



# Citrix XenServer® 7.0 管理员指南

出版日期 六月 2016  
1.0 版



## Citrix XenServer® 7.0 管理员指南

版权所有 © 2016 Citrix Systems, Inc. 保留所有权利。  
版本：7.0

Citrix, Inc.  
851 West Cypress Creek Road  
Fort Lauderdale, FL 33309  
United States of America

免责声明. 本文档“按原样”提供。Citrix, Inc. 不承诺与本文档相关的所有保证，包括但不限于对适销性和特定用途适用性的默示保证。本文档可能含有技术或其他方面的错误或印刷错误。Citrix, Inc. 保留随时修订本文档中的信息的权利，并且如有更改，恕不另行通知。本文档及本文档中介绍的软件属 Citrix, Inc. 及其许可发放方的机密信息，依据 Citrix, Inc 的许可证提供。

Citrix Systems, Inc.、Citrix 徽标、Citrix XenServer 和 Citrix XenCenter 是 Citrix Systems, Inc. 和/或其一家或多家附属机构的商标，这些商标可能已在美国专利和商标局及其他国家/地区注册。所有其他商标和注册商标均归其各自所有者所有。

商标. Citrix®  
XenServer®  
XenCenter®

## 目录

1. 文档概述 .....	1
1.1. XenServer 简介 .....	1
1.1.1. 使用 XenServer 的好处 .....	1
1.1.2. 管理 XenServer .....	2
1.2. XenServer 文档 .....	2
2. 管理用户 .....	3
2.1. 使用 Active Directory (AD) 验证用户 .....	3
2.1.1. 配置 Active Directory 身份验证 .....	4
2.1.2. 用户身份验证 .....	6
2.1.3. 删除用户的访问权限 .....	7
2.1.4. 退出 AD 域 .....	8
2.2. 基于角色的访问控制 .....	8
2.2.1. 角色 .....	9
2.2.2. RBAC 角色和权限定义 .....	10
2.2.3. 结合使用 RBAC 和 CLI .....	14
2.2.3.1. 列出 XenServer 中所有可用的已定义角色 .....	14
2.2.3.2. 显示当前使用者的列表 : .....	15
2.2.3.3. 将使用者添加到 RBAC .....	15
2.2.3.4. 为创建的使用者分配 RBAC 角色 .....	15
2.2.3.5. 更改使用者的 RBAC 角色 : .....	16
2.2.4. 审核 .....	16
2.2.4.1. 审核日志 xe CLI 命令 .....	16
2.2.4.2. 获取池中的所有审核记录 .....	16
2.2.4.3. 获取自精确到毫秒的时间戳开始的池审核记录 .....	16
2.2.4.4. 获取自精确到分钟的时间戳开始的池审核记录 .....	17
2.2.5. XenServer 如何计算会话的角色 ? .....	17
3. XenServer 主机和资源池 .....	18

3.1. 主机和资源池概述 .....	18
3.2. 创建资源池的要求 .....	18
3.3. 创建资源池 .....	19
3.4. 创建异构资源池 .....	20
3.5. 添加共享存储 .....	20
3.6. 从资源池移除 XenServer 主机 .....	21
3.7. 为进行维护准备 XenServer 主机池 .....	21
3.8. 导出资源池数据 .....	22
3.8.1. 导出资源数据 .....	24
3.9. 高可用性 .....	24
3.9.1. 高可用性概述 .....	24
3.9.1.1. 使用过量 .....	25
3.9.1.2. 使用过量警告 .....	25
3.9.1.3. 主机保护 .....	25
3.9.2. 配置要求 .....	25
3.9.3. 重新启动优先级 .....	26
3.10. 对 XenServer 池启用高可用性功能 .....	27
3.10.1. 使用 CLI 启用高可用性功能 .....	27
3.10.2. 使用 CLI 解除 VM 的高可用性保护 .....	28
3.10.3. 恢复无法访问的主机 .....	28
3.10.4. 在启用高可用性功能的情况下关闭主机 .....	28
3.10.5. 关闭受高可用性功能保护的 VM .....	28
3.11. 主机启动 .....	29
3.11.1. 远程启动主机 .....	29
3.11.2. 使用 CLI 管理主机启动功能 .....	29
3.11.2.1. 使用 CLI 启用主机启动功能 .....	29
3.11.2.2. 使用 CLI 远程打开主机 .....	29
3.11.3. 配置 XenServer 主机开机功能的自定义脚本 .....	29
3.11.3.1. 键/值对 .....	30

3.11.3.1.1. host.power_on_mode .....	30
3.11.3.1.2. host.power_on_config .....	30
3.11.3.2. 示例脚本 .....	30
3.12. 与 XenServer 主机和资源池通信 .....	31
<b>4. 网络连接 .....</b>	<b>32</b>
4.1. 网络连接支持 .....	32
4.2. vSwitch 网络 .....	32
4.3. XenServer 网络概述 .....	33
4.3.1. 网络对象 .....	34
4.3.2. 网络 .....	34
4.3.3. VLAN .....	34
4.3.3.1. 结合使用 VLAN 和管理接口 .....	34
4.3.3.2. 使用带有虚拟机的 VLAN .....	34
4.3.3.3. 使用具有专用存储 NIC 的 VLAN .....	35
4.3.3.4. 将管理接口和来宾 VLAN 组合到单一主机 NIC 上 .....	35
4.3.4. 巨型帧 .....	35
4.3.5. NIC 绑定 .....	35
4.3.5.1. 主动-主动绑定 .....	37
4.3.5.2. 主动-被动绑定 .....	39
4.3.5.3. LACP 链路聚合控制协议绑定 .....	40
4.3.5.4. 交换机配置 .....	43
4.3.5.4.1. 使用 LACP 绑定时的交换机配置 .....	43
4.3.6. 设置后的初始网络配置 .....	44
4.3.7. 更改网络配置 .....	44
4.3.7.1. 修改网络对象 .....	44
4.3.8. 更改绑定的最高延迟 .....	45
4.4. 管理网络配置 .....	45
4.4.1. 跨服务器专用网络 .....	45
4.4.2. 在独立服务器中创建网络 .....	46

4.4.3. 在资源池中创建网络 .....	46
4.4.4. 创建 VLAN .....	47
4.4.5. 在独立主机上创建 NIC 绑定 .....	47
4.4.5.1. 创建 NIC 绑定 .....	47
4.4.5.2. 控制绑定的 MAC 地址 .....	48
4.4.5.3. 还原 NIC 绑定 .....	48
4.4.6. 在资源池中创建 NIC 绑定 .....	48
4.4.6.1. 将 NIC 绑定添加到新资源池 .....	49
4.4.6.2. 将 NIC 绑定添加到现有池 .....	49
4.4.7. 配置专用存储 NIC .....	49
4.4.8. 使用已启用 SR-IOV 的 NIC .....	50
4.4.9. 控制传出数据的速率 (QoS) .....	51
4.4.10. 更改网络连接配置选项 .....	52
4.4.10.1. 主机名 .....	52
4.4.10.2. DNS 服务器 .....	52
4.4.10.3. 更改独立主机的 IP 地址配置 .....	52
4.4.10.4. 更改资源池中的 IP 地址配置 .....	52
4.4.10.5. 管理接口 .....	53
4.4.10.6. 禁用管理访问 .....	54
4.4.10.7. 添加新的物理 NIC .....	54
4.4.10.8. 使用交换机端口锁定 .....	54
4.4.10.8.1. 要求 .....	54
4.4.10.8.2. 备注 .....	54
4.4.10.8.3. 实施说明 .....	55
4.4.10.8.4. 示例 .....	55
4.4.10.8.5. 交换机端口锁定的工作原理 .....	56
4.4.10.8.6. VIF 锁定模式状态 .....	56
4.4.10.8.7. 配置交换机端口锁定 .....	57
4.4.10.8.8. 阻止虚拟机从特定网络发送或接收通信 .....	58

4.4.10.8.9. 删除 VIF 对 IP 地址的限制 .....	59
4.4.10.8.10. 简化云中的 VIF 锁定模式配置 .....	59
4.4.10.8.11. 使用网络设置过滤 VIF 通信 .....	60
4.5. 网络故障排除 .....	60
4.5.1. 诊断网络损坏 .....	60
4.5.2. 紧急网络重置 .....	61
4.5.2.1. 验证网络重置 .....	61
4.5.2.2. 使用 CLI 执行网络重置 .....	61
4.5.2.2.1. 池主服务器命令行示例 .....	63
4.5.2.2.2. 池成员命令行示例 .....	63
5. 存储 .....	65
5.1. 存储概述 .....	65
5.1.1. 存储库 (SR) .....	65
5.1.2. 虚拟磁盘映像 (VDI) .....	65
5.1.3. 物理块设备 (PBD) .....	65
5.1.4. 虚拟块设备 (VBD) .....	65
5.1.5. 存储对象摘要 .....	65
5.1.6. 虚拟磁盘数据格式 .....	66
5.1.6.1. VDI 类型 .....	66
5.1.6.2. 使用 xe CLI 创建原始虚拟磁盘 .....	66
5.1.6.3. 在 VDI 格式间转换 .....	66
5.1.6.4. 基于 VHD 的 VDI .....	67
5.2. 存储库格式 .....	67
5.2.1. 本地 LVM .....	68
5.2.1.1. LVM 性能注意事项 .....	68
5.2.1.2. 创建本地 LVM SR (lvm) .....	68
5.2.2. 本地 EXT3 .....	69
5.2.2.1. 创建本地 EXT3 SR (ext) .....	69
5.2.3. udev .....	69

5.2.4. ISO .....	69
5.2.5. 软件 iSCSI 支持 .....	69
5.2.5.1. XenServer 主机 iSCSI 配置 .....	70
5.2.6. 软件 FCoE 存储 .....	70
5.2.6.1. 创建软件 FCoE SR .....	70
5.2.7. 硬件主机总线适配器 (HBA) .....	71
5.2.7.1. QLogic iSCSI HBA 设置示例 .....	71
5.2.7.2. 删除基于 HBA 的 SAS、FC 或 iSCSI 设备条目 .....	71
5.2.8. 共享 LVM 存储 .....	71
5.2.8.1. 使用软件 iSCSI 发起程序创建基于 iSCSI 的共享 LVM SR (lvmoiscsi) .....	71
5.2.8.2. 创建基于光纤通道/以太网光纤通道/ iSCSI HBA 或 SAS 的共享 LVM SR (lvmohba) .....	72
5.2.9. NFS 和 SMB .....	74
5.2.9.1. 创建共享 NFS SR (NFS) .....	75
5.2.9.2. 创建共享 SMB SR (SMB) .....	75
5.2.10. 硬件 HBA 上的 LVM .....	76
5.3. 存储配置 .....	76
5.3.1. 创建存储库 .....	76
5.3.2. 探测 SR .....	77
5.4. 存储多路径 .....	79
5.5. XenServer 和 IntelliCache .....	80
5.5.1. IntelliCache 部署 .....	81
5.5.1.1. 安装主机时启用 .....	81
5.5.1.2. 将现有主机转换为使用精简置备 .....	81
5.5.1.3. VM 引导行为 .....	83
5.5.1.3.1. VM 缓存行为设置 .....	83
5.5.1.4. 实现详细信息和故障排除 .....	83
5.6. 存储读取缓存 .....	84
5.6.1. 启用和禁用 .....	84



5.6.2. 限制 .....	85
5.6.3. 与 IntelliCache 比较 .....	85
5.6.4. 设置读取缓存大小 .....	85
5.6.4.1. 如何查看当前 dom0 内存分配 .....	85
5.6.4.2. XenCenter 显示说明 .....	86
5.7. 管理存储库 .....	86
5.7.1. 卸载 SRs .....	86
5.7.2. 引入 SR .....	87
5.7.3. 实时 LUN 扩展 .....	87
5.7.4. 实时 VDI 迁移 .....	88
5.7.4.1. 限制和注意事项 .....	88
5.7.4.2. 使用 XenCenter 移动虚拟磁盘 .....	88
5.7.5. SR 之间的冷 VDI 迁移 ( 脱机迁移 ) .....	88
5.7.5.1. 将 VM 的所有 VDI 复制到另一个 SR .....	88
5.7.5.2. 将各个 VDI 复制到另一个 SR .....	88
5.7.6. 将本地光纤通道 SR 转换为共享 SR .....	89
5.7.7. 使用丢弃功能回收备用阵列上基于块的存储的空间 .....	89
5.7.8. 删除快照时自动回收空间 .....	90
5.7.8.1. 使用脱机合并工具回收空间 .....	90
5.7.9. 调整磁盘 IO 调度程序 .....	91
5.7.10. 虚拟磁盘 QoS 设置 .....	91
6. 配置 VM 内存 .....	93
6.1. 什么是动态内存控制 (DMC) ? .....	93
6.1.1. 动态范围的概念 .....	93
6.1.2. 静态范围的概念 .....	93
6.1.3. DMC 行为 .....	94
6.1.4. DMC 的工作原理 .....	94
6.1.5. 内存限制 .....	94
6.2. xe CLI 命令 .....	95

6.2.1. 显示 VM 的静态内存属性 .....	95
6.2.2. 显示 VM 的动态内存属性 .....	95
6.2.3. 更新内存属性 .....	95
6.2.4. 更新各个内存属性 .....	96
6.3. 升级问题 .....	96
<b>7. XenServer 内存使用率 .....</b>	<b>97</b>
7.1. 控制域内存 .....	97
7.1.1. 更改分配给控制域的内存量 .....	97
7.1.2. VM 的可用内存量 .....	97
<b>8. 灾难恢复和备份 .....</b>	<b>99</b>
8.1. 了解 XenServer DR .....	99
8.2. DR 基础结构要求 .....	99
8.3. 部署注意事项 .....	101
8.3.1. 灾难发生之前执行的步骤 .....	101
8.3.2. 灾难发生之后执行的步骤 .....	101
8.3.3. 恢复之后执行的步骤 .....	101
8.4. 在 XenCenter 中启用灾难恢复 .....	101
8.5. 发生灾难时恢复 VM 和 vApp (故障转移) .....	102
8.6. 在灾难之后将 VM 和 vApp 还原到主站点 (故障恢复) .....	102
8.7. 测试故障转移 .....	103
8.8. vApp .....	104
8.8.1. 使用 XenCenter 的“管理 vApp”对话框 .....	105
8.9. 备份和还原 XenServer 主机和 VM .....	105
8.9.1. 备份虚拟机元数据 .....	106
8.9.1.1. 备份单台主机环境 .....	106
8.9.1.2. 备份池安装 .....	107
8.9.2. 备份 XenServer 主机 .....	107
8.9.3. 备份 VM .....	108

8.10. VM 快照 .....	108
8.10.1. 常规快照 .....	108
8.10.2. 静态快照 .....	109
8.10.3. 包含内存数据的快照 .....	109
8.10.4. 创建 VM 快照 .....	109
8.10.5. 创建包含内存数据的快照 .....	109
8.10.6. 列出 XenServer 池的所有快照 .....	110
8.10.7. 列出特殊 VM 的快照 .....	110
8.10.8. 将 VM 还原为上一状态 .....	111
8.10.8.1. 删除快照 .....	111
8.10.9. 快照模板 .....	112
8.10.9.1. 通过快照创建模板 .....	112
8.10.9.2. 将快照导出到模板 .....	112
8.10.9.3. 静态快照高级说明 .....	113
8.11. 处理计算机故障 .....	114
8.11.1. 成员故障 .....	114
8.11.2. 主服务器故障 .....	115
8.11.3. 池故障 .....	115
8.11.4. 处理因配置错误导致的故障 .....	115
8.11.5. 物理机故障 .....	115
9. 监视和管理 XenServer .....	117
9.1. 监视 XenServer 性能 .....	117
9.1.1. 可用主机指标 .....	117
9.1.2. 可用 VM 指标 .....	120
9.1.3. 分析和显示 XenCenter 中的指标 .....	123
9.1.3.1. 配置性能图表 .....	123
9.1.3.1.1. 配置图表类型 .....	123
9.1.4. 配置指标 .....	124
9.1.5. 使用 RRDs .....	125

9.1.5.1. 使用 HTTP 分析 RRD .....	125
9.1.5.2. 使用 rrd2csv 分析 RRD .....	126
9.2. 警报 .....	126
9.2.1. 使用 XenCenter 查看警报 .....	126
9.2.1.1. XenCenter 性能警报 .....	126
9.2.1.1.1. 配置性能警报 .....	127
9.2.1.2. XenCenter 警报 .....	127
9.2.1.3. XenCenter 软件更新警报 .....	128
9.2.2. 使用 xe CLI 配置性能警报 .....	129
9.2.2.1. 通用示例配置 .....	131
9.3. 配置电子邮件警报 .....	131
9.3.1. 使用 XenCenter 启用电子邮件警报 .....	131
9.3.2. 使用 xe CLI 启用电子邮件警报 .....	132
9.3.2.1. 通过经过身份验证的 SMTP 服务器发送电子邮件警报 .....	132
9.3.2.1.1. 其他配置选项 .....	132
9.4. 自定义字段和标记 .....	133
9.5. 自定义搜索 .....	134
9.6. 确定物理总线适配器的吞吐量 .....	134
10. 故障排除 .....	135
10.1. 支持 .....	135
10.2. 运行状况检查 .....	135
10.3. XenServer 主机日志 .....	136
10.3.1. 将主机日志消息发送到中央服务器 .....	136
10.4. XenCenter 日志 .....	136
10.5. 对 XenCenter 和 XenServer 主机之间的连接进行故障排除 .....	136
命令行接口 .....	138
1. 基本 xe 语法 .....	138
2. 特殊字符和语法 .....	139

3. 命令类型 .....	139
3.1. 参数类型 .....	140
3.2. 低级参数命令 .....	141
3.3. 低级列表命令 .....	141
4. xe 命令参考 .....	142
4.1. 设备命令 .....	142
4.1.1. 设备参数 .....	142
4.1.2. appliance-assert-can-be-recovered .....	142
4.1.3. appliance-create .....	142
4.1.4. appliance-destroy .....	143
4.1.5. appliance-recover .....	143
4.1.6. appliance-shutdown .....	143
4.1.7. appliance-start .....	143
4.2. 审核命令 .....	143
4.2.1. audit-log-get 参数 .....	143
4.2.2. audit-log-get .....	143
4.3. 绑定命令 .....	144
4.3.1. 绑定参数 .....	144
4.3.2. bond-create .....	144
4.3.3. bond-destroy .....	144
4.4. CD 命令 .....	144
4.4.1. CD 参数 .....	144
4.4.2. cd-list .....	145
4.5. 控制台命令 .....	146
4.5.1. 控制台参数 .....	146
4.6. 灾难恢复 (DR) 命令 .....	146
4.6.1. drtask-create .....	146
4.6.2. drtask-destroy .....	146
4.6.3. vm-assert-can-be-recovered .....	147

4.6.4. appliance-assert-can-be-recovered .....	147
4.6.5. appliance-recover .....	147
4.6.6. vm-recover .....	147
4.6.7. sr-enable-database-replication .....	147
4.6.8. sr-disable-database-replication .....	147
4.6.9. 示例用法 .....	147
4.7. 事件命令 .....	148
4.7.1. 事件类 .....	148
4.7.2. event-wait .....	148
4.8. GPU 命令 .....	149
4.8.1. 物理 GPU (pGPU) 参数 .....	149
4.8.2. pGPU 操作 .....	150
4.8.2.1. pgpu-param-set .....	150
4.8.2.2. pgpu-param-get-uuid .....	150
4.8.2.3. pgpu-param-set-uuid .....	150
4.8.2.4. pgpu-param-add-uuid .....	150
4.8.3. GPU 组参数 .....	150
4.8.3.1. GPU 组操作 .....	151
4.8.4. 虚拟 GPU 参数 .....	152
4.8.5. 虚拟 GPU 类型参数 .....	152
4.8.6. 虚拟 GPU 操作 .....	153
4.8.6.1. vgpu-create .....	153
4.8.6.2. vgpu-destroy .....	153
4.8.6.3. 为使用虚拟 GPU 的 VM 禁用 VNC .....	153
4.9. 主机命令 .....	153
4.9.1. 主机选择器 .....	153
4.9.2. 主机参数 .....	154
4.9.3. host-backup .....	156
4.9.4. host-bugreport-upload .....	157

4.9.5. host-crashdump-destroy .....	157
4.9.6. host-crashdump-upload .....	157
4.9.7. host-disable .....	157
4.9.8. host-dmesg .....	157
4.9.9. host-emergency-management-reconfigure .....	158
4.9.10. host-enable .....	158
4.9.11. host-evacuate .....	158
4.9.12. host-forget .....	158
4.9.13. host-get-system-status .....	158
4.9.14. host-get-system-status-capabilities .....	159
4.9.15. host-is-in-emergency-mode .....	160
4.9.16. host-apply-edition .....	160
4.9.17. host-license-add .....	160
4.9.18. host-license-view .....	160
4.9.19. host-logs-download .....	160
4.9.20. host-management-disable .....	160
4.9.21. host-management-reconfigure .....	161
4.9.22. host-power-on .....	161
4.9.23. host-get-cpu-features .....	161
4.9.24. host-set-cpu-features .....	161
4.9.25. host-set-power-on .....	161
4.9.26. host-reboot .....	162
4.9.27. host-restore .....	162
4.9.28. host-set-hostname-live .....	162
4.9.29. host-shutdown .....	162
4.9.30. host-syslog-reconfigure .....	163
4.9.31. host-data-source-list .....	163
4.9.32. host-data-source-record .....	163
4.9.33. host-data-source-forget .....	163

4.9.34. host-data-source-query .....	164
4.10. 消息命令 .....	164
4.10.1. 消息参数 .....	164
4.10.2. message-create .....	164
4.10.3. message-destroy .....	164
4.10.4. message-list .....	164
4.11. 网络命令 .....	165
4.11.1. 网络参数 .....	165
4.11.2. network-create .....	166
4.11.3. network-destroy .....	166
4.12. 修补程序 ( 更新 ) 命令 .....	166
4.12.1. 修补程序参数 .....	166
4.12.2. patch-apply .....	167
4.12.3. patch-pool-clean .....	167
4.12.4. patch-pool-apply .....	167
4.12.5. patch-precheck .....	167
4.12.6. patch-upload .....	167
4.13. PBD 命令 .....	167
4.13.1. PBD 参数 .....	167
4.13.2. pbd-create .....	168
4.13.3. pbd-destroy .....	168
4.13.4. pbd-plug .....	168
4.13.5. pbd-unplug .....	168
4.14. PIF 命令 .....	168
4.14.1. PIF 参数 .....	169
4.14.2. pif-forget .....	171
4.14.3. pif-introduce .....	171
4.14.4. pif-plug .....	171
4.14.5. pif-reconfigure-ip .....	171



4.14.6. pif-scan .....	171
4.14.7. pif-unplug .....	172
4.15. 池命令 .....	172
4.15.1. 池参数 .....	172
4.15.2. pool-designate-new-master .....	173
4.15.3. pool-dump-database .....	173
4.15.4. pool-eject .....	173
4.15.5. pool-emergency-reset-master .....	173
4.15.6. pool-emergency-transition-to-master .....	173
4.15.7. pool-ha-enable .....	173
4.15.8. pool-ha-disable .....	174
4.15.9. pool-join .....	174
4.15.10. pool-recover-slaves .....	174
4.15.11. pool-restore-database .....	174
4.15.12. pool-sync-database .....	174
4.16. 存储管理器命令 .....	174
4.16.1. SM 参数 .....	174
4.17. SR 命令 .....	175
4.17.1. SR 参数 .....	175
4.17.2. sr-create .....	176
4.17.3. sr-destroy .....	176
4.17.4. sr-enable-database-replication .....	176
4.17.5. sr-disable-database-replication .....	176
4.17.6. sr-forget .....	177
4.17.7. sr-introduce .....	177
4.17.8. sr-probe .....	177
4.17.9. sr-scan .....	177
4.18. 任务命令 .....	177
4.18.1. 任务参数 .....	177

4.18.2. task-cancel .....	178
4.19. 模板命令 .....	178
4.19.1. VM 模板参数 .....	179
4.19.2. template-export .....	184
4.20. 更新命令 .....	184
4.20.1. update-upload .....	184
4.21. 用户命令 .....	184
4.21.1. user-password-change .....	184
4.22. VBD 命令 .....	184
4.22.1. VBD 参数 .....	185
4.22.2. vbd-create .....	186
4.22.3. vbd-destroy .....	186
4.22.4. vbd-eject .....	186
4.22.5. vbd-insert .....	187
4.22.6. vbd-plug .....	187
4.22.7. vbd-unplug .....	187
4.23. VDI 命令 .....	187
4.23.1. VDI 参数 .....	187
4.23.2. vdi-clone .....	188
4.23.3. vdi-copy .....	189
4.23.4. vdi-create .....	189
4.23.5. vdi-destroy .....	189
4.23.6. vdi-forget .....	189
4.23.7. vdi-import .....	189
4.23.8. vdi-introduce .....	190
4.23.9. vdi-pool-migrate .....	190
4.23.10. vdi-resize .....	190
4.23.11. vdi-snapshot .....	190
4.23.12. vdi-unlock .....	190

4.24. VIF 命令 .....	190
4.24.1. VIF 参数 .....	191
4.24.2. vif-create .....	192
4.24.3. vif-destroy .....	192
4.24.4. vif-plug .....	193
4.24.5. vif-unplug .....	193
4.25. VLAN 命令 .....	193
4.25.1. vlan-create .....	193
4.25.2. pool-vlan-create .....	193
4.25.3. vlan-destroy .....	193
4.26. VM 命令 .....	193
4.26.1. VM 选择器 .....	193
4.26.2. VM 参数 .....	194
4.26.3. vm-assert-can-be-recovered .....	199
4.26.4. vm-cd-add .....	199
4.26.5. vm-cd-eject .....	200
4.26.6. vm-cd-insert .....	200
4.26.7. vm-cd-list .....	200
4.26.8. vm-cd-remove .....	200
4.26.9. vm-clone .....	200
4.26.10. vm-compute-maximum-memory .....	200
4.26.11. vm-copy .....	201
4.26.12. vm-crashdump-list .....	201
4.26.13. vm-data-source-list .....	201
4.26.14. vm-data-source-record .....	202
4.26.15. vm-data-source-forget .....	202
4.26.16. vm-data-source-query .....	202
4.26.17. vm-destroy .....	202
4.26.18. vm-disk-add .....	202

4.26.19. vm-disk-list .....	203
4.26.20. vm-disk-remove .....	203
4.26.21. vm-export .....	203
4.26.22. vm-import .....	203
4.26.23. vm-install .....	204
4.26.24. vm-memory-shadow-multiplier-set .....	204
4.26.25. vm-migrate .....	204
4.26.26. vm-reboot .....	205
4.26.27. vm-recover .....	205
4.26.28. vm-reset-powerstate .....	205
4.26.29. vm-resume .....	206
4.26.30. vm-shutdown .....	206
4.26.31. vm-start .....	206
4.26.32. vm-suspend .....	206
4.26.33. vm-uninstall .....	206
4.26.34. vm-vcpu-hotplug .....	207
4.26.35. vm-vif-list .....	207

## 第 1 章 文档概述

本文档是 XenServer 的系统管理员指南，该产品是 Citrix 推出的完整服务器虚拟化平台。本指南所述的操作过程将指导您对 XenServer 部署进行配置。具体来说，本指南将重点介绍如何设置存储、网络连接和资源池，以及如何使用 xe 命令行接口管理 XenServer 主机。

本文档涵盖以下主题：

- 通过 Active Directory 及基于角色的访问控制来管理用户
- 创建资源池并设置高可用性
- 配置和管理存储库
- 使用动态内存控制配置虚拟机内存
- 在 XenServer 主机上设置控制域内存
- 配置网络连接
- 备份数据以及使用灾难恢复功能恢复虚拟机
- 监视 XenServer 性能指标与配置警报
- XenServer 故障排除
- 使用 XenServer xe 命令行接口

### 1.1. XenServer 简介

XenServer 是 Citrix 推出的完全服务器虚拟化平台。XenServer 软件包中包含了创建和管理在 Xen（接近本机性能的开放源代码半虚拟化虚拟机管理程序）上运行的虚拟 x86 计算机部署所需的所有内容。XenServer 已同时针对 Windows 和 Linux 虚拟服务器进行了优化。

XenServer 直接在服务器硬件上运行而不需要底层操作系统，因而是一种高效且可扩展的系统。XenServer 的工作方式是从物理机提取元素（例如硬盘驱动器、资源和端口），然后将其分配给物理机上运行的虚拟机。

虚拟机（VM）是指完全由能够运行自己的操作系统和应用程序的软件组成的计算机，就好像是一台物理机。VM 的运行方式与物理机十分相似，并且包含自己的虚拟（基于软件）CPU、RAM、硬盘和网络接口卡（NIC）。

XenServer 可用于创建 VM、生成 VM 磁盘快照以及管理 VM 工作负载。有关 XenServer 的主要功能的完整列表，请访问 [www.citrix.com/xenserver](http://www.citrix.com/xenserver)。

#### 1.1.1. 使用 XenServer 的好处

使用 XenServer 时，可以通过以下方式降低成本：

- 将多个 VM 合并到物理服务器上
- 减少需要管理的单独磁盘映像的数量
- 允许与现有网络和存储基础结构方便地集成

使用 XenServer 时，可以通过以下方式提高灵活性：

- 允许使用 XenMotion 在 XenServer 主机之间实时迁移 VM，在确保零停机时间的情况下安排维护工作
- 使用高可用性功能配置相应策略（当一个 XenServer 主机发生故障时在另一个主机上重新启动 VM），从而提高 VM 的可用性



- 将一个 VM 映像用于一系列的部署基础结构中，从而提高 VM 映像的可移植性

### 1.1.2. 管理 XenServer

可以通过两种方法管理 XenServer：XenCenter 和 XenServer 命令行界面 (CLI)。

XenCenter 是一种基于 Windows 的图形用户界面。XenCenter 允许您从 Windows 台式机管理 XenServer 主机、池和共享存储，以及部署、管理和监视 VM。

XenCenter 联机帮助是非常有用的 XenCenter 入门资源，可以从中获得上下文相关帮助。

通过 XenServer 命令行接口 (CLI)，可以使用基于 Linux 的 xe 命令来管理 XenServer。

## 1.2. XenServer 文档

此版本附带的 XenServer 文档包括：

- XenServer 发行说明介绍 XenServer 7.0 中的新增功能以及影响本版本的任何建议和已知问题。
- 《XenServer 快速启动指南》提供针对 XenServer 环境和组件的新用户的简介。本指南分步介绍要使 XenServer 和 XenCenter 管理控制台迅速启动并运行需要执行的基本安装和配置步骤。完成安装后，本指南还会演示如何创建 Windows VM、VM 模板以及 XenServer 主机池。本指南还介绍了基本管理任务和高级功能，例如共享存储、VM 快照和 XenMotion 实时迁移。
- 《XenServer 安装指南》分步介绍 XenServer 和 XenCenter 管理控制台的安装、配置及初始操作。
- 《XenServer 虚拟机用户指南》介绍如何在 XenServer 环境中安装 Windows 和 Linux VM。本指南介绍如何从安装介质、XenServer 软件包中包含的 VM 模板以及现有物理机 (P2V) 创建新 VM。本指南说明如何导入磁盘映像以及如何导入和导出设备。
- 《XenServer 管理员指南》深入介绍在配置 XenServer 部署的过程中所涉及的任务（包括设置存储、网络连接和池）。本指南介绍如何使用 xe 命令行接口管理 XenServer。
- 《vSwitch Controller 用户指南》是适用于 XenServer 的 vSwitch Controller 的综合性用户指南。
- 《增补包和 DDK》介绍 XenServer 驱动程序开发工具包，可用来修改和扩展 XenServer 的功能。
- 《XenServer 软件开发工具包指南》概要介绍 XenServer SDK。本指南中包括演示如何写入与 XenServer 主机交互的应用程序的代码示例。
- 《XenAPI 规范》是面向编程人员的 XenServer API 参考指南。

有关更多资源，请访问 [Citrix 知识中心](#)。

## 第 2 章 管理用户

通过定义用户、组、角色和权限，可以控制有权访问 XenServer 主机和池的用户及其可执行的操作。

首次安装 XenServer 时，会自动将一个用户帐户添加到 XenServer 中。此帐户是本地超级用户 (LSU) 或 root 用户，由 XenServer 计算机在本地进行身份验证。

LSU 或 root 是一个用于系统管理的特殊用户帐户，该帐户具有所有权限。在 XenServer 中，LSU 是安装时的默认帐户。LSU 仅通过 XenServer 进行身份验证，无需任何外部身份验证服务。如果外部身份验证服务失败，LSU 仍可登录并管理系统。LSU 始终可以通过 SSH 访问 XenServer 物理服务器。

您可以通过 XenCenter 的用户选项卡或 xe CLI 添加 Active Directory 帐户来创建其他用户。如果您的环境不使用 Active Directory，则您就只能使用 LSU 帐户。



### 注意

创建新用户时，XenServer 不会自动为新创建的用户帐户分配 RBAC 角色。因此，在您为这些帐户分配角色之前，它们对 XenServer 池没有任何访问权限。

这些权限通过角色授予，如[第 2.1 节“使用 Active Directory \(AD\) 验证用户”](#)中所述。

### 2.1. 使用 Active Directory (AD) 验证用户

如果您希望在服务器或池上具有多个用户帐户，则必须使用 Active Directory 用户帐户进行身份验证。这样使 XenServer 用户能够使用其 Windows 域凭据登录到池。

可以为特定用户配置不同访问权限级别的唯一方法是：启用 Active Directory 身份验证，添加用户帐户，然后为这些帐户分配角色。

Active Directory 用户可以使用 xe CLI ( 传递适当的 -u 和 -pw 参数 )，也可以使用 XenCenter 连接到该主机。身份验证基于每个资源池执行。

访问可通过使用使用者来控制。XenServer 中的一个主题对应目录服务器上的一个实体 ( 用户或组 )。启用外部身份验证后，将首先针对本地 root 用户的凭据来检查用于创建会话的凭据 ( 以防目录服务器不可用 )，然后针对使用者列表进行检查。要允许访问，您必须为授权访问的用户或组创建一个使用者条目。这可以通过使用 XenCenter 或 xe CLI 实现。

如果您熟悉 XenCenter，则请注意，XenServer CLI 提及 Active Directory 和用户帐户功能时，使用的术语略有差别：

XenCenter 术语	XenServer CLI 术语
用户	使用者
添加用户	添加使用者

## 了解 XenServer 环境中的 Active Directory 身份验证

虽然 XenServer 是基于 Linux 的，但 XenServer 允许您使用 Active Directory 帐户作为 XenServer 用户帐户。为此，XenServer 会将 Active Directory 凭据传递到 Active Directory 域控制器。

添加到 XenServer 后，Active Directory 用户和组即成为 XenServer 使用者，在 XenCenter 中通常简单地称之为用户。将使用者注册到 XenServer 后，用户/组在登录时会通过 Active Directory 进行身份验证，并且不需要用域名来限定其用户名。



### 注意

默认情况下，如果您未限定用户名（例如，输入 `mydomain\myuser` 或 `myuser@mydomain.com`），XenCenter 始终会尝试使用用户当前加入的域将用户登录到 Active Directory 身份验证服务器。LSU 帐户属例外情况，对于 LSU 帐户，XenCenter 始终首先在本地（即在 XenServer 上）对其进行身份验证。

外部身份验证工作流程如下：

1. 将连接到服务器时提供的凭据传递到 Active Directory 域控制器，进行身份验证。
2. 域控制器对凭据进行检查。如果凭据无效，身份验证立即失败。
3. 如果凭据有效，会对 Active Directory 控制器进行查询，以获取与凭据相关联的使用者标识符和组成员。
4. 如果使用者标识符与 XenServer 中存储的某个标识符相匹配，即可成功完成身份验证。

您加入域后，可以为池启用 Active Directory 身份验证。但是，池加入域后，只有该域（或与该域具有信任关系的域）中的用户才能连接到该池。



### 注意

不支持手动更新 DHCP 配置的网络 PIF 的 DNS 配置，且手动配置可能会导致 Active Directory 集成失败或停止工作，从而导致用户身份验证失败或停止工作。

## 2.1.1. 配置 Active Directory 身份验证

XenServer 支持运行 Windows 2003 或更新版本的 Active Directory 服务器。

XenServer 主机的 Active Directory 身份验证要求 Active Directory 服务器（配置为允许互操作）和 XenServer 主机使用相同的 DNS 服务器。在某些配置中，Active Directory 服务器本身可能会提供 DNS。使用 DHCP 来为 XenServer 主机提供 IP 地址和 DNS 服务器列表，或使用手动静态配置的情况下通过设置 PIF 对象的值或使用安装程序，都可以实现上述目标。

Citrix 建议您启用 DHCP 来广播主机名称。尤其不应将主机名称 `localhost` 或 `linux` 分配给主机。



### 警告

XenServer 主机名在 XenServer 部署中应唯一。

请注意以下问题：



- XenServer 使用其主机名在 AD 数据库中标记其 AD 条目。因此，如果两台 XenServer 主机具有相同的主机名，且加入相同的 AD 域，则无论这两台主机位于相同的池中还是不同的池中，第二台 XenServer 都会覆盖第一台 XenServer 的 AD 条目，因而导致第一台 XenServer 上的 AD 身份验证停止工作。

如果这两台 XenServer 主机加入不同的 AD 域，则可以使用相同的主机名。

- XenServer 主机可以位于不同的时区，因为比较的是 UTC 时间。为确保正确的同步，您可以为 XenServer 池和 Active Directory 服务器选择相同的 NTP 服务器。
- 不支持混合身份验证池（即，不能将池中的某些服务器配置为使用 Active Directory，而另一些不使用）。
- XenServer Active Directory 集成使用 Kerberos 协议与 Active Directory 服务器进行通信。因此，XenServer 不支持与未使用 Kerberos 的 Active Directory 服务器进行通信。
- 要确保使用 Active Directory 的外部身份验证成功，使 XenServer 主机上的时钟与 Active Directory 服务器上的时钟保持同步很重要。在 XenServer 加入 Active Directory 域时，将会对时钟同步进行检测，而且身份验证将在服务器时钟偏差过大时失败。



#### 警告

主机名必须由不超过 63 个字母数字字符组成，不得使用纯数字名称。

启用 Active Directory 身份验证之后，如果随后向该池添加服务器，系统会提示您在要加入池的服务器上配置 Active Directory。系统提示您在要加入的服务器上输入凭据后，输入具有足够权限的 Active Directory 凭据，以将服务器添加到该域。

Active Directory 集成。请确保以下防火墙端口对出站通信流开放，以便 XenServer 能够访问域控制器。

端口	协议	使用
53	UDP/TCP	DNS
88	UDP/TCP	Kerberos 5
123	UDP	NTP
137	UDP	NetBIOS 名称服务
139	TCP	NetBIOS 会话 (SMB)
389	UDP/TCP	LDAP
445	TCP	TCP 上的 SMB
464	UDP/TCP	计算机密码更改
3268	TCP	全局目录搜索



#### 注意

要使用 iptables 在 Linux 计算机上查看防火墙规则，请运行以下命令：iptables -nL。

**注意**

XenServer 使用 PowerBroker Identity Services (PBIS) 对 AD 服务器中的 AD 用户进行身份验证，以及加密与 AD 服务器的通信。

XenServer 如何管理用于 AD 集成的计算机帐户密码？. 与 Windows 客户机类似，PBIS 将自动更新计算机帐户密码，每 30 天续订一次，或者按 AD 服务器中的计算机帐户密码续订策略的规定进行续订。

启用池的外部身份验证：

- 可通过 XenCenter 或 CLI 使用以下命令配置使用 Active Directory 的外部身份验证。

```
xe pool-enable-external-auth auth-type=AD \
  service-name=<full-qualified-domain> \
  config:user=<username> \
  config:pass=<password>
```

指定的用户需要具有 Add/remove computer objects or workstations ( 添加/删除计算机对象或工作站 ) 权限，这是域管理员的默认权限。

**注意**

如果您未使用 Active Directory 和您的 XenServer 主机正在使用的网络上的 DHCP，则可以使用两种方法设置 DNS：

1. 设置域 DNS 后缀搜索顺序，用以解析非 FQDN：

```
xe pif-param-set uuid=<pif-uuid_in_the_dns_subnet> \
  "other-config:domain=suffix1.com suffix2.com suffix3.com"
```

2. 配置 XenServer 主机上使用的 DNS 服务器：

```
xe pif-reconfigure-ip mode=static dns=<dnshost> ip=<ip> \
  gateway=<gateway> netmask=<netmask> uuid=<uuid>
```

3. 手动设置使用 ( 与您的 DNS 服务器处于同一网络的 ) PIF 的管理接口：

```
xe host-management-reconfigure pif-uuid=<pif_in_the_dns_subnet>
```

**注意**

外部身份验证是每个主机都具有的属性。但是，Citrix 建议您按池启用和禁用此身份验证，这样，XenServer 就可以处理在对特定主机启用身份验证时出现的任何故障，并根据需要回滚所做的任何更改，从而确保整个池的配置保持一致。使用 host-param-list 命令检查主机属性，并通过检查相关字段的值确定外部身份验证的状态。

禁用外部身份验证：

- 使用 XenCenter 或下列 xe 命令可禁用外部身份验证：

```
xe pool-disable-external-auth
```

### 2.1.2. 用户身份验证

要允许用户访问您的 XenServer 主机，必须为该用户或其所在的组添加一个使用者。（仍以常规方法检查过渡组成员，例如：为组 A 添加一个使用者，其中组 A 包含组 B 且用户 1 是组 B 的成员之一，这会允许对用户 1 进行访问）。如果您要管理 Active Directory 中的用户权限，可以创建一个组，然后在该组中添加和删除用户；或者，可以根据您的身份验证需求从 XenServer 中添加和删除单个用户或用户和组的组合。主体列表可由 XenCenter 或按如下所示使用 CLI 进行管理。

对一个用户进行身份验证时，首先针对本地 root 帐户检查凭据，以支持您恢复 AD 服务器失败的系统。如果凭据（即用户名、密码）不匹配/未通过身份验证，则会向 AD 服务器提出身份验证请求；如果此请求成功，则会检索用户信息，并对照本地使用者列表验证用户信息，否则访问将被拒绝。如果用户或用户的组成员身份在使用者列表中，针对使用者列表的身份验证将成功。



#### 注意

使用 Active Directory 组授予需要主机 SSH 访问权限的池管理员用户所需的访问权限时，Active Directory 组中的用户数不得超过 500。

允许用户使用 CLI 访问 XenServer:

- 将 AD 使用者添加到 XenServer :

```
xe subject-add subject-name=<entity name>
```

实体名称应是要授予访问权限的用户或组的名称。虽然在无需消除歧义的情况下行为会相同，但您仍然可以选择包括实体的域（例如，“<xendtluser1>”而非“<user1>”）。

使用 CLI 撤消用户访问权限：

1. 查找用户的使用者标识符，即，该用户或包含该用户的组（删除一个组将删除该组中所有用户的访问权限，前提是这些用户未在使用者列表中指定）。要执行此操作，请使用 subject list 命令：

```
xe subject-list
```

此时将返回一个含有所有用户的列表。

您可能希望在列表中应用过滤器，例如，要获得 testad 域中名称为 user1 的用户的使用者标识符，可以使用下列命令：

```
xe subject-list other-config:subject-name='<testad\user1>'
```

2. 使用 subject-remove 命令删除用户，传递上一步中获得的使用者标识符：

```
xe subject-remove subject-uuid=<subject-uuid>
```

3. 您可能希望终止此用户已进行身份验证的所有当前会话。有关终止会话的更多信息,请参阅[使用 xe 终止所有已经过身份验证的会话](#)和[使用 xe 终止单个用户的会话](#)。如果不终止会话，已被撤销权限的用户可以在注销前继续访问系统。

列出有访问权限的使用者:

- 要识别具有访问 XenServer 主机或池权限的用户和组的列表，请使用下列命令：

```
xe subject-list
```

### 2.1.3. 删除用户的访问权限

用户通过身份验证之后，即有权访问服务器，直至用户自己终止其会话或其他用户终止其会话。从主体列表中删除一个用户或从主体列表中的某个组删除用户不会自动撤销该用户已经身份验证的会话，这意味着用户可以继续使用 XenCenter 或其他已创建的 API 会话访问池。为强制终止这些会话，XenCenter 和 CLI 提供终止单个或所有当前活动会话的工具。请参阅 XenCenter 帮助，了解使用 XenCenter 的步骤，或以下使用 CLI 的步骤。

使用 xe 终止所有已经过身份验证的会话:

- 执行以下 CLI 命令：

```
xe session-subject-identifier-logout-all
```

使用 xe 终止单个用户的会话:

1. 确定要注销的会话的使用者标识符。使用 session-subject-identifier-list 或 subject-list xe 命令查找该标识符（第一个命令显示具有会话的用户，第二个命令显示所有用户，但可以使用诸如 xe subject-list other-config:subject-name=xendtl\user1 这样的命令来过滤 – 视所用 shell 而定，您可能需要使用如例中所示的双反斜杠）。
2. 使用 session-subject-logout 命令，将上一步确定的使用者标识符作为参数传递，例如：

```
xe session-subject-logout subject-identifier=<subject-id>
```

## 2.1.4. 退出 AD 域



### 警告

退出域（即，禁用 Active Directory 身份验证，并断开池或服务器与其域之间的连接）时，通过 Active Directory 凭据进行身份验证以连接到池或服务器的所有用户均会断开连接。

使用 XenCenter 退出 AD 域。有关详细信息，请参阅 XenCenter 帮助。也可以运行 pool-disable-external-auth 命令，在需要的情况下指定池 uuid。



### 注意

退出域不会使主机对象从 AD 数据库删除。要了解相关的详细信息以及如何删除禁用的主机实体的信息，请参阅[此](#)知识库文章。

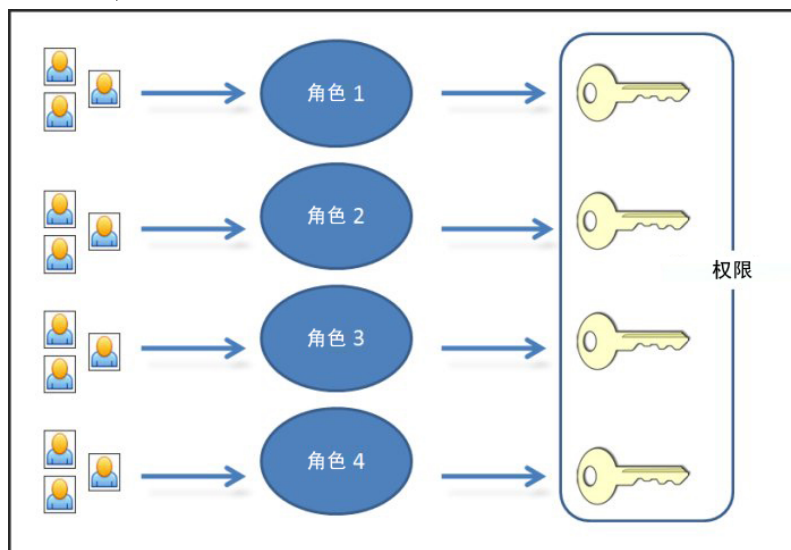
## 2.2. 基于角色的访问控制

通过 XenServer 基于角色的访问控制（RBAC），您可以分配用户、角色和权限，以控制有权访问您的 XenServer 的用户及其可执行的操作。XenServer RBAC 系统将用户（或用户组）映射到定义的角色（权限的命名集），随后这些角色将具有相关联的 XenServer 权限（执行特定操作的能力）。

由于不会为用户直接分配权限，而是通过为其分配的角色来获取权限，因此，要管理单个用户的权限，只需将用户分配到适当的角色即可；这样便简化了常规操作。XenServer 会维护授权用户及其角色的列表。

通过 RBAC，您可以轻松限制不同的用户组可以执行的操作，从而降低了缺乏经验的用户造成事故的可能性。

为便于开展合规和审核工作，RBAC 还提供了“审核日志”功能。



RBAC 基于 Active Directory 提供身份验证服务。具体而言，XenServer 会保留基于 Active Directory 用户和组帐户的授权用户列表。因此，您必须将池加入域并添加 Active Directory 帐户，然后才能分配角色。

本地超级用户 (LSU) 或 root 用户是一个用于系统管理的特殊用户帐户，具有完全权限。在 XenServer 中，本地超级用户是安装时的默认帐户。LSU 通过 XenServer 而非外部身份验证服务进行身份验证，因此，如果外部身份验证服务失败，LSU 仍可登录并管理系统。LSU 始终可通过 SSH 访问 XenServer 物理主机。

## RBAC 流程

实施 RBAC 并为用户或组分配角色的标准流程如下：

1. 加入域。请参阅 [启用池的外部身份验证](#)。
2. 将 Active Directory 用户或组添加到池。该用户或组将成为使用者。请参阅第 2.2.3.3 节“[将使用者添加到 RBAC](#)”。
3. 分配（或修改）使用者的 RBAC 角色。请参阅第 2.2.3.4 节“[为创建的使用者分配 RBAC 角色](#)”。

### 2.2.1. 角色

XenServer 出厂时具有以下六种预先设立的角色：

- 池管理员 - 与本地 root 用户相同。可执行所有操作。



#### 注意

本地超级用户（用户名为 root）将始终具有“池管理员”角色。池管理员角色与本地 root 用户具有相同的权限。

- 池操作员 - 可执行除添加/删除用户及修改其角色以外的所有操作。此角色主要管理主机和池（即创建存储、构建池、管理主机等）。
- 虚拟机超级管理员（VM 超级管理员）- 创建和管理虚拟机。此角色主要配置 VM 操作员所使用的 VM。
- 虚拟机管理员（VM 管理员）- 与 VM 超级管理员类似，但不能迁移 VM 或执行快照。
- 虚拟机操作员（VM 操作员）- 与 VM 管理员类似，但不能创建/销毁 VM，可执行启动/停止生命周期操作。
- 只读角色 - 可查看资源池和性能数据。



#### 注意

您无法在此版本的 XenServer 中添加、删除或修改角色。



#### 警告

如果您希望 AD 组的用户具有 SSH 访问权限，则不能将池管理员角色分配给成员数超过 500 的 AD 组。

有关每个角色可以具有的权限汇总以及每种权限可执行的操作的详细信息，请参阅第 2.2.2 节“[RBAC 角色和权限定义](#)”。

需要为所有 XenServer 用户分配相应的角色。默认情况下，将为所有新用户分配池管理员角色。可以为一个用户分配多个角色；在这种情况下，该用户将具有为其分配的所有角色的所有权限。

可以通过以下两种方式更改用户的角色：

1. 修改使用者 -> 角色映射（这需要分配/修改角色权限，仅适用于池管理员）。

2. 在 Active Directory 中修改包含组成员的用户角色。

### 2.2.2. RBAC 角色和权限定义

下表汇总了每个角色适用的权限。有关每种权限可执行的操作的详细信息，请参阅“权限定义”。

表 2.1. 适用于每个角色的权限:

角色权限	池管理员	池操作员	VM 超级管理员	VM 管理员	VM 操作员	只读
分配/修改角色	X					
登录到 (物理) 服务器控制台 (通过 SSH 和 XenCenter)	X					
服务器备份/还原	X					
导入/导出 OVF/OVA 包和磁盘映像	X					
使用 XenServer Conversion Manager 转换虚拟机	X					
注销活动的用户连接	X	X				
创建和取消警报	X	X				
取消任何用户的任务	X	X				
池管理	X	X				
交换机端口锁定	X	X				
VM 高级操作	X	X	X			
VM 创建/销毁操作	X	X	X	X		
VM 更改 CD 介质	X	X	X	X	X	

角色权限	池管理员	池操作员	VM 超级管理员	VM 管理员	VM 操作员	只读
查看 VM 控制台	X	X	X	X	X	
XenCenter 视图管理操作	X	X	X	X	X	
取消自己的任务	X	X	X	X	X	X
阅读审核日志	X	X	X	X	X	X
连接到池并阅读所有池元数据	X	X	X	X	X	X

## 权限的定义

下表提供了关于权限的其他详细信息：

表 2.2. 权限的定义：

权限	允许被授权人执行的操作	说明/备注
分配/修改角色	<ul style="list-style-type: none"> <li>添加/删除用户</li> <li>添加/删除用户的角色</li> <li>启用和禁用 Active Directory 集成 ( 加入域 )</li> </ul>	<p>此权限允许用户向自己授予任何权限或执行任何任务。</p> <p>警告：此角色允许用户禁用 Active Directory 集成以及从 Active Directory 添加的所有主体。</p>
登录到服务器控制台	<ul style="list-style-type: none"> <li>通过 SSH 访问服务器控制台</li> <li>通过 XenCenter 访问服务器控制台</li> </ul>	警告：具备对 root shell 的访问权限之后，被授权人可以随意重新配置整个系统 ( 包括 RBAC )。
服务器备份/还原 VM 创建/销毁操作	<ul style="list-style-type: none"> <li>备份和还原服务器</li> <li>备份和还原池元数据</li> </ul>	还原备份的能力使被授权人能够还原 RBAC 配置更改。
导入/导出 OVF/OVA 包和磁盘映像	<ul style="list-style-type: none"> <li>导入 OVF 和 OVA 包</li> <li>导入磁盘映像</li> <li>将 VM 导出为 OVF/OVA 包</li> </ul>	
注销活动的用户连接	<ul style="list-style-type: none"> <li>断开已登录用户的连接的能力</li> </ul>	



