 will be expanded using the
‘btrfs-filesystem-resize’ method.

/dev/sda3: This partition will be resized from 9.0G to 11.0G.

**********
[   3.2] Setting up initial partition table on /opt/vms/sles12sp3/new0.qcow2
[   3.9] Copying /dev/sda1
[   4.3] Copying /dev/sda2
[   6.0] Copying /dev/sda3
 100% [
[  32.0] Expanding /dev/sda1 using the ‘mkswap’ method
[  32.1] Expanding /dev/sda2 using the ‘btrfs-filesystem-resize’ method

Resize operation completed with no errors.  Before deleting the old disk,
carefully check that the resized disk boots and works correctly.
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ sudo qemu-img info /opt/vms/sles12sp3/new0.qcow2
image: /opt/vms/sles12sp3/new0.qcow2
file format: qcow2
virtual size: 21G (22548578304 bytes)
disk size: 2.6G
cluster_size: 65536
Format specific information:
    compat: 1.1
    lazy refcounts: false
    refcount bits: 16
    corrupt: false
suse:~/projects/libguestfs$
```

变更虚拟机配置，使用新镜像启动：



可以看到，新镜像容量为 21GB, /home 的容量是 5+3=8GB, swap 是 1+1=2GB,未分配的扩展分区容量是 2GB

当然，如果用户仅仅是想给虚拟机镜像增加未分配空间，而不是增加某分区的空间，那么此时使用更简单的'qemu-img resize'就可以满足需求了。

例 9：virt-builder 快速创建镜像:

注：作为演示我们在这里并未创建位于本地的虚拟机镜像模板仓库，而是使用默认配置，也就是说做为模板的镜像都在远端。

当用户运行 **`sudo run virt-builder -list`** 来列出所支持的镜像列表时，默认情况下，`virt-builder` 会去加载位于 `libguestfs/builder/test-website/virt-builder/repos.d` 下的`*.conf`,按照这些`.conf` 文件中描述的 `index.asc` 或 `index` 中的内容组合成一个所支持的虚拟机镜像列表。

这个行为可以通过`--source` 参数去改变，通过命令行让 `virt-builder` 动态地加载用户指定的 `index.asc` 文件。自查.

```
suse:~/projects/libguestfs$ ll builder/test-website/virt-builder/repos.d/*.conf
-rw-r--r-- 1 suse users 162 Apr  5 23:12 builder/test-website/virt-builder/repos.d/libguestfs.conf
-rw-r--r-- 1 suse users 176 Apr  6 13:15 builder/test-website/virt-builder/repos.d/opensuse.conf
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ cat builder/test-website/virt-builder/repos.d/libguestfs.conf
[libguestfs.org]
uri=file:///home/suse/projects/libguestfs/website/download/builder/index.asc
gpgkey=file:///home/suse/projects/libguestfs/builder/libguestfs.gpg
suse:~/projects/libguestfs$
suse:~/projects/libguestfs$ cat builder/test-website/virt-builder/repos.d/opensuse.conf
[opensuse.org]
uri=http://download.opensuse.org/repositories/Virtualization:/virt-builder-images/images/index
gpgkey=file:///home/suse/projects/libguestfs/builder/opensuse.gpg
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ egrep 'debian-8|fedora-27' /home/suse/projects/libguestfs/website/download/builder/index.asc
[debian-8]
file=debian-8.xz
[fedora-27]
file=fedora-27-aarch64.xz
    http://libguestfs.org/download/builder/fedora-27-aarch64-nvram.xz
[fedora-27]
file=fedora-27-armv7l.xz
[fedora-27]
file=fedora-27-i686.xz
[fedora-27]
file=fedora-27-ppc64.xz
[fedora-27]
file=fedora-27-ppc64le.xz
[fedora-27]
file=fedora-27.xz
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ sudo ./run virt-builder --list
opensuse-13.1            x86_64     openSUSE 13.1
opensuse-13.2            x86_64     openSUSE 13.2
opensuse-42.1            x86_64     openSUSE Leap 42.1
opensuse-tumbleweed      x86_64     openSUSE Tumbleweed
centos-6                 x86_64     CentOS 6.6
centos-7.0               x86_64     CentOS 7.0
centos-7.1               x86_64     CentOS 7.1
centos-7.2               aarch64    CentOS 7.2 (aarch64)
centos-7.2               x86_64     CentOS 7.2
centos-7.3               x86_64     CentOS 7.3
centos-7.4               x86_64     CentOS 7.4
cirros-0.3.1             x86_64     CirrOS 0.3.1
cirros-0.3.5             x86_64     CirrOS 0.3.5
debian-6                 x86_64     Debian 6 (Squeeze)
debian-7                 sparc64    Debian 7 (Wheezy) (sparc64)
debian-7                 x86_64     Debian 7 (wheezy)
debian-8                 x86_64     Debian 8 (jessie)
debian-9                 x86_64     Debian 9 (stretch)
fedora-18                x86_64     Fedora® 18
fedora-19                x86_64     Fedora® 19
fedora-20                x86_64     Fedora® 20
fedora-21                aarch64    Fedora® 21 Server (aarch64)
fedora-21                armv7l     Fedora® 21 Server (armv7l)
fedora-21                ppc64      Fedora® 21 Server (ppc64)
fedora-21                ppc64le    Fedora® 21 Server (ppc64le)
```

当使用 virt-builder 构建虚拟机镜像时，默认情况下， virt-builder 在下载镜像模板的同时它会在$HOME/.cache/virt-builder/下保存一份复本当做 cache,以后再构建时当有 cache 存在的前提下，就不用去远程重新下载模板了。

默认情况下，virt-builder 认为镜像模板与 index.asc 是位于同一个位置的，所以：

当首次创建一个 fedora-27 虚拟机的镜像时,因为该镜像信息是在 /home/suse/projects/libguestfs/website/download/builder/index.asc 中描述的,所以 virt-builder 就去 /home/suse/projects/libguestfs/website/download/builder/找 fedora-27.xz 下载并保存一份复本在$HOME/.cache/virt-builder/

当首次创建一个 opensuse-42.1 虚拟机的镜像时,因为该镜像信息是在 http://download.opensuse.org/repositories/Virtualization:/virt-builder-images/images/index 中描述的，所以 virt-builder 就去 http://download.opensuse.org/repositories/Virtualization:/virt-builder-images/images/找 openSUSE-Leap-42.1.x86_64-0.0.1-Build1.9.qcow2.xz

suse:~/projects/libguestfs$ sudo ./run **virt-builder** fedora-27
或
suse:~/projects/libguestfs$ sudo ./run **virt-builder** opensuse-42.1

查看/删除 cache 可以这样：

suse:~/projects/libguestfs$ sudo ./run **virt-builder** --print-cache

