

虚拟机 xml 文件使用

1 文档说明及概念说明.....	3
1.1 文档说明.....	3
1.2 概念说明.....	3
2 元素(Element)介绍.....	4
2.1 通用信息.....	4
2.2 内存分配.....	5
2.3 CPU.....	5
2.4 时钟设置.....	7
2.5 系统信息.....	8
2.6 操作系统.....	9
2.7 设备	11
2.7.1 总线控制器.....	11
2.7.2 输入设备.....	11
2.7.3 图形化选项.....	11
2.7.4 显卡.....	12
2.7.5 网卡.....	12
2.7.6 磁盘.....	13

1 文档说明及概念说明

1.1 文档说明

本文档是创建一台虚拟机所需 xml 文件的最少项的介绍文档, 如果需要查看更多详细的文档介绍, 请访问: <https://libvirt.org/format.html>

1.2 概念说明

镜像和映像的区别

镜像(mirror):是把很多文件打包放在一起, 一个磁盘上的数据在另一个磁盘上存在一个完全相同的副本就称为镜像。它是一种文件存储形式。使用 GHOST 等软件来进行构成镜像和拆镜像的功能。

映像(image): 将资料和程序结合的文件, 是数据的物理结构, 现在基本上特指 VIM 格式(主要用在系统封装和引导盘上)。可以用来做还原。

磁盘按接口分

ide 磁盘: 把控制器和盘体集成在一起的磁盘驱动器, IDE 接口硬盘多用于家用产品中, 也部分应用于服务器。

sata 磁盘: SATA 口的硬盘又叫串口硬盘. SATA 以它串行的数据发送方式得名。在数据传输的过程中, 数据线和信号线独立使用, 并且传输的时钟频率保持独立。是目前比较流行的的硬盘接口类型, 目前市场上最普及的接口类型。

scsi 磁盘: SCSI 并不是专为硬盘设计的, 实际上它是一种总线型接口。独立于系统总线工作。SCSI 接口的硬盘则主要应用于服务器市场。

virtio: 半虚拟化 io 设备。

2 元素 (Element) 介绍

2.1 通用信息

```
<domain type='kvm' id='1'>
  <name>MyGuest</name>
  <uuid>4dea22b3-1d52-d8f3-2516-782e98ab3fa0</uuid>
  <title>A short description - title - of the domain</title>
  <description>Some human readable description</description>
</domain>
```

domain

所有虚拟的 xml 都是从 domain 开始的，它有两个属性，一个是 type，代表 Hypervisor (管理应用程序的类型) 它可以是 "kvm", "xen", "qemu", "lxc", 以上的这几个都是不同的虚拟化的平台 (qemu 可以结合 kvm 使用，也可以单独使用，只是与 kvm 结合性能虚拟化性能更好)。第二个属性是 id，它是正在运行的虚拟机的唯一标识，没有运行的虚拟机没有这个值。它可以省略。

domain 下的元素

name

虚拟机的名称，只能是由字母和数字组成，在它所在的物理机范围内，这个名字必须是唯一的。

uuid

为虚拟机提供一个全局的唯一标识符。它的格式必须满足 RFC 4122 格式，uuid 的全称是 Universally Unique Identifier. 如果没有写的话，会随机生成一个 uuid 出来。

title

标题，注意不能出现换行符。

description

对虚拟机的描述。

2.2 内存分配

```
<domain>
...
  <maxMemory slots='16' unit='KiB'>1524288</maxMemory>
  <memory unit='KiB'>524288</memory>
  <currentMemory unit='KiB'>524288</currentMemory>
...
</domain>
```

memory

虚拟机在开机的时候分配的最大内存。它有一个属性，unit 代表单位的意思。如果不填，默认是 KiB（即常说的 KB），还有的单位有 b 或者 bytes 代表字节数。KB 代表 10^3 字节，k 或 KiB 代表 (2^{10}) 字节，MB (10^6 字节) 等一系列单位。

maxMemory

虚拟机运行时分配的最大内存。它是这条虚拟机内存的限值（最多也就能加到这么多了）。它有两个属性，一个是 slot，代表内存槽，即允许为这个虚拟机添增加“内存条”的槽的个数。unit 同上。

currentMemory

虚拟机的实际内存。这个值可以小于最大分配容量，以适应后期内存的动态增长。如果不设，它跟 memory 一模一样（包括数值和单位）

2.3CPU

```
<domain>
...
  <vcpu placement='static' cpuset="1-4,^3,6" current="1">2</vcpu>
  <cpu mode='host-model'>
    <model fallback='allow'>core2duo</model>
  <features>
    <acpi />
    <apic />
    <pae />
  </features>
...
</domain>
```