权限	允许被授权人执行的操作	说明/备注
创建/取消警报		<p>警告：具有此权限的用户可以取消整个池的警报。</p> <p>注意：查看警报的能力属于“连接到池并读取所有池元数据”权限的一部分。</p>
取消任何用户的任务	<ul style="list-style-type: none"> <li>取消任何用户正在运行的任务</li> </ul>	此权限允许用户请求 XenServer 取消任何用户启动的正在执行的任务。
池管理	<ul style="list-style-type: none"> <li>设置池属性（命名、默认 SR）</li> <li>启用、禁用和配置高可用性功能</li> <li>设置每个 VM 高可用性功能重新启动优先级</li> <li>在池中添加和删除服务器</li> <li>紧急转换到主服务器</li> <li>紧急主服务器地址</li> <li>紧急恢复从属服务器</li> <li>指定新的主服务器</li> <li>管理池和服务证书</li> <li>修补</li> <li>设置服务器属性</li> <li>配置服务器日志记录</li> <li>启用和禁用服务器</li> <li>关闭、重新引导和打开服务器</li> <li>系统状态报告</li> <li>应用许可证</li> <li>在使用维护模式或 HA 功能时将服务器上的所有其他 VM 实时迁移到其他服务器</li> <li>配置服务器管理接口和辅助接口</li> <li>禁用服务器管理</li> <li>删除故障转储</li> <li>添加、编辑和删除网络</li> <li>添加、编辑和删除 PBD/PIF/VLAN/Bond/SR</li> <li>添加、删除和检索机密信息</li> </ul>	<p>此权限包括维护池需执行的所有操作。</p> <p>注意：如果管理接口无法使用，则除本地 root 登录外，其他登录均无法通过身份验证。</p>



权限	允许被授权人执行的操作	说明/备注
VM 高级操作	<ul style="list-style-type: none"> <li>• 调整 VM 内存 ( 通过动态内存控制 )</li> <li>• 创建包含内存数据的 VM 快照、生成 VM 快照及回滚 VM</li> <li>• 迁移 VM</li> <li>• 启动 VM , 包括指定物理服务器</li> <li>• 恢复 VM</li> </ul>	此权限向被授权人提供了足够的权限, 使其能够在对所选的 XenServer 服务器不满意时在其他服务器上启动 VM。
VM 创建/销毁操作	<ul style="list-style-type: none"> <li>• 安装或删除</li> <li>• 克隆 VM</li> <li>• 添加、删除和配置虚拟磁盘/ CD 设备</li> <li>• 添加、删除和配置虚拟网络设备</li> <li>• 导入/导出 VM</li> <li>• VM 配置更改</li> </ul>	
VM 更改 CD 介质	<ul style="list-style-type: none"> <li>• 弹出当前的 CD</li> <li>• 插入新 CD</li> </ul>	
VM 更改电源状态	<ul style="list-style-type: none"> <li>• 启动 VM ( 自动放置 )</li> <li>• 关闭 VM</li> <li>• 重新引导 VM</li> <li>• 挂起 VM</li> <li>• 恢复 VM ( 自动放置 )</li> </ul>	此权限不包括启动、恢复和迁移, 这三种权限属于 VM 高级操作权限。
查看 VM 控制台	<ul style="list-style-type: none"> <li>• 查看 VM 控制台并与之交互</li> </ul>	此权限不允许用户查看服务器控制台。
XenCenter 视图管理操作	<ul style="list-style-type: none"> <li>• 创建和修改全局 XenCenter 文件夹</li> <li>• 创建和修改全局 XenCenter 自定义字段</li> <li>• 创建和修改全局 XenCenter 搜索</li> </ul>	文件夹、自定义字段和搜索在访问池的所有用户间共享
取消自己的任务	<ul style="list-style-type: none"> <li>• 允许用户取消自己的任务</li> </ul>	
阅读审核日志	<ul style="list-style-type: none"> <li>• 下载 XenServer 审核日志</li> </ul>	

权限	允许被授权人执行的操作	说明/备注
连接到池并阅读所有池元数据	<ul style="list-style-type: none"> <li>• 登录到池</li> <li>• 查看池元数据</li> <li>• 查看历史性能数据</li> <li>• 查看登录的用户</li> <li>• 查看用户和角色</li> <li>• 查看消息</li> <li>• 注册和接收事件</li> </ul>	



### 注意

在某些情况下，具有只读权限的用户无法将资源移动到 XenCenter 中的文件夹，即使在接收到提示并提供了更具特权的用户的凭据之后也是如此。在这种情况下，以更具特权的用户身份登录到 XenCenter 并重试该操作。

## 2.2.3. 结合使用 RBAC 和 CLI

### 2.2.3.1. 列出 XenServer 中所有可用的已定义角色

- 运行以下命令：xe role-list

此命令返回当前已定义角色的列表，例如：

```

uuid( RO): 0165f154-ba3e-034e-6b27-5d271af109ba
name ( RO): pool-admin
description ( RO): The Pool Administrator role has full access to all
features and settings, including accessing Dom0 and managing subjects,
roles and external authentication

uuid ( RO): b9ce9791-0604-50cd-0649-09b3284c7dfd
name ( RO): pool-operator
description ( RO): The Pool Operator role manages host- and pool-wide resources,
including setting up storage, creating resource pools and managing patches, and
high availability (HA).

uuid( RO): 7955168d-7bec-10ed-105f-c6a7e6e63249
name ( RO): vm-power-admin
description ( RO): The VM Power Administrator role has full access to VM and
template management and can choose where to start VMs and use the dynamic memory
control and VM snapshot features

uuid ( RO): aaa00ab5-7340-bfbc-0d1b-7cf342639a6e
name ( RO): vm-admin
description ( RO): The VM Administrator role can manage VMs and templates

uuid ( RO): fb8d4ff9-310c-a959-0613-54101535d3d5
name ( RO): vm-operator
description ( RO): The VM Operator role can use VMs and interact with VM consoles

uuid ( RO): 7233b8e3-eacb-d7da-2c95-f2e581cdbf4e
name ( RO): read-only
description ( RO): The Read-Only role can log in with basic read-only access

```

**注意**

此角色列表是静态列表；无法在其中添加、删除或修改角色。

**2.2.3.2. 显示当前使用者的列表：**

- 运行命令 `xe subject-list`

此命令将返回 XenServer 用户、其 uuid 及其相关联的角色的列表：

```
uuid ( RO): bb6dd239-1fa9-a06b-a497-3be28b8dca44
subject-identifier ( RO): S-1-5-21-1539997073-1618981536-2562117463-2244
other-config (MRO): subject-name: example01\user_vm_admin; subject-upn: \
  user_vm_admin@XENDT.NET; subject-uid: 1823475908; subject-gid: 1823474177; \
  subject-sid: S-1-5-21-1539997073-1618981536-2562117463-2244; subject-gecos: \
  user_vm_admin; subject-displayname: user_vm_admin; subject-is-group: false; \
  subject-account-disabled: false; subject-account-expired: false; \
  subject-account-locked: false; subject-password-expired: false
roles (SRO): vm-admin

uuid ( RO): 4fe89a50-6a1a-d9dd-afb9-b554cd00c01a
subject-identifier ( RO): S-1-5-21-1539997073-1618981536-2562117463-2245
other-config (MRO): subject-name: example02\user_vm_op; subject-upn: \
  user_vm_op@XENDT.NET; subject-uid: 1823475909; subject-gid: 1823474177; \
  subject-sid: S-1-5-21-1539997073-1618981536-2562117463-2245; \
  subject-gecos: user_vm_op; subject-displayname: user_vm_op; \
  subject-is-group: false; subject-account-disabled: false; \
  subject-account-expired: false; subject-account-locked: \
  false; subject-password-expired: false
roles (SRO): vm-operator

uuid ( RO): 8a63fbf0-9ef4-4fef-b4a5-b42984c27267
subject-identifier ( RO): S-1-5-21-1539997073-1618981536-2562117463-2242
other-config (MRO): subject-name: example03\user_pool_op; \
  subject-upn: user_pool_op@XENDT.NET; subject-uid: 1823475906; \
  subject-gid: 1823474177; subject-s id:
  S-1-5-21-1539997073-1618981536-2562117463-2242; \
  subject-gecos: user_pool_op; subject-displayname: user_pool_op; \
  subject-is-group: false; subject-account-disabled: false; \
  subject-account-expired: false; subject-account-locked: \
  false; subject-password-expired: false
roles (SRO): pool-operator
```

**2.2.3.3. 将使用者添加到 RBAC**

为使现有的 AD 用户能够使用 RBAC，您将需要在 XenServer 中直接为 AD 用户或为 AD 用户的其中一个包含组创建主体实例：

1. 运行命令 `xe subject-add subject-name=<AD user/group>`

此命令会添加一个新的使用者实例。

**2.2.3.4. 为创建的使用者分配 RBAC 角色**

添加使用者之后，您可以为其分配 RBAC 角色。可以通过角色的 uuid 或名称指向该角色：

1. 运行以下命令：

```
xe subject-role-add uuid=<subject uuid> role-uuid=<role uuid>
```

或

```
xe subject-role-add uuid=<subject uuid> role-name=<role_name>
```

例如，以下命令会将 uuid 为 b9b3d03b-3d10-79d3-8ed7-a782c5ea13b4 的使用者添加到池管理员角色：

```
xe subject-role-add uuid=b9b3d03b-3d10-79d3-8ed7-a782c5ea13b4 role-name=pool-admin
```

### 2.2.3.5. 更改使用者的 RBAC 角色：

要更改用户的角色，需要将用户从现有角色中删除，然后再将其添加到新角色：

1. 运行以下命令：

```
xe subject-role-remove uuid=<subject uuid> role-name= \
  <role_name_to_remove>
xe subject-role-add uuid=<subject uuid > role-name= \
  <role_name_to_add>
```

为确保新角色生效，应将用户注销，然后再次登录（这需要具有“注销活动用户连接”权限，池管理员或池操作员具有该权限）。



#### 警告

添加或删除池管理员使用者之后，可能会有几秒钟的延迟，以便池的所有主机都能接受与此使用者相关联的 SSH 会话。

### 2.2.4. 审核

RBAC 审核日志将记录登录的用户执行的所有操作。

- 该消息将详细记录与调用该操作的会话相关联的使用者 ID 和用户名。
- 如果在使用者没有相关授权的情况下调用操作，RBAC 审核日志会记录这一情况。
- 如果操作成功，RBAC 审核日志会记录这一情况；如果操作失败，RBAC 审核日志会记录错误代码。

#### 2.2.4.1. 审核日志 xe CLI 命令

```
xe audit-log-get [since=<timestamp>] filename=<output filename>
```

此命令将池中 RBAC 审核文件的所有可用记录下载到某个文件中。如果可选参数“since”存在，该命令将仅下载自该特定时间点开始的记录。

#### 2.2.4.2. 获取池中的所有审核记录

运行以下命令：

```
xe audit-log-get filename=/tmp/auditlog-pool-actions.out
```

#### 2.2.4.3. 获取自精确到毫秒的时间戳开始的池审核记录

运行以下命令：

```
xe audit-log-get since=2009-09-24T17:56:20.530Z \
filename=/tmp/auditlog-pool-actions.out
```

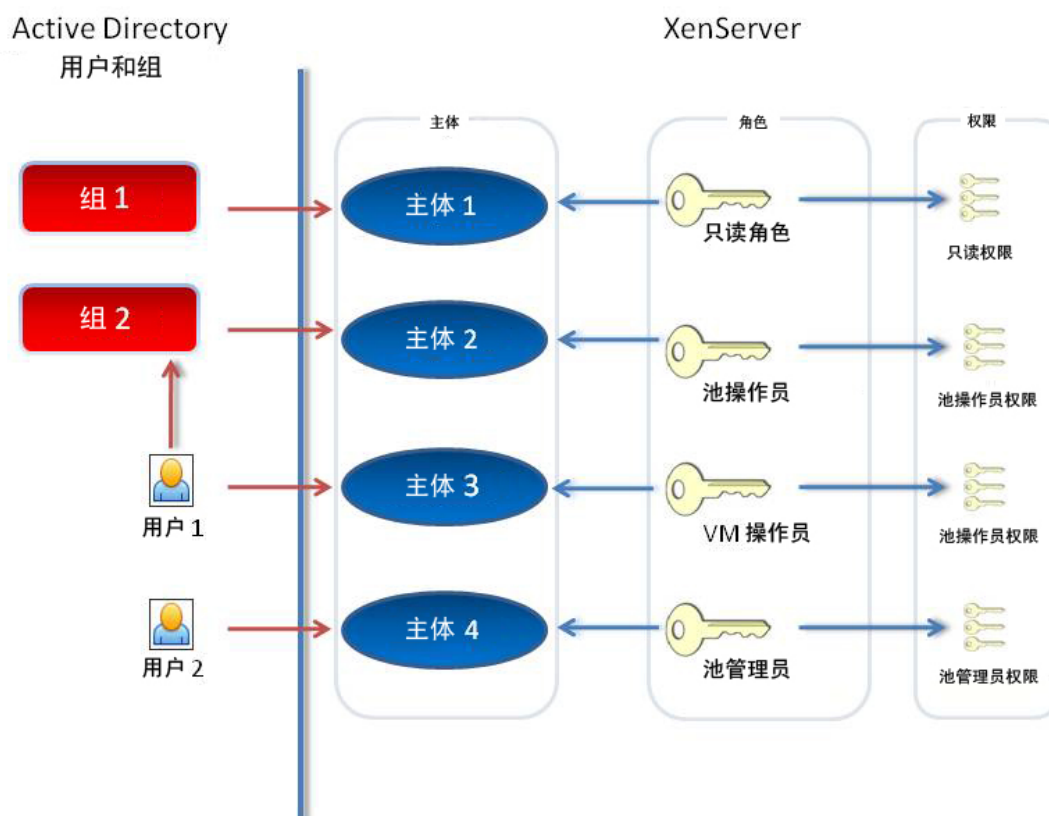
#### 2.2.4.4. 获取自精确到分钟的时间戳开始的池审核记录

运行以下命令：

```
xe audit-log-get since=2009-09-24T17:56Z \
filename=/tmp/auditlog-pool-actions.out
```

#### 2.2.5. XenServer 如何计算会话的角色？

1. 使用者通过 Active Directory 服务器进行身份验证，以确定该使用者还可能属于哪些包含组。
2. XenServer 接下来会确定已将哪些角色同时分配给该主体及其包含组。
3. 由于使用者可以是多个 Active Directory 组的成员，因此，使用者会继承关联角色的所有权限。



在上图中，由于使用者 2 (组 2) 是池操作员，而用户 1 是组 2 的成员，因此，使用者 3 (用户 1) 尝试登录时，会同时继承使用者 3 (VM 操作员) 和组 2 (池操作员) 角色。由于池操作员角色级别更高，因此，使用者 3 (用户 1) 的最终角色是池操作员而非 VM 操作员。

## 第 3 章 XenServer 主机和资源池

本章通过一系列示例介绍如何使用 `xe` 命令行接口 (CLI) 创建资源池。下面将给出一个简单的基于 NFS 的共享存储配置，并讨论多个简单的 VM 管理示例。还将介绍处理物理节点故障的过程。

### 3.1. 主机和资源池概述

资源池包括多个 XenServer 主机安装，这些主机安装绑定在一起形成可以承载虚拟机的单一托管实体。与共享存储组合后，资源池允许 VM 在具有足够内存的任何 XenServer 主机上启动；并允许 VM 在保持运行状态（停机时间极短）的情况下在 XenServer 主机之间动态移动 (XenMotion)。如果单个 XenServer 主机发生硬件故障，则管理员可以在同一资源池中的另一个 XenServer 主机上重新启动出现故障的 VM。如果在资源池上启用了高可用性 (HA)，则 VM 会在其主机发生故障时自动移动。每个资源池最多支持 16 个主机（尽管此限制并不强制执行）。

池总是至少包含一个物理节点，称为主节点。只有主节点才会公开（XenCenter 和 XenServer 命令行接口 [称为 `xe` CLI] 使用的）管理接口；主节点会根据需要向各个成员转发命令。



#### 注意

如果池的主节点出现故障，只有在启用了高可用性的情况下才会重新选择主节点。

### 3.2. 创建资源池的要求

资源池是一台或多台同类 [第 3.4 节“创建异构资源池”](#) 主机（或受限制的异构池，请参阅 XenServer）的聚合，每个池最多包含 16 台主机。同类主机的定义是：

- 要加入池的服务器上的 CPU 与池中已有服务器上的 CPU 相同（在供应商、型号和功能方面）。
- 要加入池的服务器运行的 XenServer 软件版本在修补程序级别与池中已有的服务器相同。

向池中加入服务器时，软件将实施附加限制，尤其是在以下情况下：

- 不是现有资源池的成员
- 未配置任何共享存储
- 要加入的 XenServer 主机中不存在任何运行中的 VM 或挂起的 VM
- VM 上不存在任何正在进行中的活动操作，例如关闭操作

您还必须确保要加入池的主机的时钟与池主节点同步（例如通过使用 NTP），同时其管理界面是非绑定的（可以在该主机成功加入池后进行配置）而且其管理 IP 地址是静态的（可以在主机上配置或使用 DHCP 服务器上适当的配置）。

资源池中的 XenServer 主机可以包括不同数量的物理网络接口和不同大小的本地存储库。实际上，由于通常很难实现多个服务器使用完全相同的 CPU，因此微小差异是允许的。如果您确信您的环境可以接受具有不同 CPU 的主机加入同一资源池，则加入池的操作可以通过传递 `--force` 参数强制执行。



#### 注意

资源池中的 XenServer 主机需要一个静态 IP 地址，这一要求同样适用于为池提供共享 NFS 或 iSCSI 存储的主机。

对创建资源池来说，虽然技术上并不严格要求池具有一个或多个共享存储库，但池的优势（例如，动态选择要运行 VM 的 XenServer 主机的能力，以及在 XenServer 主机之间动态移动 VM 的能力）只有在池具有一个或多个共享存储库时才会体现出来。如果可能，推迟 XenServer 主机的创建，直到共享存储可用。添

加共享存储后，Citrix 建议您将磁盘位于本地存储的现有 VM 移动到共享存储中。使用 `xe vm-copy` 命令或者使用 XenCenter 可实现此操作。

### 3.3. 创建资源池

可使用 XenCenter 管理控制台或 CLI 创建资源池。将新主机加入到资源池中时，加入的主机将其本地数据库与池范围内的数据库同步，并从池继承某些设置：

- 将 VM、本地和远程存储配置添加到池范围内的数据库中。所有这些内容仍然绑定到池中加入的主机，除非您在加入完成后采取明确措施使资源共享。
- 加入的主机继承池中现有的共享存储库并创建相应的 PBD 记录，因此新主机可以自动访问现有共享存储库。
- 新加入池的主机会部分继承网络信息：全部继承 NIC、VLAN 和绑定接口的结构详细信息，但不包括策略信息。策略信息必须重新配置，包括：
  - 管理 NIC 的 IP 地址，这类地址通过原始配置保留。
  - 管理接口的位置，保持与原始配置相同。例如，如果其他池主机的管理接口位于绑定的接口，则加入的主机在加入后必须立即显式迁移到绑定。
  - 专用存储 NIC，必须通过 XenCenter 或 CLI 为新加入的主机重新分配，并相应地重新插入 PBD 以路由通信。这是因为加入到池的操作中并不包含分配 IP 地址这一步骤，而没有正确配置 IP 地址，存储 NIC 就没有用处。有关如何通过 CLI 指定专用存储 NIC 的详细信息，请参阅第 4.4.7 节“配置专用存储 NIC”。

使用 CLI 将 XenServer 主机 host1 和 host2 加入到资源池：

1. 在 XenServer 主机 host2 中打开控制台。
2. 运行以下命令，指示 XenServer 主机 host2 加入位于 XenServer 主机 host1 上的池中：

```
xe pool-join master-address=<host1> master-username=<administrators_username> \
master-password=<password>
```

master-address 必须设置为 XenServer 主机 host1 的完全限定域名，password 必须是安装 XenServer 主机 host1 时设置的管理员密码。

命名资源池：

- 默认情况下，XenServer 主机属于未命名池。要创建您的第一个资源池，请重命名现有的无名称的池。使用 `tab-complete` 查找 `pool_uuid`：

```
xe pool-param-set name-label=<"New Pool"> uuid=<pool_uuid>
```



### 3.4. 创建异构资源池

XenServer 允许将完全不同的主机硬件加入资源池（称为异构资源池），从而简化了不断扩展部署的过程。异构资源池利用可提供 CPU“屏蔽”或“调配”的 Intel (FlexMigration) 和 AMD (Extended Migration) CPU 中的技术来实现。通过这些功能，可以将 CPU 配置为看起来提供与实际不同的样式、型号或功能。这样，将可以创建异构池，尽管这些池具有完全不同的 CPU，但仍能安全地支持实时迁移。



#### 注意

要加入异构池的 XenServer 的 CPU 的供应商（即 AMD、Intel）必须与池中已有服务器相同，但具体类型（系列、型号和步进编号）无需相同。

XenServer 7.0 简化了异构池支持功能。在以前版本的 XenServer 中，如果池的新成员的功能集与池的功能集不同，用户必须先对该新成员的 CPU 执行屏蔽操作，然后才能将其加入池。现在，不论主机的底层 CPU 属于何种类型，都可以将其加入到现有资源池（只要 CPU 来自同一个供应商系列）。在以下情况下，每次都会对池功能集进行动态计算：

- 新主机加入池
- 池成员离开池
- 池成员在重新启动之后重新连接

池功能集发生任何变化都不会影响当前在池中运行的 VM。正在运行的 VM 将继续使用启动时所应用的功能集。此功能集在启动时确定，而且在历经迁移、挂起和恢复操作后一直保持不变。如果功能有所欠缺的主机加入池，导致池的功能水平下降，正在运行的 VM 可以迁移到池中除新添加的主机以外的任何主机。尝试在池中或跨池将 VM 移动或迁移到另一台主机时，XenServer 会执行迁移检查，比较 VM 的功能集与目标主机的功能集。如果发现两者的功能集兼容，则允许迁移 VM。如此便可在池中和跨池自由移动 VM，而不论 VM 正在使用何种 CPU 功能。如果使用 Workload Balancing (WLB) 为要迁移的 VM 选择最佳目标主机，则不建议将功能集不兼容的主机作为目标主机。

### 3.5. 添加共享存储

有关支持的共享存储类型的完整列表，请参阅“[存储](#)”一章。本部分说明如何在现有 NFS 服务器中创建共享存储（表示为存储库）。

使用 CLI 将 NFS 共享存储添加到资源池：

1. 在池中任意 XenServer 主机上打开控制台。
2. 通过运行以下命令在 `<server:/path>` 上创建存储库：

```
xe sr-create content-type=user type=nfs name-label=<"Example SR"> shared=true \
  device-config:server=<server> \
  device-config:serverpath=<path>
```

device-config:server 指 NFS 服务器的主机名，device-config:serverpath 指 NFS 服务器上的路径。如果 shared 设置为 true，则共享存储将自动连接到池中的每台 XenServer 主机，并且随后加入的任何 XenServer 主机也会连接到该存储。已创建存储库的全局唯一标识符 (UUID) 将显示在屏幕上。

3. 通过以下命令查找池的 UUID：

```
xe pool-list
```

4. 使用以下命令将共享存储设置为池范围内的默认值：

```
xe pool-param-set uuid=<pool_uuid> default-SR=<sr_uuid>
```

由于共享存储已设置为池范围内的默认共享存储，所以默认情况下，将来的所有 VM 都会在共享存储上创建自己的磁盘。有关创建其他类型的共享存储的信息，请参阅第 5 章 [存储](#)。



### 3.6. 从资源池移除 XenServer 主机



#### 注意

在从池中移除 XenServer 主机之前，请确保关闭该主机上正在运行的所有 VM。否则，您可能会看到一条警告消息，指示无法移除该主机。

从池中移除（删除）XenServer 主机时，机器将重新引导、重新初始化，最终达到的状态等效于全新安装后的状态。如果本地磁盘中有重要数据，一定不要错过从池中删除 XenServer 主机。

使用 CLI 从资源池移除主机：

1. 在池中任意主机上打开控制台。
2. 运行以下命令，获取主机的 UUID：

```
xe host-list
```

3. 从池中删除所需的主机：

```
xe pool-eject host-uuid=<host_uuid>
```

XenServer 主机将删除并最终达到全新安装状态。



#### 警告

如果主机中包含存储在本地磁盘中的重要数据，请不要从资源池删除该主机。从池中删除后，将清除所有数据。如果要保留这些数据，请先使用 XenCenter 或 `xe vm-copy` CLI 命令将 VM 复制到池中的共享存储。

从池中删除包含本地存储的 VM 的 XenServer 主机时，这些 VM 仍将在池数据库中显示，并且可以被其他 XenServer 主机看到。但这些 VM 不能启动，除非将其关联的虚拟磁盘更改为指向能被池中其他 XenServer 主机看到的共享存储或直接删除。因此，强烈建议您在完成加入池的操作后立即将任何本地存储移动到共享存储，以便删除单个 XenServer 主机（或其发生物理故障）时不会丢失数据。

### 3.7. 为进行维护准备 XenServer 主机池

在资源池的某台 XenServer 主机上执行维护操作之前，应该先禁用该主机（这可以阻止在此主机上启动任何 VM），然后将其 VM 迁移到该池中的另一台 XenServer 主机上。通过使用 XenServer 将 XenCenter 主机置于维护模式，可以非常轻松地完成此操作。有关详细信息，请参阅 XenCenter 帮助。



#### 注意

如果将主服务器主机置于维护模式，则脱机 VM 的最后 24 小时 RRD 更新将会丢失。这是因为备份同步每 24 小时进行一次。



#### 警告

Citrix 强烈建议先重新引导所有 XenServer，再安装更新，然后再验证配置。这是因为有些配置更改只有在重新引导 XenServer 后才会生效，因此，重新引导可以发现可能导致更新失败的配置问题。

使用 CLI 准备池中的 XenServer 主机以执行维护操作：

1. 运行以下命令：

```
xe host-disable uuid=<xenserver_host_uuid>
xe host-evacuate uuid=<xenserver_host_uuid>
```

这将禁用 XenServer 主机，然后将任何正在运行的 VM 迁移到池中的其他 XenServer 主机上。

2. 执行所需的维护操作。
3. 维护操作完成后，启用 XenServer 主机：

```
xe host-enable
```

重新启动所有已停止的 VM 并/或恢复所有挂起的 VM。

## 3.8. 导出资源池数据



### 注意

导出资源池数据功能适用于 XenServer Enterprise Edition 客户或者能够通过 XenApp/XenDesktop 授权访问 XenServer 的客户。要了解有关 XenServer 版本以及升级方式的详细信息，请单击[此处](#)访问 Citrix Web 站点。有关许可的详细信息，请参阅 [XenServer 7.0 许可常见问题解答](#)。

使用导出资源数据选项可为您的池生成一份资源数据报告，并可将该报告导出为 .xls 或 .csv 文件。此报告提供有关池内各种资源（例如服务器、网络、存储、虚拟机、VDI 和 GPU）的详细信息。该功能允许管理员根据各种工作负载（例如 CPU、存储和网络）跟踪、计划和分配资源。

下表提供了报告内包含的资源和各种类型资源数据的列表：

资源	资源数据
服务器	<ul style="list-style-type: none"> <li>• 名称</li> <li>• 池主服务器</li> <li>• UUID</li> <li>• 地址</li> <li>• CPU 使用率</li> <li>• 网络（平均/最大 KB/秒）</li> <li>• 已用内存</li> <li>• 存储</li> <li>• 运行时间</li> <li>• 说明</li> </ul>
网络	<ul style="list-style-type: none"> <li>• 名称</li> <li>• 链接状态</li> <li>• MAC</li> <li>• MTU</li> <li>• VLAN</li> <li>• 类型</li> <li>• 位置</li> </ul>

资源	资源数据
VDI Edition	<ul style="list-style-type: none"> <li>• 名称</li> <li>• 类型</li> <li>• UUID</li> <li>• 大小</li> <li>• 存储</li> <li>• 说明</li> </ul>
存储	<ul style="list-style-type: none"> <li>• 名称</li> <li>• 类型</li> <li>• UUID</li> <li>• 大小</li> <li>• 位置</li> <li>• 说明</li> </ul>
VM	<ul style="list-style-type: none"> <li>• 名称</li> <li>• 电源状态</li> <li>• 运行平台</li> <li>• 地址</li> <li>• MAC</li> <li>• NIC</li> <li>• 操作系统</li> <li>• 存储</li> <li>• 已用内存</li> <li>• CPU 使用率</li> <li>• UUID</li> <li>• 运行时间</li> <li>• 模板</li> <li>• 说明</li> </ul>

资源	资源数据
GPU	<p>注意：仅当有 GPU 与您的 XenServer 主机连接时才有 GPU 的信息。</p> <ul style="list-style-type: none"> <li>• 名称</li> <li>• 服务器</li> <li>• PCI 总线路径</li> <li>• UUID</li> <li>• 电源使用</li> <li>• 温度</li> <li>• 已用内存</li> <li>• 计算机利用率</li> </ul>

### 3.8.1. 导出资源数据

1. 在 XenCenter“导航”窗格中，单击基础结构，然后单击相应的池
2. 单击池菜单，然后选择导出资源数据。
3. 浏览到您希望保存报告的位置，然后单击保存。

## 3.9. 高可用性

### 3.9.1. 高可用性概述

高可用性是一组设计用于为导致 XenServer 主机出现故障或使其无法访问的问题做出规划以及从这些问题中安全恢复的自动功能。例如，物理损坏的网络连接或主机硬件故障。



#### 注意

高可用性设计为与存储多路径和网络绑定结合使用，以创建能够适应硬件故障以及从硬件故障中恢复的系统。高可用性应始终与多路径存储和绑定的网络连接结合使用。

首先，高可用性可确保在主机无法访问或不稳定时，已知在该主机上运行的 VM 关闭并且能够在其他位置重新启动。这样可避免出现 VM（手动或自动）在新主机上启动和稍后启动的情况，原始主机能够恢复。这种情况会导致同一 VM 的两个实例在不同的主机上运行，并且 VM 磁盘损坏和数据丢失的可能性相对较高。

其次，在池主服务器无法访问或不稳定时，高可用性可以恢复对池的管理控制能力。高可用性可确保自动恢复管理控制能力，不需要任何手动干预。

此外，高可用性还可以自动执行在已知状态良好的主机上重新启动 VM 的过程，不需要手动干预。可以安排这些 VM 成组重新启动，以留出启动服务的时间。这使得基础结构 VM 能够在其从属 VM 启动之前启动（例如，DHCP 服务器在其从属 SQL Server 启动之前启动）。

**警告**

高可用性设计与多路径存储和绑定的网络连接结合使用，应在尝试设置高可用性之前对此进行配置。未设置多路径网络连接和存储的客户在基础结构不稳定时可能会看到意外的主机重新启动行为（自主防护）。有关详细信息，请参阅 [CTX134880 - 设计 XenServer 网络配置](#) 和 [CTX134881 - 为 XenServer 配置 iSCSI 多路径支持](#)

**3.9.1.1. 使用过量**

如果按照用户定义的故障数无法在其他位置重新启动当前正在运行的 VM，则会认为该池被过量使用。

如果池中没有任何的可用内存来运行这些出现故障的 VM，即会发生这种情况。但是，还存在一些更加细微的更改，这些更改会使高可用性保证无法持续：对虚拟块设备 (VBD) 和网络所做的更改对哪些 VM 可能会在哪些主机上重新启动会产生影响。目前，XenServer 无法在所有操作发生之前对其进行检查，也无法确定这些操作是否会违反高可用性要求。但是，如果高可用性变为无法持续，将发送异步通知。

XenServer 会动态维护故障转移方案，该方案详细说明了如果池中的一组主机在任意给定时间出现故障时应执行的操作。您需要了解的一个重要概念是允许的主机故障数值，该值作为高可用性配置的一部分进行定义。该值确定在不丢失任何服务的情况下所允许的故障数。例如，如果资源池包括 16 台主机，而允许的故障数的设置为 3，则池将计算故障转移方案，该方案允许任意 3 台主机出现故障，并仍然能够在其他主机上重新启动 VM。如果找不到方案，则会认为该池使用过量。方案根据 VM 生命周期操作和移动动态地进行重新计算。如果所做的更改（例如将新 VM 添加到池）导致池使用过量，则会发送警报（通过 XenCenter 或电子邮件）。

**3.9.1.2. 使用过量警告**

如果您尝试启动或恢复 VM，且该操作导致池使用过量，则会发出警告性警报。此警告将显示在 XenCenter 中并作为消息实例通过 Xen API 提供。此消息还可以发送到电子邮件地址（如果已配置）。然后，您可以取消操作，也可以选择仍要继续。继续操作会导致池过载。不同优先级的 VM 所使用的内存量将在池和主机级别显示。

**3.9.1.3. 主机保护**

如果出现服务器故障（如网络连接断开）或发生控制堆栈问题，XenServer 主机将进行自行保护以确保 VM 没有同时在两个服务器上运行。采取保护措施后，服务器会突然立即重新启动，使在此服务器上运行的所有 VM 停止。其他服务器会检测到 VM 不再运行，而这些 VM 将根据分配到的重新启动优先级重新启动。受保护的服务器将进入重新引导序列，并在重新启动后尝试重新加入资源池。

**3.9.2. 配置要求****注意**

Citrix 建议您仅对至少包含 3 个 XenServer 主机的池启用高可用性。有关在池中两个主机之间的检测信号丢失时高可用性功能将如何响应的详细信息，请参阅 Citrix 知识库文章 [CTX129721](#)。

要使用高可用性功能，您需要具有：

- 共享存储，其中包含至少一个大小为 356 MB 或更大的 iSCSI 或光纤通道 LUN - 检测信号。高可用性功能机制在检测信号 SR 中创建两个卷：

4 MB 检测信号卷  
用于检测信号。

256 MB 元数据卷  
存储池主服务器元数据，以便在主服务器故障转移时使用。

**注意**

为最大程度提高可靠性，Citrix 强烈建议您使用专用的 iSCSI 存储库作为高可用性功能检测信号磁盘，该磁盘不得用于任何其他用途。

如果您使用的是 NetApp 或 EqualLogic SR，请在阵列中手动置备 iSCSI LUN 作为检测信号 SR 使用。

- XenServer 池。此功能提供单一资源池内服务器级别的高可用性。
- 所有主机的静态 IP 地址。

**警告**

如果启用高可用性功能同时更改服务器 IP 地址，高可用性功能将假定该主机的网络失败，并可能会保护主机，使其处于不能引导的状态。要补救这种情况，使用 `host-emergency-ha-disable` 命令禁用高可用性功能，使用 `pool-emergency-reset-master` 重新设定池主服务器，然后重新启用高可用性功能。

对于受高可用性功能保护的 VM，该 VM 必须灵活。这意味着：

- 虚拟机的虚拟磁盘必须置于共享存储中（可以使用任何共享存储类型；只在存储检测信号时要求使用 iSCSI 或光纤通道 LUN，您可以根据个人喜好为虚拟磁盘存储使用 iSCSI 或光纤通道 LUN，对此没有强制性要求）。
- 虚拟机一定不能连接到配置的本地 DVD 驱动器。
- 虚拟机的虚拟网络接口应位于池范围内的网络中。

如果启用 HA，Citrix 强烈建议为池中的服务器使用绑定的管理接口，为检测信号 SR 使用多路径存储。

如果您从 CLI 创建 VLAN 和绑定的接口，则即使已创建，也可能不插入和激活它们。在这种情况下，VM 可能显得不够灵活，无法由高可用性功能保护。如果发生这种情况，请使用 CLI `pif-plug` 命令建立 VLAN 和绑定 PIF，以使 VM 获得灵活性。还可以使用 `xe diagnostic-vm-status` CLI 命令分析 VM 放置限制来准确确定 VM 不灵活的原因，并在需要时执行修复操作。

### 3.9.3. 重新启动优先级

为虚拟机分配了重新启动优先级，并使用一个标志指示虚拟机是否应由高可用性功能保护。启用高可用性功能时，每项工作都旨在确保受保护的虚拟机保持运行。如果指定了重新启动优先级，将自动启动任何处于停止状态的受保护的 VM。如果服务器失败，则其中的 VM 将在另一台服务器上启动。

下面是重新启动优先级的说明：

高可用性功能重新启动优先级	重新启动说明
0	首先尝试启动具有此优先级的 VM
1	仅在尝试重新启动优先级为 0 的所有 VM 之后，才尝试启动具有此优先级的 VM
2	仅在尝试重新启动优先级为 1 的所有 VM 之后，才尝试启动具有此优先级的 VM
3	仅在尝试重新启动优先级为 2 的所有 VM 之后，才尝试启动具有此优先级的 VM

高可用性功能重新启动优先级	重新启动说明
best-effort	仅在尝试重新启动优先级为 3 的所有 VM 之后，才尝试启动具有此优先级的 VM

高可用性功能始终运行	说明
True	重新启动计划中包括具有此设置的 VM
False	重新启动计划中不包括具有此设置的 VM

重新启动优先级决定出现故障时 XenServer 尝试启动 VM 的顺序。在允许服务器失败次数大于零（如 GUI 中的高可用性功能面板所示，或 CLI 内池对象的 ha-plan-exists-for 字段所示）的指定配置中，具有 0、1、2 或 3 重新启动优先级的 VM 保证能在发生指定的特定服务器失败次数时重新启动。具有 best-effort 优先级设置的 VM 不是故障转移方案的一部分，并且由于没有为它们保留容量，不能保证始终处于运行状态。如果池遇到服务器故障并进入某种允许的故障数降为零的状态，将不再保证重新启动受保护的 VM。如果出现这种情况，将生成系统警报。在这种情况下，如果发生其他故障，所有设置了重新启动优先级的 VM 将根据 best-effort 行为运行。

如果发生服务器故障时无法重新启动某个受保护的 VM（例如，发生故障时过度使用池），将在池状态更改时再次尝试启动此 VM。这意味着，如果池中其他容量变为可用（例如，如果关闭非基本的 VM 或添加其他服务器），就会进行重新启动受保护的 VM 的全新尝试，并且可能立即成功。



#### 注意

要为即将重新启动 always-run=true 的 VM 释放资源，任何运行中的 VM 都不能停止或迁移。

## 3.10. 对 XenServer 池启用高可用性功能

可使用 XenCenter 或命令行接口对池启用高可用性功能。无论使用哪种方法，都需要指定一组优先级，用来确定当池被过度使用时赋予哪些 VM 最高重新启动优先级。



#### 警告

启用高可用性功能后，可能禁用某些会影响重新启动 VM 的方案的操作，例如从池中移除服务器。要执行这些操作，可以暂时禁用高可用性功能，也可以对受高可用性功能保护的 VM 解除保护。

### 3.10.1. 使用 CLI 启用高可用性功能

1. 确认您具有连接到池的兼容的存储库 (SR)。iSCSI 或光纤通道都是兼容的 SR 类型。有关如何使用 CLI 配置此类存储库的详细信息，请参阅第 5.3 节“存储配置”。
2. 对每台要保护的 VM 设置重新启动优先级。可按如下方式执行此操作：

```
xe vm-param-set uuid=<vm_uuid> ha-restart-priority=<1> ha-always-run=true
```

3. 在池上启用高可用性，然后选择性指定超时：

```
xe pool-ha-enable heartbeat-sr-uuids=<sr_uuid> ha-config:timeout=<timeout in seconds>
```

超时是指网络连接或存储不能被池中的主机访问的期限。如果在启用高可用性时未指定超时，XenServer 将使用默认 30 秒超时。如果任何 XenServer 主机在超时期限内都无法访问网络连接或存储，则将进行自主防护并重新启动。



4. 运行 `pool-ha-compute-max-host-failures-to-tolerate` 命令。此命令返回允许的故障主机最大数目，超过此数目，资源将不足以运行池中所有受保护的 VM。

```
xe pool-ha-compute-max-host-failures-to-tolerate
```

允许的故障数量决定何时发送警报：当池状态更改时，系统将重新计算故障转移方案，系统使用此计算结果来确定池容量以及确保受保护的 VM 正常运行的最大允许故障数。当计算的值低于指定的 `ha-host-failures-to-tolerate` 值时，将生成系统警报。

5. 为允许故障数参数指定一个数目。此值应该小于或等于计算的值：

```
xe pool-param-set ha-host-failures-to-tolerate=<2> uuid=<pool-uuid>
```

### 3.10.2. 使用 CLI 解除 VM 的高可用性保护

要禁用 VM 的高可用性功能，请使用 `xe vm-param-set` 命令将 `ha-always-run` 参数设置为 `false`。这不会清除 VM 重新启动优先级设置。通过将 `ha-always-run` 参数设置为 `true`，可以对 VM 再次启用高可用性功能。

### 3.10.3. 恢复无法访问的主机

如果由于某些原因，主机无法访问高可用性功能的状态文件，则主机可能变为无法访问。要恢复 XenServer 安装，可能需要使用 `host-emergency-ha-disable` 命令禁用高可用性功能：

```
xe host-emergency-ha-disable --force
```

如果主机为池主服务器，则应禁用高可用性功能以常规方式启动。从属服务器应重新连接并自动禁用高可用性功能。如果主机是池从属服务器并且无法与主服务器联系，则可能需要强制主机作为池主服务器重新引导 (`xe pool-emergency-transition-to-master`) 或告诉它新主服务器的位置 (`xe pool-emergency-reset-master`)：

```
xe pool-emergency-transition-to-master uuid=<host_uuid>
xe pool-emergency-reset-master master-address=<new_master_hostname>
```

成功重新启动所有主机后，重新启用高可用性功能：

```
xe pool-ha-enable heartbeat-sr-uuid=<sr_uuid>
```

### 3.10.4. 在启用高可用性功能的情况下关闭主机

如果启用高可用性功能，需要特别注意关闭或重新引导主机时的情况，以防止高可用性功能机制假定主机已失败。要在启用高可用性功能的环境中完全关闭主机，请首先 `disable` 主机，然后 `evacuate` 主机，最后使用 XenCenter 或 CLI `shutdown` 主机。要通过命令行在启用高可用性功能的环境中关闭主机，可使用以下命令：

```
xe host-disable host=<host_name>
xe host-evacuate uuid=<host_uuid>
xe host-shutdown host=<host_name>
```

### 3.10.5. 关闭受高可用性功能保护的 VM

如果 VM 受高可用性方案保护并设置为自动重新启动，则无法在此保护处于活动状态时关闭。要关闭 VM，首先禁用其高可用性功能保护，然后执行 CLI 命令。如果单击受保护的 VM 的 `Shutdown` (关闭) 按钮，XenCenter 将提供一个用来自动禁用保护的对话框。



#### 注意

如果关闭来宾系统内的某个 VM，并且此 VM 受保护，则它将在高可用性功能故障的情况下自动重新启动。这有助于确保操作错误 (或某个错误程序错误地关闭了 VM 时) 不会导致受保护的 VM 意外关闭。如果要关闭此 VM，应首先禁用其高可用性功能保护。



## 3.11. 主机启动

### 3.11.1. 远程启动主机

可以使用 XenServer 主机启动功能，从 XenCenter 中或使用 CLI 远程打开和关闭服务器。

要启用主机电源，主机必须具有以下电源控制解决方案之一：

- 启用了“LAN 唤醒”的网卡。
- Dell Remote Access Cards (DRAC)。要将 XenServer 与 DRAC 一起使用，您必须安装 Dell 增补包以获取 DRAC 支持。DRAC 支持需要在带有远程访问控制器的服务器上安装 RACADM 命令行实用程序，并启用 DRAC 及其接口。RACADM 通常包含在 DRAC 管理软件中。有关详细信息，请参阅 Dell 的 DRAC 文档。
- Hewlett-Packard Integrated Lights-Out (iLO)。要将 XenServer 与 iLO 一起使用，您必须在主机上启用 iLO，并将接口连接到网络。有关详细信息，请参阅 HP 的 iLO 文档。
- 基于 XenAPI 且使您能够通过 XenServer 打开和关闭电源的自定义脚本。有关详细信息，请参阅 [第 3.11.3 节“配置 XenServer 主机开机功能的自定义脚本”](#)。

使用主机开机功能时需要执行以下两项任务：

1. 确保池中的主机支持远程控制电源（即，这些主机具有“LAN 唤醒”功能、DRAC 或 iLO 卡，或您创建的自定义脚本）。
2. 使用 CLI 或 XenCenter 启用主机启动功能。

### 3.11.2. 使用 CLI 管理主机启动功能

可以使用 CLI 或 XenCenter 管理主机启动功能。此主题介绍与使用 CLI 管理主机启动功能有关的信息。

主机启动功能在主机级别（即在每台 XenServer 上）启用。

启用主机启动功能之后，可以使用 CLI 或 XenCenter 打开主机。

#### 3.11.2.1. 使用 CLI 启用主机启动功能

1. 运行以下命令：

```
xe host-set-power-on host=<host uuid>\
power-on-mode=("","wake-on-lan",
"iLO","DRAC","custom")
power-on-config:key=value
```

对于 iLO 和 DRAC，键为 power\_on\_ip、power\_on\_user、power\_on\_password\_secret。如果要使用加密功能，可以使用 power\_on\_password\_secret 指定密码。

#### 3.11.2.2. 使用 CLI 远程打开主机

1. 运行以下命令：

```
xe host-power-on host=<host uuid>
```

### 3.11.3. 配置 XenServer 主机开机功能的自定义脚本

如果服务器的远程电源解决方案使用默认情况下不受支持的协议（例如响铃唤醒或 Intel 主动管理技术），可以创建自定义 Linux Python 脚本，以远程打开 XenServer 计算机。但是，您还可以为 iLO、DRAC 和 LAN 唤醒远程电源解决方案创建自定义脚本。

本主题介绍了与以下内容有关的信息：使用与 XenServer API 调用 `host.power_on` 相关联的键/值对配置主机启动的自定义脚本。

创建自定义脚本时，在每次要在 XenServer 上远程控制电源时从命令行运行该脚本。或者，可以在 XenCenter 中指定该脚本，并使用 XenCenter UI 功能与之交互。

《[Citrix XenServer 管理 API]》文档中介绍了 XenServer API，该文档可从 Citrix 网站获取。



#### 警告

请勿修改 `/etc/xapi.d/plugins/` 目录中提供的默认脚本。您可以在此目录中加入新脚本，但安装后不得修改此目录中包含的脚本。

### 3.11.3.1. 键/值对

要使用主机启动功能，必须配置 `host.power_on_mode` 和 `host.power_on_config` 键。这两个键的值在下面提供。

还有一个 API 调用使您能够同时设置所有字段：

```
void host.set_host_power_on_mode(string mode, Dictionary<string,string> config)
```

#### 3.11.3.1.1. `host.power_on_mode`

- 定义：此 API 调用包含用于指定远程电源解决方案类型（例如 Dell DRAC）的项/值对。
- 可能的值：
  - 空字符串，表示电源控制处于禁用状态
  - "iLO" -使您能够指定 HP iLO。
  - "DRAC" -使您能够指定 Dell DRAC。要使用 DRAC，您必须已安装 Dell 增补包。
  - "wake-on-lan" -使您能够指定“LAN 唤醒”。
  - 任何其他名称（用于指定自定义启动脚本）。此选项用于指定电源管理的自定义脚本。
- 类型：字符串

#### 3.11.3.1.2. `host.power_on_config`

- 定义：此项包含用于模式配置的项/值对。为 iLO 和 DRAC 提供其他信息。
- 可能的值：
  - 如果您配置“iLO”或“DRAC”作为远程电源解决方案的类型，还必须指定下面的某个键：
    - "power\_on\_ip" -此键为您指定配置为与电源控制卡进行通信的 IP 地址。或者，您可以输入配置 iLO 或 DRAC 的网络接口的域名。
    - "power\_on\_user" -此键为与管理处理器相关联的 iLO 或 DRAC 用户名，您可能已更改其出厂时的默认设置，也可能未更改。
    - "power\_on\_password\_secret" -指定使用加密功能来保护密码的安全。
  - 要使用加密功能存储密码，请指定键“power\_on\_password\_secret”。
- 类型：Map (string,string)

### 3.11.3.2. 示例脚本

此示例脚本会导入 XenServer API，将脚本本身定义为自定义脚本，然后将特定的参数传递给您要远程控制的主机。必须在所有自定义脚本中定义参数 `session`、`remote_host` 和 `power_on_config`。

结果仅在脚本不成功时显示。

```
import XenAPI
def custom(session,remote_host,
power_on_config):
result="Power On Not Successful"
for key in power_on_config.keys():
result=result+"
key="+key+"
value="+power_on_config[key]
return result
```



#### 注意

创建完成后，使用 .py 扩展名将脚本保存到 /etc/xapi.d/plugins 中。

## 3.12. 与 XenServer 主机和资源池通信

XenServer 7.0 使用 TLS 协议加密 XenAPI 流量。这意味着 XenServer 与 XenAPI 客户端（或设备）之间的任何通信现在都将默认使用 TLS 1.2 协议。但是，如果 XenAPI 客户端或设备不使用 TLS 1.2 进行通信，则可能会使用早期的协议进行通信。

XenServer 使用以下密码套件：

- TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA256
- TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA
- TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA
- TLS\_RSA\_WITH\_RC4\_128\_SHA
- TLS\_RSA\_WITH\_RC4\_128\_MD5
- TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA

XenServer 还允许您将主机或资源池配置为允许通过仅限 TLS 1.2 进行通信。此选项允许 XenServer 与 XenAPI 客户端（或设备）之间的通信使用 TLS 1.2 协议。“仅限 TLS 1.2”选项使用密码套件 TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA256。



#### 警告

只有在确保与 XenServer 池通信的所有 XenAPI 客户端和设备均与 TLS 1.2 兼容之后，才能选择仅限 TLS 1.2 选项。

## 第 4 章 网络连接

本章将概括介绍 XenServer 的网络连接（包括网络、VLAN 和 NIC 绑定）。此外，还将讨论如何管理网络连接配置以及进行故障排除。



### 重要

XenServer 的默认网络堆栈是 vSwitch；但是，如果需要，也可按照第 4.2 节“vSwitch 网络”中的说明使用 Linux 网络堆栈。

如果您已经熟悉了 XenServer 网络概念，可能希望提前跳到下列部分之一：

- 要为独立 XenServer 主机创建网络，请参阅第 4.4.2 节“在独立服务器中创建网络”。
- 要跨多个 XenServer 主机创建专用网络，请参阅第 4.4.1 节“跨服务器专用网络”。
- 要为资源池中配置的 XenServer 主机创建网络，请参阅第 4.4.3 节“在资源池中创建网络”。
- 要为 XenServer 主机（独立主机或资源池的成员）创建 VLAN，请参阅第 4.4.4 节“创建 VLAN”。
- 要为独立 XenServer 主机创建绑定，请参阅第 4.4.5 节“在独立主机上创建 NIC 绑定”。
- 要为资源池中配置的 XenServer 主机创建绑定，请参阅第 4.4.6 节“在资源池中创建 NIC 绑定”。

有关网络连接和网络设计的其他信息，请参阅 Citrix 知识中心中的“Designing XenServer Network Configuration”（设计 XenServer 网络配置）。

为了与此版本中的 XenCenter 术语变更保持一致，本章现在使用术语管理接口来表示支持 IP 且传送管理通信的 NIC。在上一版本中，本章使用术语主管理接口来表示这一含义。同样，本章现在使用术语辅助接口来表示支持 IP 且配置用于存储通信的 NIC。

### 4.1. 网络连接支持

对于每个 XenServer 主机，XenServer 最多支持 16 个物理网络接口（或最多支持 8 个已绑定的网络接口）；对于每个 VM，最多支持 7 个虚拟网络接口。



### 注意

XenServer 使用 xe 命令行接口 (CLI) 提供 NIC 的自动配置和管理。与早期的 XenServer 版本不同，在大多数情况下，不应直接编辑主机的网络配置文件；如果 CLI 命令可用，请不要编辑基础文件。

### 4.2. vSwitch 网络

与控制器设备结合使用时，vSwitch 网络支持开放流并提供额外的功能，例如访问控制列表 (ACL)。XenServer vSwitch 的控制器设备称为 vSwitch 控制器：允许您通过图形用户界面监视网络。vSwitch 控制器：

- 支持细化的安全策略，以控制传入和传出 VM 的通信流。
- 详细展示虚拟网络环境中所传输的所有通信的行为和性能。

vSwitch 大大简化了虚拟网络环境中的 IT 管理 — 即使 VM 从资源池中的一个物理主机迁移到另一个主机，所有的 VM 配置和统计信息仍绑定到该 VM。有关详细信息，请参阅《XenServer vSwitch 控制器用户指南》。

要确定当前配置的网络堆栈，请运行以下命令：

```
xe host-list params=software-version
```

在命令输出中，请查找 `network_backend`。如果将 vSwitch 配置为网络堆栈，则输出显示：

```
network_backend: openvswitch
```