```
cache directory: /root/.cache/virt-builder
...
debian-9            x86_64     no
fedora-18           x86_64     no
fedora-19           x86_64     no
fedora-20           x86_64     cached
fedora-21           aarch64    no
fedora-21           armv7l     no
...
```

suse:~/projects/libguestfs$ sudo ./run **virt-builder** --delete-cache

```
[   0.0] Deleting: /root/.cache/virt-builder
```

参数化构建虚拟机镜像，然后将它传给 virt-install 创建一个虚拟机：

下面 virt-builder 命令行中的参数中，挑几个略作说明 --root-password 用于变更镜像中的 root 密码，file: 是说密码位于文件中；--size 是变更镜像尺寸为 10GB(通常做为模板的镜像的尺寸都较小,便于快速下载)；--run-command 是构建镜像时，用户想要执行的命令；--firstboot-command 是首次用生成的镜像启动时自动被执行的命令。

```
suse:~/projects/libguestfs$ sudo ./run virt-builder --print-cache
[sudo] password for root:
cache directory: /root/.cache/virt-builder
opensuse-13.1             x86_64     no
opensuse-13.2             x86_64     no
opensuse-42.1             x86_64     no
opensuse-tumbleweed       x86_64     no
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ mkdir /opt/vms/leap42.1
suse:~/projects/libguestfs$ echo 123456 > /tmp/root_password.txt
suse:~/projects/libguestfs$ sudo ./run virt-builder opensuse-42.1 \
> --output /opt/vms/leap42.1/disk0.qcow2 \
> --format qcow2 \
> --size 10G \
> --root-password file:/tmp/rootpw \
> --hostname guest01 \