vcpu

vcpu 的 placement 属性有两种方式 static 或者 auto ,static 表示 CPU 的绑定策略 (vCPU 到 pCPU) 是从域 xml 中指定。(就是根据 cpuset 的值来定, 如果 cpuset 没定义, 那么将绑定到全部的 pCPU 上)。 auto 方式表示通过询问 numad 来进行配置, 这个时候的 cpuset 就算有定义也不起作用。

vcpu 的 cpuset 属性代表物理 cpu 的序号, 它可以是一个单独的序号, 也可以是一个连续的序列, 还可以是一个^后面一个序号, 表示排除这个序号。上面的代表 1 2, 4 ,6 号 CPU 可以用来做绑定。

vcpu 的 current 属性代表当前有几个 CPU 来使用。
Vcpu 的值就代表几个虚拟 CPU。

cpu

cpu 的 mode 属性, host-model 代表把主机的 xml 拷贝到当前虚拟机的 xml。

cpu 的二级子项 model

指定虚拟机指定 cpu 的型号。fallback 的意思是自动回退到虚拟机管理程序支持的最接近的模型, 同时保持 CPU 功能列表, 如果 allow 代表允许这样, forbid 代表禁止。

Features（处理器特性）

Pae: 物理地址拓展模式：即运行 32 位的虚拟机的寻址空间大于 4G.

Acpi: 是高级配置与电源管理，它帮助操作系统合理控制和分配计算机硬件设备的电量。

Apic: 是高级程序中断控制器，它可以把 IRQ(中断请求)增多，可以解决设备过多而造成中断冲突的问题。

2.4 时钟设置

```
<domain>
...
    <clock offset="utc" />
...
</domain>
```

clock offset

是虚拟机时间同步的方式。它有两种方式，一个是 utc，一个是 localtime

utc: 虚拟机开机的时候通过 UTC 来同步时间(utc 是 0 时区的时间)。

localtime: 虚拟机根据所在主机配置的时区来同步时间。(localtime 是本地的时间)

2.5 系统信息

```
<sysinfo type="smbios">
  <bios>
    <entry name="vendor">DATASURE</entry>
  </bios>
  <system>
    <entry name="manufacturer">ACESURE6.0_0_0sda_1sdb</entry>
    <entry name="product">Server_4</entry>
    <entry name="serial">20210729143449488785149</entry>
  </system>
</sysinfo>
```

sysinfo 是一级元素，它的 type 属性用来确定二级元素布局。

sysinfo

它的 type 属性可选 `smbios` 和 `fwcfg`。如果选 `smbios`，表示二级子元素调用特定的 SMBIOS 值，与 `<os>` 的 `<smbios mode='sysinfo'>` 二级子元素一起使用，才会生效。每个二级子元素代表一个 SMBIOS block，每一个 block 下面都可以有一个 entry 的 list，用来描述这个 block。

二级元素

bios

其中的 vendor：供应商名字

system

其中的 manufacturer：生产厂家

其中的 product：产品名字

其中的 serial：序列号

2.6 操作系统

这个元素可以用来设置启动项

Bios 引导

```
<os>
  <type arch="x86_64">hvm</type>
  <smbios mode="sysinfo" />
  <boot dev='hd' />
</os>
```

type

这是<os>的二级子元素，指定在虚拟机中引导的 operating system 的类型。type 的值可以是：

hvm：用于未修改的操作系统，表明操作系统是设计用于在裸机上运行的，hvm 代表全虚拟化。

linux：(遗留名称)用于支持 Xen 3 hypervisor guest ABI 的操作系统。

xen：用于 XEN PV。

xenpvh：用于 XEN PVH。

exe：用于基于容器的虚拟化

type 的属性 arch 用于指定虚拟化的 CPU 架构。缺省时，对于大多数 hypervisor 驱动程序，将选择主机的 arch(X86 代表 32 位操作系统，X86_64 代表 64 位操作系统。type 的属性 machine：机器类型。

smbios

注意和 sysinfo 的联合关系

boot

它的属性 dev 表示怎么启动：如“fd”表示从文件启动，“hd”从硬盘启动，“cdrom”从光驱启动 和 “network”从网络启动。

Uefi 引导

只需要在上面的基础上多加一个 loader 的元素就可以。

```
<os>
  <loader>/usr/share/edk2.git/ovmf-x64/OVMF-pure-efi.fd</loader>
</os>
```

loader

指向启动程序的绝对路径。

直接内核引导

```
<os>
  <type>hvm</type>
  <loader>/usr/lib/xen/boot/hvmloader</loader>
  <kernel>/root/f8-i386-vmlinuz</kernel>
  <initrd>/root/f8-i386-initrd</initrd>
  <cmdline>console=ttyS0 ks=http://example.com/f8-i386/os/</cmdline>
</os>
```

kernel

指定在主机中的内核映像的路径。

initrd

指定在主机中的 ram 驱动器的路径。

cmdline

用来把参数传给内核。

2.7 设备

2.7.1 总线控制器

```
<devices>
  <controller type="usb" index="0" />
</devices>
```

controller

每个控制器有一个 type 属性，这个属性的值必须是 “ide”，“usb”，“pci”等总线类型的一种，并且后边有一个 index，这个 index 必须是十进制的整数，然后它表示遇到总线控制器的顺序。如果没有写，会默认分配一个最小的未分配的值。