如果将 Linux 桥接配置为网络堆栈，则输出显示：

```
network_backend: bridge
```



#### 注意

要还原到 Linux 网络堆栈，请运行以下命令：

```
xe-switch-network-backend bridge
```

在运行此命令之后重新引导主机。



#### 警告

Linux 网络堆栈不支持开放流和跨服务器专用网络，不能由 XenServer vSwitch 控制器进行管理。

## 4.3. XenServer 网络概述

本部分介绍 XenServer 环境中网络的一般概念。

在 XenServer 安装期间，系统会为每个物理网络接口卡 (NIC) 创建一个网络。向资源池添加服务器时，这些默认网络会进行合并，以便设备名称相同的所有物理 NIC 均连接到同一网络。

通常情况下，如果您想要创建内部网络，使用现有 NIC 设置新 VLAN，或创建 NIC 绑定，只需添加一个新网络即可。

在 XenServer 中您可以配置四种不同类型的网络：

- 外部网络，与物理网络接口相关联，可在虚拟机与连接到网络的物理网络接口之间提供桥接，从而使虚拟机能够通过服务器的物理网络接口卡连接到可用的资源。
- 绑定的网络，可在两个 NIC 之间创建一个绑定，以在虚拟机与网络之间创建一个高性能通道。
- 单服务器专用网络，其与物理网络接口无关联，可用于提供指定主机上的虚拟机之间的连接，而与外界无连接。
- 跨服务器专用网络，对单服务器专用网络这一概念进行了扩展，允许不同主机上的 VM 使用 vSwitch 相互通信。



#### 注意

某些网络选项在用于独立 XenServer 主机与用于资源池时具有不同的行为。本章中的各节首先介绍同时适用于独立主机和池的一般信息，之后介绍分别适用于这两种类型的特定信息。

### 4.3.1. 网络对象

本章使用三种类型的服务器端软件对象来表示网络连接实体。这些对象包括：

- PIF，表示 XenServer 主机上的一个物理 NIC。PIF 对象具有名称和说明、全局唯一 UUID、它们所表示的 NIC 的参数，以及连接到的网络和服务器。
- VIF，表示虚拟机上的一个虚拟 NIC。VIF 对象具有名称和说明、全局唯一 UUID 和连接到的网络和 VM。
- 网络，即 XenServer 主机上的虚拟以太网交换机。网络对象具有名称和说明、全局唯一 UUID 以及连接到的 VIF 和 PIF 的集合。

XenCenter 和 xe CLI 都允许配置网络选项、控制使用哪个 NIC 管理操作，以及创建高级网络功能（如虚拟局域网 (VLAN) 和 NIC 绑定）。

### 4.3.2. 网络

每台 XenServer 主机都有一个或多个网络，即虚拟以太网交换机。不与 PIF 相关联的网络被视为内部网络，仅可用于提供给定 XenServer 主机上 VM 之间的连接，不与外部连接。与 PIF 相关联的网络被视为外部网络，在连接到网络的 PIF 和 VIF 之间提供了一个桥，从而能够通过 PIF 的 NIC 连接到资源。

### 4.3.3. VLAN

按照 IEEE 802.1Q 标准的有关定义，虚拟局域网 (VLAN) 允许单一物理网络支持多个逻辑网络。XenServer 主机可采用多种方式来使用 VLAN。



#### 注意

所有受支持的 VLAN 配置均可应用于池、独立主机、绑定配置和非绑定配置。

#### 4.3.3.1. 结合使用 VLAN 和管理接口

交换机端口经过配置可执行 802.1Q VLAN 标记/去除标记，它们通常称为本机 VLAN 端口或访问模式端口，这些端口与管理接口合用可将管理通信放到目标 VLAN。在这种情况下，XenServer 主机不会觉察到任何 VLAN 配置。

管理接口不能通过中继端口分配给 XenServer VLAN。

#### 4.3.3.2. 使用带有虚拟机的 VLAN

交换机端口配置为 802.1Q VLAN 中继端口后，与 XenServer VLAN 功能结合使用，可将来宾虚拟网络接口 (VIF) 连接到特定的 VLAN。在这种情况下，XenServer 主机将为来宾执行 VLAN 标记/去标记功能，而来宾不会觉察到任何 VLAN 配置。

XenServer VLAN 由另外的 PIF 对象（代表与特定 VLAN 标记相应的 VLAN 接口）来表示。然后，XenServer 网络就能连接到代表物理 NIC 的 PIF 来查看此 NIC 上的全部通信，或者连接到代表 VLAN 的 PIF 来查看只具有特定 VLAN 标记的通信。

有关如何为 XenServer 主机（独立主机或资源池中的主机）创建 VLAN 的步骤，请参阅第 4.4.4 节“[创建 VLAN](#)”。

#### 4.3.3.3. 使用具有专用存储 NIC 的 VLAN

专用存储 NIC ( 又称为支持 IP 的 NIC 或简称管理接口 ) 可配置为将上述的本机 VLAN/访问模式端口用于管理接口, 或者与上述中继端口和 XenServer VLAN 一起用于虚拟机。要配置专用存储 NIC, 请参阅第 4.4.7 节 “配置专用存储 NIC”。

#### 4.3.3.4. 将管理接口和来宾 VLAN 组合到单一主机 NIC 上

一个交换机端口可使用中继端口和本地 VLAN 来配置, 这就使得一个主机 NIC 可用作管理接口 ( 在本机 VLAN 上 ), 并可用于将来 VIF 连接到特定的 VLAN ID。

#### 4.3.4. 巨型帧

可以使用巨型帧来优化存储通信的性能。巨型帧是包含超过 1500 字节有效负载的以太网帧。巨型帧通常用于实现更高的吞吐量, 降低系统总线内存上的负载, 并减少 CPU 开销。



#### 注意

仅在 vSwitch 用作池中所有主机上的网络堆栈时, XenServer 才支持巨型帧。

#### 使用巨型帧的要求

使用巨型帧时, 客户应当注意以下事项:

- 在池级别配置巨型帧
- vSwitch 必须配置为池中所有主机上的网络后端
- 子网上的每个设备必须配置为使用巨型帧
- 建议客户仅在专用存储网络上启用巨型帧
- 在管理网络上启用巨型帧的配置不受支持
- 不支持在 VM 上使用巨型帧

要使用巨型帧, 客户应将最大传输单位 (MTU) 设置为介于 1500 和 9216 之间的一个值。这可以通过使用 XenCenter 或 xe CLI 实现。有关配置网络和巨型帧的详细信息, 请参阅 Citrix 知识中心中的 “Designing XenServer Network Configurations” (设计 XenServer 网络配置)。

#### 4.3.5. NIC 绑定

NIC 绑定 ( 有时也称作 NIC 成组 ) 可以让管理员将两个或更多 NIC 配置在一起, 在本地当作一个逻辑网卡使用, 从而提高 XenServer 主机的恢复能力和/或带宽。所有绑定的 NIC 共享同一 MAC 地址。

如果绑定中的 NIC 失败, 则主机的网络通信会自动通过另一个 NIC 重定向。XenServer 最多支持八个绑定网络。

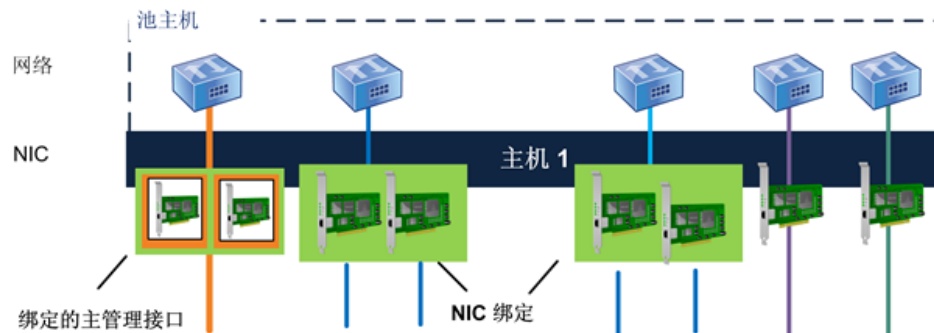
XenServer 支持主动-主动、主动-被动和 LACP 绑定模式。受支持 NIC 数量和受支持绑定模式因网络堆栈而异:

- LACP 绑定仅适用于 vSwitch, 而主动-主动和主动-被动适用于 vSwitch 和 Linux 桥接。



- 当 vSwitch 为网络堆栈时，您可以将两个、三个或四个 NIC 绑定在一起。
- 当 Linux 桥接为网络堆栈时，您只能绑定两个 NIC。

在下图中，管理接口位于绑定的 NIC 对上。XenServer 将使用此 NIC 绑定对进行管理通信。



此图显示的主机配置了一个位于绑定上的管理接口，而且还绑定了两对 NIC 用于来宾系统通信。除了管理接口绑定，XenServer 还将另外两个绑定的 NIC 和两个非绑定的 NIC 用于 VM 通信。

所有绑定模式都支持故障转移；但是，并非所有模式都允许所有链路在使用所有类型通信时保持活动状态。XenServer 支持将下列类型的 NIC 绑定在一起：

- NIC (非管理)。您可以绑定由 XenServer 专用于 VM 通信的 NIC。绑定这些 NIC 不仅提供恢复能力，而且还会在 NIC 之间平衡来自多个 VM 的通信。
- 管理接口。您可以将管理接口绑定到另一个 NIC，以便该 NIC 为管理通信提供故障转移。虽然配置 LACP 链路聚合绑定可以为管理通信提供负载平衡，但主动-主动 NIC 绑定不提供这一功能。
- 辅助接口。您可以绑定已配置为辅助接口的 NIC (例如，用于存储)。但是，对于大多数 iSCSI 软件发起程序存储，Citrix 建议配置多路径而不是 NIC 绑定，如“Designing XenServer Network Configurations” (设计 XenServer 网络配置) 中所述。

在本节中，术语基于 IP 的存储通信是 iSCSI 和 NFS 通信的统称。

如果 VIF 已在使用将绑定在一起的某一接口，则可以创建一个绑定，此时 VM 通信将自动迁移到新绑定的接口。

在 XenServer 中，NIC 绑定由额外的 PIF 表示。XenServer NIC 绑定完全包含基础物理设备 (PIF)。



### 注意

不支持创建仅包含一个 NIC 的绑定。

关于 IP 寻址的要点

绑定的 NIC 可以有一个或没有 IP 地址，如下所示：

- 管理和存储网络。
  - 如果您绑定管理接口或辅助接口，则系统会为绑定分配一个 IP 地址。即，每个 NIC 没有其自己的 IP 地址；XenServer 将这两个 NIC 视为一个逻辑连接。
  - 将绑定用于非 VM 通信 (以连接到共享网络存储或 XenCenter 进行管理) 时，必须为该绑定配置一个 IP 地址。但是，如果您已将 IP 地址分配给绑定中的某个 NIC (即，已创建管理接口或辅助接口)，系统会自动将此 IP 地址分配给整个绑定。



- 如果将管理接口或辅助接口绑定到无 IP 地址的 NIC，则从 XenServer 6.0 开始，此绑定自动使用相应接口的 IP 地址。
- VM 网络。当绑定的 NIC 用于 VM（来宾系统）通信时，不需要为该绑定配置 IP 地址。这是因为绑定在 OSI 的第 2 层（即数据链路层）上运行，而此层上不使用 IP 地址。虚拟机的 IP 地址与 VIF 相关联。

### 绑定类型

XenServer 提供三种不同类型的绑定，而且它们都可以使用 CLI 或 XenCenter 配置：

- 主动/主动模式（具有在绑定的 NIC 之间平衡 VM 通信的功能）。请参阅第 4.3.5.1 节“主动-主动绑定”。
- 主动/被动模式（仅一个 NIC 主动传输通信）。请参阅第 4.3.5.2 节“主动-被动绑定”。
- LACP 链路聚合（在交换机和服务器之间协商主动和备用 NIC）。请参阅第 4.3.5.3 节“LACP 链路聚合控制协议绑定”。



### 注意

将绑定设置为最高延迟 31000 毫秒，最低延迟 200 毫秒。最高延迟的时间看似有点长，但这是有意如此，因为某些交换机要真正启用端口需要花费一些时间。如果没有这么长的最高延迟时间，链接在失败后恢复时，绑定会在交换机准备好传递通信之前重新平衡其上的通信。如果要将两个连接都移动到另外一台不同的交换机，请移动一个连接，等待 31 秒以留出再次使用它的时间，然后再移动另一连接。有关更改延迟的信息，请参阅第 4.3.8 节“更改绑定的最高延迟”。

### 绑定状态

XenServer 在每个主机的事件日志中提供绑定状态。如果绑定中的一个或多个链路发生故障或恢复，这一情况会记录在事件日志中。同样，您可以使用以下示例中显示的 links-up 参数，查询绑定链路状态：

```
xe bond-param-get uuid=<bond_uuid> param-name=links-up
```

XenServer 每隔 5 秒左右便会检查绑定中链路的状态。因此，如果绑定中又有链路在五秒窗口中发生故障，则只有在下次检查状态时才会将此故障记录到日志中。

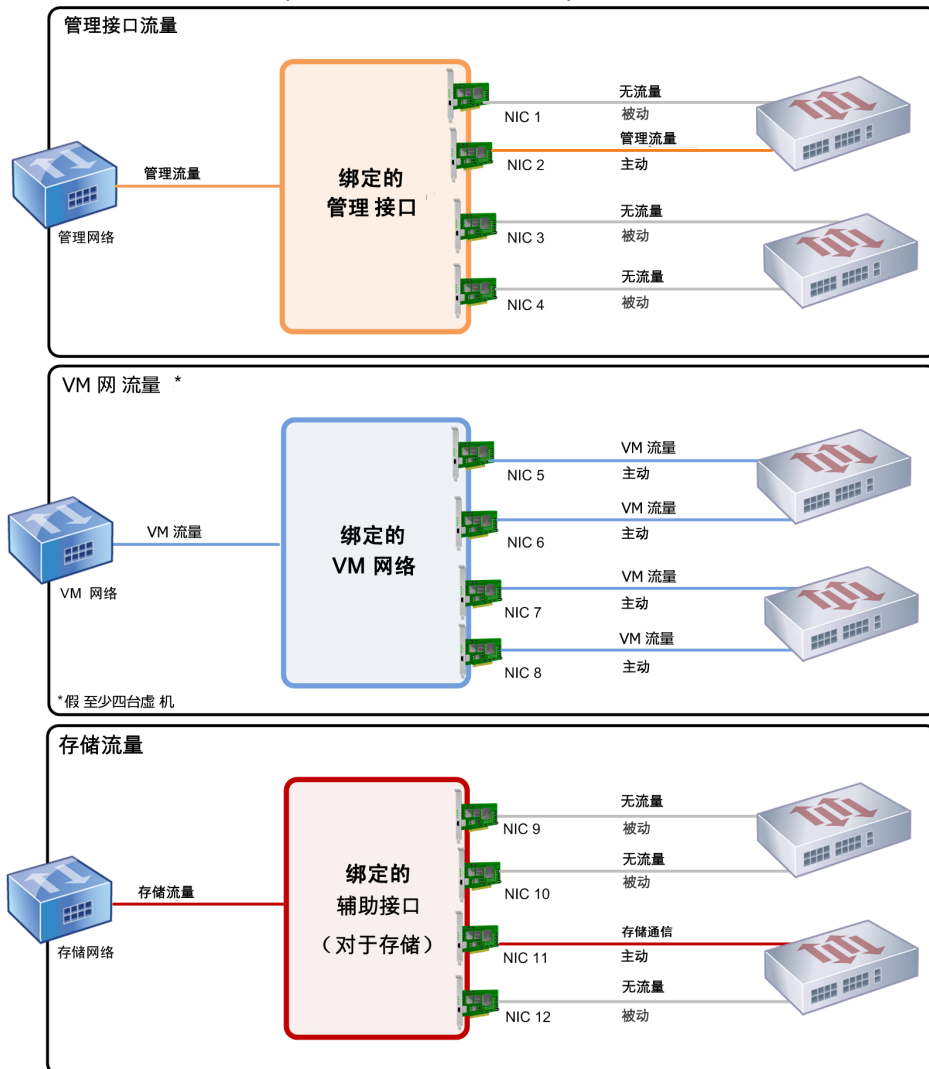
绑定事件日志在 XenCenter 日志选项卡中显示。对于不运行 XenCenter 的用户，事件日志也会在每个主机的 /var/log/xensource.log 中显示。

#### 4.3.5.1. 主动-主动绑定

主动-主动是适用于来宾系统通信的主动/主动配置，在此配置中，两个 NIC 可以同时路由 VM 通信。当绑定用于管理通信时，只有其中一个 NIC 可以路由通信，而另一个 NIC 保持未使用状态并提供故障转移支持。启用 Linux 桥接或 vSwitch 网络堆栈时，主动-主动模式为默认绑定模式。

将主动-主动绑定与 Linux 桥接一起使用时，您只能绑定两个 NIC。将 vSwitch 用作网络堆栈时，您可以在主动-主动模式下绑定两个、三个或四个 NIC。但是，在主动-主动模式下，绑定三个或四个 NIC 通常仅对 VM 通信有利，如下图所示。

## 主动-主动式绑定 ( 虚拟交换机网络堆叠 )



此图显示了绑定四个 NIC 时为何只有来宾系统通信可能会受益。在最上图的管理网络中，NIC 2 处于主动状态，NIC 1、3 和 4 处于被动状态。对于 VM 通信，绑定中的所有四个 NIC 都处于主动状态；但是，这假设系统中至少有四个 VM。对于存储通信，只有 NIC 11 处于活动状态。

当多个 MAC 地址与绑定关联时，XenServer 只能在两个或更多 NIC 上传输通信。XenServer 可以使用 VIF 中的虚拟 MAC 地址跨多个链路传输通信。特别说明：

- VM 通信。假设您对仅传输 VM (来宾系统) 通信的 NIC 启用绑定，则所有链路都处于活动状态，因而 NIC 绑定可以在各 NIC 之间平衡 VM 通信。单个 VIF 的通信从不在各 NIC 之间拆分。
- 管理或存储通信。绑定中只有一个链路 (NIC) 处于活动状态，其他 NIC 保持未使用状态，除非将通信故障转移到这些 NIC。在绑定网络上配置管理接口或辅助接口可以提供恢复能力。
- 混合通信。如果绑定的 NIC 同时传输基于 IP 的存储通信和来宾系统通信，则仅对来宾系统通信和控制域通信实施负载平衡。从本质上说，控制域是一个虚拟机，因此，它可以像其他来宾系统一样使用 NIC。XenServer 像平衡 VM 通信一样平衡控制域通信。

通信平衡

XenServer 使用数据包的源 MAC 地址在 NIC 之间对通信进行平衡。由于传输管理通信时仅呈现一个源 MAC 地址，所以主动-主动模式仅使用一个 NIC，且不平衡通信。通信平衡基于两个因素：

- 虚拟机及其关联的发送或接收通信的 VIF
- 发送的数据量（以千字节为单位）。

XenServer 会评估每个 NIC 发送和接收的数据量（以千字节为单位）。如果在一个 NIC 上发送的数据量超出在另一个 NIC 上发送的数据量，XenServer 会重新平衡各 VIF 使用的 NIC。传输 VIF 的全部负载；一个 VIF 的负载从不在两个 NIC 之间拆分。

尽管主动-主动 NIC 绑定可以为来自多个 VM 的通信提供负载平衡，但是不能让单个 VM 利用两个 NIC 的吞吐量。任何给定 VIF 一次只能使用绑定中的一个链路。当 XenServer 定期重新平衡通信时，VIF 不会永久分配给绑定中的某个特定 NIC。

主动-主动模式有时称为源负载平衡 (SLB) 绑定，因为 XenServer 使用 SLB 在绑定的网络接口之间共享负载。SLB 派生自开源自适应负载平衡 (ALB) 模式并重用 ALB 功能在各 NIC 之间动态重新平衡负载。

重新平衡时，系统会跟踪给定时间段内通过每个从属（接口）的字节数。发送包含新的源 MAC 地址的数据包时，该数据包会分配给具有最低利用率的从属接口。系统会定期重新平衡通信。

每个 MAC 地址都有对应的负载，而且 XenServer 能够在 NIC 之间转移整体负载，具体情况取决于 VM 发送和接收的数据量。对于主动-主动通信，来自一个 VM 的所有通信只能在一个 NIC 上传输。



#### 注意

主动-主动绑定不要求交换机支持 EtherChannel 或 802.3ad (LACP)。

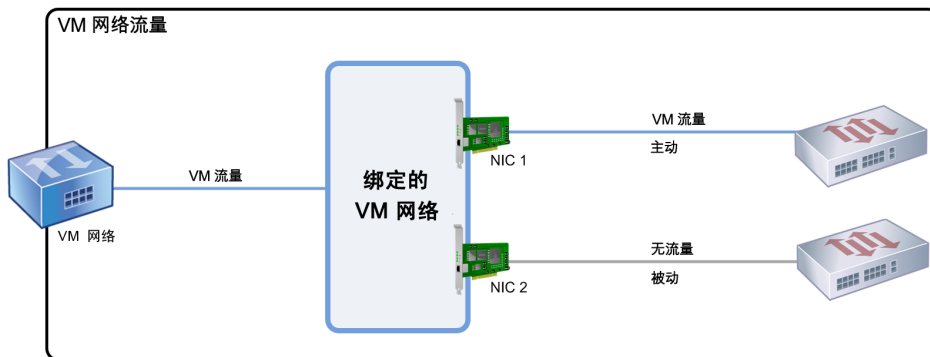
#### 4.3.5.2. 主动-被动绑定

主动-被动绑定仅在一个 NIC 上路由通信，因此，如果主动 NIC 断开网络连接，通信将故障转移到绑定中的另一个 NIC。主动-被动绑定在主动 NIC 上路由通信，而且仅当主动 NIC 发生故障时，通信才转移到被动 NIC。

主动-被动绑定可在 Linux 桥接和 vSwitch 网络堆栈中可用。与 Linux 桥接一起使用时，您可以绑定两个 NIC。与 vSwitch 一起使用时，只能绑定两个、三个或四个 NIC。但是，无论通信类型如何，在主动-被动模式下绑定 NIC 时，只有一个链路处于活动状态，且链路之间不会实施负载平衡。

下图显示了在主动-被动模式下配置的两个绑定的 NIC。

## 主动-被动式绑定



此图显示了在主动-被动模式下绑定的两个 NIC。NIC 1 处于活动状态。此绑定包含一个连接到第二台交换机用于故障转移的 NIC。此 NIC 仅在 NIC 1 发生故障时使用。

因为主动-主动模式是 XenServer 中的默认绑定配置，如果使用 CLI 配置绑定，必须为主动-被动模式指定一个参数，或者将绑定创建为主动-主动模式。但是，您不必仅仅因为网络传输管理通信或存储通信而配置主动-被动模式。

主动-被动模式是获得恢复能力的良好选择，因为它可以提供多种优势。使用主动-被动绑定时，通信不会在 NIC 之间移动。同样，主动-被动绑定可以让您配置两台交换机以实现冗余，但不需要堆栈。（如果管理交换机停机，堆栈交换机会成为单一故障点。）

主动-被动模式不要求交换机支持 EtherChannel 或 802.3ad (LACP)。

当您不需要负载平衡或当您只希望在一个 NIC 上传输通信时，请考虑配置主动-被动模式。

**重要**

在创建了 VIF 或者池投入使用之后，在对绑定进行更改或者创建新绑定时一定要格外小心。

## 4.3.5.3. LACP 链路聚合控制协议绑定

LACP 链路聚合控制协议是一个绑定类型，它可将一组端口捆绑在一起，当作一个逻辑通道使用。LACP 绑定可以提供故障转移，并能增加可用带宽总量。

不同于其他绑定模式，LACP 绑定要求配置链路两端，也就是在主机上创建绑定，并在交换机上为每个绑定创建链路聚合组 (LAG)，如第 4.3.5.4.1 节“使用 LACP 绑定时的交换机配置”中所述。要使用 LACP 绑定，必须将 vSwitch 配置为网络堆栈。此外，交换机还必须支持 IEEE 802.3ad 标准。

下表比较了主动-主动 SLB 绑定和 LACP 绑定：

	优势	注意事项
主动-主动 SLB 绑定	<ul style="list-style-type: none"> <li>可以与“XenServer 硬件兼容性列表”中的任何交换机一起使用。</li> <li>不要求交换机支持堆栈。</li> <li>支持四个 NIC。</li> </ul>	<ul style="list-style-type: none"> <li>要实现最佳负载平衡，每个 VIF 至少要使用一个 NIC。</li> <li>存储或管理通信无法在多个 NIC 上拆分。</li> <li>负载平衡仅在有多个 MAC 地址时实施。</li> </ul>

	优势	注意事项
LACP 绑定	<ul style="list-style-type: none"> <li>无论通信类型如何，所有链路都可以处于活动状态。</li> <li>通信平衡不依赖于源 MAC 地址，因此，所有通信类型都可以平衡。</li> </ul>	<ul style="list-style-type: none"> <li>交换机必须支持 IEEE 802.3ad 标准。</li> <li>要求在交换机端执行配置。</li> <li>仅支持 vSwitch。</li> <li>要求使用一台交换机或堆栈交换机。</li> </ul>

## 通信平衡

XenServer 支持两个 LACP 绑定散列类型（术语散列指的是 NIC 和交换机分发通信的方式），分别是 (1) 基于源和目标地址的 IP 和端口的负载平衡和 (2) 基于源 MAC 地址的负载平衡。

根据散列类型和通信模式，LACP 绑定有可能比主动-主动 NIC 绑定更均匀地分布通信。



### 注意

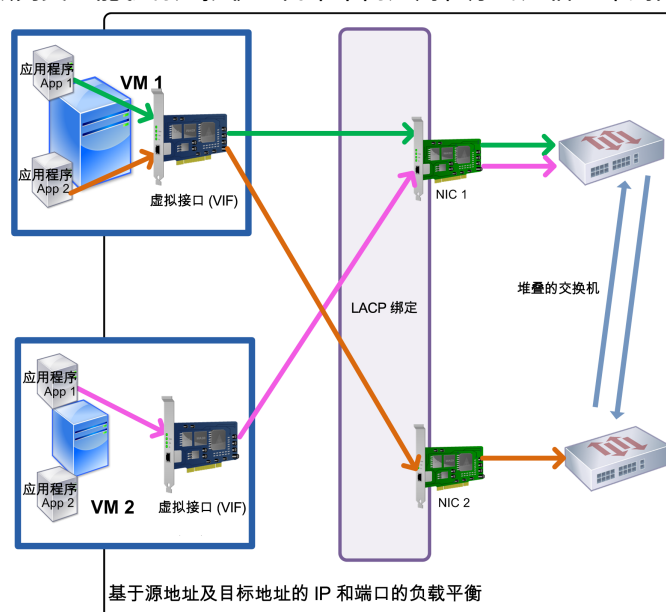
您可以在主机和交换机上分别为传出和传入通信配置设置，而且两端的配置不一定匹配。

基于源和目标地址的 IP 和端口的负载平衡。

此散列类型是默认的 LACP 绑定散列算法。只要源或目标 IP 或端口号有变化，来自一个来宾系统的通信便可以分布到两个链路上。

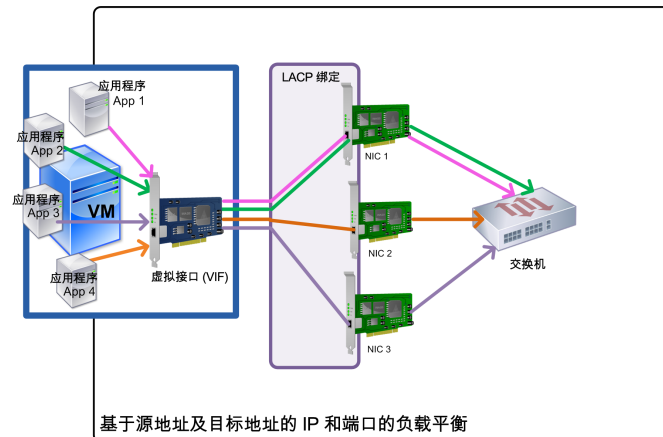
如果一个虚拟机正在运行多个应用程序，而且每个应用程序使用不同的 IP 或端口号，此散列类型就可以在若干链路上分布通信，使来宾系统有可能使用聚合吞吐量。此散列类型可以让一个来宾系统使用多个 NIC 的整体吞吐量。

同样，如下图所示，此散列类型能够将虚拟机上两个不同应用程序的通信分布到两个不同 NIC 上。



下图显示，如果您使用 LACP 绑定并启用将基于源和目标的 IP 与端口的负载平衡作为散列类型的 LACP，则来自 VM1 上两个不同应用程序的通信可分布到两个 NIC 上。

当您想平衡同一 VM 上两个不同应用程序的通信时（例如，仅将一个虚拟机配置为使用三个 NIC 构成的绑定），根据源和目标地址的 IP 和端口配置 LACP 绑定很有用。

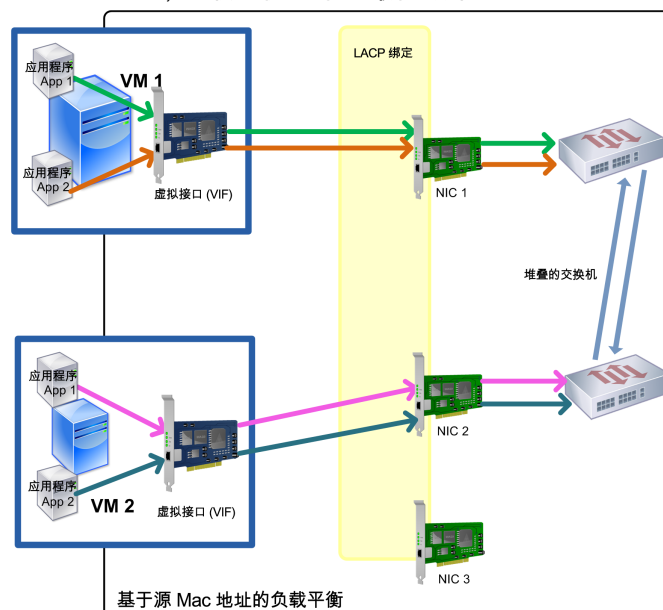


此图显示，如果使用 LACP 绑定并启用将基于源和目标 IP 与端口的负载均衡作为散列类型的 LACP，XenServer 能够通过绑定中三个 NIC 中的某个 NIC 传输虚拟机中每个应用程序的通信，即便 NIC 数量超出 VIF 数量。

此散列类型的平衡算法使用五个因素在 NIC 上分布通信，即源 IP 地址、源端口号、目标 IP 地址、目标端口号和源 MAC 地址。

基于源 MAC 地址的负载均衡。

当同一主机上有多个虚拟机时，此类型的负载均衡可以发挥很好的作用。通信平衡根据发起通信的 VM 的虚拟 MAC 地址实施。XenServer 使用同一算法发送传出通信，就像在主动-主动绑定模式下一样。来自同一来宾系统的通信不会在多个 NIC 之间拆分。因此，如果 VIF 的数量少于 NIC，此散列类型不适合使用，原因是由于通信无法在 NIC 之间拆分，负载均衡达不到最佳效果。



此图显示，如果使用 LACP 绑定并启用将基于源 MAC 地址作为散列类型的 LACP，那么如果 NIC 的数量超过 VIF 的数量，则不会使用所有 NIC。因为有三个 NIC 和两个 VM，所以只有两个 NIC 能同时使用，因而无法达到最大绑定吞吐量。来自一个 VM 的数据包无法在多个 VM 之间拆分。



#### 4.3.5.4. 交换机配置

根据您的冗余需求，可以将绑定中的每个 NIC 连接到同一个或不同的堆栈交换机。如果将其中一个 NIC 连接到另一个冗余的交换机，则在某个 NIC 或交换机出现故障时，通信将故障转移到另一个 NIC。添加第二个交换机可以通过以下方式防止配置中出现单点故障：

- 当您绑定管理接口中的某个链路连接到第二个交换机时，如果此交换机出现故障，管理网络依然保持联机，而且主机依然能够相互通信。
- 当您将一个链路（适用于任何通信类型）连接到第二个交换机时，如果 NIC 或交换机出现故障，虚拟机仍保持在网络上，因为它们的通信会故障转移到另一个 NIC/交换机。

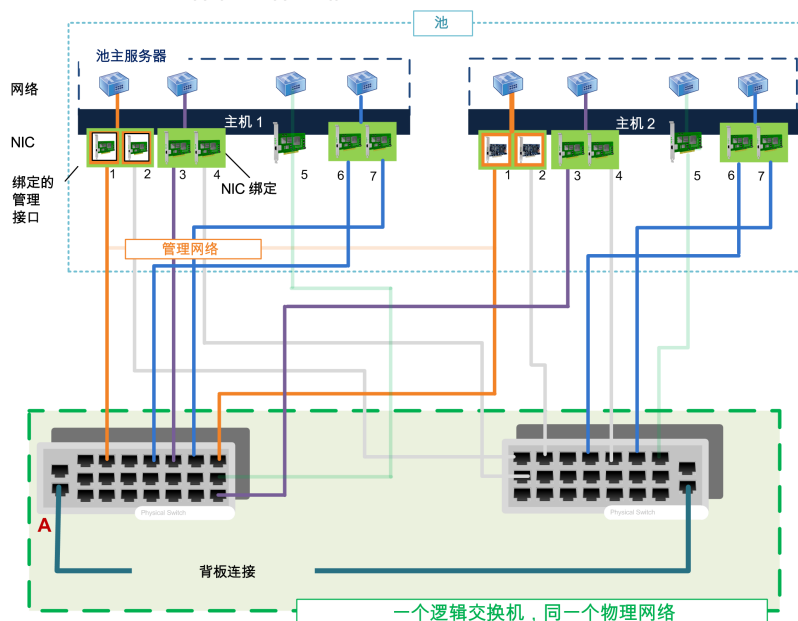
当您想将绑定的 NIC 连接到多个交换机并且已配置 LACP 绑定模式，必须使用堆栈交换机。术语堆栈交换机指的是将多个物理交换机配置为一个逻辑交换机。您必须根据交换机制造商提供的指南，利用交换机管理软件，以物理方式将交换机连接在一起，使多个交换机作为一个逻辑交换装置运行。通常情况下，交换机堆栈仅通过专有扩展实现，而且交换机供应商可能会使用不同的术语宣传此功能。



#### 注意

如果使用主动-主动绑定时遇到问题，可能需要使用堆栈交换机。主动-被动绑定不需要堆栈交换机。

下图说明绑定中 NIC 的电缆和网络配置的配对情况。



该图说明一个绑定对中的两个 NIC 如何使用相同的网络设置（由每个主机中的网络表示）。绑定的两个 NIC 连接到不同的交换机以实现冗余。

##### 4.3.5.4.1. 使用 LACP 绑定时的交换机配置

虽然交换机配置的具体详情因制造商而异，但配置交换机以便与 LACP 绑定配套使用时，需要记住几个关键点：

- 交换机必须支持 LACP 和 IEEE 802.3ad 标准。
- 在交换机上创建 LAG 组时，必须为主机上的每个 LACP 绑定创建一个 LAG 组。这意味着，如果您有一个五主机池，并且已在每个主机上的 NIC 4 和 5 上创建了 LACP 绑定，则必须在交换机上创建五个 LAG 组。每个端口集的一个组对应于主机上的 NIC。



您可能还需要将 VLAN ID 添加到 LAG 组中。

- XenServer LACP 绑定要求将 LAG 组中的“静态模式”设置设为已禁用。

如先前在[第 4.3.5.4 节“交换机配置”](#)中提及的一样，需要通过堆栈交换机将 LACP 绑定连接到多个交换机。

#### 4.3.6. 设置后的初始网络配置

XenServer 主机的网络配置是在主机初始安装期间指定的。IP 地址配置（DHCP/静态）、用作管理接口的 NIC 和主机名等选项是根据安装期间提供的值设置的。

当主机包含多个 NIC 时，安装后进行的配置取决于安装期间选择用于管理操作的 NIC：

- 为主机中的每个 NIC 创建 PIF
- 使用安装期间指定的 IP 地址选项配置选择用作管理接口的 NIC 的 PIF
- 为每个 PIF 创建网络（“network 0”、“network 1”等等）
- 将每个网络分别连接到一个 PIF
- 不配置所有其他 PIF 的 IP 地址选项

当 XenServer 主机包含单一 NIC 时，安装后将进行以下配置：

- 创建对应于该主机单一 NIC 的一个 PIF
- 使用安装期间指定的 IP 地址选项配置 PIF 并启用主机的管理
- 设置 PIF 以用于主机管理操作
- 创建单一网络，即网络 0
- 将网络 0 连接到 PIF 以启用到 VM 的外部连接

在这两种情况下，生成的网络配置允许 XenServer、xe CLI 及任何其他在独立计算机上运行的管理软件通过管理接口的 IP 地址连接到 XenCenter 主机。配置还为在主机上创建的 VM 提供外部网络。

用于管理操作的 PIF 是在 XenServer 安装期间唯一使用 IP 地址配置的 PIF。VM 的外部网络通过使用网络对象（用作虚拟以太网交换机）将 PIF 桥接到 VIF 来实现。

网络功能（如 VLAN、NIC 绑定和将 NIC 专用于存储通信）的步骤将在后面各节中介绍。

#### 4.3.7. 更改网络配置

您可以通过修改网络对象，更改网络配置。为此，请运行影响网络对象或 VIF 的命令。

##### 4.3.7.1. 修改网络对象

您可以使用 `xe network-param-set` 命令及其相关参数修改网络的不同方面，例如帧大小（MTU）、名称标签、名称说明以及其他值。

运行 `xe network-param-set` 命令时，唯一必需参数是 `uuid`。

可选参数为：

- `default_locking_mode`。请参阅[第 4.4.10.8.10 节“简化云中的 VIF 锁定模式配置”](#)。



- name-label
- name-description
- MTU
- other-config:

如果未给定参数值，此参数将被设为空值。要在映射参数中设置(键，值)对，请使用语法“map-param:key=value”。

#### 4.3.8. 更改绑定的最高延迟

如第 4.3.5 节 “NIC 绑定”中所述，默认情况下，为绑定设置 31000ms 的最高延迟，以免通信在发送故障后重新平衡到 NIC 上。虽然看似较长，但最高延迟对于所有绑定模式都非常重要，而不仅限于主动-主动模式。

但是，如果了解为环境选择的相应的设置，可以使用下面的过程更改绑定的最高延迟。

更改绑定的最高延迟：

1. 设置最高延迟（毫秒）：

```
xe pif-param-set uuid=<<uuid of bond master PIF>> other-config:bond-updelay=<<delay in ms>>
```

2. 要使更改生效，必须拔出物理接口，然后重新插入：

```
xe pif-unplug uuid=<<uuid of bond master PIF>>
```

```
xe pif-plug uuid=<<uuid of bond master PIF>>
```

### 4.4. 管理网络配置

根据您将要配置的服务器是独立服务器还是资源池中的服务器，本部分介绍的某些网络配置步骤将有所不同。

#### 4.4.1. 跨服务器专用网络

先前版本的 XenServer 允许您创建单服务器专用网络，该网络允许同一主机上运行的 VM 彼此通信。跨服务器专用网络功能，对单服务器专用网络这一概念进行了扩展，允许不同主机上的 VM 彼此通信。跨服务器专用网络将单服务器专用网络的相同隔离属性与在资源池中分布主机的额外功能结合在一起。实现这种结合后，VM 可以通过连接跨服务器专用网络来使用各种 VM 灵活性功能（例如，XenMotion 实时迁移）。

跨服务器专用网络是彻底隔离的。未连接该专用网络的 VM 无法探查通信流或将通信流注入到网络中，即使它们位于同一个物理主机上，并且通过同一个基础物理网络设备 (PIF) 上的 VIF 连接到网络。VLAN 提供相似的功能，但与 VLAN 不同，跨服务器专用网络通过使用通用路由封装 (GRE) IP 隧道协议，无需配置物理交换机光纤即可提供隔离。

专用网络无需物理交换机即可提供以下益处：

- 单服务器专用网络的隔离属性
- 跨资源池分布功能，允许连接到同一个专用网络的 VM 依赖于同一资源池中的多个主机
- 与 XenMotion 等功能兼容

跨服务器专用网络必须在管理接口或辅助接口上创建，因为这些网络需要可进行 IP 寻址的 NIC。任何支持 IP 的 NIC 都可用作底层网络传输。如果选择将跨服务器专用网络通信放置在辅助接口上，该辅助接口必须在单独的子网上。

如果任何管理或辅助接口在同一子网上，通信将不会被正确路由。



#### 注意

要创建跨服务器专用网络，必须满足以下条件：

- 池中的所有主机必须使用 XenServer 6.0 或更高版本。
- 池中的所有主机都必须将 vSwitch 用于网络堆栈。
- vSwitch Controller 必须正在运行，并且必须已添加池。（池必须配置 vSwitch Controller，用以处理 vSwitch 连接所需的初始化和配置任务。）
- 跨服务器专用网络必须在配置为管理接口的 NIC 上创建。该 NIC 可以是管理接口，也可以是专门为该目的而配置的辅助接口（支持 IP 的 PIF），但前提是它位于一个单独的子网上。

有关配置 vSwitch 的详细信息，请参阅《XenServer vSwitch 控制器用户指南》。有关使用 UI 配置专用网络的过程，请参阅 XenCenter 帮助。

### 4.4.2. 在独立服务器中创建网络

安装主机期间会为每个 PIF 创建外部网络，因此仅在执行下列操作时才需要创建其他网络：

- 使用专用网络
- 支持高级操作，如 VLAN 或 NIC 绑定

要使用 XenCenter 添加或删除网络，请参阅 XenCenter 联机帮助。

使用 CLI 添加新网络：

1. 打开 XenServer 主机文本控制台。
2. 使用 network-create 命令创建网络，该命令将返回新创建的网络的 UUID：

```
xe network-create name-label=<mynetwork>
```

此时，该网络尚未连接到 PIF，因此属于内部网络。

### 4.4.3. 在资源池中创建网络

虽然在将 XenServer 主机加入资源池时，并未强制要求资源池中的所有 XenServer 主机都必须具有同等数量的物理网络接口卡 (NIC)，但这些主机应该符合这一要求。

使池中的 XenServer 主机具有相同的物理网络配置是非常重要的，因为池中的所有主机共享通用的 XenServer 网络集。根据设备名称将各个主机上的 PIF 连接到池范围内的网络。例如，池中具有 eth0 NIC 的所有 XenServer 主机都将相应的 PIF 插入池范围内的 Network 0 网络。这对于具有 eth1 NIC 和 Network 1 的主机以及至少在池中的一个 XenServer 主机上出现的其他 NIC 同样适用。

如果某个 XenServer 主机具有的 NIC 数量与池中其他主机具有的 NIC 数量不同，情况就复杂了，因为并不是所有池网络对所有池主机都有效。例如，如果主机 host1 和 host2 位于同一池中，但 host1 有四个 NIC，而 host2 只有两个 NIC，则只有连接到与 eth0 和 eth1 对应的 PIF 的网络才对 host2 有效。VIF 连接到与 eth2 和 eth3 对应的网络的 host1 将无法迁移到主机 host2。

#### 4.4.4. 创建 VLAN

对于资源池中的服务器，您可以使用 `pool-vlan-create` 命令。该命令创建 VLAN，并在池中的主机上自动创建和插入所需的 PIF。有关详细信息，请参阅第 4.25.2 节“[pool-vlan-create](#)”。

使用 CLI 将网络连接到外部 VLAN:

1. 打开 XenServer 主机控制台。
2. 创建用于 VLAN 的新网络。将返回新网络的 UUID：

```
xe network-create name-label=network5
```

3. 使用 `pif-list` 命令找到与物理 NIC (支持所需的 VLAN 标记) 对应的 PIF 的 UUID。返回所有 PIF 的 UUID 和设备名称，其中包括任何现有的 VLAN：

```
xe pif-list
```

4. 创建 VLAN 对象，该对象指定要连接到新 VLAN 的所有 VM 上所需的物理 PIF 和 VLAN 标记。创建新的 PIF 并将其插入指定的网络。返回新 PIF 对象的 UUID。

```
xe vlan-create network-uuid=<network_uuid> pif-uuid=<pif_uuid> vlan=5
```

5. 将 VM VIF 连接到新网络。有关详细信息，请参阅第 4.4.2 节“[在独立服务器中创建网络](#)”。

#### 4.4.5. 在独立主机上创建 NIC 绑定

Citrix 建议您使用 XenCenter 创建 NIC 绑定。有关说明，请参阅 XenCenter 帮助。

本节介绍如何使用 `xe` CLI 绑定不属于池的 XenServer 主机上的 NIC 接口。有关使用 `xe` CLI 在组成资源池的第 4.4.6 节“[在资源池中创建 NIC 绑定](#)”主机上创建 NIC 绑定的详细信息，请参阅 XenServer。

##### 4.4.5.1. 创建 NIC 绑定

在绑定 NIC 时，绑定会包含当前用作管理接口的 PIF/NIC。从 XenServer 6.0 开始，管理接口会自动移动到该绑定的 PIF。

绑定两个或四个 NIC:

1. 使用 `network-create` 命令创建用于绑定的 NIC 的新网络。将返回新网络的 UUID：

```
xe network-create name-label=<bond0>
```

2. 使用 `pif-list` 命令确定要在绑定中使用的 PIF 的 UUID：

```
xe pif-list
```

3. 执行以下操作之一：

- 要在主动-主动模式 (默认模式) 下配置绑定，请使用 `bond-create` 命令创建绑定。指定新创建的网络 UUID 和要绑定的各 PIF 的 UUID，以逗号分隔各参数：

```
xe bond-create network-uuid=<network_uuid> pif-uuids=<pif_uuid_1>,<pif_uuid_2>,<pif_uuid_3>,<pif_uuid_4>
```

如果绑定两个 NIC，则输入两个 UUID；如果绑定四个 NIC，则输入四个 UUID。在运行该命令之后，将返回该绑定的 UUID。

- 要在主动-被动或 LACP 绑定模式下配置绑定，请使用相同的语法，但是添加可选的 `mode` 参数并指定 `lacp` 或 `active-backup`：

```
xe bond-create network-uuid=<network_uuid> pif-uuids=<pif_uuid_1>,<pif_uuid_2>,<pif_uuid_3>,<pif_uuid_4> /
mode=<balance-slb | active-backup | lacp>
```



#### 注意

在早期版本中，可通过指定 `other-config:bond-mode` 来更改绑定模式。虽然此命令依然可用，但在未来版本中可能不受支持，而且不像 `mode` 参数那样高效。`other-config:bond-mode` 要求运行 `pif-unplug` 和 `pif-plug`，以使模式更改生效。

#### 4.4.5.2. 控制绑定的 MAC 地址

绑定管理接口时，该绑定将包含当前正用作管理接口的 PIF/NIC。如果主机使用 DHCP，则多数情况下，该绑定的 MAC 地址与当前正使用的 PIF/NIC 的 MAC 地址相同，而且管理接口的 IP 地址可能会保持不变。

可以更改该绑定的 MAC 地址，使其不同于（当前）管理接口 NIC 的 MAC 地址。但是，当启用了该绑定而正在使用的 MAC/IP 地址发生更改时，主机的现有网络会话将停止。

可以通过以下两种方式控制该绑定的 MAC 地址：

- 可以在 `bond-create` 命令中指定 `mac` 可选参数。使用此参数可以将该绑定的 MAC 地址设置为任意地址。
- 从 XenServer 7.0 开始，如果未指定 `mac` 参数，XenServer 将使用管理接口的 MAC 地址（如果管理接口是绑定中的接口之一）。如果管理接口不是绑定的一部分，但另一个管理接口是，则绑定将使用后者的 MAC 地址（以及 IP 地址）。如果绑定中的所有 NIC 都不是管理接口，则绑定将使用第一个指定 NIC 的 MAC 地址。

#### 4.4.5.3. 还原 NIC 绑定

请注意，如果将 XenServer 主机还原为非绑定配置，`bond-destroy` 命令会自动将主-从属节点配置为要用于管理接口的接口。因此，所有的 VIF 都将移动到管理接口。

术语“主-从属节点”是指在创建绑定时用来复制 MAC 和 IP 配置的 PIF。在绑定两个 NIC 时，主-从属节点是：

1. 管理接口 NIC（如果管理接口是绑定的 NIC 之一）。
2. 任何其他具有 IP 地址的 NIC（如果管理接口不是绑定的一部分）。
3. 第一个指定的 NIC。可以通过运行以下命令来确定它是哪个 NIC：

```
xe bond-list params=all
```

#### 4.4.6. 在资源池中创建 NIC 绑定

如果可能，请尽量在创建初始资源池的过程中先创建 NIC 绑定，然后再将其他主机加入池或创建 VM。这样，可以在主机加入池时将绑定配置自动复制到其中，并减少所需的步骤。将 NIC 绑定添加到现有池中需要执行下列操作之一：

- 使用 CLI 在主服务器上配置绑定，然后在池的每个成员上配置绑定。
- 使用 CLI 在主服务器上配置绑定，然后重新启动池的每个成员，以便从池主服务器继承设置。
- 使用 XenCenter 在主服务器上配置绑定。XenCenter 会自动将成员服务器上的网络连接设置与主服务器同步，这样，您就不需要重新引导成员服务器。

为简单起见并且为了避免错误配置，Citrix 建议使用 XenCenter 创建 NIC 绑定。有关详细信息，请参阅 XenCenter 帮助。

本部分介绍如何使用 xe CLI 在组成资源池的 XenServer 主机上创建绑定的 NIC 接口。有关使用 xe CLI 在独立的 [第 4.4.5.1 节“创建 NIC 绑定”](#) 主机上创建 NIC 绑定的详细信息，请参阅 XenServer。



#### 警告

不要在启用高可用性功能的情况下尝试创建网络绑定。创建绑定的过程将干扰正在进行的高可用性功能信号检测并导致主机进行自我保护（自行关闭）；随后这些主机可能无法正常重新引导，并且需要使用 `host-emergency-ha-disable` 命令才能恢复。

#### 4.4.6.1. 将 NIC 绑定添加到新资源池

1. 选择要作为主服务器的主机。默认情况下，主服务器主机属于未命名的池。要使用 CLI 创建资源池，请重命名现有的无名称的池：

```
xe pool-param-set name-label=<"New Pool"> uuid=<pool_uuid>
```

2. 按照 [第 4.4.5.1 节“创建 NIC 绑定”](#) 中的说明创建 NIC 绑定。
3. 打开要加入到池的主机中的控制台，然后运行命令：

```
xe pool-join master-address=<host1> master-username=root master-password=<password>
```

网络和绑定信息将自动复制到新主机中。管理接口会自动从最初配置给绑定 PIF 时所在的主机 NIC 移出（即，管理接口现在包含到绑定中，使整个绑定充当管理接口）。

- 使用 `host-list` 命令找到正在配置的主机的 UUID：

```
xe host-list
```

#### 4.4.6.2. 将 NIC 绑定添加到现有池



#### 警告

不要在启用高可用性功能的情况下尝试创建网络绑定。创建绑定的过程会干扰正在进行的高可用性功能信号检测并导致主机进行自我保护（自行关闭）；随后这些主机可能无法正常重新引导，并且需要运行 `host-emergency-ha-disable` 命令才能恢复。



#### 注意

如果您不是使用 XenCenter 进行 NIC 绑定，创建池范围内的 NIC 绑定的最快方法是在主服务器上创建绑定，然后重启其他池成员。或者您可以使用 `service xapi restart` 命令。这会使每台主机继承主服务器上的绑定和 VLAN 设置。但是，每台主机的管理接口必须手动重新配置。

按照以上几节所述的过程创建 NIC 绑定，请参阅 [第 4.4.6.1 节“将 NIC 绑定添加到新资源池”](#)。

#### 4.4.7. 配置专用存储 NIC

可以使用 XenCenter 或 xe CLI 为 NIC 分配 IP 地址，并将其专用于特定功能（如存储通信）。在为 NIC 配置 IP 地址时，可以通过创建辅助接口来进行配置。（用于管理且支持 IP 的 NIC XenServer 称为管理接口。）

当您希望将辅助接口专用于特定用途时，必须确保拥有适当的网络配置，以确保 NIC 仅用于所需的通信。例如，要将 NIC 专用于存储通信，则必须对 NIC、存储目标、交换机和/或 VLAN 进行配置，以便只能通过指定的 NIC 访问此目标。如果您的物理配置和 IP 配置不限制可以通过存储 NIC 发送的通信，则可以通过辅助接口来发送其他通信（如管理通信）。

为存储通信创建新的辅助接口时，必须为其指定符合以下条件的 IP 地址：(a) 与存储控制器位于同一子网上（如果适用），且 (b) 不与任何其他辅助接口或管理接口位于同一子网上。

配置辅助接口时，每个辅助接口必须位于单独的子网上。例如，如果想为存储配置两个额外的辅助接口，则要求 IP 地址位于三个不同子网上 – 一个用于管理接口的子网，一个用于辅助接口 1 的子网，以及一个用于辅助接口 2 的子网。

如果正在使用绑定为存储通信提供恢复能力，您可能考虑使用 LACP 而非 Linux 桥接绑定。要使用 LACP 绑定，必须将 vSwitch 配置为网络连接堆栈。有关详细信息，请参阅第 4.2 节“vSwitch 网络”。