> --run-command 'mkdir /tmp/abc/' \
> --firstboot-command 'ip addr show eth0 > /tmp/info'
[   3.6] Downloading: http://download.opensuse.org/repositories/Virtualiz
[   4.5] Planning how to build this image
[   4.5] Uncompressing
[  13.3] Resizing (using virt-resize) to expand the disk to 10.0G
[  25.4] Opening the new disk
[  33.3] Setting a random seed
[  33.4] Setting the hostname: guest01
[  33.5] Running: mkdir /tmp/abc/
[  33.5] Installing firstboot command: ip addr show eth0 > /tmp/info
[  33.6] Setting passwords
[  35.5] Finishing off
                   Output file: /opt/vms/leap42.1/disk0.qcow2
                   Output size: 10.0G
                 Output format: qcow2
            Total usable space: 9.7G
                    Free space: 8.9G (91%)
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ sudo ls -lh /root/.cache/virt-builder/
total 291M
-rw-r--r-- 1 root root 291M Apr  6 01:06 opensuse-42.1.x86_64.119
suse:~/projects/libguestfs$
```

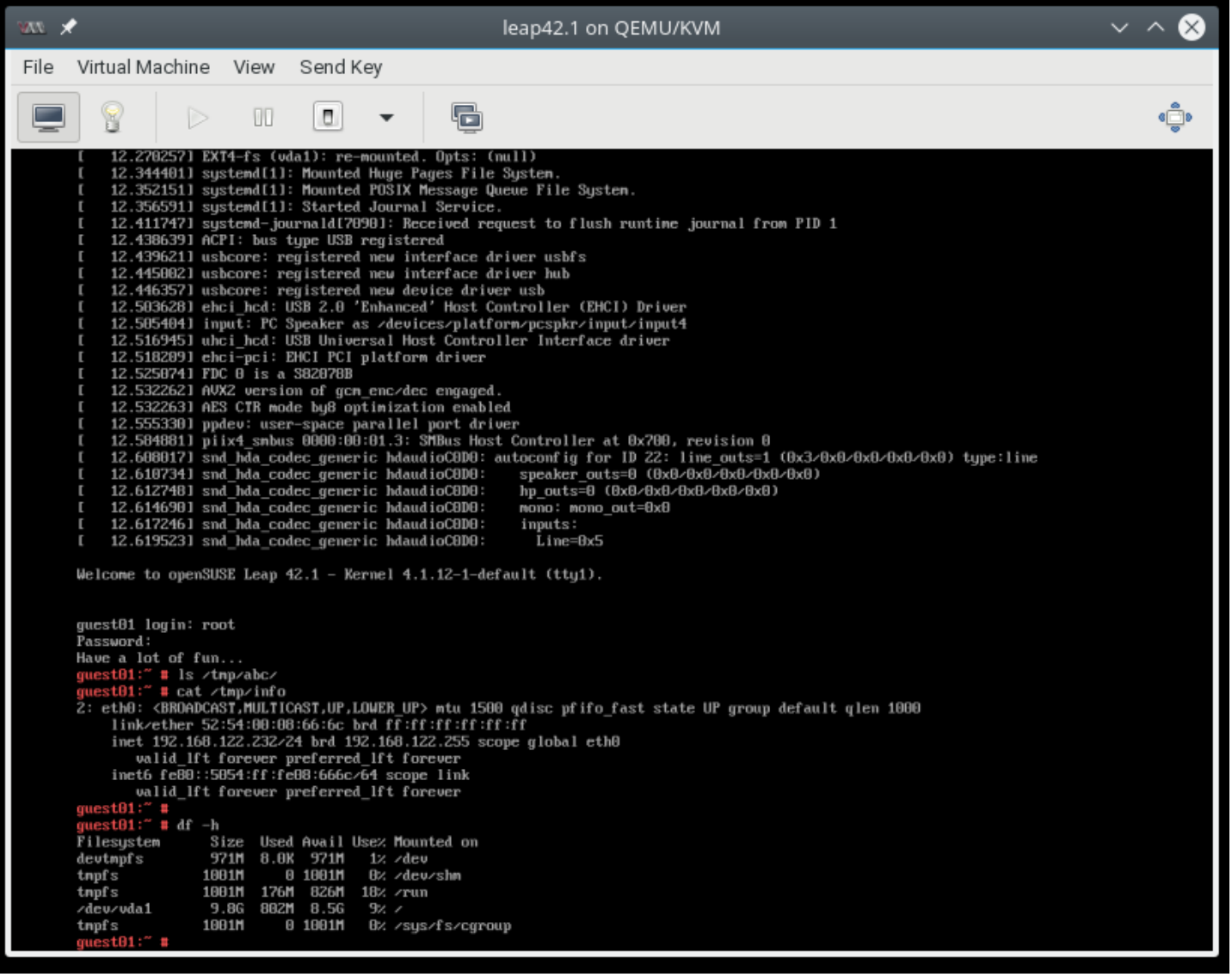
```
suse:~/projects/libguestfs$ sudo ./run virt-builder --print-cache
cache directory: /root/.cache/virt-builder
opensuse-13.1            x86_64     no
opensuse-13.2            x86_64     no
opensuse-42.1            x86_64     cached
opensuse-tumbleweed      x86_64     no
suse:~/projects/libguestfs$
```