2.7.2 输入设备

```
<devices>
  <input type="tablet" bus="usb" />
</devices>
```

input

可选的 type 属性有 mouse, keyboard, tablet 等，tablet 是触摸板、写字板的意思，它提供绝对的光标移动，鼠标提供的是相对的光标移动。bus 属性集总线类型用于细化确定的设备类型。它的值可以是 “xen”（半虚拟化），“ps2”（实际上是 ps/2 接口），usb(usb 接口)，virtio(虚拟 io)。

2.7.3 图像化选项

可以理解成显示平台（显示方式）

```
<devices>
```

```

    <graphics type="vnc" autoport="yes">
      <listen type="address" address="0.0.0.0" />
    </graphics>
  </devices>

```

graphics

graphics 的 type 属性有 “sdl”, “vnc”, “spice”, “rdp”, “desktop”, “egl-headless”等等。其中 vnc 表示启动一个 VNC 的服务器, porta 属性用来指定 TCP 的端口号, autoport 属性代表自动分配 TVP 端口。

graphics 的 listen 元素

listen 拥有设置从何处监听客户端。它的 type 属性是必填的。只有 vnc, spice 和 rdp 支持 listen 元素。这个 type 可以是 address, network , socket, none。

listen 可以填 IP 地址和主机名, 如果 IP 地址是 0, 0, 0, 0(是本机上的所有 IPV4 地址, 如果一个主机有两个 IP 地址, 192.168.1.1 和 10.1.2.1, 并且该主机上的一个服务监听的地址是 0.0.0.0, 那么通过两个 ip 地址都能够访问该服务)代表监听者两个 IP 地址, 监听服务端的 IP 地址, 看有没有客户端连过来, 而不是监听服务端 IP 地址。

2.7.4 显卡

```

<devices>
  <video>
    <model type="cirrus" vram="32768" />
  </video>
</devices>

```

vedio

用来定义 视频设备（显卡），它的 type 有“vga”, “cirrus”, “qxl”, “none”等。其中 cirrus: qemu2.2 之前默认使用的 video 类型。不能支持高清显示并且色彩采用 24 位。兼容度高。vga:如果想使用高分辨率，使用这个选项。qxl: 半虚拟化显卡。

model 的 vram 属性用来设置视频内存的大小，以千字节为单位。

2.7.5 网卡

```

<interface type="network">
  <mac address="00:0c:29:d1:10:ea" />

```

```

    <source network="default" />
    <model type="e1000" />
</interface>

```

interface

用来设置网络接口，它的 type 属性，有三种方式，“direct”：直接连到物理接口，“network”：虚拟网络，“bridge”：桥接到局域网。

mac

它的 address 属性：mac 地址。

model

它的 type 属性：设置网卡驱动

2.7.6 磁盘

```

<disk type="file" device="disk">
    <driver name="qemu" type="qcow2" />
    <boot order="1" />
    <source file="/dev_sdb1/StoreBackup/M6FJJGS516C9M9I3LPU
HZD4KFQ1VHK8J/hd0/0_20210729-143030-00001_compute_temp.dsf" />
    <target dev="sda" bus="sata" />
</disk>

```

disk

用来描述备份磁盘的信息。它的 type 属性：可以是“file”，“block”，“dir”，“network”，or “volume”（卷）or “nvme”（磁盘的一种），or “vhostuser”（通用的共享内存通信方案）。它的 device 属性：可以是“floppy”（磁片），“disk”（磁盘），“cdrom”（光碟），and “lun”（逻辑单元号，用来扩充 Target ID，一个 Target ID 可以有多个 lun），默认是 disk。

driver

qemu 只支持 name 属性为“qemu”，type 属性可以是“raw”，“bochs”，“qcow2”，and “qed”。

raw 格式镜像文件又被称为 原始镜像 或 裸设备镜像，从这些称谓可以看出，raw 格式镜像文件能够直接当作一个块设备，以供 虚拟机使用。

qcow2 格式包含一些特性，包括支持多重快照，占用更小的存储空间（不支持稀疏特性，也就是不会预先分配指定 size 的空间），可选的 AES 加密和可选的 zlib 压缩方式

source

file 属性指定源文件保持的路径

target

dev 的属性表示“逻辑”设备名称。不能保证指定的实际设备名称映射到来宾操作系统中的设备名称。bus 属性指定要模拟的磁盘设备类型；可能的值是特定于驱动程序的，典型值是“ide”、“scsi”、“virtio”、“xen”、“usb”、“sata”。这个属性可以省略，如果省略就会根据设备名称推断总线类型。