#### 注意

在选择要配置为辅助接口以与 iSCSI 或 NFS SR 一起使用的 NIC 时，请确保该专用 NIC 使用不能从管理接口路由的单独 IP 子网。如果无法确保上述条件，则由于初始化网络接口的顺序，很可能在主机重新引导后通过主管理接口定向存储通信。

使用 xe CLI 分配 NIC 功能:

1. 确保 PIF 位于独立子网中或将路由配置为适合您的网络拓扑，以通过所选 PIF 实现所需通信。
2. 为 PIF 设置 IP 配置，为模式参数添加适当的值；如果使用静态 IP 寻址，还要设置 IP、子网掩码、网关和 DNS 参数：

```
xe pif-reconfigure-ip mode=<DHCP | Static> uuid=<pif-uuid>
```

3. 将 PIF 的 disallow-unplug 参数设置为 true：

```
xe pif-param-set disallow-unplug=true uuid=<pif-uuid>
```

```
xe pif-param-set other-config:management_purpose="Storage" uuid=<pif-uuid>
```

如果希望将辅助接口用于也可以从管理接口路由的存储（请记住，此配置并非最佳实践），则有两种选择：

- 主机重新启动后，确保辅助接口正确配置，并使用 xe pbd-unplug 和 xe pbd-plug 命令重新初始化主机上的存储连接。这会重新启动存储连接并通过正确的接口对其进行路由。
- 或者，您可以使用 xe pif-forget 从 XenServer 数据库中删除存储接口，然后在控制域中手动配置该接口。这是高级选项，要求您熟悉如何手动配置 Linux 网络。

#### 4.4.8. 使用已启用 SR-IOV 的 NIC

Single Root I/O Virtualization (SR-IOV) 是一种 PCI 设备虚拟化技术，允许单一 PCI 设备在物理 PCI 总线上显示为多个 PCI 设备。实际物理设备称为物理功能 (PF)，而其他设备称为虚拟功能 (VF)。这样做的目的在于让虚拟机管理程序使用 SR-IOV 技术直接为一个虚拟机 (VM) 分配一个或多个 VF：来宾系统随后会将这些 VF 用作任何其他直接分配的 PCI 设备。

如果将一个或多个 VF 分配给一个 VM，该 VM 将可以直接利用硬件。每个 VM 在经过这种配置后，就像直接使用 NIC 一样，这样会降低处理开销并改善性能。



#### 警告

如果您的 VM 有一个 SR-IOV VF，则需要移动 VM 的功能（例如实时迁移、池滚动升级、高可用性和灾难恢复）将不可用。这是由于 VM 直接绑定到支持 SR-IOV 的物理 NIC VF。此外，通过 SR-IOV VF 发送的 VM 网络通信会绕过 vSwitch，因此无法创建 ACL 或查看 QoS。

为 VM 分配 SR-IOV NIC VF:



#### 注意

只有 XenServer [硬件兼容性列表](#) 上所列的支持 SR-IOV 的 NIC 才支持 SR-IOV 技术，而且该技术仅适用于 Windows Server 2008 来宾操作系统。



1. 在 XenServer 主机上打开本地命令 shell。
2. 运行 lspci 命令以显示虚拟功能 (VF) 的列表。例如：

```
07:10.0 Ethernet controller: Intel Corporation 82559 \
Ethernet Controller Virtual Function (rev 01)
```

在上例中，07:10.0 是 VF 的 bus:device.function 地址。

3. 通过运行以下命令为目标 VM 分配所需的 VF：

```
xe vm-param-set other-config:pci=0/0000:<bus:device.function> uuid=<vm-uuid>
```

4. 启动 VM，然后为特定的硬件安装相应的 VF 驱动程序。



#### 注意

可以为一个 VM 分配多个 VF，但是不能在多个 VM 之间共享同一个 VF。

### 4.4.9. 控制传出数据的速率 (QoS)

要限制 VM 每秒可以发送的传出数据量，可以针对 VM 虚拟接口 (VIF) 设置可选的服务质量 (QoS) 值。该设置允许您为传出的数据包指定最大传输速率（以每秒千字节为单位）。

QoS 值将限制源自 VM 的传输速率。QoS 设置不会限制 VM 可以接收的数据量。如果需要限制接收量，Citrix 建议您限制网络中较高层（例如，交换机层）传入数据包的速率。

根据在池中配置的网络堆栈，可以按照下表中的说明，在以下两个位置之一针对 VM 虚拟接口 (VIF) 设置服务质量 (QoS) 值：a) 在 vSwitch 控制器上；b) 在 XenServer 中（使用 CLI 或 XenCenter）：

网络堆栈	可用的配置方法
vSwitch	<ul style="list-style-type: none"> <li>• vSwitch 控制器。如果 vSwitch 是网络堆栈，则使用控制器是为 VIF 设置最大传输速率的首选方法。在使用 vSwitch 堆栈时，XenCenter QoS 选项不可用。</li> <li>• xe 命令。可以使用下例中的命令设置 QoS 传输速率。但是，首选方法是通过 vSwitch 控制器 UI，因为 vSwitch 控制器 UI 提供更细化的控制。</li> </ul>
Linux 桥	<ul style="list-style-type: none"> <li>• XenCenter。可以在虚拟接口的属性对话框中设置 QoS 传输速率限制值。</li> <li>• xe 命令。可以在 CLI 中使用下一节介绍的命令来设置 QoS 传输速率。</li> </ul>



#### 重要

当 vSwitch 配置为网络堆栈时，可能会无意中在 vSwitch 控制器上和 XenServer 主机内部都配置了 QoS 值。在这种情况下，XenServer 会使用您设置的最低速率来限制传出通信。

用于设置 QoS 的 CLI 命令的示例：

要使用 CLI 将 VIF 的最大传输速率限制为 100 kb/s，请使用 vif-param-set 命令：

```
xe vif-param-set uuid=<vif_uuid> qos_algorithm_type=ratelimit
xe vif-param-set uuid=<vif_uuid> qos_algorithm_params:kbps=100
```

**注意**

如果要使用 vSwitch 控制器，Citrix 建议在 vSwitch 控制器中而不是使用上面的 CLI 命令来设置传输速率限制。有关在 vSwitch 控制器中设置 QoS 速率限制的说明，请参阅《vSwitch 控制器用户指南》。

#### 4.4.10. 更改网络连接配置选项

本部分讨论如何更改 XenServer 主机的网络配置，其中包括：

- 更改主机名（即，域名系统 (DNS) 名称）
- 添加或删除 DNS 服务器
- 更改 IP 地址
- 更改将哪个 NIC 用作管理接口
- 将新的物理 NIC 添加到服务器
- 启用 ARP 过滤（交换机端口锁定）

##### 4.4.10.1. 主机名

系统主机名（又称为域名或 DNS 名称）在池级数据库中定义，可以使用以下 `xe host-set-hostname-live` CLI 命令进行修改：

```
xe host-set-hostname-live host-uuid=<host_uuid> host-name=<host-name>
```

基础控制域主机名也将动态更改以反映新主机名。

##### 4.4.10.2. DNS 服务器

要在 XenServer 主机的 IP 寻址配置中添加或删除 DNS 服务器，请使用 `pif-reconfigure-ip` 命令。例如，对于具有静态 IP 的 PIF：

```
pif-reconfigure-ip uuid=<pif_uuid> mode=static DNS=<new_dns_ip>
```

##### 4.4.10.3. 更改独立主机的 IP 地址配置

网络接口配置可以使用 `xe` CLI 进行更改。请勿直接修改基础网络配置脚本。

要修改 PIF 的 IP 地址配置，请使用 `pif-reconfigure-ip` CLI 命令。有关 `pif-reconfigure-ip` 命令参数的详细信息，请参阅第 4.14.5 节“[pif-reconfigure-ip](#)”。

**注意**

有关更改资源池中的主机 IP 地址的详细信息，请参阅第 4.4.10.4 节“[更改资源池中的 IP 地址配置](#)”。

##### 4.4.10.4. 更改资源池中的 IP 地址配置

资源池中的 XenServer 主机具有用于管理和与池中的其他主机之间的相互通信的单一管理 IP 地址。对于主节点主机和其他主机来说，更改主机的管理接口的 IP 地址所需的步骤是不同的。

**注意**

更改服务器的 IP 地址和其他网络参数时要十分谨慎。根据网络拓扑和进行的更改，可能会丢失网络存储连接。如果丢失了网络存储连接，必须使用 XenCenter 中的修复存储功能或使用 CLI 执行



pbid-plug 命令来重新连接该存储。因此，建议先将 VM 从服务器中迁移出来，然后再更改服务器的 IP 配置。

更改成员主机（非池主服务器）的 IP 地址：

1. 根据需要，使用 pif-reconfigure-ip CLI 命令设置 IP 地址。有关 pif-reconfigure-ip 命令参数的详细信息，请参阅[命令行接口](#)：

```
xe pif-reconfigure-ip uuid=<pif_uuid> mode=DHCP
```

2. 使用 host-list CLI 命令，以通过检查池中的所有其他 XenServer 主机是否均可见，来确认该成员主机是否已成功重新连接到主服务器主机：

```
xe host-list
```

每个成员主机均使用池主服务器主机建议的 IP 地址进行通信，当其 IP 地址发生更改后，将不知如何与主服务器主机联系。因此，更改主服务器 XenServer 主机的 IP 地址还需要其他步骤。

请尽可能使用在池主服务器的池生命周期内不会轻易发生更改的专用 IP 地址。

更改池主服务器的 IP 地址：

1. 根据需要，使用 pif-reconfigure-ip CLI 命令设置 IP 地址。有关 pif-reconfigure-ip 命令参数的详细信息，请参阅[命令行接口](#)：

```
xe pif-reconfigure-ip uuid=<pif_uuid> mode=DHCP
```

2. 池主服务器主机的 IP 地址发生更改后，所有成员主机在无法与主服务器主机联系时，都将进入紧急模式。
3. 在主服务器 XenServer 主机上，使用 pool-recover-slaves 命令强制主服务器与每个成员主机联系并通知它们新的主服务器 IP 地址：

```
xe pool-recover-slaves
```

#### 4.4.10.5. 管理接口

在具有多个 NIC 的主机上安装 XenServer 后，会选择其中的一个 NIC 作为管理接口。管理接口用于 XenCenter 与主机之间的连接以及主机到主机的通信。

更改用作管理接口的 NIC：

1. 使用 pif-list 命令确定哪个 PIF 与用作管理接口的 NIC 对应。将返回每个 PIF 的 UUID。

```
xe pif-list
```

2. 使用 pif-param-list 命令验证将用作管理接口的 PIF 的 IP 寻址配置。如有必要，请使用 pif-reconfigure-ip 命令配置要使用的 PIF 的 IP 寻址。有关可供 pif-reconfigure-ip 命令使用的选项的更多详细信息，请参阅[命令行接口](#)。

```
xe pif-param-list uuid=<pif_uuid>
```

3. 使用 host-management-reconfigure CLI 命令更改用作管理接口的 PIF。如果此主机属于资源池，则必须在成员主机控制台上发出此命令：

```
xe host-management-reconfigure pif-uuid=<pif_uuid>
```



#### 警告

不支持将管理接口放置到 VLAN 网络中。

#### 4.4.10.6. 禁用管理访问

要完全禁用对管理控制台的远程访问，请使用 `host-management-disable` CLI 命令。



##### 警告

禁用管理接口后，必须登录物理主机控制台来执行管理任务，XenCenter 等外部界面将不再起作用。

#### 4.4.10.7. 添加新的物理 NIC

以通常方式在 XenServer 主机上安装新的物理 NIC。然后，重新启动服务器后，运行 `xe CLI` 命令 `pif-scan` 以为新的 NIC 创建新的 PIF 对象。

#### 4.4.10.8. 使用交换机端口锁定

使用 XenServer 交换机端口锁定功能，您可以限制未知、不受信任或可能存有敌意的 VM 假装拥有未分配给它们的 MAC 或 IP 地址的能力，从而控制从这些 VM 发送的流量。您可以使用此功能中的端口锁定命令，在默认情况下阻止网络上的所有流量，也可以定义特定 IP 地址，以便允许个别 VM 从中发送流量。

交换机端口锁定功能是为环境中关注内部威胁的公有云服务提供商设计的。此功能可以为公有云服务提供商提供帮助，在这些提供商的网络架构中，每个 VM 都有一个与 Internet 连接的公用 IP 地址。由于云租户往往是不可信任的，因此，可能需要采取一些安全措施（例如，欺骗防御）来确保租户无法攻击云中的其他虚拟机。

通过交换机端口锁定功能，您可以使所有租户或来宾系统使用同一个第 2 层网络，从而简化网络配置。

端口锁定命令的一种最重要的功能就是，它们可以限制不受信任的来宾系统可以发送的流量，进而限制来宾系统假装拥有实际并不拥有的 MAC 或 IP 地址的能力。具体而言，您可以使用这些命令防止来宾系统：

- 申请 XenServer 管理员指定可以使用的 IP 或 MAC 地址以外的 IP 或 MAC 地址
- 拦截、欺骗或中断其他 VM 的流量

##### 4.4.10.8.1. 要求

- XenServer 交换机端口锁定功能支持 Linux 桥接和 vSwitch 网络连接堆栈。
- 如果在您的环境中启用了基于角色的访问控制 (RBAC)，则配置交换机端口锁定功能的用户必须使用至少拥有池操作员或池管理员角色的帐户进行登录。如果在您的环境中未启用 RBAC，则用户必须使用池主服务器的 `root` 帐户进行登录。
- 运行交换机端口锁定命令时，网络可以联机或脱机。
- 在 Windows 来宾系统中，断开连接的网络图标仅在 XenServer Tools 安装在来宾系统中时显示。

##### 4.4.10.8.2. 备注

如果没有任何交换机端口锁定配置，VIF 将被设为“`network_default`”，网络被设为“`unlocked`”。

在环境中使用 vSwitch 控制器和其他第三方控制器时，配置交换机端口锁定不受支持。

交换机端口锁定不阻止云租户：

- 在另一个租户/用户上执行 IP 层攻击。但是，如果它们尝试使用以下手段执行 IP 层攻击并且配置交换机端口锁定：a) 模拟云中的另一个租户或用户，或者 b) 启动对面向另一用户的通信的拦截。

- 耗尽网络资源。
- 通过正常的交换机阻塞行为接收一些面向其他虚拟机的通信（对于广播 MAC 地址或未知目标 MAC 地址）。

同样，交换机端口锁定不限制 VM 向哪里发送通信。

#### 4.4.10.8.3. 实施说明

您可以使用命令行或 XenServer API 实施交换机端口锁定功能。但是，在注重自动化的大环境中，最典型的实施方法可能是使用 API。

#### 4.4.10.8.4. 示例

本节提供了交换机端口锁定如何阻止特定类型的攻击的示例。在这些示例中，VM-c 是一个敌对租户（租户 C）租用和用于攻击的虚拟机。VM-a 和 VM-b 是非攻击性租户租用的虚拟机。

##### 示例 1：交换机端口锁定如何阻止 ARP 欺骗防御

ARP 欺骗指的是攻击者尝试将他或她的 MAC 地址与另一节点的 IP 地址相关联，这可能会导致节点的通信发送给攻击者。为实现该目标，攻击者将假冒（欺骗）ARP 消息发送给以太网 LAN。

方案：

虚拟机 A (VM-a) 想通过将 IP 通信寻址到虚拟机 B (VM-b)，将 IP 通信从 VM-a 发送到 VM-b。虚拟机 C 的所有者希望使用 ARP 欺骗伪装成他的 VM 即 VM-c（实际是 VM-b）。

1. VM-c 将推测性 ARP 答复流发送到 VM-a。这些 ARP 答复要求将答复 (c\_MAC) 中的 MAC 地址与 IP 地址 b\_IP 关联起来。

结果：因为管理员启用了交换机端口锁定，所以这些数据包全部被丢弃，因为启用交换机端口锁定可以防止模拟。

2. VM-b 将 ARP 答复发送到 VM-a，要求将答复 (b\_MAC) 中的 MAC 地址与 IP 地址 b\_IP 关联起来。

结果：VM-a 接收 VM-b 的 ARP 响应。

##### 示例 2：IP 欺骗防御

IP 地址欺骗是一个通过使用伪造源 IP 地址创建 Internet 协议 (IP) 数据包，隐藏数据包身份的过程。

方案：

租户 C 正在尝试使用其主机 Host-C 在远程系统上执行拒绝服务攻击，隐瞒其身份。

##### 尝试 1

租户 C 将 Host-C 的 IP 地址和 MAC 地址设为 VM-a 的 IP 和 MAC 地址 (a\_IP 和 a\_MAC)。租户 C 指示 Host-C 将 IP 通信发送到远程系统。

结果：Host-C 数据包被丢弃。这是因为管理员启用了交换机端口锁定；Host-C 数据包被丢弃，因为启用交换机端口锁定可以防止模拟。

##### 尝试 2

租户 C 将 Host-C 的 IP 地址设为 VM-a 的 IP 地址 (a\_IP)，并保存其原始 c\_MAC。

租户 C 指示 Host-C 将 IP 通信发送到远程系统。

结果：Host-C 数据包被丢弃。这是因为管理员启用了防止模拟的交换机端口锁定。

### 示例 3：Web 托管

方案：

Alice 是一名基础结构管理员。

租户 B 是她的租户之一，正在从他的 VM 即 VM-b 托管多个网站。每个网站都需要一个在同一虚拟网络接口 (VIF) 上托管的不同 IP 地址。

Alice 将 Host-B 的 VIF 重新配置为锁定到单一 MAC 和多个 IP 地址。

#### 4.4.10.8.5. 交换机端口锁定的工作原理

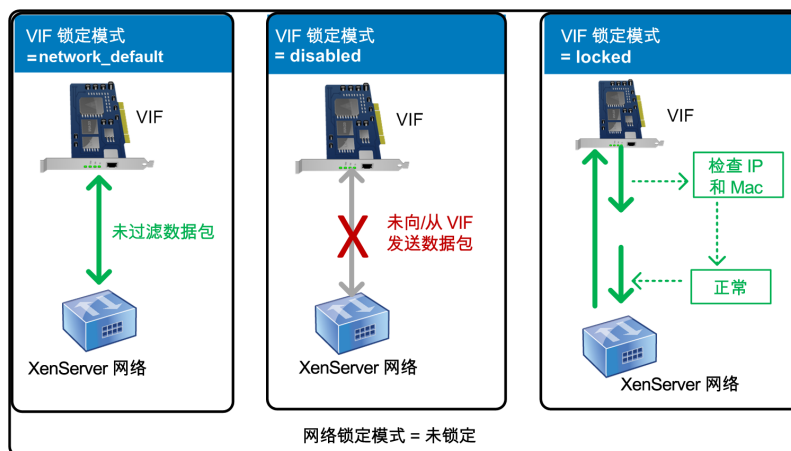
交换机端口锁定功能可以让您在一个或两个层面上控制数据包过滤：

- VIF 层。您在 VIF 上配置的设置可以确定数据包的过滤方式。您可以设置 VIF，使其阻止 VM 发送任何通信，限制 VIF，以便它只能使用分配给它的 IP 地址发送通信，或者允许 VM 将通信发送到连接 VIF 的网络上的任何 IP 地址。
- 网络层。XenServer 网络可以确定数据包的过滤方式。当 VIF 的锁定模式设为 network\_default 时，它引用网络层锁定设置确定允许哪些通信。

无论您使用哪个网络连接堆栈，此功能的运行方式都相同。但是，如以下各节的详细描述一样，Linux 桥接不完全支持 IPv6 中的交换机端口锁定。

#### 4.4.10.8.6. VIF 锁定模式状态

XenServer 交换机端口锁定功能提供一种锁定模式，让您将 VIF 配置为四种不同的状态。这些状态仅在 VIF 插入正在运行的虚拟机中时适用。



此图显示将网络锁定模式设为已解除锁定和配置 VIF 状态时，三种不同的 VIF 锁定模式状态的行为方式。在第一张图中，VIF 状态被设为默认设置，因此，不过滤来自 VM 的通信。在第二张图中，VIF 不发送或接收任何数据包，因为此锁定模式被设为禁用。在第三张图中，VIF 状态被设为锁定，因此，如果这些数据包包含正确的 MAC 和 IP 地址，VIF 只能发送数据包。

- Network\_default。当 VIF 状态被设为 network\_default 时，XenServer 使用网络的 default-locking-mode 参数，确定是否以及如何过滤流经 VIF 的数据包。此行为因相关联的网络将网络默认锁定模式参数设为禁用还是已解除锁定而异。

- default-locking-mode=disabled，XenServer 应用过滤规则，使 VIF 丢弃所有通信。
- default-locking-mode=unlocked，XenServer 删除所有与 VIF 相关联的过滤规则。默认情况下，默认锁定模式参数被设为 unlocked。

有关 default-locking-mode 参数的信息，请参阅第 4.11 节“网络命令”。

网络的默认锁定模式不会影响锁定状态未被设为 network\_default 的已连接 VIF。



#### 注意

无法更改连接了活动 VIF 的网络的 default-locking-mode。

- 锁定。XenServer 应用过滤规则，以便仅允许通过 VIF 向外发送向或从指定 MAC 和 IP 地址发送的通信。在此模式下，如果不指定 IP 地址，VM 则无法通过该 VIF（该网络上的）发送任何通信。

要指定 VIF 将接受来自哪些 IP 地址的通信，请通过使用 ipv4\_allowed 或 ipv6\_allowed 参数使用 IPv4 和/或 IPv6 地址。但是，如果您已配置了 Linux 桥接，则无需输入 IPv6 地址。

XenServer 可以让您在 Linux 桥接处于活动状态时输入 IPv6 地址；但是，XenServer 无法根据输入的 IPv6 地址过滤。（原因是 Linux 桥接没有过滤 Neighbor Discovery Protocol (NDP) 数据包的模块，因此，无法实施完整保护，而且来宾系统能够通过伪造 NDP 数据包模拟另一个来宾系统。）结果，甚至您指定一个 IPv6 地址，XenServer 也可以让所有 IPv6 通信经过 VIF。如果不指定任何 IPv6 地址，XenServer 不会让任何 IPv6 通信经过 VIF。

- 已解除锁定。所有网络通信都可以经过 VIF。也就是说，不对任何传入 VIF 或从 VIF 传出的通信应用过滤器。
- 已禁用。不允许任何通信经过 VIF。（也就是说，XenServer 应用过滤规则，使 VIF 丢弃所有通信。）

#### 4.4.10.8.7. 配置交换机端口锁定

本节提供三种不同的过程：

- 限制 VIF 使用特定 IP 地址
- 向现有的受限列表添加 IP 地址（例如，如果您需要在 VM 正在运行且已连接到网络时向 VIF 添加 IP 地址（例如，如果您临时让网络处于脱机状态））
- 从现有的受限列表删除 IP 地址

如果 VIF 的锁定模式被设为 locked，它只能使用在 ipv4-allowed 或 ipv6-allowed 参数中指定的地址。

因为在某些相对罕见的情况下，VIF 可能有多个 IP 地址，所以可以为 VIF 指定多个 IP 地址。

您可以在插入 VIF（或启动 VM）之前或之后执行这些过程。

限制 VIF 使用特定 IP 地址：

1. 运行以下命令，将默认锁定模式更改为已锁定（如果尚未使用该模式）：

```
xe vif-param-set uuid=<vif-uuid> locking-mode=locked
```

vif-uuid 表示您想允许发送通信的 VIF 的 UUID。要获取 UUID，请在主机上运行 xe vif-list 命令。vm-uuid 指示显示哪个虚拟机的信息。设备 ID 指示 VIF 的设备编号。

2. 运行 vif-param-set 命令，指定虚拟机可以从哪些 IP 地址发送通信。执行以下一项或多项操作：

- 指定一个或多个 IPv4 IP 地址目标。例如：

```
xe vif-param-set uuid=<vif-uuid> ipv4-allowed=<comma separated list of ipv4-addresses>
```

- 指定一个或多个 IPv6 IP 地址目标。例如：

```
xe vif-param-set uuid=<vif-uuid> ipv6-allowed=<comma separated list of ipv6-addresses>
```

您可以指定多个 IP 地址，使用逗号将其隔开，如上例所示。

#### 限制 VIF 使用其他 IP 地址:

执行上述过程限制 VIF 使用特定 IP 地址后，您可以添加一个或多个 VIF 能够使用的 IP 地址。

- 运行 vif-param-add 命令，向现有列表添加 IP 地址。执行以下一项或多项操作：

- 指定 IPv4 IP 地址。例如：

```
xe vif-param-add uuid=<vif-uuid> ipv4-allowed=<comma separated list of ipv4-addresses>
```

- 指定 IPv6 IP 地址。例如：

```
xe vif-param-add uuid=<vif-uuid> ipv6-allowed=<comma separated list of ipv6-addresses>
```

#### 从 VIF 的 IP 地址列表中删除 IP 地址:

如果限制 VIF 使用两个或更多 IP 地址，您可以从列表中删除一个 IP 地址。

- 运行 vif-param-remove 命令，从现有列表中删除 IP 地址。执行以下一项或多项操作：

- 指定要删除的 IPv4 IP 地址。例如：

```
xe vif-param-remove uuid=<vif-uuid> ipv4-allowed=<comma separated list of ipv4-addresses>
```

- 指定要删除的 IPv6 IP 地址。例如：

```
xe vif-param-remove uuid=<vif-uuid> ipv6-allowed=<comma separated list of ipv6-addresses>
```

#### 4.4.10.8.8. 阻止虚拟机从特定网络发送或接收通信

以下过程阻止虚拟机通过特定 VIF 进行通信。因为 VIF 连接到特定的 XenServer 网络，所以您可以使用此过程阻止虚拟机从特定网络发送或接收任何通信。这可以比禁用整个网络提供更精细的控制级别。

如果使用 CLI 命令，无需拨下 VIF 即可设置 VIF 的锁定模式；此命令可以在 VIF 运行（在线）时更改过滤规则。在此情况下，网络连接依然显示为存在；但是，VIF 会丢弃 VM 尝试发送的任何数据包。



#### 提示

要查找 VIF 的 UUID，请在主机上运行 xe vif-list 命令。设备 ID 指示 VIF 的设备编号。

#### 阻止 VIF 接收通信:

- 禁用已连接到您想阻止 VM 从其接收通信的网络的 VIF：

```
xe vif-param-set uuid=<vif-uuid> locking-mode=disabled
```

此外，您也可以通过在 VM 的网络连接选项卡中选择虚拟网络接口，单击取消激活，在 XenCenter 中禁用 VIF。



#### 4.4.10.8.9. 删除 VIF 对 IP 地址的限制

要返回到默认（原始）锁定模式状态，请使用以下过程。默认情况下，创建 VIF 时，XenServer 可以对其进行配置，不限制它使用特定 IP 地址。

使 VIF 返回到已解除锁定状态：

- 运行以下命令，将 VIF 默认锁定模式更改为已解除锁定（如果尚未使用该模式）：

```
xe vif-param-set uuid=<vif_uuid> locking-mode=unlocked
```

#### 4.4.10.8.10. 简化云中的 VIF 锁定模式配置

不对每个 VIF 运行 VIF 锁定模式命令，您可以确保所有 VIF 在默认情况下处于禁用状态。为此，您必须在网络层修改数据包过滤，这会使 XenServer 网络确定数据包的过滤方式，如第 4.4.10.8.5 节“交换机端口锁定的工作原理”中所述。

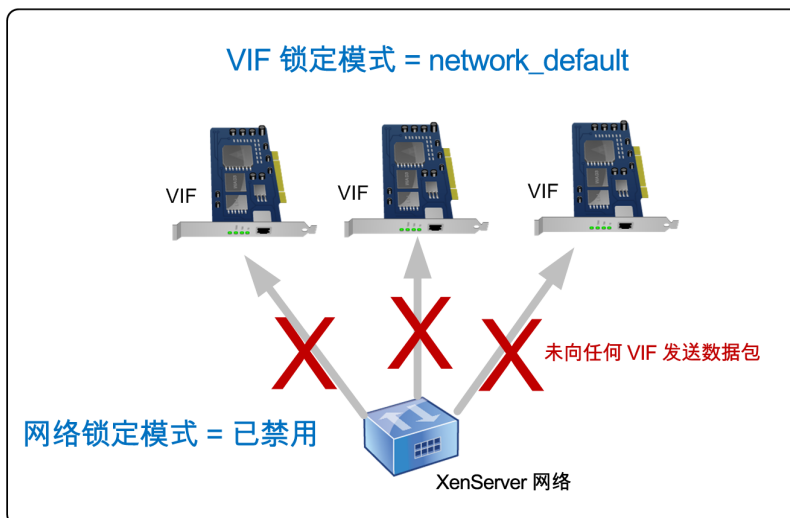
具体而言，网络的 default-locking-mode 设置可以确定采用默认设置的新 VIF 的行为方式。无论 VIF 的 locking-mode 被设为 default，VIF 引用网络锁定模式 (default-locking-mode) 确定是否以及如何过滤流经 VIF 的数据包：

- 已解除锁定。当网络 default-locking-mode 参数被设为 unlocked 时，XenServer 可以让 VM 将通信发送到 VIF 连接到的网络上的任何 IP 地址。
- 已禁用。当 default-locking-mode 参数被设为 disabled 时，XenServer 应用过滤规则，使 VIF 丢弃所有通信。

默认情况下，在 XenCenter 中创建并且使用 CLI 的所有网络的 default-locking-mode 被设为 unlocked。

通过将 VIF 的锁定模式设为其默认设置 (network\_default)，您可以使用此设置为所有连接特定网络的新建 VIF 创建基础默认配置（在网络层）。

此图显示，当 VIF 的 locking-mode 设为其默认设置 (network\_default) 时，VIF 如何使用网络 default-locking-mode 确定其行为。



此图显示，当 VIF 配置为默认设置 (locking-mode=network\_default) 时，如何查看与 default-locking-mode 相关联的设置。在此图中，网络被设为 default-locking-mode=disabled，使任何通信都无法经过 VIF。

例如，默认情况下，由于创建了 VIF 并且它们的 locking-mode 被设为 network\_default，如果设置网络的 default-locking-mode=disabled，您未为其配置锁定模式的任何新 VIF 都将被禁用，直到您 (a) 更改各个

VIF 的 `locking-mode` 参数或者 (b) 将 VIF 的 `locking-mode` 显式设为 `unlocked` 为止 (例如, 如果您足够信任某个特定 VM, 根本不想过滤其通信)。

更改网络的默认锁定模式设置:

- 创建网络后, 运行以下命令, 更改默认锁定模式:

```
xe network-param-set uuid=<network-uuid> default-locking-mode=[unlocked|disabled]
```



#### 注意

要获得网络的 UUID, 请运行 `xe network-list` 命令。此命令在您运行它的主机上显示所有网络的 UUID。

查看网络的默认锁定模式设置:

- 运行以下命令之一:

```
xe network-param-get uuid=<network-uuid> param-name=default-locking-mode
```

或

```
xe network-list uuid=<network-uuid> params=default-locking-mode
```

#### 4.4.10.8.11. 使用网络设置过滤 VIF 通信

以下过程指示虚拟机上的 VIF 在 XenServer 网络本身上使用网络 `default-locking-mode` 设置, 确定如何过滤通信。

使用网络设置过滤 VIF 通信:

- 运行以下命令, 将 VIF 锁定状态更改为 `network_default` (如果尚未使用该模式):

```
xe vif-param-set uuid=<vif-uuid> locking-mode=network_default
```

- 运行以下命令, 将默认锁定模式更改为 `unlocked` (如果尚未使用该模式):

```
xe network-param-set uuid=<network-uuid> default-locking-mode=unlocked
```

## 4.5. 网络故障排除

如果在配置网络时遇到问题, 请先确保您未曾直接修改过任何控制域 `ifcfg-*` 文件。这些文件直接由控制域主机代理管理, 任何更改都会被覆盖。

### 4.5.1. 诊断网络损坏

某些网卡型号在负载不足或启用了某些优化的情况下需要使用供应商提供的固件升级才能够稳定运行。如果看到与 VM 之间的通信出现损坏, 您应该首先尝试从供应商获得建议的最新固件并应用 BIOS 更新。

如果问题仍然存在, 则可以使用 CLI 禁用物理接口上的接收/发送卸载优化。



#### 警告

禁用接收/发送卸载优化可能会导致性能降低和/或 CPU 使用增加。

首先, 确定物理接口的 UUID。可以根据 `device` 字段进行过滤, 如下所示:

```
xe pif-list device=eth0
```



然后，在 PIF 上设置以下参数以禁用 TX 卸载：

```
xe pif-param-set uuid=<pif_uuid> other-config:ethtool-tx=off
```

最后，重新连接 PIF 或重新引导主机以使更改生效。

#### 4.5.2. 紧急网络重置

错误的网络连接设置会导致网络连接丢失，而且 XenServer 主机可能变得无法通过 XenCenter 或远程 SSH 访问。紧急网络重置可以提供简单的主机网络连接恢复和重置机制。

此功能可以使用 `xe-reset-networking` 命令，在 `xsconsole` 的网络和管理接口部分从命令行接口 (CLI) 获取。

导致网络连接丢失的错误设置包括重命名网络接口，创建绑定或 VLAN，或者更改管理接口时发生的错误（例如，输入错误的 IP 地址）。除此之外，如果池滚动升级、手动升级、修补程序安装或驱动程序安装导致网络连接丢失，或者如果资源池中的池主服务器或主机无法连接其他主机，您可能想运行该实用程序。

该实用程序仅应在紧急模式下使用，因为它将删除所有与主机相关联的 PIF、绑定、VLAN 和通道的配置。保存来宾系统网络和 VIF。作为该实用程序的一部分，VM 将被强行关闭，在可能的情况下，运行该命令之前，应当彻底地关闭 VM。应用重置之前，用户可以更改管理接口，并指定应当使用哪个 IP 配置（DHCP 或静态）。

如果池主服务器要求网络重置，必须在对任何其他池成员进行网络重置之前执行。然后，对池中的所有剩余主机进行网络重置，确保池的网络连接配置保持同类。对于 XenMotion，这是一个特别重要的因素。



##### 注意

如果池主服务器的 IP 地址（管理接口）因网络重置或 `xe host.management_reconfigure` 而发生改变，还必须对池中的其他主机应用网络重置命令，以便它们能够重新连接采用新 IP 地址的池主服务器。在此情况下，必须指定池主服务器的 IP 地址。

如果高可用性 (HA) 已启用，则不支持网络重置。要在此情况下重置网络配置，必须首先手动禁用 HA，然后运行网络重置命令。

##### 4.5.2.1. 验证网络重置

指定要在网络重置后使用的配置模式之后，`xsconsole` 和 CLI 将显示将在主机重新启动后应用的设置。这提供了在应用紧急网络重置命令之前执行任何修改的最后机会。重新启动后，可以在 XenCenter 和 `xsconsole` 中验证新网络配置。在 XenCenter 中，选中主机，单击网络连接选项卡，将显示新网络配置。在 `xsconsole` 中，此信息显示在网络和管理接口部分。



##### 注意

此外，也应当对其他池成员应用紧急网络重置，以便从池主服务器的新配置复制绑定、VLAN 或通道。

##### 4.5.2.2. 使用 CLI 执行网络重置

下表显示了可与 `xe-reset-networking` 命令一起使用的可选参数。



##### 警告

用户负责确保参数对 `xe-reset-networking` 命令的有效性，因此，需要仔细检查参数。如果指定无效参数，网络连接和配置将丢失。在此情况下，Citrix 建议客户重新运行命令 `xe-reset-networking`，而不使用任何参数。



必须首先在池主服务器上重置整个池的网络连接配置，然后在池中的所有其他主机上执行网络重置。

参数	必需/可选	说明
-m、--master	可选	池主服务器的管理接口的 IP 地址。默认为已知的最新池主服务器 IP 地址。
--device	可选	管理接口的设备名称。默认为在安装期间指定的设备名称。
--mode=static	可选	启用下列四个网络连接参数，用于管理接口的静态 IP 配置。如果不指定，将使用 DHCP 配置网络连接。
--ip	如果模式为静态，则为必需	主机管理接口的 IP 地址。如果模式为静态，则仅为有效。
--netmask	如果模式为静态，则为必需	管理接口的网络掩码。如果模式为静态，则仅为有效。
--gateway	可选	管理接口的网关。如果模式为静态，则仅为有效。
--dns	可选	管理接口的 DNS 服务器。如果模式为静态，则仅为有效。

#### 4.5.2.2.1. 池主服务器命令行示例

可在池主服务器上应用的命令的示例：  
重置 DHCP 配置的网络连接：

```
xe-reset-networking
```

重置静态 IP 配置的网络连接：

```
xe-reset-networking --mode= static --ip=<ip-address> \  
--netmask=<netmask> --gateway=<gateway> \  
--dns=<dns>
```

如果另一接口在初始设置后变为管理接口，则重置 DHCP 配置的网络连接：

```
xe-reset-networking --device=<device-name>
```

如果另一接口在初始设置后变为管理接口，则重置静态 IP 配置的网络连接：

```
xe-reset-networking --device=<device-name> --mode=static \  
--ip=<ip-address> --netmask=<netmask> \  
--gateway=<gateway> --dns=<dns>
```

#### 4.5.2.2.2. 池成员命令行示例

所有上述示例也适用于池成员。除此之外，可以指定池主服务器的 IP 地址（如果已更改，则需要指定。）  
重置 DHCP 配置的网络连接：

```
xe-reset-networking
```

如果池主服务器的 IP 地址已被修改，则重置 DHCP 的网络连接：

```
xe-reset-networking --master=<master-ip-address>
```

假设池主服务器的 IP 地址未更改，重置静态 IP 配置的网络连接：

```
xe-reset-networking --mode=static --ip=<ip-address> --netmask-<netmask> \  
--gateway=<gateway> --dns=<dns>
```



如果管理接口和池主服务器的 IP 地址已在初始设置后修改，则重置 DHCP 配置的网络连接： .

```
xe-reset-networking --device=<device-name> --master<master-ip-address>
```

## 第 5 章 存储

本章介绍将物理存储硬件映射到虚拟机 (VM) 的方式，以及 XenServer 主机 API 用来执行存储相关任务的软件对象。在每个支持的存储类型的详细信息部分中，包括使用 CLI 为 VM 创建存储的过程、特定类型的设备配置选项，生成用于备份的快照，以及一些在 XenServer 主机环境中管理存储的最佳做法。最后，将介绍虚拟磁盘 QoS (服务质量) 设置。

### 5.1. 存储概述

本部分介绍有哪些 XenServer 存储对象，以及它们之间的关系。

#### 5.1.1. 存储库 (SR)

存储库 (SR) 是一个特定存储目标，用于存储虚拟机 (VM) 虚拟磁盘映像 (VDI)。VDI 是一个表示虚拟硬盘 (HDD) 的存储抽象。

SR 具有对本地连接的 IDE、SATA、SCSI 和 SAS 驱动器和远程连接的 iSCSI、NFS、SAS 和光纤通道的内置支持，因而非常灵活。SR 和 VDI 抽象允许在支持高级存储功能 (如精简置备、VDI 快照和快速克隆) 的存储目标上提供这些功能。对于本身不直接支持高级操作的存储子系统，会根据实现这些功能的 Microsoft 虚拟硬盘 (VHD) 规范提供软件堆栈。

SR 命令提供用于创建、销毁、克隆、连接、查找它们包含的各个 VDI 及调整这些 VDI 大小的操作。

存储库是一个永久性的磁盘上数据结构。对于使用基本块设备的 SR 类型，创建新 SR 的过程都包括清除指定存储目标上的所有现有数据。其他存储类型 (如 NFS) 则会在与现有 SR 平行的存储阵列上创建一个新容器。

每台 XenServer 主机可以同时使用多个 SR 和不同的 SR 类型。可以在主机之间共享这些 SR，也可以将其专用于特定主机。共享存储会加入一个已定义的资源池，并由该池内的多台主机共用。共享 SR 必须可由每台主机通过网络访问。单个资源池中的所有主机必须至少有一个共用的共享 SR。

用于管理存储库的 CLI 操作在 [第 4.17 节 “SR 命令”](#) 中介绍。

#### 5.1.2. 虚拟磁盘映像 (VDI)

虚拟磁盘映像 (VDI) 是一个表示虚拟硬盘 (HDD) 的存储抽象。VDI 是 XenServer 中的虚拟化存储的基本单元。VDI 是磁盘上持久对象，独立于 XenServer 主机而存在。用于管理 VDI 的 CLI 操作在 [第 4.23 节 “VDI 命令”](#) 中说明。数据在磁盘上的实际表现形式不同于 SR 类型，并通过针对每个 SR 的专用存储插件界面 (称为 SM API) 来管理。

#### 5.1.3. 物理块设备 (PBD)

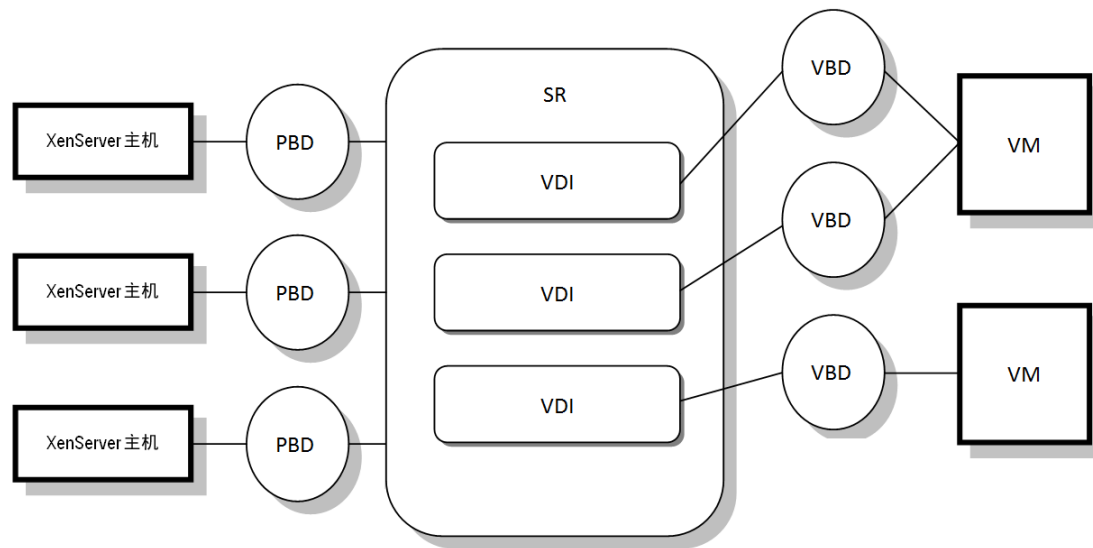
物理块设备表示物理服务器与连接的 SR 之间的接口。PBD 是连接器对象，允许将给定 SR 映射到 XenServer 主机。PBD 存储一些设备配置字段，用于与给定的存储目标进行连接和交互。例如，NFS 设备配置包括 NFS 服务器的 IP 地址和 XenServer 主机装载的关联路径。PBD 对象管理给定 SR 与给定 XenServer 主机之间的运行时连接。与 PBD 相关的 CLI 操作在 [第 4.13 节 “PBD 命令”](#) 中介绍。

#### 5.1.4. 虚拟块设备 (VBD)

虚拟块设备是连接器对象 (与上述的 PBD 类似)，可用于在 VDI 和 VM 之间进行映射。除了作为 VDI 和 VM 之间的连接 (也称为插入) 机制外，VBD 还可用于微调给定 VDI 的 QoS (服务质量)、统计数据 and 可引导性等相关参数。与 VBD 相关的 CLI 操作在 [第 4.22 节 “VBD 命令”](#) 中介绍。

#### 5.1.5. 存储对象摘要

下图概括说明了目前提供的存储对象之间的关系：



存储库和相关对象的图形概述

### 5.1.6. 虚拟磁盘数据格式

一般来说，针对 VDI 的物理存储的映射包括以下两种：

1. LUN 上基于逻辑卷的 VHD；默认情况下，基于块设备的 XenServer 存储将插入磁盘上的逻辑卷管理器，可以是逻辑上连接的设备（LVM 类型的 SR），也可以是通过光纤通道（LVMoHBA 类型的 SR）、iSCSI（LVMoISCSI 类型的 SR）或 SAS（LVMoHBA 类型的 SR）连接的 SAN LUN。VDI 表示为卷管理器中的卷并以 VHD 格式保存，以支持快照和克隆上引用节点的精简置备。
2. 文件系统上基于文件的 VHD；VM 映像作为精简置备的 VHD 格式文件在本地非共享文件系统（EXT 类型 SR）或共享 NFS 目标（NFS 类型 SR）上存储。

#### 5.1.6.1. VDI 类型

通常状况下将创建 VHD 格式的 VDI。在创建 VDI 时可以选择使用原始格式；这只能通过使用 xe CLI 来实现。

要检查 VDI 是否以 type=raw 格式创建，请检查其 sm-config 映射。sr-param-list 和 vdi-param-list xe 命令可分别用于实现此目的。

#### 5.1.6.2. 使用 xe CLI 创建原始虚拟磁盘

1. 根据您想要放置虚拟磁盘的 SR 的 UUID，运行以下命令创建一个 VDI：

```
xe vdi-create sr-uuid=<sr-uuid> type=user virtual-size=<virtual-size> \
  name-label=<VDI name> sm-config:type=raw
```

2. 将新建虚拟磁盘连接到 VM，使用 VM 中的常用磁盘工具分区和格式化，或者使用新磁盘。您可以使用 vbd-create 命令创建新 VBD，以将该虚拟磁盘映射到 VM 中。

#### 5.1.6.3. 在 VDI 格式间转换

原始和 VHD 格式之间不能实现直接转换。相反，您可以创建新 VDI（原始 VDI（如上所述）或 VHD），然后将数据从现有卷复制到该 VDI 中。Citrix 建议您使用 xe CLI 来确保新 VDI 的虚拟大小至少与从中复制数

据的 VDI 相同（可以查看其虚拟大小字段，例如使用 `vdi-param-list` 命令）。然后您可以将该新建 VDI 连接到 VM，并使用 VM 中您的首选工具（Windows 中标准的磁盘管理工具或 Linux 的 `dd` 命令）来直接执行数据块复制。如果新建卷为 VHD 卷，使用能够避免向磁盘中写入空扇区的工具十分重要，这样可以让基础存储库中的空间得到最佳使用。这种情况下，基于文件的复制方法更为合适。

#### 5.1.6.4. 基于 VHD 的 VDI

可以链接 VHD 文件，从而允许两个 VDI 共享通用数据。如果克隆基于 VHD 的 VM，生成的 VM 将共享克隆时的通用磁盘数据。每个 VM 将在 VDI 的单独写入时复制 (CoW) 版本中继续进行各自的更改。此功能允许从模板快速克隆基于 VHD 的 VM，便于极快地置备和部署新 VM。

这会导致出现一种情况，即在克隆 VM 及其关联的 VDI 时会不断创建链接 VDI 的树。删除链中的一个 VDI 时，XenServer 会合理化链中的其他 VDI 以删除不必要的 VDI。此合并过程异步运行。回收的磁盘空间量和执行此过程所花费的时间取决于 VDI 的大小和共享数据量。仅一个合并过程将始终对 SR 保持活动状态。此进程线程在 SR 主服务器主机上运行。

如果有重要的 VM 在池的主服务器上运行，且 IO 由于此进程偶尔会出现缓慢情况，您可以采取措施来缓解这一现象：

- 将 VM 迁移到 SR 主节点以外的主机。
- 将磁盘 IO 优先级设置为较高的级别，并调整调度程序。有关详细信息，请参阅第 5.7.10 节“虚拟磁盘 QoS 设置”。

由 XenServer 中基于 LVM 和基于文件的 SR 类型使用的 VHD 格式使用精简置备。当 VM 将数据写入到磁盘时，映像文件将自动在大小为 2MB 的块中扩展。对于基于文件的 VHD，VM 映像文件仅在物理存储中占用所需的块，这将带来极大的好处。通过基于 LVM 的 VHD，基础逻辑卷容器必须与 VDI 的虚拟磁盘大小相同，但是当出现快照或克隆时，基础 CoW 实例磁盘上的闲置空间将被收回。两种行为之间的差异描述如下：

- 对于基于 LVM 的 VHD，链中的差异磁盘节点 (difference disk node) 所消耗的空间与写入磁盘的数据量相同，但叶节点 (VDI 克隆) 则完全扩充至虚拟磁盘的大小。快照叶节点 (VDI 快照) 在未使用时保持压缩状态，并且可以附加只读存储以保持压缩分配。附加读写存储的快照节点将在附加后完全扩大，在分离后压缩。
- 对于基于文件的 VHD，所有节点所消耗的空间与写入的数据量相同，叶节点文件所占空间则随数据的实时写入而增长。如果为新 VM 分配 100GB 的 VDI 并且安装操作系统，VDI 文件的物理大小仅是已写入到磁盘的操作系统数据加上一些小的元数据开销的大小。

基于单个 VHD 模板克隆 VM 时，每个子 VM 会形成一个链，其中新更改将写入新 VM，并且直接从父模板读取旧块。如果将新 VM 转化为深层模板并且克隆更多 VM，生成的链将导致性能降级。XenServer 支持的最大链长度为 30，但如果没有充分的理由，一般不建议接近此限制。如果存有疑问，使用 XenCenter 或使用 `vm-copy` 命令“复制”VM，这会将链长度重置回 0。

## 5.2. 存储库格式

新存储库可以在 XenCenter 中使用新建存储库向导创建。此向导将指导您完成各个配置步骤。或者，使用 CLI 和 `sr-create` 命令。此命令在存储基底中创建新的 SR（可能销毁任何现有数据），并创建 SR API 对象和相应的 PBD 记录，使 VM 能够使用存储。成功创建 SR 后，自动插入 PBD。如果已设置 `SR shared=true` 标志，则创建 PBD 记录并为资源池中的每台 XenServer 插入该记录。

如果您正在为基于 IP 的存储 (iSCSI 或 NFS) 创建 SR，您可以将下列 NIC 之一配置为存储网络：处理管理流量的 NIC 或面向存储流量的新 NIC。要为 NIC 分配 IP 地址，请参阅第 4.4.7 节“配置专用存储 NIC”。

所有 XenServer SR 类型都支持 VDI 重新调整大小、快速克隆和快照。基于 LVM SR 类型（本地、iSCSI 或 HBA）的 SR 提供用于快照和隐藏父节点的精简置备。其他 SR 类型支持完整的精简置备，包括活动虚拟磁盘的精简置备。



**警告**

当 VHD VDI 未连接到 VM 时，例如是 VDI 快照，它们会在默认情况下以精简置备方式存储。尝试重新连接 VDI 时，客户应当确保有足够的磁盘空间，用于对 VDI 进行精简置备。VDI 克隆采用厚置备。

支持的最大 VDI 大小为：

存储库格式	最大 VDI 大小
EXT3	2 TB
LVM	2 TB
NFS	2 TB
iSCSI	2 TB
HBA	2 TB

### 5.2.1. 本地 LVM

本地 LVM 类型表示磁盘处于一个本地连接的卷组中。

默认情况下，XenServer 使用安装它的物理主机的本地磁盘。Linux 逻辑卷管理器 (LVM) 用于管理 VM 存储。在指定大小的 LVM 逻辑卷中，VDI 采用 VHD 格式实施。

#### 5.2.1.1. LVM 性能注意事项

XenServer 5.5 和更新版本提供的快照和快速克隆功能（用于基于 LVM 的 SR）具有固有的性能系统开销。如果想要获得最佳性能，XenServer 支持创建原始格式以及默认 VHD 格式的 VDI。原始 VDI 不支持 XenServer 快照功能。

**注意**

使用默认的 Windows VSS 提供程序的不可传送快照适用于所有类型 VDI。

**警告**

请勿尝试为连接了 type=raw 磁盘的 VM 生成快照。这可能会导致创建部分快照。这种情况下，您可以通过检查 snapshot-of 字段确定孤立快照 VDI，然后删除这些快照。

#### 5.2.1.2. 创建本地 LVM SR (lvm)

默认情况下，在主机安装期间创建 LVM SR。

lvm SR 的 Device-config 参数为：

参数名称	说明	是否必需？
设备	用于 SR 的本地主机上的设备名称	是

要在 /dev/sdb 上创建本地 lvm SR，请使用以下命令。

```
xe sr-create host-uuid=<valid_uuid> content-type=user \
name-label=<"Example Local LVM SR"> shared=false \
device-config:device=/dev/sdb type=lvm
```



### 5.2.2. 本地 EXT3

使用 EXT3 在本地存储上启用精简置备。但是，默认存储库类型为 LVM，因为它可以提供一致的写性能，防止存储过度使用。使用 EXT3 的客户在执行 VM 生命周期操作（例如，VM 创建、挂起/恢复）和初次从 VM 中创建大文件时，可能会看到性能下降。

必须使用 XenServer CLI 配置本地磁盘 EXT SR。

#### 5.2.2.1. 创建本地 EXT3 SR (ext)

ext SR 的 Device-config 参数：

参数名称	说明	是否必需？
Device	用于 SR 的本地主机上的设备名称	是

要在 /dev/sdb 上创建本地 ext SR，请使用以下命令：

```
xe sr-create host-uuid=<valid_uuid> content-type=user \
  name-label=<"Example Local EXT3 SR"> shared=false \
  device-config:device=/dev/sdb type=ext
```