```
suse:~/projects/libguestfs$ qemu-img info /opt/vms/leap42.1/disk0.qcow2
image: /opt/vms/leap42.1/disk0.qcow2
file format: qcow2
virtual size: 10G (10737418240 bytes)
disk size: 888M
cluster_size: 65536
Format specific information:
    compat: 1.1
    lazy refcounts: false
    refcount bits: 16
    corrupt: false
suse:~/projects/libguestfs$
```

下图在 guest 中验证了 virt-builder 命令行选项的效果，即 root 登陆时使用的密码就是用户自定义的 123456(相信我，上图中我输入的密码真的是 123456), 主机名是 guest01, 生成镜像后自动在其中创建了/tmp/abc 文件夹，扩展其尺寸至 10GB, 首次启动时自动运行”ip addr show eth0 > /tmp/info”.

推荐在 virt-install 阶段指定一个 os-variant，它可以提供比较适合所选 guest 的配置集合。
如果 virt-install 阶段不清楚用什么 os-variant,可以运行 osinfo-query os --fields=short-id,release-date,vendor,eol-date vendor="openSUSE" 或干脆用 osinfo-query os 来查找。

```
suse:~/projects/virt-manager$ sudo ./virt-install \
>     --virt-type kvm \
>     --os-variant opensuse42.1 \
>     --name leap42.1 \
>     --memory 2048 \
>     --vcpu=2 \
>     --import \
>     --disk path=/opt/vms/leap42.1/disk0.qcow2,format=qcow2,bus=virtio \
>     --network model=virtio,bridge=virbr0 \
>     --noautoconsole

Starting install...
Domain creation completed.
suse:~/projects/virt-manager$
```



例 10：**virt-rescue** 进入 rescue shell 去执行一些任务，就像一张 Rescue CD 所起的作用，像检测磁盘，修复某些错误或变更磁盘分区等等。

现在，我们假设某个云客户忘记了虚拟机的 root 密码，管理员帮其变更 root 用户的密码为 123456:

```
suse:~/projects/libguestfs$ sudo ./run virt-rescue -d sles12sp3
supermin: mounting /proc
supermin: ext2 mini initrd starting up: 5.1.18 glibc
Starting /init script ...
[/usr/lib/tmpfiles.d/legacy.conf:14] Unknown group 'lock'.
[/usr/lib/tmpfiles.d/systemd.conf:11] Unknown group 'utmp'.
[/usr/lib/tmpfiles.d/systemd.conf:19] Unknown user 'systemd-network'.
[/usr/lib/tmpfiles.d/systemd.conf:20] Unknown user 'systemd-network'.
[/usr/lib/tmpfiles.d/systemd.conf:21] Unknown user 'systemd-network'.
[/usr/lib/tmpfiles.d/systemd.conf:25] Unknown group 'systemd-journal'.
[/usr/lib/tmpfiles.d/systemd.conf:26] Unknown group 'systemd-journal'.
starting version 234
mdadm: No arrays found in config file or automatically
  WARNING: Failed to connect to lvmetad. Falling back to device scanning.
mdadm: No arrays found in config file or automatically
/init: line 144: ldmtool: command not found

The virt-rescue escape key is '^]'.  Type '^] h' for help.

------------------------------------------------------------

Welcome to virt-rescue, the libguestfs rescue shell.

Note: The contents of / (root) are the rescue appliance.
You have to mount the guest's partitions under /sysroot
before you can examine them.

><rescue> mount /dev/sda5 /sysroot
><rescue> chroot /sysroot
><rescue> oldpwd=$(awk -F ':' '/^root/{print $2}' /etc/shadow)
><rescue> newpwd=$(openssl passwd -1 -salt justdoit 123456)
><rescue> sed -i '/^root/s%'$oldpwd'%'$newpwd'%' /etc/shadow
><rescue> exit
exit
><rescue> exit
exit

virt-rescue: Syncing the disk now before exiting ...

Rebooting.
[   40.583438] reboot: Restarting system
suse:~/projects/libguestfs$
```

如此，在 host 上变更虚拟机 root 密码的操作就完成了。

例 11：**虚拟机镜像的挂载与卸载** `guestmount / guestunmount`

记得在虚拟化技巧文章的第一集中，我曾提到过利用 qemu-nbd 以 nbd 的方式呈现虚拟机镜像给 host 并挂载，下面是另一种方式，或许较前者更方便，因为它用法极其简单，我就不详细举例了：

suse:~/projects/libguestfs$ sudo ./run **guestmount** -a /opt/vms/sles12sp3/disk0.qcow2 -m /dev/sda5 --ro /mnt/root

suse:~/projects/libguestfs$ sudo ./run **guestmount** -a /opt/vms/sles12sp3/disk0.qcow2 -m /dev/sda2 --rw /mnt/home

suse:~/projects/libguestfs$ sudo ./run **guestunmount** /mnt/root

suse:~/projects/libguestfs$ sudo ./run **guestunmount** /mnt/home