### 5.2.3. udev

udev 类型表示插入的设备使用 udev 设备管理器作为 VDI。

XenServer 具有两个代表可移动存储的 udev 类型的 SR。一个用于 XenServer 主机的物理 CD 或 DVD-ROM 驱动器中的 CD 或 DVD 磁盘。另一个用于插入到 XenServer 主机的 USB 端口的 USB 设备。插入磁盘或 U 盘，代表介质的 VDI 出现；移除磁盘或 U 盘，VDI 消失。

### 5.2.4. ISO

ISO 类型处理以 ISO 格式文件存储的 CD 映像。此 SR 类型在创建共享 ISO 库时十分有用。对于用于存储 ISO 库的存储库，必须将 content-type 参数设置为 iso。

例如：

```
xe sr-create host-uuid=<valid_uuid> content-type=iso \
  type=iso name-label=<"Example ISO SR"> \
  device-config:location=<nfs server:path>
```

### 5.2.5. 软件 iSCSI 支持

XenServer 提供了对 iSCSI LUN 上的共享 SR 的支持。由于使用 Open-iSCSI 软件 iSCSI 发起程序或使用受支持的 iSCSI 主机总线适配器 (HBA)，iSCSI 得到支持。使用 iSCSI HBA 的步骤与使用光纤通道 HBA 的步骤相同，这两种步骤在 [第 5.2.8.2 节“创建基于光纤通道/以太网光纤通道/iSCSI HBA 或 SAS 的共享 LVM SR \(lvmohba\)”](#) 中均有说明。

使用软件 iSCSI 发起程序的共享 iSCSI 支持基于 Linux 卷管理器 (LVM) 实现，并且提供的性能优势与本地磁盘库中的 LVM VDI 提供的相同。使用基于软件的主机启动程序的共享 iSCSI SR 能够灵活地支持使用 XenMotion 的 VM：VM 可以在资源池中的任意 XenServer 主机上启动，并且可以在不会导致明显停机时间的情况下在主机间迁移。

iSCSI SR 利用在创建期间指定的整个 LUN，但不能跨多个 LUN。此外，还为数据路径初始化阶段和 LUN 检测阶段进行的客户端身份验证提供 CHAP 支持。



#### 注意

iSCSI LUN 的块大小必须为 512 字节。

### 5.2.5.1. XenServer 主机 iSCSI 配置

所有 iSCSI 启动程序和目标都必须具有唯一的名称，以确保可以在网络上对它们进行唯一标识。启动程序具有 iSCSI 启动程序地址，目标具有 iSCSI 目标地址。这些名称统称为 iSCSI 限定名称 (IQN)。

XenServer 主机支持在主机安装期间使用随机 IQN 自动创建并配置的单个 iSCSI 启动程序。可以使用该单个启动程序并发连接到多个 iSCSI 目标。

iSCSI 目标通常使用 iSCSI 启动程序 IQN 列表提供访问控制，因此，必须将可以由 XenServer 主机访问的所有 iSCSI 目标/LUN 配置为允许该主机的启动程序 IQN 访问。同样，必须将用作共享 iSCSI SR 的目标/LUN 配置为允许资源池中的所有主机 IQN 访问。



#### 注意

不提供访问控制的 iSCSI 目标通常默认为仅允许单个启动程序访问 LUN 以确保数据完整性。若要将 iSCSI LUN 用作资源池中多个 XenServer 主机的共享 SR，请确保为指定的 LUN 启用多启动程序访问。

在使用 iSCSI 软件启动程序时，可以通过 XenServer 或者通过 CLI 使用以下命令来调整 XenCenter 主机 IQN 值：

```
xe host-param-set uuid=<valid_host_id> other-config:iscsi_iqn=<new_initiator_iqn>
```



#### 警告

每个 iSCSI 目标和启动程序必须具有唯一的 IQN。如果使用非唯一的 IQN 标识符，则可能导致数据损坏和/或 LUN 访问被拒绝。



#### 警告

不要更改连接了 iSCSI SR 的 XenServer 主机 IQN，否则可能导致无法连接到新目标或现有 SR。

### 5.2.6. 软件 FCoE 存储

软件 FCoE 提供了一个标准框架，硬件提供商可以在此框架中插入具备 FCoE 功能的 NIC，获得与基于硬件的 FCoE 相同的优势。这可避免使用昂贵的 HBA。

创建新的软件 FCoE 存储之前，必须手动完成向主机公开 LUN 所需的配置，包括配置 FCoE 架构以及向 SAN 公共全球名称 (PWWN) 分配 LUN。完成此配置之后，应将可用的 LUN 作为 SCSI 设备装载到主机的 CNA。然后，可以使用该 SCSI 设备访问 LUN，就像它是本地连接的 SCSI 设备一样。有关配置支持 FCoE 的物理交换机和阵列的信息，请参阅供应商提供的文档。



#### 注意

可以将软件 FCoE 与 Open vSwitch 和 Linux 桥接一起用作网络后端。

#### 5.2.6.1. 创建软件 FCoE SR

创建软件 FCoE SR 之前，客户必须确保将具有 FCoE 功能的 NIC 连接到主机。

FCoE SR 的 Device-config 参数如下：

参数名称	说明	是否必需？
SCSId	目标 LUN 的 SCSI 总线 ID	是

运行以下命令可创建共享 FCoE SR：

```
xe sr-create type=lvmofocoe \
name-label=<"FCoE SR"> shared=true device-config:SCSId=<SCSI_id>
```

## 5.2.7. 硬件主机总线适配器 (HBA)

本部分介绍管理 SAS、光纤通道和 iSCSI HBA 所需的各种操作。

### 5.2.7.1. QLogic iSCSI HBA 设置示例

有关配置 QLogic 光纤通道和 iSCSI HBA 的完整详细信息，请访问 [QLogic Web 站点](#)。

将 HBA 物理安装到 XenServer 主机后，请执行下列步骤配置 HBA：

1. 设置 HBA 的 IP 网络配置。本示例假设 DHCP 和 HBA 端口为 0。如果使用静态 IP 寻址或多端口 HBA，请指定适当的值。

```
/opt/QLogic_Corporation/SANsurferiCLI/iscli -ipdhcp 0
```

2. 将永久性 iSCSI 目标添加到 HBA 的端口 0。

```
/opt/QLogic_Corporation/SANsurferiCLI/iscli -pa 0 <iscsi_target_ip_address>
```

3. 使用 `xe sr-probe` 命令强制重新扫描 HBA 控制器并显示可用的 LUN。有关更多详细信息，请参阅第 5.3.2 节“探测 SR”和第 5.2.8.2 节“创建基于光纤通道/以太网光纤通道/ iSCSI HBA 或 SAS 的共享 LVM SR (lvmoahba)”。

### 5.2.7.2. 删除基于 HBA 的 SAS、FC 或 iSCSI 设备条目



#### 注意

此步骤不是必需步骤。Citrix 建议仅在必要时由高级用户执行此步骤。

每个基于 HBA 的 LUN 都在 `/dev/disk/by-scsibus` 下具有相应的全局设备路径条目（格式为 `SCSIid-adapter:bus:target:lun`），并在 `/dev` 下具有标准设备路径。要删除不再用作 SR 的 LUN 的设备条目，请执行下列步骤：

1. 根据需要使用 `sr-forget` 或 `sr-destroy` 从 XenServer 主机数据库中删除 SR。有关详细信息，请参阅第 5.7.1 节“卸载 SRs”。
2. 删除 SAN 中的将所需 LUN 的区域划分给所需主机的区域划分配置。
3. 使用 `sr-probe` 命令确定要与要删除的 LUN 对应的 ADAPTER、BUS、TARGET 和 LUN 值。有关详细信息，请参阅第 5.3.2 节“探测 SR”。
4. 使用以下命令删除设备条目：

```
echo "1" > /sys/class/scsi_device/<adapter>:<bus>:<target>:<lun>/device/delete
```



#### 警告

务必保证您确定要删除哪个 LUN。意外删除主机操作所需的 LUN（如引导设备或根设备）将使该主机不可用。

## 5.2.8. 共享 LVM 存储

共享 LVM 类型将磁盘表示为在 iSCSI（FC 或 SAS）LUN 上创建的卷组中的逻辑卷。



#### 注意

iSCSI LUN 的块大小必须为 512 字节。

### 5.2.8.1. 使用软件 iSCSI 发起程序创建基于 iSCSI 的共享 LVM SR (lvmoiscsi)

lvmoiscsi SR 的 Device-config 参数：

参数名称	说明	是否必需？
target	SR 所在的 iSCSI 文件管理器的 IP 地址或主机名	是
targetIQN	SR 所在的 iSCSI 文件管理器的 IQN 目标地址	是
SCSIid	目标 LUN 的 SCSI 总线 ID	是
chapuser	用于 CHAP 身份验证的用户名	否
chappassword	用于 CHAP 身份验证的密码	否
port	在其上查询目标的网络端口号	否
usediscoverynumber	要使用的特定 iscsi 记录索引	否
incoming_chapuser	iSCSI 过滤器用来针对主机进行身份验证的用户名	否
incoming_chappassword	iSCSI 过滤器用来针对主机进行身份验证的密码	否

要在 iSCSI 目标的特定 LUN 上创建共享 lvmoiscsi SR，请使用以下命令。

```
xe sr-create host-uuid=<valid_uuid> content-type=user \
name-label=<"Example shared LVM over iSCSI SR"> shared=true \
device-config:target=<target_ip> device-config:targetIQN=<target_iqn> \
device-config:SCSIid=<scsi_id> \
type=lvmoiscsi
```

#### 5.2.8.2. 创建基于光纤通道/以太网光纤通道/ iSCSI HBA 或 SAS 的共享 LVM SR (lvmohba)

lvmohba 类型的 SR 只能使用 xe CLI 或 XenCenter 创建和管理。

lvmohba SR 的 Device-config 参数：

参数名称	说明	是否必需？
SCSIid	设备 SCSI ID	是

要创建共享 lvmohba SR，请在池中的每个主机上执行下列步骤：

1. 将一个或多个 LUN 中的区域划分给池中的每个 XenServer 主机。此过程特定于所使用的 SAN 设备。请参阅 SAN 文档了解详情。
2. 如有必要，请使用 XenServer 主机中包含的 HBA CLI 来配置 HBA：
  - Emulex:/bin/sbin/ocmanager
  - QLogic FC:/opt/QLogic\_Corporation/SANsurferCLI
  - QLogic iSCSI:/opt/QLogic\_Corporation/SANsurferCLI

有关 QLogic iSCSI HBA 配置的示例，请参阅第 5.2.7 节“[硬件主机总线适配器 \(HBA\)](#)”。有关光纤通道和 iSCSI HBA 的详细信息，请访问 [Emulex](#) 和 [QLogic](#) Web 站点。

3. 使用 sr-probe 命令确定 HBA LUN 的全局设备路径。sr-probe 强制重新扫描系统中安装的 HBA，以检测其区域已划分给主机的任何新 LUN，然后返回找到的每个 LUN 的属性列表。指定 host-uuid 参数以确保在所需主机上执行探测。



作为 <path> 属性返回的全局设备路径适用于池中的所有主机。因此，当创建 SR 时，必须将该路径用作 device-config:device 参数的值。

如果提供了多个 LUN，则使用供应商、LUN 大小、LUN 序列号或 <path> 属性中包含的 SCSI ID 确定所需的 LUN。

```
xe sr-probe type=lvmotha \
host-uuid=1212c7b3-f333-4a8d-a6fb-80c5b79b5b31
Error code: SR_BACKEND_FAILURE_90
Error parameters: , The request is missing the device parameter, \
<?xml version="1.0" ?>
<Devlist>
  <BlockDevice>
    <path>
      /dev/disk/by-id/scsi-360a9800068666949673446387665336f
    </path>
    <vendor>
      HITACHI
    </vendor>
    <serial>
      730157980002
    </serial>
    <size>
      80530636800
    </size>
    <adapter>
      4
    </adapter>
    <channel>
      0
    </channel>
    <id>
      4
    </id>
    <lun>
      2
    </lun>
    <hba>
      qla2xxx
    </hba>
  </BlockDevice>
  <Adapter>
    <host>
      Host4
    </host>
    <name>
      qla2xxx
    </name>
    <manufacturer>
      QLogic HBA Driver
    </manufacturer>
    <id>
      4
    </id>
  </Adapter>
</Devlist>
```

4. 在池的主服务器主机上创建 SR，同时指定在 sr-probe 的 <path> 属性中返回的全局设备路径。将创建 PBD 并将其自动插入池中的每个主机。

```
xe sr-create host-uuid=<valid_uuid> \
content-type=user \
name-label=<"Example shared LVM over HBA SR"> shared=true \
device-config:SCSIid=<device_scsi_id> type=lvmoaha
```



### 注意

可以使用 XenCenter 中的修复存储库功能重试 sr-create 操作的 PBD 创建和插入部分。创建 SR 后，如果 LUN 区域划分对于池中的一个或多个主机不正确，则此功能非常有用。更正受影响的主机的区域划分并使用修复存储库功能，而不是删除并重新创建 SR。

## 5.2.9. NFS 和 SMB

NFS 和 SMB 类型将磁盘作为 VHD 文件存储在远程文件系统中。

XenServer 允许通过 TCP/IP 将 NFS 服务器上的共享（支持 NFSv4 或 NFSv3）或 SMB 服务器上的共享（支持 SMB 3.0）立即用作虚拟磁盘 (VDI) 的存储库。VDI 仅以 Microsoft VHD 格式存储。此外，由于可以共享这些 SR，因此，共享 SR 上存储的 VDI 允许执行以下操作：

- 在资源池中的任何 XenServer 主机上启动 VM
- 使用 XenMotion 在资源池中的 XenServer 主机之间迁移 VM（无明显停机）



### 重要

- 对 SMB 3.0 的支持功能限制为能够使用 3.0 协议连接到共享。透明故障转移等其他功能取决于上游 Linux 内核中的功能的可用性，但在 XenServer 7.0 中不受支持。
- 对于 NFSv4，只有身份验证类型 AUTH\_SYS 在 XenServer 7.0 中受支持。



### 注意

SMB 存储适用于 XenServer Enterprise Edition 客户或通过 XenApp/XenDesktop 授权访问 XenServer 的客户。要了解有关 XenServer 版本以及升级方式的详细信息，请单击[此处](#)访问 Citrix Web 站点。有关许可的详细信息，请参阅 [XenServer 7.0 许可常见问题解答](#)。

基于文件的 SR 上存储的 VDI 是精简置备的。当 VM 向磁盘写入数据时分配映像文件。VM 映像文件仅在存储中占用所需的空間，这将带来极大的好处。例如，如果为新 VM 分配 100 GB 的 VDI 并且安装操作系统，VDI 文件将只反映已写入到磁盘的操作系统数据的大小，而不会反映整个 100 GB。

还可以链接 VHD 文件，从而允许两个 VDI 共享通用数据。如果克隆基于文件的 VM，生成的 VM 将共享克隆时的通用磁盘数据。每个 VM 将在 VDI 的单独写入时复制版本中继续进行各自的更改。此功能允许从模板快速克隆基于文件的 VM，便于极快地置备和部署新 VM。



### 注意

支持的最长 VHD 链长度为 30。

XenServer 的基于文件的 SR 和 VHD 实现假定它们可以完全控制文件服务器上的 SR 目录。管理员不应该修改 SR 目录的内容，因为这可能损坏 VDI 的内容。

XenServer 已针对使用稳定 RAM 的企业级存储进行了优化，以快速确认写入请求，同时保留针对故障的高级数据保护。XenServer 已针对使用 Data OnTap 7.3 和 8.1 的 Network Appliance FAS2020 和 FAS3210 存储进行了广泛测试。



### 警告

由于基于文件的 SR 上的 VDI 创建为精简配置文件，因此，管理员必须确保在基于文件的 SR 上为所有必需的 VDI 保留足够的磁盘空间。XenServer 主机并不强制在基于文件的 SR 上实际存在 VDI 所需的空間。

### 5.2.9.1. 创建共享 NFS SR (NFS)

要创建 NFS SR，需要提供 NFS 服务器的主机名或 IP 地址。可以在任何有效的目标路径下创建 SR；使用 `sr-probe` 命令可显示服务器导出的有效目标路径列表。

XenServer 与低端存储配合使用时，在将确认传递到来宾系统 VM 之前，它会谨慎地等待所有写入得到确认。这可能会导致性能显著下降，但通过设置存储以将 SR 装载点呈现为异步模式导出，可以解决这个问题。异步导出确认实际不在磁盘上的写入，因此，管理员应该认真考虑这些情况中的故障风险。



#### 注意

NFS 服务器必须配置为将指定的路径导出到池中的所有 XenServer 主机，否则创建 SR 和插入 PBD 记录将失败。

默认情况下，XenServer NFS 实现使用 TCP。如果条件允许，您可以将该实现配置为在可能对性能有益的情况下使用 UDP。为此，创建 SR 时，请指定 `device-config` 参数 `useUDP=true`。

NFS SR 的 Device-config 参数：

参数名称	说明	是否必需？
server	NFS 服务器的 IP 地址或主机名称	是
serverpath	SR 在 NFS 服务器上的路径，包括 NFS 装载点	是

例如，要在 192.168.1.10:/export1 上创建共享 NFS SR，请使用以下命令：

```
xe sr-create content-type=user \
name-label=<"shared NFS SR"> shared=true \
device-config:server=192.168.1.10 device-config:serverpath=/export1 type=nfs \
nfsversion=<"3", "4">
```

要创建非共享 NFS SR，请使用以下命令：

```
xe sr-create host-uuid=<host_uuid> content-type=user \
name-label=<"Non-shared NFS SR"> \
device-config:server=192.168.1.10 device-config:serverpath=/export1 type=nfs \
nfsversion=<"3", "4">
```

### 5.2.9.2. 创建共享 SMB SR (SMB)

要创建 SMB SR，需要提供 SMB 服务器的主机名或 IP 地址、导出的共享的完整路径以及相应的凭据。



#### 注意

SMB SR 已针对运行 OnTap 8.3 和 Windows Server 2012 R2 的网络设备存储进行测试。

SMB SR 的 Device-config 参数：

参数名称	说明	是否必需？
server	服务器上的共享的完整路径	是
username	对共享具有读写访问权限的用户帐户	可选
password	用户帐户的密码	可选



例如，要在 192.168.1.10:/share1 上创建共享 SMB SR，请使用以下命令：

```
xe sr-create content-type=user \
name-label=<"Example shared SMB SR"> shared=true \
device-config:server=//192.168.1.10/share1 \
device-config:username=<valid_username> device-config:password=<valid_password> type=smb
```

要创建非共享 SMB SR，请使用以下命令：

```
xe sr-create host-uuid=<host_uuid> content-type=user \
name-label=<"Non-shared SMB SR"> \
device-config:server=//192.168.1.10/share1 \
device-config:username=<valid_username> device-config:password=<valid_password> type=smb
```

### 5.2.10. 硬件 HBA 上的 LVM

硬件 HBA 上 LVM 类型表示磁盘作为卷组中逻辑卷上的 VHD 在提供的 HBA LUN 上创建，例如，基于硬件的 iSCSI 或 FC 支持。

XenServer 主机通过 Emulex 或 QLogic 主机总线适配器 (HBA) 支持光纤通道 (FC) 存储区域网络 (SAN)。必须手动完成向主机公开 FC LUN 所需的所有 FC 配置，其中包括存储设备、网络设备和 XenServer 主机中的 HBA。完成所有 FC 配置后，HBA 将向主机公开一个由 FC LUN 支持的 SCSI 设备。然后，可以使用该 SCSI 设备访问 FC LUN，就像它是本地连接的 SCSI 设备一样。

使用 sr-probe 命令可以列出主机上存在的由 LUN 支持的 SCSI 设备。此命令强制扫描新的由 LUN 支持的 SCSI 设备。sr-probe 返回的由 LUN 支持的 SCSI 设备的路径值在能够访问该 LUN 的所有主机上是一致的，因此当创建资源池中的所有主机都可访问的共享 SR 时，必须使用该路径值。

同样的功能也适用于 QLogic iSCSI HBA。

有关创建基于 HBA 的共享 FC 和 iSCSI SR 的详细信息，请参阅第 5.3.1 节“创建存储库”。



#### 注意

XenServer 支持光纤通道，但不支持将 LUN 直接映射到 VM。必须将基于 HBA 的 LUN 映射到主机并指定将其用于 SR。SR 中的 VDI 作为标准块设备向 VM 公开。

## 5.3. 存储配置

本部分介绍了创建存储库类型并使其可用于 XenServer 主机。提供的示例显示如何使用 CLI 完成此操作。有关使用新建存储库向导的详细信息，请参阅 XenCenter 帮助。

### 5.3.1. 创建存储库

本部分介绍了如何创建不同类型的存储库 (SR) 并使其可用于 XenServer 主机。提供的示例涉及使用 xe CLI 创建 SR。有关使用 XenCenter 通过新建存储库向导添加 SR 的详细信息，请参阅 XenCenter 帮助。



#### 注意

lvm 和 ext3 类型的本地 SR 只能使用 xe CLI 创建。创建后，所有 SR 类型可由 XenCenter 或 xe CLI 进行管理。

使用 CLI 创建在 XenServer 主机上使用的新存储库包括两个基本步骤：

1. 探测 SR 类型以确定任何必需参数的值。
2. 创建 SR 以初始化 SR 对象及关联的 PBD 对象、插入 PBD 以及激活 SR。

根据创建的 SR 类型，这些步骤稍有差异。在所有示例中，如果成功，sr-create 命令将返回创建的 SR 的 UUID。



如果不再使用 SR，可以将它销毁以释放物理设备的空间，或忽略该 SR 以将其与某个 XenServer 主机分离而连接到另一个主机上。有关详细信息，请参阅第 5.7.1 节“卸载 SRs”。

### 5.3.2. 探测 SR

sr-probe 命令有两种使用方式：

1. 确定未知参数以供创建 SR。
2. 返回现有 SR 的列表。

在这两种情况下，sr-probe 通过指定 SR 类型和该 SR 类型的一个或多个 device-config 参数起作用。如果提供的参数集不完整，则 sr-probe 命令将返回错误消息，指示参数缺失以及缺失参数的可能选项。如果提供的参数集完整，将返回现有 SR 的列表。所有 sr-probe 输出将作为 XML 返回。

例如，通过指定已知 iSCSI 目标的名称或 IP 地址，可以探测该目标，并将返回该目标上可用的 IQN 集：

```
xe sr-probe type=lvmoiscsi device-config:target=<192.168.1.10>
```

```
Error code: SR_BACKEND_FAILURE_96
Error parameters: , The request is missing or has an incorrect target IQN parameter, \
<?xml version="1.0" ?>
<iscsi-target-iqns>
  <TGT>
    <Index>
      0
    </Index>
    <IPAddress>
      192.168.1.10
    </IPAddress>
    <TargetIQN>
      iqn.192.168.1.10:filer1
    </TargetIQN>
  </TGT>
</iscsi-target-iqns>
```

再次探测上述目标并指定名称/IP 地址和所需的 IQN 将返回该目标/IQN 上的可用 SCSIId (LUN) 集。

```
xe sr-probe type=lvmoiscsi device-config:target=192.168.1.10 \
device-config:targetIQN=iqn.192.168.1.10:filer1
```

```
Error code: SR_BACKEND_FAILURE_107
Error parameters: , The SCSIId parameter is missing or incorrect, \
<?xml version="1.0" ?>
<iscsi-target>
  <LUN>
    <vendor>
      IET
    </vendor>
    <LUNid>
      0
    </LUNid>
    <size>
      42949672960
    </size>
    <SCSIId>
      149455400000000000000000002000000b70200000f000000
    </SCSIId>
  </LUN>
</iscsi-target>
```

探测上述目标并提供所有三个参数将返回在 LUN 上存在 ( 如果有 ) 的 SR 的列表。

```
xe sr-probe type=lvmoiscsi device-config:target=192.168.1.10 \
device-config:targetIQN=192.168.1.10:file1 \
device-config:SCSIid=149455400000000000000000002000000b70200000f000000

<?xml version="1.0" ?>
<SRlist>
  <SR>
    <UUID>
      3f6e1ebd-8687-0315-f9d3-b02ab3adc4a6
    </UUID>
    <Devlist>
      /dev/disk/by-id/scsi-149455400000000000000000002000000b70200000f000000
    </Devlist>
  </SR>
</SRlist>
```

可以针对每个 SR 类型探测以下参数：

SR 类型	device-config 参数 ( 按依赖关系排序 )	是否可以探测 ?	sr-create 是否需要该参数 ?
lvmoiscsi	target	否	是
	chapuser	否	否
	chappassword	否	否
	targetIQN	是	是
	SCSIid	是	是
lvmohba	SCSIid	是	是
NetApp	target	否	是
	username	否	是
	password	否	是
	chapuser	否	否
	chappassword	否	否
	aggregate	否*	是
	FlexVol	否	否
	allocation	否	否
nfs	asis	否	否
	server	否	是
	serverpath	是	是
lvm	device	否	是

SR 类型	device-config 参数 (按依赖关系排序)	是否可以探测?	sr-create 是否需要该参数?
ext	device	否	是
EqualLogic	target	否	是
	username	否	是
	password	否	是
	chapuser	否	否
	chappassword	否	否
	storagepool	否 <sup>†</sup>	是

\* 只有在运行 sr-create 时才能执行聚合探测。需要在创建时执行该探测，以便在创建 SR 时指定聚合。

<sup>†</sup> 只有在运行 sr-create 时才能执行存储池探测。需要在创建时执行该探测，以便在创建 SR 时指定聚合。

## 5.4. 存储多路径

光纤通道和 iSCSI 存储后端存在动态多路径支持。默认情况下，XenServer 使用循环模式负载平衡，因此，在正常操作期间，两个路由通道上都具有活动通信。您可以在 XenCenter 中或 xe CLI 上启用多路径。有关多路径的其他信息，请参阅 CTX134881 —[Configuring Multipathing for XenServer](#) (为 XenServer 配置多路径)。



### 小心

在尝试启用多路径之前，请确认您的存储服务器上提供了多个目标。例如，查询给定门户上的 sendtargets 的 iSCSI 存储后端应返回多个目标，如下例所示：

```
iscsiadm -m discovery --type sendtargets --portal 192.168.0.161
192.168.0.161:3260,1 iqn.strawberry:litchie
192.168.0.204:3260,2 iqn.strawberry:litchie
```

要启用存储多路径，请使用 xe CLI:

1. 拔出主机上的所有 PBD：

```
xe pbd-unplug uuid=<pbid_uuid>
```

2. 设置主机的 other-config:multipathing 参数：

```
xe host-param-set other-config:multipathing=true uuid=host_uuid
```

3. 将主机的 other-config:multipathhandle 设置为 dmp：

```
xe host-param-set other-config:multipathhandle=dmp uuid=host_uuid
```

4. 如果主机上存在以单一路径模式运行但具有多路径的现有 SR：

- 迁移或挂起在受影响的 SR 中存在虚拟磁盘的任何正在运行的来宾系统
- 拔出并重新插入所有受影响的 SR 的 PBD 以使用多路径重新对其进行连接：

```
xe pbd-plug uuid=<pbid_uuid>
```

要禁用多路径，请先拔出 VBD，接着将主机 `other-config:multipathing` 参数设置为 `false`，然后按上述方式重新插入 PBD。不要修改 `other-config:multipathhandle` 参数，它将自动进行更改。

XenServer 中的多路径支持基于设备映射器 `multipathd components`（多路径组件）。激活和取消激活多路径节点由存储管理器 API 自动处理。与 Linux 中的标准 `dm-multipath` 工具不同，并非系统上的所有 LUN 都会自动创建设备映射器节点，只有 LUN 在由存储管理器层实时使用时，才会置备新设备映射器节点。不必使用任何 `dm-multipath` CLI 工具查询或刷新 XenServer 中的 DM 表节点。如果需要手动查询设备映射器的状态，或列出系统中活动设备映射器多路径节点，请使用 `mpathutil` 实用程序：

- `mpathutil` 列表
- `mpathutil` 状态



#### 注意

由于与集成多路径管理体系结构不兼容，标准 `dm-multipath` CLI 实用程序不应与 XenServer 结合使用。请使用 `mpathutil` CLI 工具查询主机上的节点状态。



#### 注意

Equallogic 阵列中的多路径支持不包括传统意义上的存储 IO 多路径。必须在网络/NIC 绑定级别处理多路径。有关为 Equallogic SR/LVMoISCSI SR 配置网络故障转移的信息，请参阅 Equallogic 文档。

## 5.5. XenServer 和 IntelliCache



#### 注意

只有在将 XenServer 与 XenDesktop 结合使用时才支持此功能。

通过将 XenServer 与 IntelliCache 结合使用，可以组合使用共享存储和本地存储，从而提高所托管虚拟桌面基础结构部署的成本效益。当多个虚拟机 (VM) 共享一个公用的操作系统映像时，优势尤其明显。既降低了存储阵列的负载，又提高了性能。此外，当本地存储从共享存储中缓存主映像时，进出共享存储的网络流量会减少。

IntelliCache 的工作原理是从 VM 主机上本地存储中的 VM 父 VDI 中缓存数据。然后，在从父项 VDI 中读取数据时，会填充此本地缓存。当多个 VM 共享一个公用父项 VDI 时（例如，全部都以一个特定主映像为基础），通过读取一个 VM 而拉入缓存的数据可供另一个 VM 使用。这意味着，不需要再访问共享存储上的主映像。

精简置备的本地 SR 是 IntelliCache 的必备条件。精简置备是一种优化可用存储空间利用率的方式。通过这种方式，可以更多地利用本地存储来代替共享存储。它依赖于数据块的按需分配，而不是预先分配所有存储块的传统方式。



#### 重要

精简置备将主机的默认本地存储类型由 LVM 改为 EXT3。要正常进行 XenDesktop 本地缓存，必须启用精简置备。

精简置备允许管理员为连接到存储库 (SR) 的 VM 提供比 SR 中实际可用的空间更多的空间。但对空间没有保证，并且在 VM 写入数据前，分配的 LUN 不会要求获得任何数据块。



#### 警告

由于空间中 VM 数量不断增加，从而占用所需磁盘容量，因此精简置备的 SR 可能会出现物理运行空间不足的情况。IntelliCache VM 处理此情况的方法是，在本地 SR 缓存已满时自动故障恢复。

到共享存储。由于 IntelliCache VM 的大小会快速增加，因此建议不要在同一个 SR 中混合使用传统的虚拟机与 IntelliCache VM。

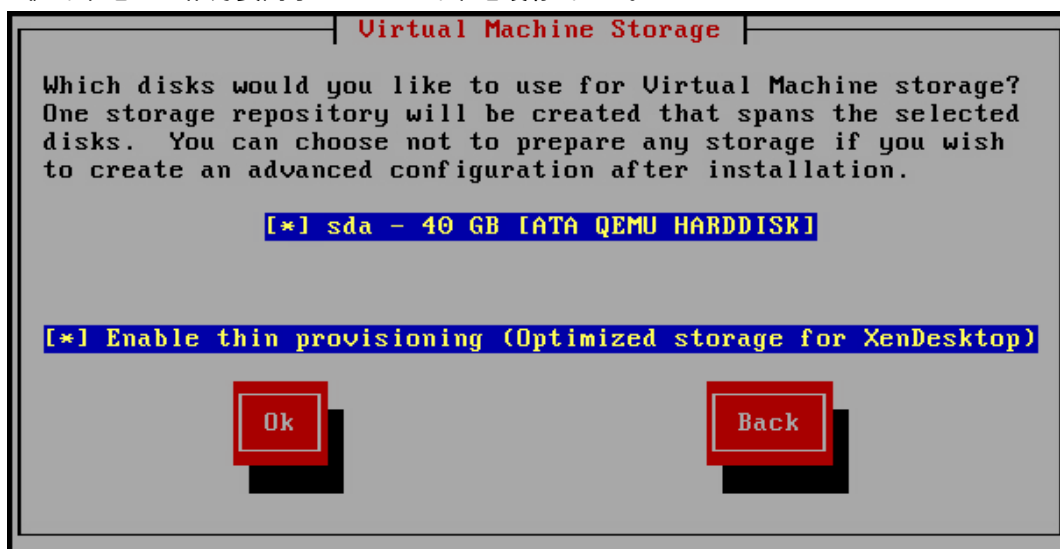
### 5.5.1. IntelliCache 部署

IntelliCache 必须在主机安装期间启用，或在主机运行期间使用 CLI 手动启用。

Citrix 建议使用高性能的本地存储设备（如固态硬盘或高性能 RAID 阵列），以保证尽可能快速的数据传输。在确定本地磁盘大小时，对数据吞吐量和存储容量均应加以考虑。用于托管源虚拟磁盘映像 (VDI) 的共享存储类型必须基于 NFS 或 EXT。

#### 5.5.1.1. 安装主机时启用

要在安装主机期间启用 IntelliCache，请在“Virtual Machine Storage”（虚拟机存储）屏幕上选择 Enable thin provisioning (Optimized storage for XenDesktop)（启用精简置备(用于 XenDesktop 的优化存储)）。此时将选择主机的本地 SR 作为要用于 VM VDI 的本地缓存的 SR。



#### 5.5.1.2. 将现有主机转换为使用精简置备

要销毁基于现有 LVM 的本地 SR，并将其替换为基于精简置备的 EXT3 的 SR，请输入以下命令。



#### 警告

这些命令将销毁现有的本地 SR，该 SR 上的 VM 将被永久删除。

```
localsr=`xe sr-list type=lvm host=<hostname> params=uuid --minimal`
echo localsr=$localsr
pbd=`xe pbd-list sr-uuid=$localsr params=uuid --minimal`
echo pbd=$pbd
xe pbd-unplug uuid=$pbd
xe pbd-destroy uuid=$pbd
xe sr-forget uuid=$localsr
sed -i "s/lvm/`ext`/" /etc/firstboot.d/data/default-storage.conf
rm -f /etc/firstboot.d/state/10-prepare-storage
rm -f /etc/firstboot.d/state/15-set-default-storage
service firstboot start
xe sr-list type=ext
```



要启用本地缓存，请输入以下命令：

```
xe host-disable host=<hostname>
  localsr=`xe sr-list type=ext host=<hostname> params=uuid --minimal`
xe host-enable-local-storage-caching host=<hostname> sr-uuid=$localsr
xe host-enable host=<hostname>
```

### 5.5.1.3. VM 引导行为

对于 VM 引导时的 VM VDI 行为，有两个选项：

#### 1. 共享桌面模式

在 VM 引导时，VDI 还原为上一次引导时的状态。VM 下次引导时，在 VM 运行时所做的所有更改都将丢失。

如果您打算交付用户无法进行永久性更改的标准化桌面，请选择此选项。

#### 2. 专有桌面模式

在 VM 引导时，VDI 处于上一次关机时的状态。

如果您打算允许用户对其桌面进行永久性更改，请选择此选项。

#### 5.5.1.3.1. VM 缓存行为设置

VDI 标志 `allow-caching` 指示缓存行为：

##### 5.5.1.3.1.1. 共享桌面模式

对于共享桌面，`on-boot` 选项设置为 `reset`，`allow-caching` 标志设置为 `true`，新的 VM 数据仅写入本地存储，而不写入共享存储。这意味着，共享存储的负载会显著减少。但是，不能在主机之间迁移 VM。

##### 5.5.1.3.1.2. 专有桌面模式

对于专有桌面，`on-boot` 选项设置为 `persist`，`allow-caching` 标志设置为 `true`，新的 VM 数据会同时写入本地和共享存储。读取缓存数据不需要共享存储的 I/O 流量，因而可以降低共享存储的负载。允许将 VM 迁移到其他主机，并且读入数据时会填充新主机上的本地缓存。

#### 5.5.1.4. 实现详细信息和故障排除

问：IntelliCache 是否与 XenMotion 和高可用性兼容？

答：当虚拟桌面处于专有模式（即，`on-boot=persist`）时，可以将 XenMotion 和高可用性功能与 IntelliCache 结合使用。



#### 警告

如果任何 VDI 的缓存行为标志设置为 `on-boot=reset` 和 `allow-caching=true`，则不能迁移该 VM。尝试迁移具有这些属性的 VM 将会失败。

问：本地缓存在本地磁盘中的什么位置？

答：该缓存位于存储库 (SR) 中。每个主机都有一个配置参数（称为 `local-cache-sr`），用于指示使用哪个（本地）SR 存储缓存文件。其通常为 EXT 类型的 SR。在运行采用 IntelliCache 的 VM 时，您将看到 SR 中名为 `<uuid>.vhdcache` 的文件。这是与具有指定 UUID 的 VDI 对应的缓存文件。XenCenter 中不会显示这些文件，查看这些文件的唯一方法是登录 `dom0` 并列出行 `/var/run/sr-mount/<sr-uuid>` 的内容。

问：如何指定要用作缓存的特定 SR？

答：主机对象字段 local-cache-sr 是指本地 SR。可通过运行以下命令来查看其值：

```
xe sr-list params=local-cache-sr,uuid,name-label
```

该字段的设置：

- 如果在主机安装程序中选择了“Enable thin provisioning”（启用精简置备）选项，可以在主机安装后设置
- 可以通过执行 `xe host-enable-local-storage-caching host=<host> sr-uuid=<sr>` 来设置。使用此命令时需要禁用指定的主机，而且必须关闭 VM。

第一个选项使用 EXT 类型的本地 SR 并且在主机安装期间创建。第二个选项使用在命令行中指定的 SR。



#### 警告

只有配置了多个本地 SR 的用户才需要执行这些步骤。

问：何时删除本地缓存？

答：只有在删除 VDI 本身时，才会删除 VDI 缓存文件。在将 VDI 附加到 VM（例如，在 VM 启动时）时，会重置缓存。如果删除 VDI 时主机处于脱机状态，则在启动时运行的 SR 同步将对缓存文件进行回收。



#### 注意

在将 VM 迁移到不同的主机或关闭时，不会从主机中删除缓存文件。

## 5.6. 存储读取缓存



#### 注意

存储读取缓存适用于 XenServer Enterprise Edition 客户或通过 XenDesktop/XenApp 授权访问 XenServer 的客户。要了解有关 XenServer 版本以及升级方式的详细信息，请单击[此处](#)访问 Citrix Web 站点。有关许可的详细信息，请参阅 [XenServer 7.0 许可常见问题解答](#)。

读取缓存可以改进 VM 的磁盘性能，因为从外部磁盘进行初始读取后，数据缓存在主机的可用内存中。在单一基础 VM 上克隆多个 VM 的情况下，它可以显著改进性能。例如，在 XenDesktop Machine Creation Service (MCS) 环境中，它将显著减少从磁盘读取的块的数量。

无论何时需要从磁盘多次读取数据，都可以看到这种性能改进，因为数据缓存在内存中。这是在繁重 I/O 情景中发生的最明显的服务降级。例如，当非常狭小的时间框架内有大量最终用户引导（引导风暴）时，或者当安排大量 VM 同时运行恶意软件扫描（杀毒风暴）时。如果您具有恰当的许可证类型，则默认启用读取缓存。

### 5.6.1. 启用和禁用

对于基于文件的 SR，例如 NFS 和 EXT3 SR 类型，读取缓存在默认情况下处于启用状态。它针对所有其他 SR 处于禁用状态。

要针对特定 SR 禁用读取缓存，请输入：

```
xe sr-param-set uuid=<sr-uuid> other-config:o_direct=true
```



### 5.6.2. 限制

- 读取缓存仅适用于 NFS 和 EXT3 SR。它不适用于其他 SR 类型。
- 读取缓存仅适用于只读 VDI 和父 VDI；它们存在于从“快速克隆”或快照磁盘创建 VM 的情况中。从单一“黄金”映像克隆 VM 时，将看到最显著的性能改进。
- 性能改进取决于主机控制域 (dom0) 中的可用内存量。增加 dom0 的内存量将允许为读取缓存分配更多内存。有关如何配置 dom0 内存的信息，请参阅 [CTX134951 - 如何在 XenServer 6.1 及更高版本中配置 dom0 内存](#)。

### 5.6.3. 与 IntelliCache 比较

IntelliCache 和基于内存的读取缓存在某种程度上互为补充。IntelliCache 不仅缓存在不同层上，还缓存写入和读取。主要差异在于，IntelliCache 通过将读取从网络缓存到本地磁盘，而内存中读取缓存可以将读取从网络或磁盘缓存到主机内存中。内存中读取缓存的优势是，内存依然比固态硬盘 (SSD) 快一个数量级，因此，引导风暴和其他繁重 I/O 情况下的性能应当得到改进。

读取缓存和 IntelliCache 可以同时启用；在此情况下，来自网络的读取由 IntelliCache 缓存到本地磁盘，来自本地磁盘的读取通过读取缓存缓存在内存中。

### 5.6.4. 设置读取缓存大小

通过为 XenServer 的控制域 (dom0) 分配更多的内存，可以优化读取缓存性能。



#### 重要

为实现优化，应当逐一在池中的全部主机上设置读取缓存大小。此外，还必须在池中的所有主机上设置读取缓存大小的任何后续更改。

在 XenServer 主机上，打开本地 shell，以 root 用户身份登录。

要设置读取缓存的大小，请运行以下命令：

```
/opt/xensource/libexec/xen-cmdline --set-xen dom0_mem=<nn>M,max:<nn>M
```

初始值和最大值应设为同一值，例如，将 dom0 内存设置为 2048 MB：

```
/opt/xensource/libexec/xen-cmdline --set-xen dom0_mem=20480M,max:20480M
```



#### 重要

对读取缓存大小执行任何更改后，必须重新启动所有主机。

#### 5.6.4.1. 如何查看当前 dom0 内存分配

要查看当前 dom0 内存设置，请输入：

```
free -m
```

free -m 的输出将显示当前 dom0 内存设置。由于各种开销，该值可能低于预期。以下示例表显示主机输出，dom0 设为 752 MB

	总数	已使用	可用	共享虚拟机	缓冲	已缓存
内存	594	486	108	0	149	78

-/+ 缓冲/缓存		258	336			
交换	511	49	462			

可以使用的值范围是什么？

因为 XenServer 7.0 控制域 (dom0) 是 64 位的，所以可以使用较大值，例如 32768 MB。Citrix 建议您不要将初始值设为低于 752 MB，因为这样做可能会导致主机无法引导。

#### 5.6.4.2. XenCenter 显示说明

管理员应当注意，可以认为整个主机的内存由 Xen 虚拟机管理程序、dom0、VM 和可用内存组成。尽管 dom0 和 VM 的内存大小通常是固定的，但 Xen 虚拟机管理程序使用可变的内存量。这取决于许多因素，包括任意时间主机上正在运行的 VM 的数量以及配置这些 VM 的方式。不可能限制 Xen 使用的内存量，因为 Xen 稍后可能耗尽内存，并阻止新 VM 启动，即使主机有可用内存。

要查看分配给主机的内存，在 XenCenter 中选择主机，然后单击内存选项卡。

XenServer 字段显示分配给 dom0 与 Xen 内存的总和。因此，显示的内存量可能高于管理员指定的值，它的大小会在启动和停止 VM 时发生改变，即使管理员为 dom0 设置了固定大小。

## 5.7. 管理存储库

本部分介绍了存储库 (SR) 日常管理中所需的各种操作，包括实时 VDI 迁移。

### 5.7.1. 卸载 SRs

存储库 (SR) 可被临时或永久删除。

- 分离：断开存储设备和池或主机之间的关联，(PBD Unplug) 和 SR (及其 VDI) 变得不可访问。VDI 和内容以及 VM 用来访问 VDI 的元信息将被保存。需要临时将 SR 置于脱机状态 (例如，为了进行维护) 时，可以使用分离。分离的 SR 随后可以重新连接。
- 忘记：在物理磁盘上保存 SR 内容，但用于将 VM 连接到它包含的 VDI 的信息将被永久删除 (PBD 和 VBD Unplug)。例如，这允许您将 SR 重新连接到另一个 XenServer 主机，而不必删除 SR 的任何内容。
- 销毁：从物理磁盘删除 SR 的内容。

在“销毁”或“忘记”的情况下，必须从主机连接到 SR 的 PBD 必须从主机上拔下。

1. 拔出 PBD 以将 SR 与相应的 XenServer 主机分离：

```
xe pbd-unplug uuid=<pbid_uuid>
```

2. 销毁 SR，即从 XenServer 主机数据库中删除 SR 及其相应的 PBD 并从物理磁盘中删除 SR 内容：

```
xe sr-destroy uuid=<sr_uuid>
```

3. 忘记 SR，即从 XenServer 主机数据库中删除 SR 及其相应的 PBD，但会保持物理介质上的实际 SR 内容不变：

```
xe sr-forget uuid=<sr_uuid>
```



#### 注意

对与 SR 相对应的软件对象进行垃圾回收可能需要一定的时间。

### 5.7.2. 引入 SR

要重新引入以前忘记的 SR，需要创建 PBD，将 PBD 手动插入相应的 XenServer 主机以激活 SR。

以下示例引入一个 lvmoiscsi 类型的 SR。

1. 探测现有 SR 以确定其 UUID：

```
xe sr-probe type=lvmoiscsi device-config:target=<192.168.1.10> \
device-config:targetIQN=<192.168.1.10:filer1> \
device-config:SCSIid=<149455400000000000000000002000000b70200000f000000>
```

2. 引入从 sr-probe 命令返回的现有 SR UUID。将返回新 SR 的 UUID：

```
xe sr-introduce content-type=user name-label=<"Example Shared LVM over iSCSI SR">
shared=true uuid=<valid_sr_uuid> type=lvmoiscsi
```

3. 创建一个随附 SR 的 PBD。将返回新 PBD 的 UUID：

```
xe pbd-create type=lvmoiscsi host-uuid=<valid_uuid> sr-uuid=<valid_sr_uuid> \
device-config:target=<192.168.0.1> \
device-config:targetIQN=<192.168.1.10:filer1> \
device-config:SCSIid=<149455400000000000000000002000000b70200000f000000>
```

4. 插入此 PBD 以连接 SR：

```
xe pbd-plug uuid=<pbd_uuid>
```

5. 验证此 PBD 插件的状态。如果插入成功，则 currently-attached 属性将为 true：

```
xe pbd-list sr-uuid=<sr_uuid>
```



#### 注意

必须为资源池中的各个主机执行步骤 3 到步骤 5，也可以使用 XenCenter 中的修复存储库功能执行这些步骤。

### 5.7.3. 实时 LUN 扩展

为了满足容量要求，可能需要为存储阵列添加容量，以增加为 XenServer 主机置备的 LUN 大小。实时 LUN 扩展允许您增加 LUN 的大小而不会出现任何 VM 停机的情况。

将更多容量添加到存储阵列后，请输入

```
xe sr-scan sr-uuid=<sr_uuid>
```

此命令会重新扫描 SR，之后将添加任何额外容量并使其可用。

此操作在 XenCenter 中可用；选择 SR 以调整大小，然后单击重新扫描。有关详细信息，请按 F1 以显示 XenCenter 联机帮助。



#### 注意

在 XenServer 的早期版本中，需要使用显式命令来重新调整 iSCSI 和 HBA SR 的物理卷组的大小。不再需要这些命令。



#### 警告

无法收缩或截断 LUN。减小存储阵列中的 LUN 大小可能导致数据丢失。

#### 5.7.4. 实时 VDI 迁移

实时 VDI 迁移允许管理员重新定位 VM 虚拟磁盘映像 (VDI)，而无需关闭 VM。此功能支持执行以下管理操作：

- 将 VM 从便宜的本地存储移至快速灵活且支持阵列的存储中。
- 将 VM 从开发环境移至生产环境。
- 在存储层中移动 VM (如果 VM 受到存储容量限制)。
- 执行存储阵列升级。

##### 5.7.4.1. 限制和注意事项

实时 VDI 迁移受以下限制和注意事项约束

- 在目标库中必须具有充足的磁盘空间。
- 无法迁移具有多个快照的 VDI。

##### 5.7.4.2. 使用 XenCenter 移动虚拟磁盘

1. 在资源窗格中，选择当前存储虚拟磁盘的 SR，然后单击存储选项卡。
2. 在虚拟磁盘列表中，选择要移动的虚拟磁盘，然后单击移动。
3. 在移动虚拟磁盘对话框中，选择要将 VDI 移至的目标 SR。



#### 注意

确保该 SR 具有可容纳另一个虚拟磁盘的充足空间：可用空间在可用 SR 列表中显示。

4. 单击移动以移动虚拟磁盘。

对于 xe cli 参考，请参阅[第 4.23.9 节 “vdi-pool-migrate”](#)。

#### 5.7.5. SR 之间的冷 VDI 迁移 (脱机迁移)

可以将与 VM 关联的 VDI 从一个 SR 复制到另一个 SR 以符合维护要求或分层存储配置。通过 XenCenter，可以将 VM 及其所有 VDI 复制到同一个或另一个 SR 中，也可以结合使用 XenCenter 和 xe CLI 来复制各个 VDI。

对于 xe cli 参考，请参阅[第 4.26.25 节 “vm-migrate”](#)。

##### 5.7.5.1. 将 VM 的所有 VDI 复制到另一个 SR

XenCenter 复制 VM 功能可以在同一个或另一个 SR 上创建所选 VM 的所有 VDI 的副本。默认情况下，源 VM 和 VDI 不会受到影响。如果只是将 VM 移动到所选 SR，而不是创建副本，请选择复制虚拟机对话框中的删除原始 VM 选项。

1. 关闭 VM。
2. 在 XenCenter 中，选择 VM，然后选择 VM > 复制 VM 菜单选项。
3. 选择所需的目标 SR。

##### 5.7.5.2. 将各个 VDI 复制到另一个 SR

可以结合使用 xe CLI 和 XenCenter 来在 SR 之间复制各个 VDI。

1. 关闭 VM。
2. 使用 xe CLI 确定要移动的 VDI 的 UUID。如果 VM 具有 DVD 驱动器，则该 VM 的 vdi-uuid 将被列为 <not in database>，可以将其忽略。

```
xe vbd-list vm-uuid=<valid_vm_uuid>
```



#### 注意

vbd-list 命令显示 VBD 和 VDI 两者的 UUID。请务必记录 VDI 的 UUID 而不是 VBD 的 UUID。

3. 在 XenCenter 中，选择该 VM 的存储选项卡。对于要移动的每个 VDI，请选择该 VDI，然后单击分离按钮。还可以使用 vbd-destroy 命令执行此步骤。



#### 注意

如果使用 vbd-destroy 命令分离 VDI 的 UUID，请务必首先检查 VBD 是否将参数 other-config:owner 设置为了 true。如果为 true，请将其设置为 false。发出设置为 other-config:owner=true 的 vbd-destroy 命令还将销毁关联的 VDI。

4. 使用 vdi-copy 命令将该 VM 的每个要移动的 VDI 复制到所需 SR 中。

```
xe vdi-copy uuid=<valid_vdi_uuid> sr-uuid=<valid_sr_uuid>
```

5. 在 XenCenter 中，选择该 VM 的存储选项卡。单击连接按钮并从新 SR 中选择 VDI。还可以使用 vbd-create 命令执行此步骤。
6. 要删除原始 VDI，请在 XenCenter 中选择原始 SR 的存储选项卡。将列出这些原始 VDI，但其 VM 字段的值为空，可以使用“删除”按钮删除这些原始 VDI。

### 5.7.6. 将本地光纤通道 SR 转换为共享 SR

使用 xe CLI 和 XenCenter 修复存储库功能，以便将当地 FC SR 转换为共享 FC SR：

1. 将资源池中的所有主机升级为 XenServer 7.0。
2. 确保池中的所有主机都对 SR 的 LUN 进行了适当的分区。有关使用 sr-probe 命令验证每台主机上都存在 LUN 的详细信息，请参阅第 5.3.2 节“探测 SR”。
3. 将 SR 转换为共享 SR：

```
xe sr-param-set shared=true uuid=<local_fc_sr>
```

4. 在 XenCenter 中，SR 将从主机级别移到池级别，以指示它现在为共享 SR。将使用红色的感叹号（！）对该 SR 进行标记，以显示当前并未在池中的所有主机上都插上了该 SR。
5. 选择该 SR，然后选择存储 > 修复存储库菜单选项。
6. 单击修复以创建并为池中的每个主机插上 PBD。

### 5.7.7. 使用丢弃功能回收备用阵列上基于块的存储的空间

空间回收允许您在由存储阵列精简置备的 LUN 上释放一些未使用的块（例如，SR 中已删除的 VDI）。释放后，回收的空间随后可由该阵列重新使用。



#### 注意

此功能仅适用于存储阵列子集。请参阅 XenServer [硬件兼容性列表](#) 和存储厂商特定文档，确定阵列是否支持此功能，以及是否需要特定配置以便运行。

使用 XenCenter 回收空间：

1. 选择基础结构视图，然后单击连接到 SR 的主机或池。
2. 单击存储选项卡。
3. 从列表中选择该 SR，然后单击回收释放的空间。
4. 要确认该操作，请单击是。
5. 单击通知，然后单击事件，查看操作状态。

有关详细信息，请在 XenCenter 中按 F1 以访问联机帮助。



#### 注意

- 这是 XenCenter 的唯一操作。
- 此操作仅适用于基于 LVM 的 SR，这些 SR 基于阵列上经过精简置备的 LUN。
- 本地 SSD 也将从空间回收中受益。
- 不需要基于文件的 SR（例如 NFS/Ext3）；XenCenter 中的回收释放的空间按钮不适用于这些 SR 类型。
- 空间回收是一种密集型操作，会导致存储阵列性能下降。因此，仅应在阵列上需要空间回收时启动。Citrix 建议您将此工作安排在非阵列需求高峰期进行。

### 5.7.8. 删除快照时自动回收空间

通过 XenServer 7.0 删除快照时，所有已分配的空间都将自动回收，并且无需重新引导 VM；这一过程称为联机合并。



#### 注意

联机合并仅适用于基于 LVM 的 SR（LVM、LVMoISCsi 和 LVMoHBA），而不适用于 EXT 或 NFS SR（这些 SR 的行为保持不变）。

某些情况下，空间自动回收可能无法继续，以下情况下建议使用“脱机合并”工具：

- VM I/O 吞吐量相当大时
- 一段时间后未回收空间时



#### 注意

由于需要执行挂起/恢复操作，因此运行“脱机合并”工具会导致 VM 停机。

在运行该工具之前，请删除不再需要的任何快照和克隆；脚本将回收尽可能多的空间留给剩余的快照/克隆。如果要回收所有的空间，请删除所有的快照和克隆。

所有 VM 磁盘必须位于一个主机的共享或本地存储上。如果 VM 的磁盘采用这两种存储类型，VM 将无法合并。

#### 5.7.8.1. 使用脱机合并工具回收空间



#### 注意

联机合并仅适用于基于 LVM 的 SR（LVM、LVMoISCsi 和 LVMoHBA），而不适用于 EXT 或 NFS SR（这些 SR 的行为保持不变）。

通过 XenCenter 启用隐藏的对象：“视图”菜单->“隐藏对象”。在“资源”窗格中，选择要获得 UUID 的 VM。UUID 将显示在“常规”选项卡中。

在“资源”窗格中，选择资源池主服务器主机（列表中的第一个主机）。UUID 将显示在“常规”选项卡中。如果未使用资源池，请选择该 VM 主机。

1. 在主机上打开控制台并运行以下命令：

```
xe host-call-plugin host-uuid=<host-UUID> \
  plugin=coalesce-leaf fn=leaf-coalesce args:vm_uuid=<VM-UUID>
```

例如，如果 VM UUID 是 9bad4022-2c2d-dee6-abf5-1b6195b1dad5，主机 UUID 是 b8722062-de95-4d95-9baa-a5fe343898ea，则应运行以下命令：

```
xe host-call-plugin host-uuid=b8722062-de95-4d95-9baa-a5fe343898ea \
  plugin=coalesce-leaf fn=leaf-coalesce args:vm_uuid=9bad4022-2c2d-dee6-abf5-1b6195b1dad5
```

2. 此命令将挂起 VM（除非 VM 已经关机），启动空间回收过程，然后恢复该 VM。



#### 注意

Citrix 建议在执行脱机合并前，手动关闭或挂起 VM（使用 XenCenter 或 XenServer CLI）。如果在运行中的 VM 上执行合并工具，该工具将自动挂起 VM，执行所需的 VDI 合并操作，然后恢复该 VM。

如果要合并的虚拟磁盘映像 (VDI) 位于共享存储上，则必须在池主服务器上执行脱机合并工具。

如果要合并的 VDI 位于本地存储上，则必须在本地存储所连的服务器上执行脱机合并工具。

### 5.7.9. 调整磁盘 IO 调度程序

为了获得一般性能，会对所有新 SR 类型应用默认磁盘调度程序 noop。noop 调度程序提供可与访问相同设备的 VM 相媲美的最佳性能。为了应用磁盘 QoS（请参阅第 5.7.10 节“虚拟磁盘 QoS 设置”），需要覆盖默认设置并将 cfq 磁盘调度程序分配给 SR。必须拔出并重新插上相应的 PBD 以使调度程序参数生效。可以使用以下命令调整磁盘调度程序：

```
xe sr-param-set other-config:scheduler=noop|cfq|anticipatory|deadline \
  uuid=<valid_sr_uuid>
```



#### 注意

这不会影响 EqualLogic、NetApp 或 NFS 存储。

### 5.7.10. 虚拟磁盘 QoS 设置

虚拟磁盘具有可选 I/O 优先级质量服务 (QoS) 设置。可以按照本部分中介绍的方法使用 xe CLI 将此设置应用到现有虚拟磁盘。

在共享 SR（即多个主机访问同一 LUN）的情况下，会将 QoS 设置应用到从同一主机访问 LUN 的 VBD，即不会跨池中的主机应用 QoS。

为 VBD 配置任何 QoS 参数之前，请确保已适当设置了 SR 的磁盘调度程序。有关如何调整调度程序的详细信息，请参阅第 5.7.9 节“调整磁盘 IO 调度程序”。必须在需要 QoS 的 SR 上将调度程序参数设置为 cfq。



#### 注意

请记住在 SR 上将该调度程序设置为 cfq，并确保已重新插上 PBD 以使调度程序更改生效。





第一个参数是 `qos_algorithm_type`。需要将此参数值设置为 `ionice`，这是此版本的虚拟磁盘唯一支持的 QoS 算法类型。

QoS 参数自身就是使用分配给 `qos_algorithm_param` 参数的键/值对进行设置的。对于虚拟磁盘，`qos_algorithm_param` 使用 `sched` 键，并且根据参数值，还需要使用类键。

`qos_algorithm_param:sched` 具有以下可能值：

- `sched=rt` 或 `sched=real-time` 将 QoS 调度参数设置为实时优先级，此优先级需要使用类参数来设置
- `sched=idle` 将 QoS 调度参数设置为空闲优先级，此优先级不需要使用类参数来设置
- `sched=<anything>` 将 QoS 调度参数设置为 best-effort 优先级，此优先级需要使用类参数来设置

`class` 的可能值：

- 以下关键词之一：最高,高,普通,低,最低
- 0 和 7 之间的一个整数，其中 7 表示最高优先级，0 表示最低优先级。因此，如果 I/O 请求的优先级为 5，则其优先级要高于优先级为 2 的 I/O 请求。

要启用磁盘 QoS 设置，您还需要将 `other-config:scheduler` 设置为 `cfq` 并重新插上所涉及的存储的 PBD。

例如，下列 CLI 命令将虚拟磁盘的 VBD 设置为使用实时优先级 5：

```
xe vbd-param-set uuid=<vbd_uuid> qos_algorithm_type=ionice
xe vbd-param-set uuid=<vbd_uuid> qos_algorithm_params:sched=rt
xe vbd-param-set uuid=<vbd_uuid> qos_algorithm_params:class=5
xe sr-param-set uuid=<sr_uuid> other-config:scheduler=cfq
xe pbd-plug uuid=<pbd_uuid>
```



## 第 6 章 配置 VM 内存

首次创建 VM 时，会为其分配固定数量的内存。要增大 XenServer 环境中物理内存的利用率，可以使用动态内存控制 (DMC)，这是一种能够在 VM 之间动态重新分配内存的内存管理功能。

XenCenter 在其内存选项卡中提供了内存使用率的图形显示。这在 XenCenter 帮助中有说明。

动态内存控制 (DMC) 具有以下优势：

- 无需重新启动即可添加或删除内存，从而为用户提供更加优异的无缝体验。
- 服务器满载后，DMC 允许您在这些服务器上启动更多 VM，从而按比例减少分配给正在运行的 VM 的内存量。

### 6.1. 什么是动态内存控制 (DMC)？

XenServer DMC 的工作原理是：自动调整正在运行的 VM 的内存，使分配给每个 VM 的内存量保持在指定的最小内存值与最大内存值之间，确保性能并允许每台服务器具有更大的 VM 密度。

如果未使用 DMC，则当服务器满载时，启动更多 VM 将因“内存不足”错误而失败：要减少现有 VM 内存分配并为更多 VM 释放空间，您必须编辑每个 VM 的内存分配，然后重新引导 VM。启用 DMC 之后，即使在服务器满载时，XenServer 仍能通过在 VM 定义的内存范围内自动减少正在运行的 VM 的当前内存分配来尝试回收内存。

#### 6.1.1. 动态范围的概念

对于每个 VM，管理员可以设置一个动态内存范围，在此范围内，可以添加/删除 VM 的内存而无需重新引导。VM 运行时，管理员可以调整动态范围。XenServer 始终确保分配给 VM 的内存量保持在动态范围内，因此，在 VM 运行时调整动态范围可能会导致 XenServer 调整分配给该 VM 的内存量。（管理员将最小/最大动态内存设置为相同的值，从而强制 XenServer 确保分配给 VM 此内存量，此为最极端的情况。）如果需要“满载”服务器上启动新 VM，正在运行的 VM 会“腾出”内存来启动新 VM。获取所需的额外内存的方法是：在正在运行的现有 VM 的预定义动态范围内按比例腾出内存。

DMC 允许您配置最小和最大动态内存级别，从而创建 VM 将在其中运行的动态内存范围 (DMR)。

- 动态最小内存：分配给 VM 的内存下限。
- 动态上限：分配给 VM 的内存上限。

例如，如果最小动态内存设置为 512 MB，最大动态内存设置为 1024 MB，则 VM 的动态内存范围 (DMR) 为 512 - 1024 MB，VM 将在此范围内运行。通过 DMC，XenServer 可确保始终在每个 VM 的指定 DMR 内为其分配内存。

#### 6.1.2. 静态范围的概念

XenServer 支持的许多操作系统不完全“理解”动态添加或删除内存的概念。因此，XenServer 必须声明 VM 在引导时需要占用的最大内存量。（这允许来宾操作系统相应调整其页表及其他内存管理结构的大小。）这在 XenServer 中引入了静态内存范围的概念。静态内存范围无法在 VM 运行时进行调整。对于特定引导，动态范围受到限制，例如始终限制在此静态范围内。请注意，最小静态内存（静态范围的下限）用于保护管理员，设置为操作系统可以在 XenServer 上运行时占用的最少内存量。



#### 注意

静态最低级别设置为每个操作系统支持的级别，因此 Citrix 建议您不要对其进行更改，有关更多详细信息，请参阅内存限制表。

设置高于最大动态内存的静态最高级别，意味着如果在将来需要将更多内存分配给 VM，您无需重新引导即可完成该操作。

### 6.1.3. DMC 行为

自动 VM 腾出内存

- 如果未启用 DMC，则当主机满载时，新 VM 会因“内存不足”错误而导致启动失败。
- 如果启用了 DMC，即使在主机满载时，XenServer 仍能尝试回收内存（方法是：在 VM 定义的动态范围内自动减少正在运行的 VM 的内存分配）。在上述方法中，正在运行的 VM 在最小动态内存与最大动态内存之间以相同的间距按比例为主机上的所有 VM 腾出内存。

启用了 DMC 时

- 主机内存足够时 - 正在运行的所有 VM 都将达到其最大动态内存级别。
- 主机内存不足时 - 正在运行的所有 VM 都将达到其最小动态内存级别。

配置 DMC 时，请记住，只将少量内存分配给 VM 会对其造成负面影响。例如，分配过少内存：

- 使用动态内存控制减少 VM 可用的物理内存量可能会致使 VM 引导缓慢。同样，如果为 VM 分配的内存过少，VM 启动可能极其缓慢。
- 如果为 VM 设置的动态内存最小值过低，可能会导致 VM 在启动时出现性能或稳定性不佳等问题。

### 6.1.4. DMC 的工作原理

使用 DMC，可以在以下两种模式中的一种模式下运行来宾虚拟机：

1. 目标模式：管理员指定来宾操作系统的内存目标。XenServer 调整来宾操作系统的内存分配以满足目标的要求。在虚拟服务器环境中，以及在您知道希望来宾操作系统使用的准确内存量的任何情况下，指定内存特别有用。XenServer 将调整来宾操作系统的内存分配，以满足您指定的目标。
2. 动态范围模式：管理员指定来宾操作系统的动态内存范围；XenServer 从该范围内选择一个目标，并调整来宾操作系统的内存分配以满足此目标。在虚拟桌面环境中，以及您希望 XenServer 动态重新分配主机内存以响应不断变化的来宾操作系统数目或不断变化的主机内存压力的任何情况下，指定动态内存范围特别有用。XenServer 会从该范围内选择一个目标，并调整来宾操作系统的内存分配以满足该目标。



#### 注意

对于任何正在运行的来宾操作系统，可以随时在目标模式与动态范围模式之间进行更改。只需指定新目标或新动态范围，XenServer 即可完成剩余的工作。

### 6.1.5. 内存限制

XenServer 允许管理员对任意来宾操作系统使用所有内存控制操作。但是，对于所有来宾操作系统，XenServer 会强制实行以下内存属性顺序限制：

$$0 \leq \text{memory-static-min} \leq \text{memory-dynamic-min} \leq \text{memory-dynamic-max} \leq \text{memory-static-max}$$

XenServer 允许管理员将来宾操作系统内存属性更改为满足此限制（受验证检查限制）的任何值。但是，除上述限制外，对于支持的每个操作系统，Citrix 仅支持特定的来宾操作系统内存配置。受支持的配置的范围取决于正在使用的来宾操作系统。XenServer 不会阻止管理员将来宾操作系统配置为超出支持的限制。但是，强烈建议客户将内存属性保持在支持的限制内，以避免出现性能或稳定性等问题。有关每个受支持操作系统的历史和最大内存限制的详细准则，请参阅《XenServer 虚拟机用户指南》。

**警告**

配置来宾系统内存时，Citrix 建议不要超出操作系统可寻址的最大物理内存数量。设置大于操作系统所支持的限制的内存最大值，可能会导致来宾操作系统中出现稳定性问题。

此外，客户还应当注意，所有受支持操作系统的最小动态值应当大约或等于最大静态值的四分之一。将下限降低到最小动态值以下也可能导致稳定性问题。建议管理员认真校准其 VM 的大小，并确保其所用应用程序集在最小动态值下可靠地工作。

## 6.2. xe CLI 命令

### 6.2.1. 显示 VM 的静态内存属性

1. 查找所需 VM 的 uuid：

```
xe vm-list
```

2. 记下该 uuid，然后运行命令 param-name=memory-static

```
xe vm-param-get uuid=<uuid> param-name=memory-static-{min,max}
```

例如，以下命令将显示 uuid 以 ec77 开头的 VM 的最大静态内存属性：

```
xe vm-param-get uuid= \
ec77a893-bff2-aa5c-7ef2-9c3acf0f83c0 \
param-name=memory-static-max;
268435456
```

上述命令显示此 VM 的最大静态内存为 268435456 字节 (256 MB)。

### 6.2.2. 显示 VM 的动态内存属性

要显示动态内存属性，请按照上述步骤进行操作，但请使用命令 param-name=memory-dynamic：

1. 查找所需 VM 的 uuid：

```
xe vm-list
```

2. 记下该 uuid，然后运行命令 param-name=memory-dynamic：

```
xe vm-param-get uuid=<uuid> param-name=memory-dynamic-{min,max}
```

例如，以下命令将显示 uuid 以 ec77 开头的 VM 的最大动态内存属性：

```
xe vm-param-get uuid= \
ec77a893-bff2-aa5c-7ef2-9c3acf0f83c0 \
param-name=memory-dynamic-max;
134217728
```

上述命令显示此 VM 的最大动态内存为 134217728 字节 (128 MB)。

### 6.2.3. 更新内存属性

**警告**

设置最小/最大静态（或最小/最大动态）参数时使用正确的顺序至关重要。此外，必须保持以下限制有效：

$$0 \leq \text{memory-static-min} \leq \text{memory-dynamic-min} \leq \text{memory-dynamic-max} \leq \text{memory-static-max}$$

更新虚拟机的静态内存范围：

```
xe vm-memory-static-range-set uuid=<uuid> min=<value> max=<value>
```

更新虚拟机的动态内存范围：

```
xe vm-memory-dynamic-range-set \
  uuid=<uuid> min=<value> \
  max=<value>
```

在虚拟服务器环境中，以及在您知道希望来宾操作系统使用的准确内存量的任何情况下，指定内存特别有用。XenServer 将调整来宾操作系统的内存分配，以满足您指定的目标。例如：

```
xe vm-target-set target=<value> vm=<vm-name>
```

更新虚拟机的所有内存限制（静态限制和动态限制）：

```
xe vm-memory-limits-set \
  uuid=<uuid> \
  static-min=<value> \
  dynamic-min=<value> \
  dynamic-max=<value> static-max=<value>
```



#### 注意

- 要将特定数量且不会变化的内存分配给 VM，请将最大动态内存和最小动态内存设置为相同的值。
- 不能将 VM 的动态内存增加到超过最大静态内存。
- 要更改 VM 的最大静态内存，需要挂起或关闭 VM。

### 6.2.4. 更新各个内存属性



#### 警告

静态最低级别设置为每个操作系统支持的级别，因此 Citrix 建议您不要对其进行更改，有关更多详细信息，请参阅内存限制表。

更新 VM 的动态内存属性。

1. 查找所需 VM 的 uuid：

```
xe vm-list
```

2. 记下该 uuid，然后使用命令 memory-dynamic-{min,max}=<value>

```
xe vm-param-set uuid=<uuid> memory-dynamic-{min,max}=<value>
```

下例将最大动态内存更改为 128 MB：

```
xe vm-param-set uuid=ec77a893-bff2-aa5c-7ef2-9c3acf0f83c0 memory-dynamic-max=128MiB
```

## 6.3. 升级问题

从 Citrix XenServer 5.5 升级之后，XenServer 会设置所有 VM 内存，使最小动态内存等于最大动态内存。

## 第 7 章 XenServer 内存使用率

XenServer 主机的内存占用由两部分构成。第一个是 Xen 虚拟机管理程序本身占用的内存；第二个是主机的控制域占用的内存。也称作“Domain0”或“dom0”，控制域是一个安全的特权 Linux VM，运行 XenServer 管理 toolstack (xapi)。除了提供 XenServer 管理功能之外，控制域还运行驱动程序堆栈，提供对物理设备的用户创建 VM 访问。

### 7.1. 控制域内存

分配给控制域的内存量可以自动调整，并且基于物理主机上的物理内存量：

主机内存 (GB)	分配的控制域内存 (MB)
20 - 24	752
24 - 48	2048
48 - 64	3072
64 - 1024	4096



#### 注意

在 XenCenter 中的 XenServer 字段中报告的内存量可能会超出这些值。这是因为报告的内存量包括由控制域 (dom0)、Xen 虚拟机管理程序本身和崩溃内核占用的内存量。对于使用更多内存的主机来说，虚拟机管理程序占用的内存量更大。

#### 7.1.1. 更改分配给控制域的内存量

在内存量较少（例如，不足 16 GB）的主机上，您可能想将分配给控制域的内存减少到默认 752 MB 以下。但是，Citrix 建议您不要将 dom0 内存减少到 400 MB 以下。

1. 在 XenServer 主机上，打开本地 shell，以 root 用户身份登录。
2. 输入以下命令：

```
/opt/xensource/libexec/xen-cmdline --set-xen dom0_mem=<nn>M,max:<nn>M
```

其中 <nn> 代表分配给 dom0 的内存量（以 MB 为单位）。

3. 在控制台上使用 XenCenter 或 reboot 命令重新启动 XenServer 主机。

重新启动主机时，在 xs 控制台上，运行 free 命令，验证新的内存设置。



#### 警告

增加控制域内存量将导致 VM 的可用内存减少。

分配给控制域的内存量可以增加到超过上面显示的值。但是，这应当只能在 Citrix 支持部门的指导下执行。

#### 7.1.2. VM 的可用内存量

要了解实际可将多少主机内存分配给 VM，请先查找该主机 memory-free 字段的值，然后输入 vm-compute-maximum-memory 命令，获取可分配给该 VM 的实际可用内存量，例如：



```
xe host-list uuid=<host_uuid> params=memory-free  
xe vm-compute-maximum-memory vm=<vm_name> total=<host_memory_free_value>
```

## 第 8 章 灾难恢复和备份

XenServer 灾难恢复 (DR) 功能旨在允许您从禁用或破坏整个池或站点的灾难性硬件故障中恢复虚拟机 (VM) 和 vApp。有关如何防止出现单服务器故障的信息，请参阅第 3.9 节“高可用性”。



### 注意

要使用此功能，必须使用用户名 root 进行登录，或具有池操作员或更高级别的角色。

### 8.1. 了解 XenServer DR

XenServer DR 的工作原理是：将恢复业务关键型 VM 和 vApp 所需的全部信息存储在存储库 (SR) 中，然后将存储库从主（生产）环境复制到备份环境。当主站点上受到保护的池出现故障时，可以从复制的存储恢复该池中的 VM 和 vApp 并在辅助 (DR) 站点上重新创建，从而最大限度地减少对应用程序或用户造成的停机时间。

如果出现灾难，可以使用 XenCenter 中的灾难恢复向导来查询该存储，并将选定的 VM 和 vApp 导入恢复池中。当 VM 在恢复池中运行之后，还会立即复制恢复池的元数据，以便在主池恢复后将 VM 设置进行的更改恢复到主池。如果 XenCenter 向导在两个或多个位置（例如，来自主站点的存储、来自灾难恢复站点的存储以及要导入数据的池）找到了有关同一个 VM 的信息，将确保使用该虚拟机的最新信息。

可通过 XenCenter 和 xe CLI 使用灾难恢复功能。有关这些命令的详细信息，请参阅第 4.6 节“灾难恢复 (DR) 命令”。



### 提示

还可以使用“灾难恢复”向导运行测试故障转移，以便对灾难恢复系统进行非中断性测试。测试故障转移与故障转移涉及的步骤基本相同，但是，VM 和 vApp 在恢复到 DR 站点之后不会启动，而是在测试完成时执行清理，以删除 DR 站点上重新创建的所有 VM、vApp 和存储。

XenServer VM 包含两个组件：

- 由 VM 使用的虚拟磁盘，存储在 VM 所在池中配置的存储库 (SR) 中。
- 描述 VM 环境的元数据。即在原始 VM 不可用或损坏时重新创建 VM 所需的全部信息。大多数元数据配置数据会在创建 VM 时写入，而且仅在对 VM 配置进行了更改时才会更新。对于池中的 VM，此元数据的副本存储在池中的每个服务器上。

在 DR 环境中，会根据池元数据（有关池中所有 VM 和 vApp 的配置信息）在辅助 (DR) 站点上重新创建 VM。每个 VM 的元数据都包含其名称、说明、通用唯一标识符 (UUID)、内存、虚拟 CPU、网络连接配置和存储配置。元数据还包含 VM 启动选项（启动顺序、延迟间隔和高可用性重新启动优先级），当在高可用性或 DR 环境中重新启动 VM 时将使用这些选项。例如，在灾难恢复期间恢复 VM 时，vApp 中的 VM 将按照 VM 元数据中指定的顺序，以指定的延迟间隔在 DR 池中重新启动。

### 8.2. DR 基础结构要求

要使用 XenServer DR，需要在主站点和辅助站点均设置适当的 DR 基础结构：

- 用于池元数据和 VM 所用虚拟磁盘的存储都必须从主（生产）环境复制到备份环境。存储复制（例如，使用镜像）最好通过存储解决方案进行处理，并且因设备而异。
- 当 VM 和 vApp 恢复到 DR 站点上的池中并且启动并运行后，还必须复制包含 DR 池元数据和虚拟磁盘的 SR，以便恢复后的 VM 和 vApp 在主站点重新联机后立即还原到主站点（故障恢复）。
- DR 站点的硬件基础结构不必与主站点的硬件基础结构一致，但是 XenServer 环境必须与主站点具有相同的版本和修补程序级别，而且应当在目标池中配置足够的资源，以便重新创建和启动所有故障转移的 VM。





### 警告

“灾难恢复”向导不控制任何存储阵列功能。

灾难恢复功能的用户必须确保元数据存储按照某种方式在两个站点之间复制。某些存储阵列包含可自动实现复制的“镜像”功能，如果使用这些功能，则在恢复站点上重新启动虚拟机之前，务必禁用镜像功能（“断开镜像”）。

### 8.3. 部署注意事项

请在启用灾难恢复之前检查以下步骤。

#### 8.3.1. 灾难发生之前执行的步骤

下一节将介绍在灾难发生之前执行的步骤。

- 配置 VM 和 vApp。
- 注意 VM 和 vApp 与 SR 的映射关系，以及 SR 与 LUN 的映射关系。特别要注意 name\_label 和 name\_description 字段的命名。如果 SR 的命名方式反映 VM 和 vApp 与 SR 以及 SR 与 LUN 的映射关系，将可以更为方便地从复制的存储恢复 VM 和 vApp。
- 安排 LUN 的复制。
- 允许将池元数据复制到这些 LUN 上的一个或多个 SR。

#### 8.3.2. 灾难发生之后执行的步骤

下一节将介绍在灾难发生之后执行的步骤。

- 中断任何现有的存储镜像，使恢复站点可以对共享存储进行读/写访问。
- 确保要用来恢复 VM 数据的 LUN 未连接到任何其他池，否则数据可能会被破坏。
- 如果要防止恢复站点出现灾难，则必须允许将池元数据复制到恢复站点的上一个或多个 SR。

#### 8.3.3. 恢复之后执行的步骤

下一节将介绍在成功恢复数据之后执行的步骤。

- 重新同步任何存储镜像。
- 在恢复站点上，彻底关闭要移回到主站点的 VM 或 vApp。
- 在主站点上，按照与上面所述故障转移过程相同的过程，将选定的 VM 或 vApp 故障恢复到主站点。
- 要防止主站点在将来发生灾难，必须重新允许将池元数据复制到所复制的 LUN 上的一个或多个 SR。

### 8.4. 在 XenCenter 中启用灾难恢复

本节介绍如何在 XenCenter 中启用灾难恢复。使用配置 DR 对话框可以指定用来存储池元数据（有关池中所有 VM 和 vApp 的配置信息）的存储库 (SR)。每当您对池中的 VM 或 vApp 配置进行更改时，池元数据都将进行更新。



#### 注意

只有当使用基于 HBA 的 LVM 或基于 iSCSI 的 LVM 时，才能启用灾难恢复。在该存储上需要少量空间来创建新 LUN，用以包含池恢复信息。

为此，您需要：

1. 在主站点上，选择要保护的池。从池菜单中，指向灾难恢复，然后单击配置。
2. 最多选择 8 个用来存储池元数据的 SR。在该存储上需要少量空间来创建新 LUN，用以包含池恢复信息。



#### 注意

将存储池中所有 VM 的信息，而不需要单独选择各个 VM 进行保护。

3. 单击确定。池现在将受保护。

## 8.5. 发生灾难时恢复 VM 和 vApp ( 故障转移 )

本节将介绍如何在辅助 ( 恢复 ) 站点上恢复 VM 和 vApp。

1. 在 XenCenter 中选择辅助池，而且在池菜单上，单击灾难恢复，然后打开灾难恢复向导。

灾难恢复向导显示三个恢复选项：故障转移、故障恢复和测试故障转移。要在辅助站点上恢复，请选择故障转移，然后单击下一步。



### 警告

如果您结合使用光纤通道共享存储和 LUN 镜像将数据复制到辅助站点，则在尝试恢复 VM 之前，必须断开镜像，以使辅助站点具有读/写访问权限。

2. 选择要恢复的 VM 和 vApp 的池元数据所在的存储库 (SR)。

默认情况下，此向导页面上的列表显示池中当前连接的所有 SR。要扫描查找更多的 SR，请选择查找存储库，然后选择要扫描的存储类型：

- 要扫描所有可用的硬件 HBA SR，请选择查找硬件 HBA SR。
- 要扫描软件 iSCSI SR，请选择查找软件 iSCSI SR，然后在出现的对话框中输入目标主机、IQN 和 LUN 详细信息。

在该向导中选择了所需的 SR 之后，单击下一步继续操作。

3. 选择要恢复的 VM 和 vApp，然后选择相应的恢复后的电源状态选项，以指定您希望 VM 和 vApp 是在恢复之后由向导立即自动启动，还是在故障转移完成后由您手动启动。

单击下一步前进到下一个向导页面，开始进行故障转移预检。

4. 此向导会在开始进行故障转移之前执行一些预检 ( 例如，确保选定 VM 和 vApp 所需的所有存储均可用 )。如果此时缺少任何存储，您可以单击此页面上的连接 SR 以查找并连接相关 SR。

解决预检页面上的任何问题，然后单击故障转移开始执行恢复过程。

5. 将出现一个进度页，显示每个 VM 和 vApp 的恢复是否成功。故障转移可能需要一段时间，具体取决于要恢复的 VM 和 vApp 的数量，因为需要将这些 VM 和 vApp 的元数据从复制的存储中导出。将在主池中重新创建所选 VM 和 vApp，包含虚拟磁盘的 SR 将连接到重新创建的 VM，然后这些 VM 将启动 ( 如果指定了启动选项 )。

6. 当故障转移完成时，单击下一步查看摘要报告。单击摘要报告页面上的完成关闭向导。

在主站点再次可用之后，如果您希望恢复 VM 在该站点上的运行，请再次执行“灾难恢复”向导，但这次应选择故障恢复选项。

## 8.6. 在灾难之后将 VM 和 vApp 还原到主站点 ( 故障恢复 )

本节介绍当主 ( 生产 ) 站点在灾难事件之后重新联机时，如何将 VM 和 vApp 从复制的存储还原到主站点上的池中。要将 VM 和 vApp 故障恢复到主站点，请使用“灾难恢复”向导。

1. 在 XenCenter 中选择辅助池，而且在池菜单上，单击灾难恢复，然后打开灾难恢复向导。

灾难恢复向导显示三个恢复选项：故障转移、故障恢复和测试故障转移。要将 VM 和 vApp 还原到主站点，请选择故障恢复，然后单击下一步。

**警告**

如果您结合使用光纤通道共享存储和 LUN 镜像将数据复制到主站点，则在尝试恢复 VM 之前，必须断开镜像，以使主站点具有读/写访问权限。

## 2. 选择要恢复的 VM 和 vApp 的池元数据所在的存储库 (SR)。

默认情况下，此向导页面上的列表显示池中当前连接的所有 SR。要扫描查找更多的 SR，请选择查找存储库，然后选择要扫描的存储类型：

- 要扫描所有可用的硬件 HBA SR，请选择查找硬件 HBA SR。
- 要扫描软件 iSCSI SR，请选择查找软件 iSCSI SR，然后在出现的对话框中输入目标主机、IQN 和 LUN 详细信息。

在该向导中选择了所需的 SR 之后，单击下一步继续操作。

## 3. 选择要还原的 VM 和 vApp，然后选择相应的恢复后的电源状态选项，以指定您希望这些 VM 和 vApp 是在恢复之后由向导立即自动启动，还是等到故障恢复完成后由您手动启动。

单击下一步前进到下一个向导页面，开始进行故障恢复预检。

## 4. 此向导会在开始进行故障恢复之前执行一些预检（例如，确保选定 VM 和 vApp 所需的所有存储均可用）。如果此时缺少任何存储，您可以单击此页面上的连接 SR 以查找并连接相关 SR。

解决预检页面上的任何问题，然后单击故障恢复开始执行恢复过程。

## 5. 将出现一个进度页，显示每个 VM 和 vApp 的恢复是否成功。故障恢复可能需要一段时间，具体取决于要还原的 VM 和 vApp 的数量，因为需要将这些 VM 和 vApp 的元数据从复制的存储中导出。将在主池中重新创建所选 VM 和 vApp，包含虚拟磁盘的 SR 将连接到重新创建的 VM，然后这些 VM 将启动（如果指定了启动选项）。

## 6. 当故障恢复完成时，单击下一步查看摘要报告。单击摘要报告页面上的完成关闭向导。

## 8.7. 测试故障转移

故障转移测试是灾难恢复规划中不可或缺的部分。您可以使用“灾难恢复”向导对灾难恢复系统执行非中断性测试。测试故障转移期间的所有步骤与故障转移基本相同，只是 VM 和 vApp 在恢复到 DR 站点之后不会启动，而是被置于暂停状态。在测试故障转移过程结束时，在 DR 站点上重新创建的所有 VM、vApp 和存储都将自动恢复。在完成初始 DR 配置后，以及在启用 DR 的池中进行重要的配置更改后，我们建议您通过执行测试故障转移来验证故障转移是否仍能正常进行。

对 VM 和 vApp 执行以辅助站点为目标的测试故障转移：

1. 在 XenCenter 中选择辅助池，在池菜单上，单击灾难恢复以打开灾难恢复向导。
2. 选择测试故障转移，然后单击下一步。

**注意**

如果您结合使用光纤通道共享存储和 LUN 镜像将数据复制到辅助站点，则在尝试恢复数据之前，必须断开镜像，以使辅助站点具有读/写访问权限。

## 3. 选择要恢复的 VM 和 vApp 的池元数据所在的存储库 (SR)。

默认情况下，此向导页面上的列表显示池中当前连接的所有 SR。要扫描查找更多的 SR，请选择查找存储库，然后选择要扫描的存储类型：

- 要扫描所有可用的硬件 HBA SR，请选择查找硬件 HBA SR。

- 要扫描软件 iSCSI SR，请选择查找软件 iSCSI SR，然后在出现的对话框中输入目标主机、IQN 和 LUN 详细信息。

在该向导中选择了所需的 SR 之后，单击下一步继续操作。

4. 选择要恢复的 VM 和 vApp，然后单击“下一步”前进到下一个向导页，开始进行故障转移预检。
5. 此向导会在开始进行测试故障转移过程之前执行一些预检（例如，确保选定 VM 和 vApp 所需的所有存储均可用）。

- 检查存储是否可用。如果缺少任何存储，可以单击此页面上的连接 SR 以查找并连接相关 SR。
- 检查在目标 DR 池上是否启用了高可用性。为了避免同一个 VM 在主池和 DR 池中同时运行，必须对辅助池禁用高可用性，以确保 VM 和 vApp 在恢复之后不会由高可用性功能自动启动。要对辅助池禁用高可用性，只需单击此页面上的禁用高可用性。（如果此时高可用性处于禁用状态，则它会在测试故障转移过程结束时自动重新启用。）

解决预检页面上的任何问题，然后单击故障转移开始执行测试故障转移。

6. 将出现一个进度页，显示每个 VM 和 vApp 的恢复是否成功。故障转移可能需要一段时间，具体取决于要恢复的 VM 和 vApp 的数量，因为需要将这些 VM 和 vApp 的元数据从复制的存储恢复。将在 DR 池中重新创建所选 VM 和 vApp，包含虚拟磁盘的 SR 将连接到重新创建的 VM。

恢复后的 VM 将被置于暂停状态：在测试故障转移期间，它们将不会在辅助站点上启动。

7. 如果测试故障转移成功执行，请单击向导中的下一步，使向导在 DR 站点上进行清理：
  - 删除在测试故障转移期间恢复的 VM 和 vApp。
  - 分离在测试故障转移期间恢复的存储。
  - 如果在预检阶段为了允许进行测试故障转移而对 DR 池禁用了高可用性功能，则高可用性功能会自动重新启用。

向导将显示清理过程的进度。

8. 单击完成关闭向导。

## 8.8. vApp

vApp 是一个或多个相关虚拟机 (VM) 的逻辑组，在发生灾难时，这些虚拟机可以作为单个实体来启动。当 vApp 启动后，其中包含的 VM 将按照用户预定义的顺序启动，使相互依赖的 VM 自动排成序列。这意味着在整个服务需要重新启动时（例如在软件更新后），管理员不再需要手动设置相关 VM 的启动顺序。vApp 中的 VM 不必位于同一个主机上，而是按照正常的规则在池内分布。在灾难恢复的情况下，vApp 功能尤其有用。此时，管理员可以选择将位于同一个存储库中或者与同一个服务级别协议 (SLA) 相关的所有 VM 组合到一起。

创建 vApp:

要将多个 VM 组合到一个 vApp 中，请按以下步骤操作：

1. 选择池，然后在池菜单上，单击管理 vApp。此时将显示管理 vApp 窗口。
2. 输入该 vApp 的名称和可选说明，然后单击下一步。

可以根据自己的喜好选择任何名称，但通常最好使用描述性的名称。尽管建议您避免为多个 vApp 使用相同的名称，但并不是必须要这样做，XenCenter 不会针对 vApp 名称实施唯一性约束。对于包含空格的名称，不必使用引号。

3. 选择要包含在新 vApp 中的 VM，然后单击下一步。

可以使用搜索框仅列出名称中包含指定字符串的 VM。

- 为该 vApp 中的 VM 指定启动顺序，然后单击下一步。

值	说明
启动顺序	指定各个 VM 在 vApp 中的启动顺序，使某些 VM 先于其他 VM 重新启动。启动顺序值为 0 (零) 的 VM 将首先启动，启动顺序值为 1 的 VM 接着启动，然后是启动顺序值为 2 的 VM，以此类推。
尝试在以下时间后启动下一个 VM	这是一段延迟时间间隔，用于指定从启动 VM 到尝试启动启动序列中下一组 VM (即，启动顺序更靠后的 VM) 之间的等待时间。

- 在该向导的最后一页上，可以检查 vApp 配置。单击上一步返回并修改任何设置，或者单击完成创建新的 vApp 并关闭向导。



#### 注意

一个 vApp 可以跨单个池中的多个服务器，但是不能跨多个池。

### 8.8.1. 使用 XenCenter 的“管理 vApp”对话框

使用 XenCenter 的管理 vApp 对话框，可以在选定池内创建、删除、修改、启动、关闭、导入和导出 vApp。当您在列表中选择某个 vApp 时，其中包含的 VM 会列在右侧的详细信息窗格中。有关详细信息，请参阅 XenCenter 联机帮助。按 F1 键或单击帮助可显示帮助。

## 8.9. 备份和还原 XenServer 主机和 VM

Citrix 建议，如果可能最好不要更改 XenServer 主机的安装状态。也就是说，不要在 XenServer 主机上安装任何其他软件包或启动其他服务，将其视为设备。最好的还原方式是从安装介质重新安装 XenServer 主机软件。如果您有多个 XenServer 主机，则最好的方法是配置 TFTP 服务器和相应的应答文件来实现此目的 (请参阅《XenServer 安装指南》)。

对于 VM，最佳方法是将其视为标准物理服务器，在上面安装备份代理。对于 Windows VM，截止到此版本，我们已测试了 CA BrightStor ARCserve Backup 以及 Symantec NetBackup 和 Backup Exec。

有关已测试的备份工具、最佳做法和常规备份的详细信息，请参阅 [Citrix 知识库](#)。

Citrix 建议您尽量经常执行下列备份过程以从可能的服务器和/或软件故障中恢复。

#### 备份池元数据:

- 运行以下命令：

```
xe pool-dump-database file-name=<backup>
```

- 运行以下命令：

```
xe pool-restore-database file-name=<backup> dry-run=true
```

此命令检查目标计算机具有一定数量的适当命名的 NIC，这是成功进行备份的必备条件。

#### 备份主机配置和软件:

- 运行以下命令：

```
xe host-backup host=<host> file-name=<hostbackup>
```



**注意**

- 不要在控制域中创建备份。
- 此过程可能创建大型备份文件。
- 要完成还原，必须重新引导到原始安装 CD。
- 只能将此数据还原到原始计算机。

**备份 VM:**

1. 确保要备份的 VM 处于脱机状态。
2. 运行以下命令：

```
xe vm-export vm=<vm_uuid> filename=<backup>
```

**注意**

此备份也将备份 VM 的所有数据。导入 VM 时，可以指定要为已备份的数据使用的存储机制。

**警告**

由于此过程将备份 VM 的所有数据，因此可能需要花费一些时间。

**仅备份 VM 元数据:**

- 运行以下命令：

```
xe vm-export vm=<vm_uuid> filename=<backup> metadata=true
```

**8.9.1. 备份虚拟机元数据**

XenServer 主机使用各个主机上的数据库存储有关 VM 和相关资源（如存储和网络）的元数据。当与存储库结合使用时，此数据库形成池中所有可用 VM 的完整视图。因此，重要的是，要了解如何备份此数据库，以便从物理硬件故障和其他灾难情况中恢复。

本部分首先介绍单主机环境元数据的备份方法，然后介绍更为复杂的池环境。

**8.9.1.1. 备份单台主机环境**

使用 CLI 备份池数据库。要获得一致的池元数据备份文件，请在 XenServer 主机上运行 `pool-dump-database` 并将结果文件归档。此备份文件将包含有关池的敏感身份验证信息，因此请确保安全存储该文件。

要还原池数据库，请对前面的转储文件使用 `xe pool-restore-database` 命令。如果您的 XenServer 主机完全死机，则必须首先进行全新安装，然后对重新安装的 XenServer 主机运行 `pool-restore-database` 命令。

还原池数据库后，某些 VM 可能仍注册为 `Suspended` 状态，但是如果包含其挂起的内存状态（在 `suspend-VDI-uuid` 字段中定义）的存储库是本地 SR，则因为该主机已重新安装，所以它将不再可用。要将这些 VM 重置为 `Halted` 状态以便可以再次启动，请使用 `xe vm-shutdown vm=vm_name -force` 命令或使用 `xe vm-reset-powerstate vm=<vm_name> -force` 命令。

**警告**

使用此方法还原的 XenServer 主机将保留其 UUID。如果在原始 XenServer 主机仍运行时还原到其他物理机，则将存在重复的 UUID。此操作主要的显著影响是 XenCenter 将拒绝连接到第二



个 XenServer 主机。不建议使用池数据库备份机制克隆物理主机；应使用自动化安装支持进行克隆（请参阅 XenServer 安装指南）。

### 8.9.1.2. 备份池安装

在池方案中，主服务器主机提供了一个同步镜像到池中所有成员主机的授权数据库。这为池提供了内置的冗余度；任何成员主机都可替换主服务器主机，因为每个成员主机都带有正确版本的池数据库。有关如何将成员主机转换为主服务器主机的详细信息，请参阅《XenServer 管理员指南》。

这种保护级别可能远远不够；例如，您的包含 VM 数据的共享存储在多个站点中进行了备份，但包含池元数据的本地服务器存储没有备份。要在仅给定一组共享存储的情况下完全重新创建池，您必须首先对主服务器主机上的 pool-dump-database 文件进行备份，然后将此文件归档。

随后在一组全新的主机上还原此备份：

1. 从安装介质中安装一组全新的 XenServer 主机，或者从 TFTP 服务器进行网络引导（如果适用）。
2. 对指定为新主服务器的主机使用 `xe pool-restore-database`。
3. 在新主服务器主机上运行 `xe host-forget` 命令以删除旧成员计算机。
4. 在成员主机上使用 `xe pool-join` 命令以将这些主机连接到新池。

### 8.9.2. 备份 XenServer 主机

本部分介绍 XenServer 主机控制域的备份和还原步骤。这些步骤不会备份用来容纳 VM 的存储库，而仅备份运行 Xen 和 XenServer 代理的特权控制域。



#### 注意

因为特权控制域最好保留安装时的状态，不使用其他软件包对其进行自定义，因此 Citrix 建议您设置网络引导环境，以便仅从 XenServer 介质执行全新安装，以此作为恢复策略。在许多情况下，您根本不需要备份此控制域，只需要保存池元数据即可（请参阅第 8.9.1 节“[备份虚拟机元数据](#)”）。此备份方法应始终被视为备份池元数据的补充方法。

可以采取的另一种方法是使用 `xe` 命令 `host-backup` 和 `host-restore`。`xe host-backup` 命令将活动分区归档到您指定的文件，`xe host-restore` 命令则将由 `xe host-backup` 创建的归档文件提取到主机上当前不活动的磁盘分区中。然后通过引导安装 CD 并选择还原相应备份来激活此分区。

完成上述步骤并重新引导主机后，您必须确保 VM 元数据已还原为一致的状态。这可以通过对 `/var/backup/pool-database- $\{DATE\}$`  运行 `xe pool-restore-database` 来实现。此文件由 `xe host-backup` 使用 `xe pool-dump-database` 命令在归档正在运行的文件系统之前创建，以便对 VM 元数据的一致状态生成快照。

备份 XenServer 主机：

- 在具有足够磁盘空间的远程主机上运行以下命令：

```
xe host-backup file-name=<filename> -h <hostname> -u root -pw <password>
```

这将在 `file-name` 参数指定的位置创建控制域文件系统的压缩映像。

还原正在运行的 XenServer 主机：

1. 如果要从特定备份还原 XenServer 主机，请在该 XenServer 主机已启动并可以连接时运行以下命令：

```
xe host-restore file-name=<filename> -h <hostname> -u root -pw <password>;
```

这个命令会将压缩映像还原到运行此命令的 XenServer 主机（而不是 filename 驻留的主机）的硬盘中。在该上下文中，“还原”有些用词不当，因为该词通常表示已完全恢复备份的状态。此处的还原命令仅对压缩的备份文件进行解压缩并将其还原到正常形式，但它写入其他分区 (/dev/sda2) 并且不会覆盖当前版本的文件系统。

2. 要使用已还原版本的根文件系统，请使用 XenServer 安装 CD 重新引导 XenServer 主机，然后选择 Restore from backup（从备份还原）选项。

从备份还原完成后，重新引导 XenServer 主机，该主机将从还原映像启动。

最后，使用以下命令还原 VM 元数据

```
xe pool-restore-database file-name=/var/backup/pool-database-*
```



#### 注意

如此处所述，从备份还原不会破坏备份分区。

重新启动已崩溃的 XenServer 主机:

1. 如果您的 XenServer 主机已崩溃且不能再连接，您需要使用 XenServer 安装 CD 进行升级安装。升级安装完成后，重新引导此计算机并确保可以使用 XenCenter 或远程 CLI 连接到您的主机。
2. 然后继续执行上述第 8.9.2 节“备份 XenServer 主机”。

### 8.9.3. 备份 VM

最好使用 VM 上独立运行的标准备份工具备份 VM。对于 Windows VM，我们已测试了 CA BrightStor ARCserve Backup。

## 8.10. VM 快照



#### 重要

虚拟机保护和恢复 (VMPR) 及其功能已从 XenServer 7.0 及更高版本中删除。依赖于 VMPR 的应用程序、代码或使用在 XenServer 7.0 及更高版本中不工作。VM 快照功能及其他依赖于 VM 快照的功能 (VMPR 除外) 不受影响。有关详细信息，请参阅 [CTX137335](#)。

XenServer 提供了一种简便的快照机制。借助该机制，可以在给定的时间生成 VM 存储和元数据的快照。生成快照时，可在需要时临时停止 IO 以确保捕获自身一致的磁盘映像。

快照操作会生成类似于模板的快照 VM。VM 快照包含所有存储信息和 VM 配置（包括已连接的 VIF），可以导出并还原这些信息和配置以进行备份。虽然所有存储类型都支持快照，但对基于 LVM 的存储类型而言，如果存储库是使用以前版本的 XenServer 创建的，则必须对其进行升级，而且卷必须采用默认格式（无法对 type=raw 卷生成快照）。

快照操作过程分为 2 个步骤：

- 将元数据捕获为模板。
- 创建磁盘的 VDI 快照。

支持三种类型的 VM 快照：常规快照、静态快照以及包含内存数据的快照

### 8.10.1. 常规快照

常规快照处于持续崩溃状态，可以在所有 VM 类型（包括 Linux VM）上执行。

### 8.10.2. 静态快照

静态快照利用 Windows 卷快照服务 (VSS)，生成和应用程序一致的实时快照。VSS 框架帮助可识别 VSS 的应用程序（例如 Microsoft Exchange 或 Microsoft SQL Server）在生成快照前将数据刷新到磁盘并为快照做好准备。

因此，还原静态快照比较安全，但当生成静态快照时，这些快照可能会对系统性能产生更大的影响。如果负载不足，它们还可能会失败，因此生成该快照可能需要进行多次尝试。

XenServer 在下列平台上支持静态快照：

- Windows Server 2012 R2 Server Core
- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2
- Windows Server 2008 ( 32/64 位 )
- Windows Server 2003 ( 32/64 位 )

不支持 Windows 8.1、Windows 8、Windows 7、Windows 2000 和 Windows Vista。有关静态快照的详细信息，请参阅第 8.10.9.3 节“静态快照高级说明”。

### 8.10.3. 包含内存数据的快照

除保存虚拟机内存（存储）和元数据外，包含内存数据的快照还保存虚拟机状态（RAM）。当您正在升级或修补软件，或想要测试新应用程序，但同时又希望能选择返回到虚拟机的当前、更改前状态（RAM）时，这种快照类型会非常有用。还原到包含内存数据的快照无需重新引导 VM。

可以通过 XenAPI、xe CLI 或使用 XenCenter 来生成包含内存数据的屏幕快照。

### 8.10.4. 创建 VM 快照

拍摄快照前，请参阅《XenServer 虚拟机用户指南》中“准备使用 VSS 克隆 Windows VM”部分，有关任何特定于特殊操作系统的配置以及需要考虑的注意事项的信息，请参阅“准备克隆 Linux VM”部分。

首先确保 VM 正在运行或已挂起，以便可以捕获内存状态。选择要执行操作的 VM 的最简单方法是应用参数 `vm=<name>` 或 `vm=<vm uuid>`。

运行 `vm-snapshot` 和 `vm-snapshot-with-quiesce` 命令以生成 VM 的快照。

```
xe vm-snapshot vm=<vm uuid> new-name-label=<vm_snapshot_name>
xe vm-snapshot-with-quiesce vm=<vm uuid> new-name-label=<vm_snapshot_name>
```

### 8.10.5. 创建包含内存数据的快照

运行 `vm-checkpoint` 命令，为包含内存数据的快照指定一个描述性名称，以便以后能识别该快照：

```
xe vm-checkpoint vm=<vm uuid> new-name-label=<name of the checkpoint>
```

XenServer 创建完包含内存数据的快照之后，系统会显示其 uuid。

例如：

```
xe vm-checkpoint vm=2d1d9a08-e479-2f0a-69e7-24a0e062dd35 \
  new-name-label=example_checkpoint_1
b3c0f369-59a1-dd16-ecd4-a1211df29886
```

包含内存数据的快照需要每个磁盘至少有 4 MB 的磁盘空间，以及加上 RAM 的大小，再加上约 20% 的开销。因此，256 MB RAM 的检查点需要约 300 MB 的存储空间。



#### 注意

在创建检查点的过程中，VM 会短时暂停，在暂停期间不能使用。

### 8.10.6. 列出 XenServer 池的所有快照

运行 snapshot-list 命令：

```
xe snapshot-list
```

这样会列出 XenServer 池中的所有快照。

### 8.10.7. 列出特殊 VM 的快照

您需要知道该特殊 VM 的 uuid，为此，请运行 vm-list 命令。

```
xe vm-list
```

结果会显示所有虚拟机的列表及其 UUID。例如：

```
xe vm-list
uuid ( RO): 116dd310-a0ef-a830-37c8-df41521ff72d
name-label ( RW): Windows Server 2003 (1)
power-state ( RO): halted

uuid ( RO): 96fde888-2a18-c042-491a-014e22b07839
name-label ( RW): Windows XP SP3 (1)
power-state ( RO): running

uuid ( RO): dff45c56-426a-4450-a094-d3bba0a2ba3f
name-label ( RW): Control domain on host
power-state ( RO): running
```

也可以通过过滤字段值上的完整 VM 列表，来指定 VM。

例如，如果指定 power-state=halted，将会选择电源状态字段为“halted”的所有虚拟机。如果有多个匹配的 VM，则必须指定选项 --multiple 才能执行操作。可匹配字段的完整列表可以通过命令 xe vm-list params=all 获得。

查找所需的 VM，然后输入以下内容：

```
xe snapshot-list snapshot-of=<vm uuid>
```

例如：

```
xe snapshot-list snapshot-of=2d1d9a08-e479-2f0a-69e7-24a0e062dd35
```

这会列出该 VM 上的当前快照：

```
uuid ( RO): d7eeb03-39bc-80f8-8d73-2ca1bab7dcff
name-label ( RW): Regular
name-description ( RW):
snapshot_of ( RO): 2d1d9a08-e479-2f0a-69e7-24a0e062dd35
snapshot_time ( RO): 20090914T15:37:00Z

uuid ( RO): 1760561d-a5d1-5d5e-2be5-d0dd99a3b1ef
name-label ( RW): Snapshot with memory
name-description ( RW):
snapshot_of ( RO): 2d1d9a08-e479-2f0a-69e7-24a0e062dd35
snapshot_time ( RO): 20090914T15:39:45Z
```

### 8.10.8. 将 VM 还原为上一状态

确保知道要还原的快照的 uuid，然后运行 snapshot-revert 命令：

为此，您需要：

1. 运行 snapshot-list 命令以查找要还原的快照或数据点的 UUID：

```
xe snapshot-list
```

2. 记下快照的 uuid，然后运行以下命令进行还原：

```
xe snapshot-revert snapshot-uuid=<snapshot uuid>
```

例如：

```
xe snapshot-revert snapshot-uuid=b3c0f369-59a1-dd16-eed4-a1211df29886
```

还原为检查点后，VM 将被挂起。



#### 注意

如果没有足够的磁盘空间可用于对快照进行厚置备，在当前的磁盘状态被释放之前，不可能还原到快照。如果发生这种情况，请重新尝试此操作。



#### 注意

不可能还原到任何快照。还原操作期间不会删除现有快照和检查点。

#### 8.10.8.1. 删除快照

确保知道要删除的检查点或快照的 UUID，然后运行以下命令：

1. 运行 snapshot-list 命令以查找要还原的快照或数据点的 UUID：

```
xe snapshot-list
```

2. 记下快照的 UUID，然后运行 snapshot-uninstall 命令将其删除：

```
xe snapshot-uninstall snapshot-uuid=<snapshot-uuid>
```

3. 此命令会警告您将要删除的 VM 和 VDI。键入 yes 确认。

例如：

```
xe snapshot-uninstall snapshot-uuid=1760561d-a5d1-5d5e-2be5-d0dd99a3b1ef
The following items are about to be destroyed
VM : 1760561d-a5d1-5d5e-2be5-d0dd99a3b1ef (Snapshot with memory)
VDI: 11a4aa81-3c6b-4f7d-805a-b6ea02947582 (0)
VDI: 43c33fe7-a768-4612-bf8c-c385e2c657ed (1)
VDI: 4c33c84a-a874-42db-85b5-5e29174fa9b2 (Suspend image)
Type 'yes' to continue
yes
All objects destroyed
```

如果您只希望删除检查点或快照的元数据，请运行以下命令：

```
xe snapshot-destroy snapshot-uuid=<snapshot-uuid>
```

例如：

```
xe snapshot-destroy snapshot-uuid=d7eefb03-39bc-80f8-8d73-2ca1bab7dcff
```

## 8.10.9. 快照模板

### 8.10.9.1. 通过快照创建模板

可以通过快照创建 VM 模板，但会删除模板的内存状态。

为此，您需要：

1. 使用命令 snapshot-copy 并为模板指定 new-name-label：

```
xe snapshot-copy new-name-label=<vm-template-name> \
  snapshot-uuid=<uuid of the snapshot>
```

例如：

```
xe snapshot-copy new-name-label=example_template_1
  snapshot-uuid=b3c0f369-59a1-dd16-ecd4-a1211df29886
```



#### 注意

这会在 SAME 池中创建一个模板对象。此模板仅存在于当前池的 XenServer 数据库中。

2. 要验证是否已创建模板，请运行命令 template-list：

```
xe template-list
```

这会列出 XenServer 主机上的所有模板。

### 8.10.9.2. 将快照导出到模板

导出 VM 快照时，VM 的完整副本（包括磁盘映像）会作为单独的文件存储到本地计算机上，文件扩展名为 .xva。

为此，您需要：

1. 使用 `snapshot-export-to-template` 命令创建新的模板文件：

```
xe snapshot-export-to-template snapshot-uuid=<snapshot-uuid> \
  filename=<template- filename>
```

例如：

```
xe snapshot-export-to-template snapshot-uuid=b3c0f369-59a1-dd16-ecd4-a1211df29886 \
  filename=example_template_export
```

可以通过多种不同的方式使用 VM 导出/导入功能：

- 作为一种便捷的 VM 备份工具。在发生灾难性事件时，可以使用导出的 VM 文件恢复整个 VM。
- 作为一种快速复制 VM 的方式，例如，多次使用的特殊用途服务器配置。您只需以希望的方式配置 VM，将其导出，然后再将其导入，即可创建原始 VM 的副本。
- 作为一种将 VM 移动到其他服务器的简单方法。

有关使用模板的详细信息，请参阅《XenServer 虚拟机用户指南》中的“创建 VM”一章，以及“XenCenter 帮助”中的“管理 VM”一节。

### 8.10.9.3. 静态快照高级说明



#### 注意

不要忘记在 Windows 来宾系统中安装 Xen VSS 提供程序以便支持 VSS。这可以使用 XenServer Tools 随附的 `install-XenProvider.cmd` 脚本来完成。有关 vApp 的详细信息，请参阅《XenServer 虚拟机用户指南》。

一般来说，使用 VSS 接口，VM 只能访问自身的 VDI 快照（而非 VDI 克隆）。有一个可由 XenServer 管理员设置的标志，其中如果将 `snapmanager=true` 属性添加到 VM 的 `other-config`，便可使该 VM 能够导入其他 VM 的 VDI 快照。



#### 警告

这将暴露一个安全漏洞，使用时需谨慎。通过此功能，管理员可以使用 VSS 层生成的来宾系统可传送快照 ID，将 VSS 快照连接到其他 VM 以进行备份。

**VSS 静止时段：**系统将 Microsoft VSS 静止时段设置为 10 秒钟的非可配置值，因此快照很有可能无法按时完成。例如，如果 XAPI 后台程序在队列中又添加了其他块任务（例如 SR 扫描），则 VSS 快照可能会超时并失败。如果发生这种情况，应该重试该操作。



#### 注意

连接到 VM 的 VBD 越多，达到此超时的可能性越大。Citrix 建议最多将 2 个 VBD 连接到 VM，以避免超时。但是，有一个解决方法可以解决此问题。如果 VM 的所有 VDI 托管在不同的 SR 上，使用超过 2 个 VBD 成功地生成基于 VSS 的 VM 快照的可能性就会成倍增加。

**VSS 快照将所有磁盘连接到 VM：**为存储在生成 VSS 快照时可用的所有数据，XAPI 管理器将为所有磁盘以及可以使用 XenServer 存储管理器 API 生成快照的 VM 元数据生成快照。如果 VSS 层请求只有一小部分磁盘的快照，那么生成的 VM 快照将不完整。

**vm-snapshot-with-quiesce 生成可引导的快照 VM 映像：**要实现这一目标，XenServer VSS 硬件提供程序将制作可写入的快照卷，其中包括引导卷的快照。



卷的 VSS 快照托管在 Windows 来宾系统的动态磁盘中：vm-snapshot-with-quietse CLI 以及 XenServer VSS 硬件提供程序不支持 Windows VM 上的动态磁盘中托管的卷的快照。



### 注意

不要忘记在 Windows 来宾系统中安装 Xen VSS 提供程序以便支持 VSS。这可以使用 XenServer Tools 随附的 install-XenProvider.cmd 脚本来完成。有关详细信息，请参阅《XenServer 虚拟机用户指南》。

## 8.11. 处理计算机故障

本部分详细介绍如何从各种故障方案中恢复。所有故障恢复方案都需要使用第 8.9 节“备份和还原 XenServer 主机和 VM”中列出的一个或多个备份类型。

### 8.11.1. 成员故障

如果缺少高可用性功能，主节点将通过接收常规检测信号消息检测成员的故障。如果在 600 秒内还没有接收到检测信号，主服务器就会认为该成员已停止活动。可通过下列两种方式从此问题中恢复：

- 修复停止活动的主机（例如，以物理方式重新引导主机）。还原与该成员的连接后，主服务器会再次将该成员标记为活动成员。
- 关闭主机并使用 `xe host-forget` CLI 命令指示主服务器忽略该成员节点。忘记该成员后，会将该成员上运行的所有 VM 标记为脱机，然后在其他 XenServer 主机上重新启动它们。请注意，确保 XenServer 主机确实脱机非常重要，否则可能会损坏 VM 数据。请注意，不要使用 `xe host-forget` 将您的池拆分为单一主机的多个池，因为这可能会导致这些池全都映射相同的共享存储并损坏 VM 数据。



### 警告

- 如果您准备将已忽略的主机再次用作 XenServer 主机，请完全重新安装 XenServer 软件。
- 如果在该池上启用高可用性功能，则请勿使用 `xe host-forget` 命令。首先请禁用高可用性功能，然后忽略主机，接着重新启用高可用性功能。

成员 XenServer 主机出现故障后，某些 VM 可能仍然以运行状态进行注册。如果您确定 XenServer 成员主机确实已关闭，请使用 `xe vm-reset-powerstate` CLI 命令将 VM 的电源状态设置为 `halted`。有关详细信息，请参阅第 4.26.28 节“`vm-reset-powerstate`”。



### 警告

如果此命令使用不当，则会导致数据损坏。请仅在必要时使用此命令。

在另一个 XenServer 主机上启动 VM 之前，还需要释放 VM 存储的锁定。SR 中的每个磁盘一次只能由一个主机使用，因此，在一个主机出现故障后，必须使磁盘可供其他 XenServer 主机访问。为此，请在池主服务器上针对任何受影响 VM 的磁盘所在的每个 SR 运行以下脚本：

```
/opt/xensource/sm/resetvdis.py <host_UUID> <SR_UUID> [master]
```

崩溃时，如果出现故障的主机是 SR 主服务器 — 池主服务器或使用本地存储的 XenServer 主机，您只需提供第三个字符串（“master”）。



### 警告

在执行此命令之前，请务必确保主机已关闭。如果此命令使用不当，则会导致数据损坏。

如果在运行上述脚本之前，尝试在另一个 XenServer 主机上启动 VM，则将收到下面的错误消息：VDI <UUID> already attached RW。

### 8.11.2. 主服务器故障

如果需要，资源池的每个成员都将包含担任主服务器角色所需的所有信息。如果主节点发生故障，将依次发生下列事件：

1. 如果禁用高可用性功能，则会自动选出另一个主服务器。
2. 如果未启用高可用性功能，则每个成员都将等待主服务器恢复。

此时如果主服务器恢复，它会与其成员重新建立通信，操作恢复正常。

如果主服务器已停用，您应该选择一个成员并在其上运行 `xe pool-emergency-transition-to-master` 命令。当该成员成为主服务器后，运行命令 `xe pool-recover-slaves`，这些成员此时将指向新的主服务器。

如果修复或替换用作原始主服务器的服务器，您只需启动服务器，安装 XenServer 主机软件，然后将它添加到池中。由于池中的 XenServer 主机强制为同类主机，因此实际上不需要将替换的服务器设为主服务器。

将成员 XenServer 主机转换为主服务器后，您还应检查默认池存储库是否设置了适当的值。通过使用 `xe pool-param-list` 命令并验证 `default-SR` 参数是否指向有效的存储库，可实现此操作。

### 8.11.3. 池故障

如果您的整个资源池不幸发生故障，就需要从头开始重新创建池数据库。请务必使用 `xe pool-dump-database` CLI 命令定期备份池元数据（请参阅第 4.15.3 节“[pool-dump-database](#)”）。

还原彻底出故障的池：

1. 安装一组全新的主机。请不要在此阶段将这些主机组成池。
2. 对于任命为主服务器的主机，使用 `xe pool-restore-database`（请参阅第 4.15.11 节“[pool-restore-database](#)”）命令从备份中还原池数据库。
3. 使用 XenCenter 连接到主服务器主机并确保所有共享存储和 VM 都再次可用。
4. 在已全新安装的其余成员主机上执行池加入操作，并在相应主机上启动 VM。

### 8.11.4. 处理因配置错误导致的故障

如果物理主机可以运行但软件或主机配置被损坏，可按以下方式处理：

还原主机软件和配置：

1. 运行以下命令：

```
xe host-restore host=<host> file-name=<hostbackup>
```

2. 重新引导到主机安装 CD 并选择 Restore from backup（从备份还原）。

### 8.11.5. 物理机故障

如果物理主机发生故障，请使用下面列出的相应过程进行恢复。



#### 警告

所有曾在发生了故障的先前成员（或先前主机）上运行的 VM，在数据库中仍将被标记为 Running。这是出于安全考虑，因为在两个不同的主机上同时启动 VM 会导致磁盘严重损坏。如果确定计算机（和 VM）处于脱机状态，您可将 VM 电源状态重置为 Halted：

```
xe vm-reset-powerstate vm=<vm_uuid> --force
```

然后可使用 XenCenter 或 CLI 重新启动 VM。

用仍处于运行状态的成员替换发生故障的主服务器：

1. 运行以下命令：

```
xe pool-emergency-transition-to-master
xe pool-recover-slaves
```

2. 如果命令成功，请重新启动 VM。

还原所有主机都发生故障的池：

1. 运行以下命令：

```
xe pool-restore-database file-name=<backup>
```



#### 警告

只有当目标计算机具有合适数量的正确命名的 NIC 时，此命令才能成功。

2. 如果目标计算机具有与原始计算机不同的存储视图（例如，具有不同 IP 地址的块镜像），请使用 pbd-destroy 命令和 pbd-create 命令修改存储配置以重新创建存储配置。有关这些命令的文档，请参阅第 4.13 节“PBD 命令”。
3. 如果已创建新的存储配置，请使用 XenCenter 中的 pbd-plug 或存储 > 修复存储库菜单项应用新配置。
4. 重新启动所有 VM。

当 VM 存储不可用时还原 VM：

1. 运行以下命令：

```
xe vm-import filename=<backup> metadata=true
```

2. 如果元数据导入失败，请运行以下命令：

```
xe vm-import filename=<backup> metadata=true --force
```

此命令将“尽最大努力”尝试还原 VM 元数据。

3. 重新启动所有 VM。

## 第 9 章 监视和管理 XenServer

XenServer 会提供详细的性能指标监视数据，包括 CPU、内存、磁盘、网络、C-state/P-state 信息以及存储。适用情况下，会针对每个主机和每个 VM 提供这些指标。这些指标可以直接获取，也可以在 XenCenter 或其他第三方应用程序中以图形方式进行访问和查看。

XenServer 也提供系统和性能警报。警报是指为响应选定系统事件而发生的通知，或在 CPU、内存使用率、网络吞吐量、存储吞吐量或 VM 磁盘活动超过托管主机、VM 或存储库上指定的阈值时发生的通知。您可以使用 xe CLI 或 XenCenter 配置这些设置，根据任何可用的主机或 VM 性能指标创建通知，请参阅第 9.2 节“警报”。

### 9.1. 监视 XenServer 性能

客户可以使用通过轮询数据库 (RRD) 公布的指标，监视其 XenServer 主机和虚拟机 (VM) 的性能。这些指标可以在 HTTP 上或通过 RRD2CSV 工具查询；此外，XenCenter 使用此数据生成系统性能图形。请参阅第 9.1.5 节“使用 RRDs”和第 9.1.3 节“分析和显示 XenCenter 中的指标”。

下表列出了所有可用的主机和 VM 指标。



#### 注意

某个时间段内的延迟被定义为此时间段中的平均操作延迟。

某些指标的可用性和实用性取决于 SR 和 CPU。

#### 9.1.1. 可用主机指标

指标名称	说明	条件	XenCenter 名称
avgqu_sz_<sr-uuid-short>	平均 I/O 队列大小 ( 请求 )。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 队列大小
cpu<cpu>-C<cstate>	CPU <cpu> 处于 C-state <cstate> 下的时间 ( 以毫秒为单位 )。	C-state 存在于 CPU 上	CPU <cpu> C-state <cstate>
cpu<cpu>-P<pstate>	CPU <cpu> 处于 P-state <pstate> 下的时间 ( 以毫秒为单位 )。	P-state 存在于 CPU 上	CPU <cpu> P-state <pstate>
cpu<cpu>	物理 CPU <cpu> 的利用率 ( 片段 )。默认情况下，此选项处于启用状态。	pCPU <cpu> 存在	CPU <cpu>
cpu_avg	物理 CPU 平均利用率 ( 片段 )。默认情况下，此选项处于启用状态。	无	平均 CPU 使用率

指标名称	说明	条件	XenCenter 名称
inflight_<sr-uuid-short>	当前正在进行的 I/O 请求。默认情况下，此选项处于启用状态。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 传输请求
io_throughput_read_<sr-uuid-short>	从 SR 读取的数据 ( MiB/秒 ) 。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 读取吞吐量
io_throughput_write_<sr-uuid-short>	写入 SR 的数据 ( MiB/秒 ) 。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 写入吞吐量
io_throughput_total_<sr-uuid-short>	所有 SR I/O ( MiB/秒 ) 。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 总吞吐量
iops_read_<sr-uuid-short>	每秒读取请求数。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 读取 IOPS
iops_write_<sr-uuid-short>	每秒写入请求数。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 写入 IOPS
iops_total_<sr-uuid-short>	每秒 I/O 请求数。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 总 IOPS
iowait_<sr-uuid-short>	I/O 等待时间百分比。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> IO 等待

指标名称	说明	条件	XenCenter 名称
latency_<sr-uuid-short>	平均 I/O 延迟 ( 毫秒 )。	主机上的 SR <sr> 中至少有一个插入的 VBD	<sr> 延迟
loadavg	Domain0 loadavg。默认情况下处于启用状态	无	控制域负载
memory_free_kib	可用内存总量 (KiB)。默认情况下，此选项处于启用状态。	无	可用内存
memory_reclaimed	通过腾出回收的主机内存 (B)。	无	回收的内存
memory_reclaimed_max	可通过腾出回收的主机内存 (B)。	无	潜在回收的内存
memory_total_kib	主机中的内存总量 (KiB)。默认情况下，此选项处于启用状态。	无	总内存
network/latency	从本地主机传输到所有在线主机的最后两个检测信号之间的时间间隔 ( 以秒为单位 )。默认情况下处于禁用状态。	已启用 HA	网络延迟
statefile/<t>/latency	从本地主机访问的最新状态文件的周转时间 ( 以秒为单位 )。默认情况下处于禁用状态。	已启用 HA	高可用性状态文件延迟
pif_<pif>_rx	物理接口上每秒接收的字节数 <pif>。默认情况下，此选项处于启用状态。	PIF<pif>存在	<xencenter-pif-name> 接收 ( 见注释 )
pif_<pif>_tx	物理接口上每秒发送的字节数 <pif>。默认情况下，此选项处于启用状态。	PIF<pif>存在	<xencenter-pif-name> 发送 ( 见注释 )
pif_<pif>_rx_errors	物理接口上每秒发生的接收错误 <pif>。默认情况下处于禁用状态。	PIF<pif>存在	<xencenter-pif-name> 接收错误 ( 见注释 )
pif_<pif>_tx_errors	物理接口上每秒发生的传输错误 <pif>。默认情况下处于禁用状态	PIF<pif>存在	<xencenter-pif-name> 发送错误 ( 见注释 )

指标名称	说明	条件	XenCenter 名称
pif_aggr_rx	所有物理接口上每秒接收的字节数。默认情况下，此选项处于启用状态。	无	NIC 接收的总数据量
pif_aggr_tx	所有物理接口上每秒发送的字节数。默认情况下，此选项处于启用状态。	无	NIC 发送的总数据量
sr_<sr>_cache_size	IntelliCache SR 的大小（以字节为单位）。默认情况下，此选项处于启用状态。	已启用 IntelliCache	IntelliCache 缓存大小
sr_<sr>_cache_hits	每秒缓存命中率。默认情况下，此选项处于启用状态。	已启用 IntelliCache	IntelliCache 缓存命中数
sr_<sr>_cache_misses	每秒缓存错失率。默认情况下，此选项处于启用状态。	已启用 IntelliCache	IntelliCache 缓存未命中数
xapi_allocation_kib	xapi 后台程序完成的内存 (KiB) 分配。默认情况下，此选项处于启用状态。	无	代理内存分配
xapi_free_memory_kib	xapi 后台程序的可用内存 (KiB)。默认情况下，此选项处于启用状态。	无	可用代理内存
xapi_healthcheck/latency_healt	本地主机上最新 xapi 状态监测调用的周转时间（以秒为单位）。默认情况下处于禁用状态	已启用 HA	XenServer 运行状况检查延迟
xapi_live_memory_kib	xapi 后台程序使用的实时内存 (KiB)。默认情况下，此选项处于启用状态。	无	活动代理内存
xapi_memory_usage_kib	xapi 后台程序使用的已分配总内存 (KiB)。默认情况下，此选项处于启用状态。	无	代理内存使用率

### 9.1.2. 可用 VM 指标

指标名称	说明	条件	XenCenter 名称
cpu<cpu>	vCPU 利用率 <cpu>（片段）。默认情况下处于启用状态	vCPU <cpu> 存在	CPU <cpu>
memory	当前分配给 VM 的内存（以字节为单位）。默认情况下处于启用状态	无	总内存
memory_target	VM 气球驱动程序的目标（以字节为单位）。默认情况下处于启用状态	无	内存目标



指标名称	说明	条件	XenCenter 名称
memory_internal_free	按来宾系统代理报告使用的内存 (KiB)。默认情况下处于启用状态	无	可用内存
runstate_fullrun	所有 VCPU 运行的时间片段。	无	vCPU 完全运行
runstate_full_contention	所有 VCPU 的可运行时间片段 (即, 等待 CPU)	无	vCPU 完全争用
runstate_concurrency_haz	有些 VCPU 正在运行和有些 VCPU 可运行的时间片段	无	vCPU 并发危险
runstate_blocked	所有 VCPU 受阻或脱机的时间片段	无	vCPU 空闲
runstate_partial_run	有些 VCPU 正在运行和有些 VCPU 受阻的时间片段	无	vCPU 部分运行
runstate_partial_contention	有些 VCPU 可运行和有些 VCPU 受阻的时间片段	无	vCPU 部分争用
vbd_<vbd>_write	每秒写入设备 <vbd> 的字节数。默认情况下处于启用状态	VBD <vbd> 存在	磁盘 <vbd> 写入
vbd_<vbd>_read	每秒从设备 <vbd> 读取的字节数。默认情况下, 此选项处于启用状态。	VBD <vbd> 存在	磁盘 <vbd> 读取
vbd_<vbd>_write_latency	写入设备 <vbd> (以毫秒为单位)。	VBD <vbd> 存在	磁盘 <vbd> 写入延迟
vbd_<vbd>_read_latency	从设备 <vbd> 读取 (以毫秒为单位)。	VBD <vbd> 存在	磁盘 <vbd> 读取延迟
vbd <vbd>_iops_read	每秒读取请求数。	主机上的非 ISO VDI 至少有一个插入的 VBD	磁盘 <vbd> 读取 IOPS
vbd <vbd>_iops_write	每秒写入请求数。	主机上的非 ISO VDI 至少有一个插入的 VBD	磁盘 <vbd> 写入 IOPS

指标名称	说明	条件	XenCenter 名称
vbd <vbd>_iops_total	每秒 I/O 请求数。	主机上的非 ISO VDI 至少有一个插入的 VBD	磁盘 <vbd> 总 IOPS
vbd <vbd>_iowait	I/O 等待时间百分比。	主机上的非 ISO VDI 至少有一个插入的 VBD	磁盘 <vbd> IO 等待
vbd <vbd>_inflight	当前正在进行的 I/O 请求。	主机上的非 ISO VDI 至少有一个插入的 VBD	磁盘 <vbd> 传输请求
vbd <vbd>_avgqu_sz	平均 I/O 队列大小。	主机上的非 ISO VDI 至少有一个插入的 VBD	磁盘 <vbd> 队列大小
vif <vif>_rx	虚拟接口上每秒接收的字节数 <vif>。默认情况下，此选项处于启用状态。	VIF <vif> 存在	<vif> 接收
vif <vif>_tx	虚拟接口每秒传输的字节数 <vif>。默认情况下处于启用状态。	VIF <vif> 存在	<vif> 发送
vif <vif>_rx_errors	虚拟接口上每秒发生的接收错误 <vif>。默认情况下，此选项处于启用状态。	VIF <vif> 存在	<vif> 接收错误
vif <vif>_tx_errors	虚拟接口上每秒发生的传输错误 <vif>。默认情况下处于启用状态。	VIF <vif> 存在	<vif> 发送错误



### 注意

<xencenter-pif-name> 可以是以下任意一项：

NIC <pif>	如果 <pif> 包含 pif_eth#，其中 # 为 0-9
<pif>	如果 <pif> 包含 pif_eth## 或 pif_xenbr# 或 pif_bond#
<Internal> 网络 <pif>	如果 <pif> 包含 pif_xapi# ( 请注意，<Internal> 按原样显示 )
TAP <tap>	如果 <pif> 包含 pif_tap#
xapi 环回	如果 <pif> 包含 pif_lo

### 9.1.3. 分析和显示 XenCenter 中的指标

XenCenter 中的性能选项卡提供整个资源池性能的实时监视统计数据，并以图表方式显示虚拟机和物理机性能的趋势。默认情况下，显示 CPU、内存、网络和磁盘 I/O 的图表包含在“性能”选项卡中，但是您可以添加其他指标，更改现有图表的外观，或者创建其他图表。请参阅第 9.1.3.1 节“配置性能图表”。

- 可以查看最长 12 个月的性能数据，并可将数据放大，以便更清楚地查看活动高峰。
- 当托管服务器、VM 或存储库上的 CPU、内存使用、网络 I/O、存储 I/O 或磁盘 I/O 使用超出指定阈值时，XenCenter 会生成性能警报。请参阅第 9.2.1 节“使用 XenCenter 查看警报”



#### 注意

为查看完整的 VM 性能数据，必须安装 XenServer 工具（半虚拟化驱动程序）。

#### 9.1.3.1. 配置性能图表

##### 添加新图表

1. 在性能选项卡上，单击操作，然后单击新建图表。此时将显示新建图表对话框。
2. 在名称字段中，输入图表的名称。
3. 从数据源列表中，选择图表中要包含的数据源所对应的复选框。
4. 单击保存。

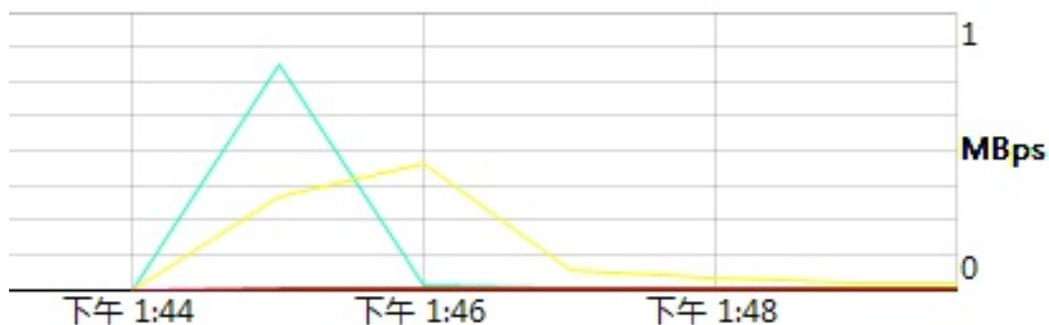
##### 编辑现有图表

1. 导航到性能选项卡，然后选择要修改的图表。
2. 在图表上单击鼠标右键，选择操作，或者单击操作按钮。然后选择编辑图表。
3. 在“图表详细信息”窗口中，执行必要更改，然后单击确定。

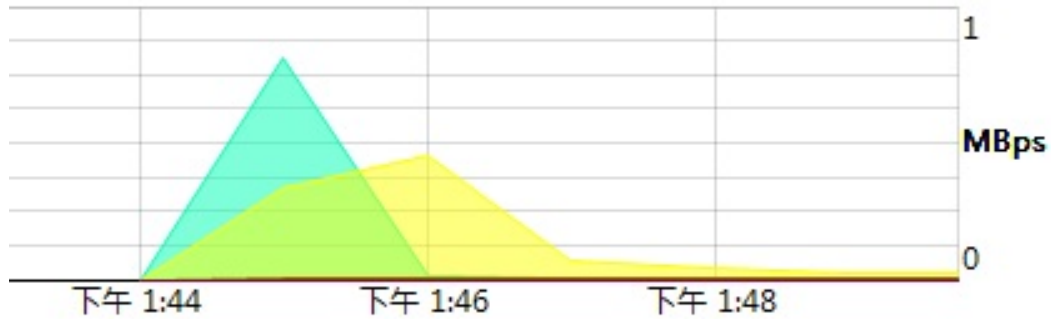
##### 9.1.3.1.1. 配置图表类型

可以采用折线图或区域图的形式来显示性能图表中的数据：

折线图：



区域图：



### 更改图表类型

1. 在工具菜单上，单击选项，选择图表。
2. 要以折线图形式查看性能数据，请单击折线图单选按钮
3. 要以区域图形式查看性能数据，请单击区域图单选按钮
4. 单击确定保存更改。

有关配置和查看 XenCenter 性能图表的详细信息，请参阅《监视系统性能》中的 XenCenter 帮助部分。

### 9.1.4. 配置指标



#### 注意

C-state 和 P-state 是某些处理器的电源管理功能。可用状态范围取决于主机的物理功能以及电源管理配置。

主机和 VM 命令可以返回以下信息：

- 数据源的完整说明
- 应用于指标的单位
- 可以使用的可能值范围

例如：

```
name_label: cpu0-C1
name_description: Proportion of time CPU 0 spent in C-state 1
  enabled: true
  standard: true
  min: 0.000
  max: 1.000
  units: Percent
```

### 启用特定指标

默认情况下，启用和收集大部分指标，要启用未启用的指标，请输入以下信息：

```
xe host-data-source-record data-source=<metric name> host=<hostname>
```

### 禁用特定指标

您可能不想定期收集某些指标。要禁用先前启用的指标，请输入以下信息：



```
xe host-data-source-forget data-source=<metric name> host=<hostname>
```

显示当前启用的主机指标的列表

要列出当前正在收集的主机指标的列表，请输入以下信息：

```
xe host-data-source-list host=<hostname>
```

显示当期启用的 VM 指标的列表

要托管当前收集的 VM 指标，请输入以下信息：

```
xe vm-data-source-list vm=<vm_name>
```

### 9.1.5. 使用 RRDs

为存储性能指标，XenServer 使用 (RRDs)。它们包含大小固定的数据库中的多个轮询档案 (RRA)。

数据库中的每个档案以指定精度抽取其特定指标样本：

- 在 10 分钟内每隔 5 秒钟
- 在过去 2 个小时内每隔 1 分钟
- 在过去 1 周内每隔 1 小时
- 在过去 1 年内每隔 1 天

每隔 5 秒钟的抽样记录实际数据点，但是，下列 RRA 则使用整合函数 (CF)。XenServer 支持的 CF：

- AVERAGE
- MIN
- MAX

RRD 为单个 VM (包括 dom0) 和 XenServer 主机而存在。VM RRD 存储在它们运行所在的主机上，不运行时，存储在池主服务器上。因此，必须知道 VM 的位置，才能检索相关联的性能数据。

有关如何使用 XenServer RRD 的详细信息，请参阅 Citrix Developer Network 文章：[使用 XenServer RRD](#)。

#### 9.1.5.1. 使用 HTTP 分析 RRD

RRD 可以通过 HTTP，从使用 /host\_rrd 或 /vm\_rrd 上注册的 HTTP 处理程序指定的 XenServer 主机下载。两个地址都要求 HTTP auth 进行身份验证，或者通过提供有效的 XenAPI 会话参考作为队列参数进行身份验证。例如：

下载主机 RRD. `wget http://<server>/host_rrd?session_id=OpaqueRef:<SESSION HANDLE>>`

下载 VM RRD. `wget http://<server>/vm_rrd?session_id=OpaqueRef:<SESSION HANDLE>>&uuid=<VM UUID>>`

两个调用都将以能够导入 rrdtool 进行分析或直接解析的格式下载 XML。

有关如何使用 XenServer RRD 和 HTTP 的详细信息，请参阅 Citrix Developer Network 文章：[使用 XenServer RRD](#)。

### 9.1.5.2. 使用 rrd2csv 分析 RRD

除了查看 XenCenter 中的性能指标，rrd2csv 工具还会将 RRD 记录为 Comma Separated Value (CSV) 格式。提供手册和帮助页面。要显示 rrd2csv 工具手册或帮助页面，请运行以下命令：

```
man rrd2csv
```

或

```
rrd2csv --help
```



#### 注意

使用多个选项时，应当单独提供。例如：要返回与 VM 或主机相关联的 UUID 和名称标签，应当调用 rrd2csv，如下所示：

```
rrd2csv -u -n
```

虽然返回的 UUID 是唯一的，而且可用作主密钥，但是，实体的名称标签可能不必是唯一的。

手册页面 (rrd2csv --help) 是工具的明确性帮助文本。

## 9.2. 警报

您可以配置 XenServer，根据任何可用的主机或 VM 指标生成警报。此外，当主机经历某些条件和状态时，XenServer 提供可以触发的预配置警报。您可以使用 XenCenter 或 xe CLI 查看这些警报。

### 9.2.1. 使用 XenCenter 查看警报

您可以在 XenCenter 中查看不同类型的警报，方法是单击通知，然后单击警报。“警报”视图会显示各种不同类型的警报，包括：

- [第 9.2.1.1 节 “XenCenter 性能警报”](#)
- [第 9.2.1.2 节 “XenCenter 警报”](#)
- [第 9.2.1.3 节 “XenCenter 软件更新警报”](#)

#### 9.2.1.1. XenCenter 性能警报

当托管主机、VM 或存储库 (SR) 上的 CPU、内存使用率、网络、存储吞吐量或 VM 磁盘活动超出指定阈值时，可生成性能警报。

默认情况下，警报重复间隔为 60 分钟，但如果需要，也可以修改此间隔。警报将在 XenCenter 的通知区域中的警报页面上显示。此外，您可以配置 XenCenter 以电子邮件形式发送任何指定的性能警报以及其他严重系统警报。此外，任何使用 xe CLI 配置的自定义警报也显示在 XenCenter 中的系统警报页面上。

每个警报都有对应的优先级/严重级别。您也可以修改这些级别，根据需要选择在触发警报时接收电子邮件。默认警报优先级/严重性级别设为 3。

优先级	名称	说明	默认电子邮件警报
1	严重	立即采取行动，否则数据可能永久丢失/损坏。	是
2	主要	立即采取行动，否则有些服务可能失败。	是

优先级	名称	说明	默认电子邮件警报
3	警告	立即采取行动，否则服务可能会中断。	是
4	次要	注意，有些事项仅得到改进。	否
5	信息	日常信息（VM 启动、停止、恢复等）	否
?	未知	未知错误	否

#### 9.2.1.1.1. 配置性能警报

配置性能警报：

1. 在资源窗格中，选择相应的主机、VM 或 SR，单击常规选项卡，然后单击属性。

2. 单击警报选项卡。

- 主机或 VM CPU 警报：选择生成 CPU 使用率警报复选框，然后设置将触发该警报的 CPU 使用率和时间阈值。
- 主机或 VM 的网络警报：选择生成网络使用率警报复选框，然后设置将触发该警报的网络使用率和时间阈值。
- 主机内存警报：选择生成内存使用率警报复选框，然后设置将触发该警报的内存使用率和时间阈值。
- VM 磁盘使用情况警报：选择生成磁盘使用情况警报复选框，然后设置将触发该警报的磁盘使用情况和时间阈值。
- SR 存储吞吐量警报：选择生成存储吞吐量警报复选框，然后设置将触发该警报的存储吞吐量和时间阈值。



#### 注意

物理块设备 (PBD) 代表特定 XenServer 主机与连接的 SR 之间的接口。当 PBD 上的总读/写 SR 吞吐量活动超过您指定的阈值时，与 PBD 连接的主机上将生成警报。与其他 XenServer 主机警报不同，此警报必须在相关的 SR 上进行配置。

- 要更改警报重复间隔，请在警报重复间隔框中输入分钟数。达到警报阈值并生成一个警报后，只有在已耗完警报重复间隔时间后，才会生成另一个警报。

3. 单击确定保存更改。

有关如何查看、过滤和配置性能警报严重性的综合详细信息，请参阅 XenCenter 联机帮助。

#### 9.2.1.2. XenCenter 警报

下表显示将触发在 XenCenter 的警报页面上显示的警报的系统事件/条件。

名称	优先级/严重性	说明
license_expires_soon	2	XenServer 许可协议即将过期。
ha-statefile_lost	2	与 HA 存储库的联系丢失，立即采取行动。
ha-heartbeat_approaching_timeout	5	HA 即将超时，主机可能重新启动，除非采取行动。



名称	优先级/严重性	说明
ha_statefile_approaching_timeout	5	HA 即将超时，主机可能重新启动，除非采取行动。
ha-xapi_healthcheck_approaching_timeout	5	HA 即将超时，主机可能重新启动，除非采取行动。
ha_network_bonding_error	3	潜在服务丢失。发送 HA 检测信号的网络丢失
ha_pool_overcommitted	3	潜在服务丢失。HA 无法保障对已配置 VM 的保护
ha_poor_drop_in_plan_exists_for	3	HA 范围已丢弃，很有可能失败，尚未出现丢失
ha_protected_vm_restart_failed	2	服务丢失。HA 无法重新启动受保护的 VM
ha_host_failed	3	HA 检测到主机发生了故障
ha_host_was_fenced	4	HA 重新启动主机，保护 VM 不损坏
redo_log_healthy	4	xapi redo 日志已从上一错误中恢复
redo_log_broken	3	xapi redo 日志遇到了错误
ip_configured_pif_can_unplug	3	使用 HA 时，已配置 IP 的 NIC 会被 xapi 拔出，这有可能导致 HA 故障。
host_sync_data_failed	3	无法同步 XenServer 性能统计信息
host_clock_skew_detected	3	主机时钟与池中其他主机时钟不同步
host_clock_went_backwards	1	主机时钟已损坏
pool_master_transition	4	已将新主机指定为池主服务器
pbd_plug_failed_on_server_start	3	引导时，主机未连接到存储。
auth_external_init_failed	2	主机无法启用外部 AD 身份验证
auth_external_pool_non-homogeneous	2	池中的主机有不同的 AD 身份验证配置。
multipath_period_alert	3	到 SR 的路径发生故障或已恢复。
bond-status-changed	3	绑定中的链路已断开或重新连接

### 9.2.1.3. XenCenter 软件更新警报

警报	说明
XenCenter 版本较旧	XenServer 需要使用较新的版本，但仍可连接到当前版本
XenCenter 过时	XenCenter 版本太旧，无法连接到 XenServer

警报	说明
XenServer 过时	XenServer 版本较旧，当前 XenCenter 无法连接到它
许可证过期警报	XenServer 许可证已过期
缺少 IQN 警报	XenServer 使用 iSCSI 存储，但主机 IQN 为空白
重复 IQN 警报	XenServer 使用 iSCSI 存储，但存在重复的主机 IQN

### 9.2.2. 使用 xe CLI 配置性能警报



#### 注意

系统将按五分钟的最小间隔检查警报触发器（这样可以避免因检查这些条件而使系统承受过高的负荷并防止误报）；如果设置的警报重复间隔小于此值，警报仍会按五分钟的最小间隔生成。

性能监视 perfmon 工具每 5 分钟运行一次，并且平均 1 分钟从 XenServer 请求一次更新。这些默认值可以在 `/etc/sysconfig/perfmon` 中更改。

perfmon 工具每 5 分钟读取一次同一主机上运行的性能变量的更新。这些变量分成两组，一组与主机本身相关，另一组与在此主机上运行的每个 VM 相关。对于每个 VM 和 XenServer 主机，perfmon 读取参数 `other-config:perfmon`，并且使用此字符串来确定监视哪些变量以及在何种情况下生成消息。

例如，下面显示了通过将 XML 字符串写入参数 `other-config:perfmon` 来配置 VM“CPU 使用率”警报的示例：

```
xe vm-param-set uuid=<vm_uuid> other-config:perfmon=\
```

```
'<config>
  <variable>
    <name value="cpu_usage"/>
    <alarm_trigger_level value="0.5"/>
  </variable>
</config>'
```



#### 注意

允许多个 `<variable>` 节点

### 有效的 VM 元素

#### name

变量名称（无默认值）。如果名称值为 `cpu_usage`、`network_usage` 或 `disk_usage` 中的任意一个，则当使用这些值时，默认情况下不要求使用 `rrd_regex` 和 `alarm_trigger_sense` 参数。

#### alarm\_priority

生成警报的优先级（默认为 3）。

#### alarm\_trigger\_level

触发警报的值的级别（无默认级别）。

#### alarm\_trigger\_sense

如果 `alarm_trigger_level` 是最大值，则为 `high`；如果 `alarm_trigger_level` 是最小值，则为 `low`（默认为 `high`）。



#### alarm\_trigger\_period

在发送警报之前，能够接收的（高于或低于警报阈值）值的秒数（默认为 60）。

#### alarm\_auto\_inhibit\_period

在发送警报后多久（秒）禁用此警报（默认为 3600）。

#### consolidation\_fn

将来自 rrd\_updates 的变量合并为一个值。对于 cpu\_usage，默认为 average；对于 fs\_usage，默认为 get\_percent\_fs\_usage；对于所有其他，默认为 sum。

#### rrd\_regex

匹配来自 xe vm-data-sources-list uuid=<vm\_uuid> 的变量的名称，以计算性能值。此参数具有下列已命名变量的默认值：

- cpu\_usage
- network\_usage
- disk\_usage

如果指定了正则表达式，将使用根据 consolidation\_fn 指定的方法，对 xe vm-data-source-list 返回的项中名称与指定的正则表达式匹配的所有项的值进行合并。

### 有效主机元素

#### name

变量名称（无默认值）。

#### alarm\_priority

生成警报的优先级（默认为 3）。

#### alarm\_trigger\_level

触发警报的值的级别（无默认级别）。

#### alarm\_trigger\_sense

如果 alarm\_trigger\_level 是最大值，则为 high；如果 alarm\_trigger\_level 是最小值，则为 low。（默认为 high）

#### alarm\_trigger\_period

在发送警报之前，能够接收（高于或低于警报阈值）的值的秒数（默认为 60）。

#### alarm\_auto\_inhibit\_period

发送警报后多久（秒）禁用警报。（默认为 3600）。

#### consolidation\_fn

将来自 rrd\_updates 的变量合并为一个值（默认为 sum 或 average）

#### rrd\_regex

与 xe vm-data-source-list uuid=<vm\_uuid> 命令返回的变量的名称相匹配的正则表达式，用于计算统计值。此参数具有下列已命名变量的默认值：

- cpu\_usage
- network\_usage
- memory\_free\_kib
- sr\_io\_throughput\_total\_xxxxxxx（其中 xxxxxxxx 是 SR-UUID 的前八个字符）。

**注意**

SR 吞吐量：必须在 SR 而非主机上配置存储吞吐量警报。例如：

```
xe sr-param-set uuid=<sr_uuid> other-config:perfmon=\
'<config>
  <variable>
    <name value="sr_io_throughput_total_per_host"/>
    <alarm_trigger_level value="0.01"/>
  </variable>
</config>'
```

### 9.2.2.1. 通用示例配置

下例显示了一个通用配置：

```
<config>
  <variable>
    <name value="NAME_CHOSEN_BY_USER"/>
    <alarm_trigger_level value="THRESHOLD_LEVEL_FOR_ALARM"/>
    <alarm_trigger_period value="RAISE_ALARM_AFTER_THIS_MANY_SECONDS_OF_BAD_VALUES"/>
    <alarm_priority value="PRIORITY_LEVEL"/>
    <alarm_trigger_sense value="HIGH_OR_LOW"/>
    <alarm_auto_inhibit_period value="MINIMUM_TIME_BETWEEN_ALARMS_FROM_THIS_MONITOR"/>
    <consolidation_fn value="FUNCTION_FOR_COMBINING_VALUES"/>
    <rrd_regex value="REGULAR_EXPRESSION_TO_CHOOSE_DATASOURCE_METRIC"/>
  </variable>

  <variable>
    ...
  </variable>

  ...
</config>
```

## 9.3. 配置电子邮件警报

客户能够配置 XenServer，使其在 XenServer 主机生成警报时发送电子邮件通知。这可以通过使用 XenCenter 或使用 xe 命令行接口 (CLI) 完成。

### 9.3.1. 使用 XenCenter 启用电子邮件警报

1. 在“资源”窗格中，在池上单击鼠标右键，选择属性。
2. 在属性窗口中，选择电子邮件选项。
3. 选择发送电子邮件警报通知复选框，然后输入电子邮件地址和 SMTP 服务器详细信息。

**注意**

输入不需要身份验证的 SMTP 服务器的详细信息。

### 9.3.2. 使用 xe CLI 启用电子邮件警报



#### 重要

使用 XenCenter 或 xe CLI 启用电子邮件通知时，客户应当输入不需要身份验证的 SMTP 服务器的详细信息。通过需要身份验证的 SMTP 服务器发送的电子邮件将不会进行发送。

要配置电子邮件警报，请指定电子邮件地址和 SMTP 服务器：

```
xe pool-param-set uuid=<pool_uuid> other-config:mail-destination=<joe.bloggs@domain.tld>
xe pool-param-set uuid=<pool_uuid> other-config:ssmtp-mailhub=<smtp.domain.tld[:port]>
```

在发送电子邮件之前，您还可以在消息中指定优先级（在 XenCenter 中也称为“严重性”）字段的最小值：

```
xe pool-param-set uuid=<pool_uuid> other-config:mail-max-priority=<level>
```

默认优先级为 4。



#### 注意

某些 SMTP 服务器只转发地址使用 FQDN 的邮件。如果您发现无法转发电子邮件，可能是由于此原因导致，在这种情况下，可以将服务器主机名设置为 FQDN，以便通过它连接到您的邮件服务器。

#### 9.3.2.1. 通过经过身份验证的 SMTP 服务器发送电子邮件警报

XenServer 中的 mail-alarm 实用程序使用 sSMTP 发送电子邮件通知。发送电子邮件通知之前，mail-alarm 实用程序会查找配置文件 mail-alarm.conf。如果此配置文件存在，文件内容将用于配置 sSMTP，否则，XAPI 数据库（使用 XenCenter 或 xe CLI 配置）中的详细信息将用于发送电子邮件警报。要通过经过身份验证的 SMTP 服务器发送电子邮件通知，客户应当在 /etc/ 中使用下列内容创建一个 mail-alarm.conf 文件：

```
root=postmaster
authUser=<username>
authPass=<password>
mailhub=<server address>[:port]
```



#### 注意

此配置文件将用于 XenServer 主机生成的所有警报。

##### 9.3.2.1.1. 其他配置选项

每个 SMTP 服务器在设置上都略有不同，并且可能需要额外配置。请参阅下面从 ssmtp.conf 手册页面解压的内容，查看正确的语法和可用选项：



#### NAME

ssmtp.conf – ssmtp configuration file

#### DESCRIPTION

ssmtp reads configuration data from /etc/ssmtp/ssmtp.conf. The file contains keyword-argument pairs, one per line. Lines starting with '#' and empty lines are interpreted as comments.

The possible keywords and their meanings are as follows (both are case-insensitive):

##### Root

The user that gets all mail for userids less than 1000. If blank, address rewriting is disabled.

##### Mailhub

The host to send mail to, in the form host | IP\_addr port [:port]. The default port is 25.

##### RewriteDomain

The domain from which mail seems to come. For user authentication.

##### Hostname

The full qualified name of the host. If not specified, the host is queried for its hostname.

##### FromLineOverride

Specifies whether the From header of an email, if any, may override the default domain. The default is "no".

##### UseTLS

Specifies whether ssmtp uses TLS to talk to the SMTP server. The default is "no".

##### UseSTARTTLS

Specifies whether ssmtp does a EHLO/STARTTLS before starting SSL negotiation. See RFC 2487.

##### TLSCert

The file name of an RSA certificate to use for TLS, if required.

##### AuthUser

The user name to use for SMTP AUTH. The default is blank, in which case SMTP AUTH is not used.

##### AuthPass

The password to use for SMTP AUTH.

##### AuthMethod

The authorization method to use. If unset, plain text is used. May also be set to "cram-md5".

## 9.4. 自定义字段和标记

XenCenter 支持创建标记和自定义字段，这样，您可以组织和快速搜索 VM、存储等对象。有关详细信息，请参阅 XenCenter 帮助。

## 9.5. 自定义搜索

XenCenter 支持创建自定义搜索。可以导出和导入搜索，还可以在导航面板中显示搜索结果。有关详细信息，请参阅 XenCenter 帮助。

## 9.6. 确定物理总线适配器的吞吐量

对于 FC、SAS 和 iSCSI HBA，您可以通过以下步骤确定 PBD 的网络吞吐量。

确定 PBD 吞吐量：

1. 列出主机上的 PBD。
2. 确定哪个 LUN 在哪个 PBD 上路由。
3. 对于每个 PBD 和 SR，列出 SR 上引用 VDI 的 VBD。
4. 针对主机上连接到 VM 的所有活动 VBD，计算组合吞吐量。

针对 iSCSI 和 NFS 存储，检查网络统计，以确定阵列上是否存在吞吐量瓶颈，或者 PBD 是否已饱和。



## 第 10 章 故障排除

### 10.1. 支持

Citrix 提供两种形式的支持服务：[Citrix 技术支持](#) Web 站点上的免费自助支持以及付费的支持服务（可以从支持站点购买）。通过 Citrix 技术支持，您可以在遇到技术问题时，打开在线支持案例或者通过电话与支持中心联系。

[Citrix 知识中心](#) 包含许多资源，当您遇到异常行为、崩溃或其他问题时，这些资源可能会对您有所帮助。资源包括：论坛、知识库文章、白皮书、产品文档、修补程序和其他更新。

本章内容旨在帮助您解决 XenServer 主机的技术问题；此外还介绍了应用程序日志的位置和其他有助于 Citrix 解决方案提供商和 Citrix 跟踪并解决问题的其他信息，如果本章内容不能帮助您解决问题，这些信息也对您有所帮助。

《XenServer 安装指南》中介绍了对安装问题的故障排除。《XenServer 虚拟机用户指南》中介绍了对虚拟机问题的故障排除。



#### 重要

我们建议您仅在 Citrix 解决方案提供商或 Citrix 技术支持人员的指导下利用本章的故障诊断信息解决问题。



#### 注意

在某些支持案例中，出于调试目的，需要串行控制台访问。因此，设置 XenServer 配置时，建议配置串行控制台访问。对于不具备物理串行端口的主机（例如刀片式服务器）或者无适当物理基础结构可用的情况，客户应当研究能否配置嵌入式管理设备，例如 Dell DRAC 或 HP iLO。有关设置串行控制台访问的详细信息，请参阅 CTX121442 [How to Set Up a Serial Cable for Troubleshooting on XenServer](#)（如何为 XenServer 故障排除设置串行线缆）。

### 10.2. 运行状况检查

通过运行状况检查功能，客户能够在 XenCenter 中自动执行生成服务器状态报告并将其上载至 Citrix Insight Services (CIS) 以及接收 CIS 分析报告的过程。

将任何符合条件的池连接到 XenCenter 时，系统都会提示您为池启用运行状况检查功能。注册过程中，可以指定希望自动向 CIS 上载服务器状态报告的计划，输入要用于与池建立连接的 XenServer 凭据以及验证向 CIS 上载的内容。成功注册池以进行运行状况检查后，您将在 XenCenter 中收到与池的运行状况有关的通知。这样，您就可以根据 CIS 生成的报告，主动监控 XenServer 系统的运行状况了。

#### 要求

要使用运行状况检查功能，必须满足以下要求：

- 池中的所有主机都应运行 XenServer 7.0
- 您应使用随 XenServer 提供的 XenCenter 连接到 XenServer 池 7.0
- XenCenter 应能够访问 Internet
- XenCenter 计算机上应安装并运行运行状况检查服务。
- 如果使用 Active Directory (AD)，您应具有池操作员或更高级别的角色

有关运行状况检查的详细信息以及注册池以进行运行状况检查的步骤说明，请参阅 XenCenter 帮助。

## 10.3. XenServer 主机日志

XenCenter 可用于收集 XenServer 主机信息。单击 Tools ( 工具 ) 菜单中的 Server Status Report ( 服务器状态报告 )，打开 Server Status Report ( 服务器状态报告 ) 向导。您可以从不同类型的信息 ( 各种日志、故障转储等 ) 列表中进行选择。这些信息将编译并下载到正在运行 XenCenter 的计算机上。有关详细信息，请参阅 XenCenter 帮助。

此外，XenServer 主机包括多个 CLI 命令，通过这些命令，可以方便地使用 xen-bugtool 实用程序整理日志输出和系统信息的各种其他比特。使用 xe 命令 host-bugreport-upload 可以收集适当的日志文件和系统信息并将其上传到 Citrix 支持 ftp 站点。有关此命令及其可选参数的完整说明，请参阅第 4.9.4 节 “host-bugreport-upload”。如果要求您将故障转储发送到 Citrix 支持，请使用 xe 命令 host-crashdump-upload。有关此命令及其可选参数的完整说明，请参阅第 4.9.6 节 “host-crashdump-upload”。



小心

可能会将敏感信息写入 XenServer 主机日志。

### 10.3.1. 将主机日志消息发送到中央服务器

您可以将 XenServer 主机配置为将日志写入远程服务器，而不是写入控制域文件系统。远程服务器必须运行 syslogd 后台程序，以接收日志并将它们正确地聚合。Syslogd 后台程序是 Linux 和 Unix 的所有版本的标准部件，Windows 和其他操作系统可以使用第三方版本。

将日志写入远程服务器：

1. 将 syslog\_destination 参数设置为要在其中写入日志的远程服务器的主机名或 IP 地址：

```
xe host-param-set uuid=<xenserver_host_uuid> logging:syslog_destination=<hostname>
```

2. 发出以下命令：

```
xe host-syslog-reconfigure uuid=<xenserver_host_uuid>
```

以强制执行更改。( 也可以通过指定 host 参数远程执行此命令。 )

## 10.4. XenCenter 日志

XenCenter 还提供客户端日志。此文件包括使用 XenCenter 时执行的所有操作以及出现的所有错误的完整描述，还包含事件的信息记录，以便为您提供发生的各种操作的审核追踪。XenCenter 日志文件存储在您的配置文件文件夹中。如果 XenCenter 安装在 Windows XP 中，则路径为

```
%userprofile%\AppData\Citrix\XenCenter\logs\XenCenter.log
```

如果 XenCenter 安装在 Windows Vista 中，则路径为

```
%userprofile%\AppData\Citrix\Roaming\XenCenter\logs\XenCenter.log
```

要快速找到 XenCenter 日志文件，例如，要打开该日志文件或通过电子邮件发送它的时候，可单击 XenCenter 帮助菜单中的查看应用程序日志文件。

## 10.5. 对 XenCenter 和 XenServer 主机之间的连接进行故障排除

如果您在将 XenServer 连接到 XenCenter 主机时遇到问题，请检查以下内容：

- XenCenter 版本是否低于尝试连接到的 XenServer 主机版本？



XenCenter 应用程序是向后兼容的，可以与旧版本的 XenServer 主机正常通信，但是旧版本的 XenCenter 无法与较新版本的 XenServer 主机正常通信。

要更正此问题，请安装与 XenCenter 主机版本相同的 XenServer 版本或更新的版本。

- 您的许可证是否有效？

您可以在 XenCenter 中的 License Details ( 许可证详细信息 ) 部分下 XenServer 主机的常规选项卡中查看许可证密钥的到期日期。

有关授权主机的详细信息，请参阅《XenServer 安装指南》中的“XenServer 许可”一章。

- XenServer 主机使用 HTTPS 通过端口 443 与 XenCenter 通信 ( 使用 XenAPI 进行命令和响应的双向连接 )，并通过端口 5900 与半虚拟化的 Linux VM 进行图形 VNC 连接。如果在 XenServer 主机和运行客户端软件的计算机之间启用了防火墙，请确保防火墙允许通过这些端口的通信。

## 命令行接口

本章介绍 XenServer 命令行接口 (CLI)。xe CLI 支持通过编写脚本来自动完成系统管理任务，并允许将 XenServer 集成到现有 IT 基础结构中。

默认情况下，xe 命令行接口安装在 XenServer 主机上并包含在 XenCenter 中。对于 Linux，还可以使用独立远程 CLI。

在 Windows 上，xe.exe CLI 可执行文件随 XenCenter 一起安装。

要使用该可执行文件，请打开 Windows 命令提示符窗口，将目录更改为该文件所在的目录（通常为 C:\Program Files\XenSource\XenCenter），或者将该文件的安装位置添加到您的系统路径。

在基于 RPM 的版本（如 Red Hat 和 CentOS）上，可以从 XenServer 主安装 ISO 上名为 xe-cli-6.00-@BUILD\_NUMBER@.i386.rpm 的 RPM 安装独立的 xe CLI 可执行文件，如下所示：

```
rpm -ivh xe-cli-6.00-@BUILD_NUMBER@.i386.rpm
```

要获得有关主机上的 CLI 命令的基本帮助，请键入：

```
xe help command
```

要显示最常使用的 xe 命令的列表，请键入：

```
xe help
```

或者，要显示所有 xe 命令的列表，请键入：

```
xe help --all
```

### 1. 基本 xe 语法

所有 XenServer xe CLI 命令的基本语法是：

```
xe <command-name> <argument=value> <argument=value> ...
```

每个特定命令都有自己的一组参数，参数形式为 argument=value。一些命令具有必需参数，大多数命令都多少有些可选参数。通常情况下，如果调用一个命令时未使用某些可选参数，该命令将假定这些可选参数使用默认值。

如果远程执行 xe 命令，则会用到其他连接和身份验证参数。这些参数也采用 argument=argument\_value 的形式。

server 参数用于指定主机名或 IP 地址。username 和 password 参数用于指定凭据。可以指定 password-file 参数，而不直接指定密码。在这种情况下，请尝试从指定文件中读取密码（如有必要，请去掉该文件末尾的回车符 (CR) 和换行符 (LF)）并用该密码进行连接。与直接在命令行指定密码相比，这样做更安全。

可选 port 参数可用于指定远程 XenServer 主机上的代理端口（默认为 443）。

示例：在本地 XenServer 主机上：

```
xe vm-list
```

示例：在远程 XenServer 主机上：

```
xe vm-list -user <username> -password <password> -server <hostname>
```

速记语法也适用于远程连接参数：

-u	username
-pw	password
-pwf	password file
-p	port
-s	server

示例：在远程 XenServer 主机上：

```
xe vm-list -u <myuser> -pw <mypassword> -s <hostname>
```

也可以通过采用逗号分隔键/值对形式的环境变量 `XE_EXTRA_ARGS` 设置参数。例如，为了在远程 XenServer 主机上运行的一个 XenServer 主机中输入命令，可以执行以下操作：

```
export XE_EXTRA_ARGS="server=jeffbeck,port=443,username=root,password=pass"
```

此命令意味着您不再需要在执行的每个 `xe` 命令中指定远程 XenServer 主机参数。

使用 `XE_EXTRA_ARGS` 环境变量也可以在对远程 XenServer 主机执行 `xe` 命令时对命令启用 Tab 键自动补齐功能，该功能在默认情况下处于禁用状态。

## 2. 特殊字符和语法

要在 `xe` 命令行中指定参数/值对，请写入：

```
argument=value
```

除非值中包含空格，否则请勿使用引号。参数名称、等号 (=) 和值之间不应存在任何空格。不符合此格式的参数都将被忽略。

对于包含空格的值，请写入：

```
argument="value with spaces"
```

如果使用 CLI 登录到 XenServer 主机，则命令将具有 Tab 键自动补齐功能，该功能类似于标准 Linux Bash Shell 中的 Tab 键自动补齐功能。如果您键入的内容可以确定一个唯一的命令，例如 `xe vm-l`，然后按 TAB 键，则该命令的其余部分也将显示出来。如果多个命令都以 `vm-l` 开头，则再次按 Tab 将列出所有可能的命令。这对在命令中指定对象 UUID 特别有用。



### 注意

如果在远程 XenServer 主机上执行命令，Tab 键自动补齐功能将无法正常运行。但是，如果将服务器、用户名和密码置于计算机上名为 `XE_EXTRA_ARGS` 的环境变量中，并从该计算机输入命令，则可以启用 Tab 键自动补齐。有关详细信息，请参阅第 1 节“基本 `xe` 语法”。

## 3. 命令类型

一般而言，CLI 命令可以分为两种：低级命令和高级命令，前一种命令侧重于 API 对象的列表和参数操作，后一种命令用于在更加抽象的级别与 VM 或主机交互。低级命令包括：

- `<class>-list`
- `<class>-param-get`
- `<class>-param-set`



- `<class>-param-list`
- `<class>-param-add`
- `<class>-param-remove`
- `<class>-param-clear`

其中 `<class>` 可以是以下任一项：

- bond
- console
- host
- host-crashdump
- host-cpu
- network
- patch
- pbd
- pif
- pool
- sm
- sr
- task
- template
- vbd
- vdi
- vif
- vlan
- vm

请注意，并不是 `<class>` 的每个值都具有完整的 `<class>-param-` 命令集；其中一些值只具有子集。

### 3.1. 参数类型

使用 `xe` 命令处理的对象具有标识这些对象并定义其状态的参数集。

大多数参数只使用单一值。例如，VM 的 `name-label` 参数包含单一字符串值。在参数列表命令（如 `xe vm-param-list`）的输出中，这样的参数都有括在括号中的指示，用来定义它们为可读和可写，还是为只读。例如，指定 VM 上的 `xe vm-param-list` 输出可能包含下列行：

```
user-version ( RW): 1
is-control-domain ( RO): false
```

第一个参数 `user-version` 为写入参数，值为 1。第二个参数 `is-control-domain` 为只读参数，值为 `false`。

其他两个类型的参数都是多值参数。`set` 参数包含一系列值。`map` 参数是一组键/值对。例如，请参阅下面指定 VM 上的 `xe vm-param-list` 输出示例的摘录：

```
platform (MRW): acpi: true; apic: true; pae: true; nx: false
allowed-operations (SRO): pause; clean_shutdown; clean_reboot; \
hard_shutdown; hard_reboot; suspend
```

platform 参数包含表示键/值对的一系列项。键名后跟冒号字符 (:)。各键/值对之间用分号字符 (;) 分隔。RW 前面的 M 表示这是一个 map 参数，既可读又可写。allowed-operations 参数包含构成一组项的列表。RO 前面的 S 表示这是一个 set 参数，可读但不可写。

在 xe 命令中，如果您希望按 map 参数过滤或者设置 map 参数，请在 map 参数名称与键/值对之间使用：( 冒号 ) 分隔符。例如，要将 VM 的 other-config 参数的 foo 键值设置为 baa，使用的命令应为：

```
xe vm-param-set uuid=<VM uuid> other-config.foo=baa
```



### 注意

在以前的版本中，短划线 ( - ) 分隔符用于指定 map 参数。此语法现在仍然有效，但不建议使用。

## 3.2. 低级参数命令

有多个命令可在对象参数上运行：`<class>-param-get`、`<class>-param-set`、`<class>-param-add`、`<class>-param-remove`、`<class>-param-clear` 和 `<class>-param-list`。其中的每个命令都利用 uuid 参数指定特殊对象。这些命令都是低级命令，必须由 UUID 而不是 VM 名称标签处理。

`<class>-param-list uuid=<uuid>`

列出所有参数及其相关的值。与 class-list 命令不同，此命令将列出相当多的字段的值。

`<class>-param-get uuid=<uuid> param-name=<parameter> [param-key=<key>]`

返回特殊参数的值。如果此参数为 map，则指定 param-key 将使该值与 map 中的键关联。如果不指定 param-key，或者该参数为 set，则它将返回 set 或 map 的字符串表示形式。

`<class>-param-set uuid=<uuid> param=<value>...`

设置一个或多个参数的值。

`<class>-param-add uuid=<uuid> param-name=<parameter> [<key>=<value>...] [param-key=<key>]`

添加到 map 或 set 参数。如果该参数为 map，则使用 `<key>=<value>` 语法添加键/值对。如果该参数为 set，则使用 `<param-key>=<key>` 语法添加键。

`<class>-param-remove uuid=<uuid> param-name=<parameter> param-key=<key>`

从 map 删除键/值对，或者从 set 删除键。

`<class>-param-clear uuid=<uuid> param-name=<parameter>`

完全清除 set 或 map。

## 3.3. 低级列表命令

`<class>-list` 命令用于列出 `<class>` 类型的对象。默认情况下，它将列出所有对象，并列出参数的子集。可以通过两种方法修改此行为：该命令可过滤对象以便仅输出子集，并且可以修改列出的参数。

要更改列出的参数，应将参数 params 指定为所需参数的以逗号分隔的列表。例如：

```
xe vm-list params=name-label,other-config
```

或者，要列出所有参数，请使用以下语法：

```
xe vm-list params=all
```

请注意，某些需要进行大量计算的参数不会通过列表命令显示出来。例如，这些参数将显示为：

```
allowed-VBD-devices (SRO): <expensive field>
```

要获取这些字段，请使用 `<class>-param-list` 或 `<class>-param-get`





要过滤列表，CLI 需要将参数值与在命令行上指定的参数值相匹配，以便只列出满足所有指定的限制的对象。例如：

```
xe vm-list HVM-boot-policy="BIOS order" power-state=halted
```

此命令将只列出满足以下条件的 VM：power-state 和 HVM-boot-policy 这两个字段的值分别为 halted 和 BIOS order。

还可以根据 map 中的键值或 set 中存在的值来过滤列表。其中的第一个值的语法是 map-name:key=value，第二个值的语法是 set-name:contains=value

对于脚本来说，一种有用的技术是在命令行上传递 --minimal，以使 xe 只列出以逗号分隔的列表中的第一个字段。举例来说，在安装有三个 VM 的 XenServer 主机上，xe vm-list --minimal 命令为这些 VM 提供了三个 UUID，例如：

```
a85d6717-7264-d00e-069b-3b1d19d56ad9,aaa3eec5-9499-bcf3-4c03-af10baea96b7,\  
42c044de-df69-4b30-89d9-2c199564581d
```

## 4. xe 命令参考

本部分提供 xe 命令参考。参考按 xe 命令处理的对象分组，并按字母顺序列出。

### 4.1. 设备命令

用来创建和修改 VM 设备（又称 vApp）的命令。有关 vApp 的详细信息，请参阅《XenServer 虚拟机用户指南》。

#### 4.1.1. 设备参数

设备命令具有下列参数：

参数名称	说明	类型
uuid	设备 uuid	必需
name-description	设备说明	可选
paused		可选
force	强制关闭	可选

#### 4.1.2. appliance-assert-can-be-recovered

```
appliance-assert-can-be-recovered uuid=<appliance-uuid> database:vdi-uuid=<vdi-uuid>
```

测试存储是否可用于恢复此 VM 设备/vApp。

#### 4.1.3. appliance-create

```
appliance-create name-label=<name-label> [name-description=<name-description>]
```

创建设备/vApp。例如：

```
xe appliance-create name-label=my_appliance
```

向设备中添加 VM：

```
xe vm-param-set uuid=<VM-UUID> appliance=<appliance-uuid> \
xe vm-param-set uuid=<VM-UUID> appliance=<appliance-uuid>
```

#### 4.1.4. appliance-destroy

```
appliance-destroy uuid=<appliance-uuid>
```

销毁设备/vApp。例如：

```
xe appliance-destroy uuid=<appliance-uuid>
```

#### 4.1.5. appliance-recover

```
appliance-recover uuid=<appliance-uuid> database:vdi-uuid=<vdi-uuid> [paused=<true|false>]
```

从提供的 VDI 中包含的数据库恢复 VM 设备/vAPP。

#### 4.1.6. appliance-shutdown

```
appliance-shutdown uuid=<appliance-uuid> [force=<true|false>]
```

关闭设备/vApp 中的所有 VM。例如：

```
xe appliance-shutdown uuid=<appliance-uuid>
```

#### 4.1.7. appliance-start

```
appliance-start uuid=<appliance-uuid> [paused=<true|false>]
```

启动设备/vApp。例如：

```
xe appliance-start uuid=<appliance-uuid>
```

### 4.2. 审核命令

审核命令下载池中 RBAC 审核文件的所有可用记录。如果可选参数 since 存在，该命令将仅下载自该特定时间点开始的记录。

#### 4.2.1. audit-log-get 参数

audit-log-get 具有下列参数：

参数名称	说明	类型
filename	将池的审核日志写入 <filename>	必需
since	特定的日期/时间点	可选

#### 4.2.2. audit-log-get

```
audit-log-get [since=<timestamp>] filename=<filename>
```

例如，要获取自某个精确的毫秒时间戳以来的池审核记录，请运行以下命令：

运行以下命令：

```
xe audit-log-get since=2009-09-24T17:56:20.530Z \
filename=/tmp/auditlog-pool-actions.out
```

### 4.3. 绑定命令

与网络绑定一起使用的命令，用于提供物理接口故障转移恢复能力。有关详细信息，请参阅第 4.4.5 节“在独立主机上创建 NIC 绑定”。

绑定对象是将主节点和成员 PIF 粘附在一起的引用对象。主节点 PIF 是必须用作整体 PIF 以引用绑定的绑定接口。成员 PIF 是组合成高级绑定接口的两个或更多个物理接口的集合。

#### 4.3.1. 绑定参数

绑定具有下列参数：

参数名称	说明	类型
uuid	绑定的唯一标识符/对象引用	只读
master	主节点绑定 PIF 的 UUID	只读
members	基础绑定 PIF 的 UUID 集	只读 set 参数

#### 4.3.2. bond-create

`bond-create network-uuid=<network_uuid> pif-uuids=<pif_uuid_1,pif_uuid_2,...>`

在通过现有 PIF 对象列表指定的网络上创建绑定网络接口。在以下情况下，该命令将失败：PIF 已位于另一绑定、其中的某个成员具有 VLAN 标记集、引用的 PIF 不在同一 XenServer 主机上，或者提供的 PIF 的数量低于两个。

#### 4.3.3. bond-destroy

`host-bond-destroy uuid=<bond_uuid>`

从 XenServer 主机删除由绑定接口的 UUID 指定的相应绑定接口。

### 4.4. CD 命令

与 XenServer 主机上的物理 CD/DVD 驱动器一起使用的命令。

#### 4.4.1. CD 参数

CD 具有下列参数：

参数名称	说明	类型
uuid	CD 的唯一标识符/对象引用	只读
name-label	CD 的名称	读取/写入
name-description	CD 的说明文本	读取/写入
allowed-operations	可对此 CD 执行的操作的列表	只读 set 参数
current-operations	当前正对此 CD 执行的操作的列表	只读 set 参数
sr-uuid	此 CD 所属的 SR 的唯一标识符/对象引用	只读
sr-name-label	此 CD 所属的 SR 的名称	只读

参数名称	说明	类型
vbd-uuids	连接到此 CD 的 VM 上的 VBD 的唯一标识符列表	只读 set 参数
crashdump-uuids	故障转储无法写入 CD，因而不能在 CD 上使用	只读 set 参数
virtual-size	CD 在 VM 中显示时的大小，以字节为单位	只读
physical-utilisation	CD 映像当前在 SR 上占用的物理空间大小，以字节为单位	只读
type	对 CD，将此参数设置为 User	只读
sharable	CD 驱动器是否可共享。默认为 false。	只读
read-only	CD 是否只读，如果为 false，则设备可写入。对 CD 始终为 true。	只读
storage-lock	如果在存储级别锁定此磁盘，则为 true	只读
parent	如果此 CD 是链的一部分，则引用父磁盘	只读
missing	如果 SR 扫描操作报告此 CD 未在磁盘上显示，则为 true	只读
other-config	为 CD 指定其他配置参数的键/值对列表	读取/写入 map 参数
location	装载设备的路径	只读
managed	如果设备是托管的，则为 true	只读
xenstore-data	插入到 xenstore 树中的数据	只读 map 参数
sm-config	存储管理器设备配置键的名称和描述	只读 map 参数
is-a-snapshot	如果此模板为 CD 快照，则为 true	只读
snapshot_of	此模板属于其快照的 CD 的 UUID	只读
snapshots	根据此 CD 生成的任何快照的 UUID	只读
snapshot_time	快照操作的时间戳	只读

#### 4.4.2. cd-list

cd-list [params=<param1,param2,...>] [parameter=<parameter\_value>...]

列出 XenServer 主机或池上的 CD 和 ISO（CD 映像文件），按可选参数 params 过滤。

如果使用可选参数 params，则 params 的值是一个字符串，其中包含您希望显示的此对象的参数列表。或者，可以使用关键字 all 显示所有参数。如果不使用 params，则返回的列表将显示所有可用参数的默认子集。

可选参数可以为 [CD 参数](#) 部分开头列出的参数中的任意数个。

## 4.5. 控制台命令

与控制台一起使用的命令。

可以使用标准对象列表命令 (xe console-list) 和标准参数命令控制的参数列出控制台对象。有关详细信息，请参阅 [第 3.2 节“低级参数命令”](#)。

### 4.5.1. 控制台参数

控制台具有下列参数：

参数名称	说明	类型
uuid	控制台的唯一标识符/对象引用	只读
vm-uuid	打开控制台的 VM 的唯一标识符/对象引用	只读
vm-name-label	打开控制台的 VM 的名称	只读
protocol	此控制台使用的协议。可能值为 vt100：VT100 终端、rfb：远程帧终端协议（用于 VNC）、或者 rdp：远程桌面协议	只读
location	控制台服务的 URI	只读
other-config	用于指定控制台其他配置参数的键/值对的列表。	读取/写入 map 参数

## 4.6. 灾难恢复 (DR) 命令

用来在发生灾难时恢复 VM 的命令

### 4.6.1. drtask-create

`drtask-create type=<type> sr-whitelist=<sr-white-list> device-config=<device-config>`

创建灾难恢复任务。例如，要连接到 iSCSI SR 以准备进行灾难恢复，请运行以下命令：

```
xe dr-task-create type=lvmoiscsi device-config:target=<target-ip-address> \
device-config:targetIQN=<targetIQN> device-config:SCSIid=<SCSIid> \
sr-whitelist=<sr-uuid-list>
```



#### 注意

sr-whitelist 将列出 SR UUID，drtask-create 将只引入并连接到具有白名单上所列某个 UUID 的 SR

### 4.6.2. drtask-destroy

`drtask-destroy uuid=<dr-task-uuid>`

销毁灾难恢复任务并忘记所引入的 SR。

#### 4.6.3. vm-assert-can-be-recovered

vm-assert-can-be-recovered uuid=<vm-uuid> database:vdi-uuid=<vdi-uuid>

测试存储是否可用于恢复此 VM。

#### 4.6.4. appliance-assert-can-be-recovered

appliance-assert-can-be-recovered uuid=<appliance-uuid> database:vdi-uuid=<vdi-uuid>

检查包含设备/vAPP 磁盘的存储是否可见。

#### 4.6.5. appliance-recover

appliance-recover uuid=<appliance-uuid> database:vdi-uuid=<vdi-uuid> [force=<true|false>]

从提供的 VDI 中包含的数据库恢复设备/vAPP。

#### 4.6.6. vm-recover

vm-recover uuid=<vm-uuid> database:vdi-uuid=<vdi-uuid> [force=<true|false>]

从提供的 VDI 中包含的数据库恢复 VM。

#### 4.6.7. sr-enable-database-replication

sr-enable-database-replication uuid=<sr\_uuid>

启用以指定 ( 共享 ) SR 为目标的 xapi 数据库复制。例如：

```
xe sr-enable-database-replication uuid=<sr-uuid>
```

#### 4.6.8. sr-disable-database-replication

sr-disable-database-replication uuid=<sr\_uuid>

禁用以指定 SR 为目标的 xapi 数据库复制。例如：

```
xe sr-enable-database-replication uuid=<sr-uuid>
```

#### 4.6.9. 示例用法

以下示例显示上下文中的 DR CLI 命令：

在主站点上，启用数据库复制：

```
xe sr-database-replication uuid=<sr=uuid>
```

发生灾难时，在辅助站点上连接到 SR ( 请注意 device-config 与 sr-probe 具有相同的字段 )：

```
xe drtask-create type=lvmoiscsi \
  device-config:target=<target ip address> \
  device-config:targetIQN=<target-iqn> \
  device-config:SCSIid=<scsi-id> \
  sr-whitelist=<sr-uuid>
```



在 SR 上查找数据库 VDI :

```
xe vdi-list sr-uuid=<sr-uuid> type=Metadata
```

在数据库 VDI 中查询存在的 VM :

```
xe vm-list database:vdi-uuid=<vdi-uuid>
```

恢复 VM :

```
xe vm-recover uuid=<vm-uuid> database:vdi-uuid=<vdi-uuid>
```

销毁 DR 任务；由 DR 任务引入但并非 VM 所需的任何 SR 都将被销毁：

```
xe drtask-destroy uuid=<drtask-uuid>
```

## 4.7. 事件命令

与事件一起使用的命令。

### 4.7.1. 事件类

下表列出了事件类：

类名称	说明
pool	物理主机的池
vm	虚拟机
host	物理主机
network	虚拟网络
vif	虚拟网络接口
pif	物理网络接口（将独立的 VLAN 表示为多个 PIF）
sr	存储库
vdi	虚拟磁盘映像
vbd	虚拟块设备
pbd	物理块设备，主机可通过这些设备访问 SR

### 4.7.2. event-wait

```
event-wait class=<class_name> [<param-name>=<param_value>] [<param-name>!=<param_value>]
```

阻止其他命令执行，直到存在符合命令行上给出的条件的对象为止。x=y 表示“等待字段 x 取值 y”，x!=y 表示“等待字段 x 取除 y 以外的任何值”。

示例：等待特定 VM 运行。



```
xe event-wait class=vm name-label=myvm power-state=running
```

在名为 myvm 的 VM 的 power-state 为“running”之前，将一直阻止其他命令。

示例：等待特定 VM 重新引导：

```
xe event-wait class=vm uuid=$VM start-time!=$(xe vm-list uuid=$VM params=start-time --minimal)
```

在 UUID 为 \$VM 的 VM 重新引导（即，具有不同的 start-time 值）之前，阻止执行其他命令。

类名称可以是在本部分开头列出的[事件类](#)中的任何类，参数可以是在 CLI 命令 class-param-list 中列出的任何参数。

## 4.8. GPU 命令

用来处理物理 GPU、GPU 组和虚拟 GPU 的命令。

可以使用标准对象列表命令（xe pgpu-list、xe gpu-group-list 和 xe vgpu-list）及由标准参数命令处理的参数来列出 GPU 对象。有关详细信息，请参阅[第 3.2 节“低级参数命令”](#)。

### 4.8.1. 物理 GPU (pGPU) 参数

pGPUs 具有下列参数：

参数名称	说明	类型
uuid	pGPU 的唯一标识符/对象引用	只读
vendor-name	pGPU 的供应商名称	只读
device-name	供应商分配给此 pGPU 型号的名称	只读
gpu-group-uuid	XenServer 已自动将此 pGPU 分配到的 GPU 组的唯一标识符/对象引用；池中各主机上相同的 pGPU 将组合到一起	只读
gpu-group-name-label	将 pGPU 分配到的 GPU 组的名称	只读
host-uuid	此 pGPU 连接到的 XenServer 主机的唯一标识符/对象引用	只读
host-name-label	此 pGPU 连接到的 XenServer 主机的名称	只读
pci-id	PCI 标识符	只读
dependencies	列出传递到同一个 VM 的相关 PCI 设备	读取/写入 map 参数
other-config	为 pGPU 指定其他配置参数的键/值对列表	读取/写入 map 参数

参数名称	说明	类型
supported-VGPU-types	底层硬件支持的虚拟 GPU 类型的列表	只读
enabled-VGPU-types	已为此 pGPU 启用的虚拟 GPU 类型的列表	读取/写入
resident-VGPUs	在此 pGPU 上运行的 vGPU 的列表	只读

#### 4.8.2. pGPU 操作

用于 pGPU 组的命令

##### 4.8.2.1. pgpu-param-set

```
pgpu-param-set uuid=<uuid_of_pgpu> gpu-group-uuid=<uuid_of_destination_group>
```

将 pGPU 移动到不同的 GPU 组。如果正在使用驻留在目标组上的虚拟 GPU 运行 VM，则不允许命令。

##### 4.8.2.2. pgpu-param-get-uuid

```
pgpu-param-get-uuid uuid=<uuid_of_pgpu> param-name=<supported-vGPU-types|enabled-vGPU-types>
```

显示此 pGPU 的受支持或已启用虚拟 GPU 类型。

##### 4.8.2.3. pgpu-param-set-uuid

```
pgpu-param-set-uuid          uuid=<uuid_of_pgpu>          enabled_VGPU-
types=<comand_separated_list_of_vgpu_type_uuids>
```

更改为此类型 pGPU 启用的虚拟 GPU 类型组。

##### 4.8.2.4. pgpu-param-add-uuid

```
pgpu-param-add-uuid          uuid=<uuid_of_pgpu>          param-name=<enabled_vgpu_types>          param-
key=<uuid_of_vgpu>
```

在此 pGPU 上启用虚拟 GPU。

#### 4.8.3. GPU 组参数

GPU 组具有下列参数：

参数名称	说明	类型
uuid	GPU 组的唯一标识符/对象引用	只读
name-label	GPU 组的名称	读取/写入
name-description	GPU 组的描述性文本	读取/写入

参数名称	说明	类型
VGPU-uuids	列出 GPU 组中虚拟 GPU 的唯一标识符/对象引用	只读 set 参数
PGPU-uuids	列出 GPU 组中 pGPU 的唯一标识符/对象引用	只读 set 参数
other-config	键/值对列表，用于指定 GPU 组的其他配置参数。	读取/写入 map 参数
supported-VGPU-types	底层硬件支持的所有虚拟 GPU 类型的联合体	只读
enabled-VGPU-types	已在底层 pGPU 上启用的所有虚拟 GPU 类型的联合体	只读
allocation-algorithm	组中 pGPU 上分配虚拟 GPU 的 Depth-first/Breadth-first 设置	读取/写入 enum 参数

#### 4.8.3.1. GPU 组操作

用于 GPU 组的命令

##### 4.8.3.1.1. gpu-group-create

```
gpu-group-create name-label=<name_for_group> [name-description=<description>]
```

创建一个可将 pGPU 迁入的新 ( 空 ) GPU 组。

##### 4.8.3.1.2. gpu-group-destroy

```
gpu-group-destroy uuid=<uuid_of_group>
```

销毁 GPU 组；仅允许空组。

##### 4.8.3.1.3. gpu-group-get-remaining-capacity

```
gpu-group-get-remaining-capacity uuid=<uuid_of_group> vgpu-type-uuid=<uuid_of_vgpu_type>
```

返回可在此 GPU 组中实例化的指定类型的虚拟 GPU 新增数量。

##### 4.8.3.1.4. gpu-group-param-set

```
gpu-group-param-set uuid=<uuid_of_group> allocation-algorithm=<breadth-first|depth-first>
```

更改 GPU 组的分配算法，将虚拟 GPU 分配给 pGPUS。

##### 4.8.3.1.5. gpu-group-param-get-uuid

```
gpu-group-param-get-uuid uuid=<uuid_of_group> param-name=<supported-vGPU-types|enabled-vGPU-types>
```

返回此 GPU 组的受支持或已启用类型。

#### 4.8.4. 虚拟 GPU 参数

虚拟 GPU 具有以下参数：

参数名称	说明	类型
uuid	虚拟 GPU 的唯一标识符/对象引用	只读
vm-uuid	将虚拟 GPU 分配到的 VM 的唯一标识符/对象引用	只读
vm-name-label	将虚拟 GPU 分配到的 VM 的名称	只读
gpu-group-uuid	虚拟 GPU 所属的 GPU 组的唯一标识符/对象引用	只读
gpu-group-name-label	虚拟 GPU 所属的 GPU 组的名称	只读
currently-attached	如果具有 GPU 传递功能的 VM 正在运行，则为 True，否则为 false	只读
other-config	为虚拟 GPU 指定其他配置参数的键/值对列表	读取/写入 map 参数
type-uuid	此虚拟 GPU 的虚拟 GPU 类型的唯一标识符/对象引用	读取/写入 map 参数
type-model-name	与虚拟 GPU 类型相关联的模型名称	只读

#### 4.8.5. 虚拟 GPU 类型参数



##### 注意

虚拟 GPU 和 GPU 直通功能与 XenMotion、Storage XenMotion 或 VM Suspend 不兼容。但是，使用 GPU 直通或虚拟 GPU 的 VM 依然可以从任何拥有相应资源的主机启动。

虚拟 GPU 类型具有以下参数：

参数名称	说明	类型
uuid	虚拟 GPU 类型的唯一标识符/对象引用	只读
vendor-name	虚拟 GPU 供应商的名称	只读
model-name	与虚拟 GPU 类型相关联的模型名称	只读

参数名称	说明	类型
freeze-frame	虚拟 GPU 类型的 Framebuffer 大小 ( 以字节为单位 )	只读
max-heads	虚拟 GPU 类型支持的最大显示数量	只读
supported-on-PGPUs	支持此虚拟 GPU 类型的 pGPU 的列表	只读
enabled-on-PGPUs	启用了此虚拟 GPU 类型的 pGPU 的列表	只读
VGPU-uuids	此类型的虚拟 GPU 的列表	只读

#### 4.8.6. 虚拟 GPU 操作

##### 4.8.6.1. vgpu-create

```
vgpu-create vm-uuid=<uuid_of_vm> gpu_group_uuid=<uuid_of_gpu_group> [vgpu-type-uuid=<uuid_of_vgpu-type>]
```

创建虚拟 GPU。此命令将 VM 连接到指定的 GPU 组并根据需要指定虚拟 GPU 类型。如果未指定虚拟 GPU 类型，则假设使用“直通”类型。

##### 4.8.6.2. vgpu-destroy

```
vgpu-destroy uuid=<uuid_of_vgpu>
```

销毁指定的虚拟 GPU。

##### 4.8.6.3. 为使用虚拟 GPU 的 VM 禁用 VNC

```
xe vm-param-add uuid=<uuid_of_vm>param-name=platform vgpu_vnc_enabled=<true|false.>
```

使用 false 禁用 VM 的 VNC 控制台，因为它可以将 disablevnc=1 传递到显示模拟器。默认情况下，启用 VNC。

#### 4.9. 主机命令

与 XenServer 主机交互的命令。

XenServer 主机是运行 XenServer 软件的物理服务器。在这些主机上运行的 VM 受具有特权的虚拟机 ( 称为控制域或域 0 ) 的控制。

可以使用标准对象列表命令 ( xe host-list、xe host-cpu-list 以及 xe host-crashdump-list ) 和标准参数命令控制的参数列出 XenServer 主机对象。有关详细信息，请参阅第 3.2 节“低级参数命令”。

##### 4.9.1. 主机选择器

此处列出的多个命令具有用于选择在其上执行操作的一个或多个 XenServer 主机的通用机制。最简单的方式是提供参数 host=<uuid\_or\_name\_label>。也可以通过按字段值过滤主机的完整列表，也可以指定

XenServer 主机。例如，指定 `enabled=true` 可选择 `enabled` 字段等于 `true` 的所有 XenServer 主机。如果有多个匹配的 XenServer 主机并且操作可以在多个 XenServer 主机上执行，则必须指定选项 `--multiple` 才能执行操作。本部分的开头介绍了可以匹配的参数的完整列表，通过运行 `xe host-list params=all` 命令可获得该列表。如果未提供选择 XenServer 主机的参数，将在所有 XenServer 主机上执行操作。

#### 4.9.2. 主机参数

XenServer 主机具有下列参数：

参数名称	说明	类型
<code>uuid</code>	XenServer 主机的唯一标识符/对象引用	只读
<code>name-label</code>	XenServer 主机的名称	读取/写入
<code>name-description</code>	XenServer 主机的描述字符串	只读
<code>enabled</code>	如果禁用了主机（这将阻止任何新 VM 在其上启动，准备使 XenServer 主机关闭或重新引导），则为 <code>false</code> ；如果当前启用了主机，则为 <code>true</code>	只读
<code>API-version-major</code>	主版本号	只读
<code>API-version-minor</code>	次版本号	只读
<code>API-version-vendor</code>	API 供应商的标识	只读
<code>API-version-vendor-implementation</code>	供应商实现的详细信息	只读 map 参数
<code>logging</code>	日志记录配置	读取/写入 map 参数
<code>suspend-image-sr-uuid</code>	挂起的映像所在 SR 的唯一标识符/对象引用	读取/写入
<code>crash-dump-sr-uuid</code>	故障转储所在的 SR 的唯一标识符/对象引用	读取/写入
<code>software-version</code>	版本控制参数及其值的列表	只读 map 参数
<code>capabilities</code>	XenServer 主机可以运行的 Xen 版本列表	只读 set 参数
<code>other-config</code>	为 XenServer 主机指定其他配置参数的键/值对列表	读取/写入 map 参数
<code>chipset-info</code>	用来指定芯片组相关信息的键/值对列表	只读 map 参数
<code>hostname</code>	XenServer 主机名	只读
<code>address</code>	XenServer 主机 IP 地址	只读

参数名称	说明	类型
license-server	<ul style="list-style-type: none"> <li>用来指定许可证服务器相关信息的键/值对列表</li> <li>与 Citrix 产品通信使用的默认端口为 27000。有关在发生冲突时更改端口号的信息，请参阅 <a href="#">Citrix 产品文档</a> Web 站点上的<a href="#">更改端口号</a>主题。</li> </ul>	只读 map 参数
supported-bootloaders	XenServer 主机支持的引导加载器列表，例如 pygrub、eliloader	只读 set 参数
memory-total	XenServer 主机上的物理 RAM 总量（以字节为单位）	只读
memory-free	可分配给 VM 的剩余物理 RAM 总量（以字节为单位）	只读
host-metrics-live	如果主机可以正常运行，则为 true	只读
logging	syslog_destination 键可设置为远程侦听 syslog 服务的主机名。	读取/写入 map 参数
allowed-operations	此种状态下允许执行的操作列表。此列表只是建议性的，并且客户端读取此字段时服务器状态可能已更改	只读 set 参数
current-operations	当前正在执行的操作列表。此列表只是建议性的，并且客户端读取此字段时服务器状态可能已更改	只读 set 参数
patches	主机修补程序集	只读 set 参数
blobs	二进制数据存储	只读
memory-free-computed	对主机上最大可用内存量的保守估算	只读
ha-statefiles	所有高可用性功能状态文件的 UUID	只读
ha-network-peers	出现故障时可承载此主机上 VM 的所有主机的 UUID	只读
external-auth-type	外部身份验证类型，例如 Active Directory。	只读

参数名称	说明	类型
external-auth-service-name	外部身份验证服务的名称	只读
external-auth-configuration	外部身份验证服务的配置信息。	只读 map 参数

XenServer 主机包含的一些其他对象也具有参数列表。

XenServer 主机上的 CPU 具有下列参数：

参数名称	说明	类型
uuid	CPU 的唯一标识符/对象引用	只读
number	XenServer 主机中的物理 CPU 内核数量	只读
vendor	表示 CPU 名称的供应商字符串，例如，“GenuineIntel”	只读
speed	CPU 时钟速度（以赫兹为单位）	只读
modelName	表示 CPU 型号的供应商字符串，例如，“Intel(R) Xeon(TM) CPU 3.00GHz”	只读
stepping	CPU 修订号	只读
flags	物理 CPU 的标志（功能字段的解码版本）	只读
Utilisation	当前 CPU 利用率	只读
host-uuid	CPU 所在主机的 UUID	只读
model	物理 CPU 的型号	只读
family	物理 CPU 系列号	只读

XenServer 主机上的故障转储具有下列参数：

参数名称	说明	类型
uuid	故障转储的唯一标识符/对象引用	只读
host	故障转储对应的 XenServer 主机	只读
timestamp	故障转储发生日期和时间的戳，格式为 yyyyymmdd-hhmmss-ABC，其中 ABC 是时区指示器，例如 GMT	只读
size	故障转储的大小（以字节为单位）	只读

#### 4.9.3. host-backup

host-backup file-name=<backup\_filename> host=<host\_name>



将指定的 XenServer 主机的控制域备份下载到调用此命令的计算机，然后将其作为名为 file-name 的文件保存在该计算机上。



小心

xe host-backup 命令在本地主机上执行（即不指定特定主机名）时可以起作用，但是不要以这种方式使用此命令。这样做将使用备份文件填充控制域分区。应该仅从保留了存储备份文件空间的远程主机外计算机上使用此命令。

#### 4.9.4. host-bugreport-upload

```
host-bugreport-upload [<host-selector>=<host_selector_value>...] [url=<destination_url>]
[http-proxy=<http_proxy_name>]
```

生成全新错误报告（使用 xen-bugtool，包括所有可选文件）并上载到 Citrix 支持 ftp 站点或某个其他位置。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

可选参数是 http-proxy：使用特定的 http 代理以及 url：上载到此目标 URL。如果没有使用可选参数，将不确定代理服务器，并且目标位置为默认的 Citrix 支持 ftp 站点。

#### 4.9.5. host-crashdump-destroy

```
host-crashdump-destroy uuid=<crashdump_uuid>
```

从 XenServer 主机删除通过 UUID 指定的主机故障转储。

#### 4.9.6. host-crashdump-upload

```
host-crashdump-upload uuid=<crashdump_uuid>
[url=<destination_url>]
[http-proxy=<http_proxy_name>]
```

将故障转储上载到 Citrix 支持 ftp 站点或其他位置。如果没有使用可选参数，将不确定代理服务器，并且目标位置为默认的 Citrix 支持 ftp 站点。可选参数是 http-proxy：使用特定的 http 代理以及 url：上载到此目标 URL。

#### 4.9.7. host-disable

```
host-disable [<host-selector>=<host_selector_value>...]
```

禁用指定的 XenServer 主机，防止在这些主机上启动任何新 VM。这样做可以为关闭或重新引导 XenServer 主机做好准备。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

#### 4.9.8. host-dmesg

```
host-dmesg [<host-selector>=<host_selector_value>...]
```

从指定的 XenServer 主机中获取 Xen dmesg（内核环缓冲的输出）。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

#### 4.9.9. host-emergency-management-reconfigure

host-emergency-management-reconfigure interface=<uuid\_of\_management\_interface\_pif>

重新配置此 XenServer 主机的管理接口。仅在 XenServer 主机处于紧急模式时使用此命令，紧急模式是指此主机所属资源池的主服务器从网络中消失，经过数次重新尝试后仍无法连接。

#### 4.9.10. host-enable

host-enable [<host-selector>=<host\_selector\_value>...]

启用指定的 XenServer 主机，以允许在这些主机上启动新 VM。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

#### 4.9.11. host-evacuate

host-evacuate [<host-selector>=<host\_selector\_value>...]

将所有正在运行的 VM 实时迁移到池上的其他适合主机。必须首先使用 host-disable 命令禁用主机。

如果退出的主机是池主服务器，那么必须选择其他主机作为池主服务器。若要在高可用性功能禁用时更改池主服务器，需要使用 pool-designate-new-master 命令。有关详细信息，请参阅第 4.15.2 节“pool-designate-new-master”。启用高可用性功能时，只能选择关闭该服务器，这将导致高可用性功能随机选择一个新主服务器。请参阅第 4.9.29 节“host-shutdown”。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

#### 4.9.12. host-forget

host-forget uuid=<XenServer\_host\_UUID>

xapi 代理忽略指定的 XenServer 主机，不与其进行显式连接。

使用 --force 参数，避免提示您确认是否确实想要执行该操作。



#### 警告

如果此池中启用了高可用性功能，请不要使用此命令。首先禁用高可用性功能，然后在您忽略了该主机后再重新启用它。



#### 提示

如果要忽略的 XenServer 主机不起作用，则此命令非常有用；但是，如果 XenServer 主机起作用并且是池的一部分，则应使用 xe pool-eject 命令。

#### 4.9.13. host-get-system-status

host-get-system-status filename=<name\_for\_status\_file>

[entries=<comma\_separated\_list>] [output=<tar.bz2 | zip>] [<host-selector>=<host\_selector\_value>...]

将系统状态信息下载到指定文件。可选参数 entries 是系统状态条目的逗号分隔列表，这些条目从 host-get-system-status-capabilities 命令返回的功能 XML 片段获取。有关详细信息，请参阅第 4.9.14 节 “host-get-system-status-capabilities”。如果未指定，则所有系统状态信息都保存在此文件中。参数 output 可以是 tar.bz2（默认）或 zip；如果未指定此参数，则文件使用 tar.bz2 格式保存。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。

#### 4.9.14. host-get-system-status-capabilities

host-get-system-status-capabilities [<host-selector>=<host\_selector\_value>...]

获取指定主机的系统状态功能。这些功能作为 XML 片段返回，类似如下内容：

```
<?xml version="1.0" ?> <system-status-capabilities>
  <capability content-type="text/plain" default-checked="yes" key="xenserver-logs" \
    max-size="150425200" max-time="-1" min-size="150425200" min-time="-1" \
    pii="maybe"/>
  <capability content-type="text/plain" default-checked="yes" \
    key="xenserver-install" max-size="51200" max-time="-1" min-size="10240" \
    min-time="-1" pii="maybe"/>
  ...
</system-status-capabilities>
```

每个功能条目具有很多属性。

属性	说明
key	功能的唯一标识符。
content-type	可以是 text/plain 或 application/data。指示 UI 是否可以呈现条目以供用户使用。
default-checked	可以是 yes 或 no。指定默认情况下 UI 是否应选择此条目。
min-size, max-size	指示此条目的大致大小范围（以字节为单位）。-1 指示大小并不重要。
min-time, max-time	指示收集此条目所需的大致时间范围（以秒为单位）。-1 指示时间并不重要。
pii	<p>个人识别信息。指示条目是否将包含用于标识系统所有者的信息，或网络拓扑的详细信息。存在以下几种值：</p> <ul style="list-style-type: none"> <li>no：这些条目中不包含 PII</li> <li>yes：这些条目中可能或一定包含 PII</li> <li>maybe：您可能希望审核这些条目是否包含 PII</li> <li>if_customized 如果文件没有修改，它们将不包含 PII，但由于我们鼓励编辑这些文件，所以 PII 可能已经通过这种自定义而引入。这尤其适用于控制域中的网络脚本。</li> </ul> <p>不管存在任何 PII 声明，任何错误报告都不会包括密码。</p>

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。

#### 4.9.15. host-is-in-emergency-mode

host-is-in-emergency-mode

如果 CLI 正在对话的主机当前处于紧急模式，则返回 true，否则返回 false。此 CLI 命令直接作用于从属主机，即使没有主服务器主机。

#### 4.9.16. host-apply-edition

host-apply-edition [host-uuid=<XenServer\_host\_UUID>] [edition=xenserver\_edition=<"free"><"per-socket"><"xendesktop">]

将 XenServer 许可证分配给主机服务器。分配许可证时，XenServer 会联系 Citrix 许可证服务器，并申请指定类型的许可证。如果有可用许可证，则会将该许可证从许可证服务器中签出。

对于 Citrix XenServer for XenDesktop 版本，请使用 <"xendesktop">。

对于初始许可配置，另请参阅 license-server-address 和 license-server-port。

#### 4.9.17. host-license-add

host-license-add [license-file=<path/license\_filename>] [host-uuid=<XenServer\_host\_UUID>]

对于 XenServer（免费版），用于解析本地许可证文件，并将其添加到指定的 XenServer 主机。

#### 4.9.18. host-license-view

host-license-view [host-uuid=<XenServer\_host\_UUID>]

显示 XenServer 主机许可证的内容。

#### 4.9.19. host-logs-download

host-logs-download [file-name=<logfile\_name>] [<host-selector>=<host\_selector\_value>...]

下载指定 XenServer 主机的日志副本。默认情况下副本保存在一个带时间戳的文件中，文件名为 hostname-yyyy-mm-dd T hh:mm:ssZ.tar.gz。您可以使用可选参数 file-name 指定其他文件名。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。



#### 小心

xe host-logs-download 命令在本地主机上执行（即不指定特定主机名）时可以起作用，但是不要以这种方法使用此命令。这样做将因日志副本而使控制域分区变得混乱。应该仅从保留了存储日志副本空间的远程主机外计算机上使用此命令。

#### 4.9.20. host-management-disable

host-management-disable

禁用侦听外部管理网络接口的主机代理，并断开所有连接的 API 客户端（如 XenCenter）。该命令直接作用于 CLI 连接的 XenServer 主机，并且应用到成员 XenServer 主机时不会转发到池主服务器。



#### 警告

在主机外使用此 CLI 命令时一定要格外小心，因为一旦运行此命令，将不可能通过网络远程连接控制域来重新启用它。

### 4.9.21. host-management-reconfigure

```
host-management-reconfigure [interface=<device>] | [pif-uuid=<uuid>]
```

重新配置 XenServer 主机以将指定网络接口用作其管理接口，即用于连接到 XenCenter 的接口。此命令重写 /etc/xen-source-inventory 中的 MANAGEMENT\_INTERFACE 键。

如果指定了接口（必须具有 IP 地址）的设备名称，则 XenServer 主机将立即重新绑定。这适用于正常模式和紧急模式。

如果指定了 PIF 对象的 UUID，则 XenServer 主机将确定重新绑定到自身的 IP 地址。执行此命令时一定不能处于紧急模式。



#### 警告

在除主机外的计算机上使用此 CLI 命令时需谨慎，并确保在新接口上有网络连接。首先使用 xe pif-reconfigure 建立一个连接。否则，后续 CLI 命令将到达 XenServer 主机。

### 4.9.22. host-power-on

```
host-power-on [host=<host_uuid>]
```

打开 XenServer 主机电源，并启用“主机启动”功能。在使用此命令之前，必须在主机上启用 host-set-power-on。

### 4.9.23. host-get-cpu-features

```
host-get-cpu-features {features=<pool_master_cpu_features>} [uuid=<host_uuid>]
```

打印代表主机物理 CPU 功能的十六进制值。

### 4.9.24. host-set-cpu-features

```
host-set-cpu-features {features=<pool_master_cpu_features>} [uuid=<host_uuid>]
```

尝试屏蔽主机的物理 CPU 功能，以与给定功能相匹配。如 host-get-cpu-features 命令所指定，给定字符串必须是 32 位的十六进制数（可以包含空格）。

### 4.9.25. host-set-power-on

```
host-set-power-on {host=<host uuid> {power-on-mode=<""> <"wake-on-lan"> <"iLO"> <"DRAC"> <"custom"> } | [power-on-config=<"power_on_ip"><"power_on_user"><"power_on_password_secret">]}
```



用于在与远程电源解决方案兼容的 XenServer 主机上启用“主机启动”功能。使用 `host-set-power-on` 命令时，必须在主机上指定电源管理解决方案的类型（即 `<power-on-mode>`）。然后使用 `<power-on-config>` 参数及其键-值对指定配置选项。要使用加密功能存储密码，请指定键“`power_on_password_secret`”。

#### 4.9.26. host-reboot

```
host-reboot [<host-selector>=<host_selector_value>...]
```

重新引导指定的 XenServer 主机。首先必须使用 `xe host-disable` 命令禁用指定的主机，否则将显示 `HOST_IN_USE` 错误消息。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

如果指定的 XenServer 主机是池的成员，则将处理在关闭时断开的连接并且池将在 XenServer 主机重新启动时恢复。如果关闭一个池成员，其他成员和主服务器将继续运行。如果关闭主服务器，池将无法正常工作，直到重新引导主服务器并使其恢复联机状态（此时成员将重新连接主服务器并与其同步），或者直到使某个成员成为主服务器。

#### 4.9.27. host-restore

```
host-restore [file-name=<backup_filename>] [<host-selector>=<host_selector_value>...]
```

还原 XenServer 主机控制软件的名为 `file-name` 的备份。请注意，此处使用的“还原”不是通常意义上的完整还原，它仅仅指压缩的备份文件已经解压缩并且解压缩到辅助分区。执行 `xe host-restore` 后，您必须引导安装 CD，并使用其 `Restore from Backup`（从备份还原）选项。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

#### 4.9.28. host-set-hostname-live

```
host-set-hostname host-uuid=<uuid_of_host> hostname=<new_hostname>
```

更改 `host-uuid` 指定的 XenServer 主机的主机名。此命令将控制域数据库中的主机名和 XenServer 主机的实际 Linux 主机名进行相同设置。请注意，`hostname` 与 `name_label` 字段的值不同。

#### 4.9.29. host-shutdown

```
host-shutdown [<host-selector>=<host_selector_value>...]
```

关闭指定的 XenServer 主机。首先必须使用 `xe host-disable` 命令禁用指定的 XenServer 主机，否则将显示 `HOST_IN_USE` 错误消息。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

如果指定的 XenServer 主机是池的成员，则将处理在关闭时断开的连接并且池将在 XenServer 主机重新启动时恢复。如果关闭一个池成员，其他成员和主服务器将继续运行。如果关闭主服务器，池将无法正常工作，直到重新引导主服务器并使其恢复在线状态（此时成员将重新连接主服务器并与其同步），或者直到使某个成员成为主服务器。如果启用了池的高可用性功能，则将自动使一个成员成为主服务器。如果禁用了





高可用性功能，您必须使用 `pool-designate-new-master` 命令手动指定所需服务器为主服务器。请参阅第 4.15.2 节 “[pool-designate-new-master](#)”。

#### 4.9.30. host-syslog-reconfigure

`host-syslog-reconfigure [<host-selector>=<host_selector_value>...]`

在指定的 XenServer 主机上重新分配 syslog 后台程序。此命令应用在主机的 logging 参数中定义的配置信息。

将使用标准选择机制来选择应在其上执行此操作的主机（请参阅上文中的[主机选择器](#)）。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。

#### 4.9.31. host-data-source-list

`host-data-source-list [<host-selectors>=<host selector value>...]`

列出可为主机记录的数据源。

使用标准选择机制（请参阅[主机选择器](#)）选择将执行此操作的主机。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。如果未提供选择主机的参数，将在所有主机上执行操作。

数据源具有两个参数 — standard 和 enabled，如此命令的输出所示。如果数据源的 enabled 设置为 true，则指标当前记录到性能数据库中。如果数据源的 standard 设置为 true，则默认情况下指标会记录到性能数据库中（因此，该数据源的 enabled 也将设置为 true）。如果数据源的 standard 设置为 false，则默认情况下指标不会记录到性能数据库中（因此，该数据源的 enabled 也将设置为 false）。

要开始将数据源的指标记录到性能数据库中，请运行 `host-data-source-record` 命令。这会将 enabled 设置为 true。要停止记录，请运行 `host-data-source-forget`。这会将 enabled 设置为 false。

#### 4.9.32. host-data-source-record

`host-data-source-record data-source=<name_description_of_data-source> [<host-selectors>=<host selector value>...]`

记录主机的指定数据源。

此操作会将数据源中的信息写入指定主机的静态性能指标数据库中。由于性能的原因，此数据库不同于普通的代理数据库。

使用标准选择机制（请参阅[主机选择器](#)）选择将执行此操作的主机。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。如果未提供选择主机的参数，将在所有主机上执行操作。

#### 4.9.33. host-data-source-forget

`host-data-source-forget data-source=<name_description_of_data-source> [<host-selectors>=<host selector value>...]`

停止为主机记录指定数据源并忽略所有已记录的数据。

使用标准选择机制（请参阅[主机选择器](#)）选择将执行此操作的主机。可选参数可以为[主机参数](#)部分开头列出的参数中的任意数个。如果未提供选择主机的参数，将在所有主机上执行操作。

#### 4.9.34. host-data-source-query

host-data-source-query data-source=<name\_description\_of\_data-source> [**<host-selectors>=<host selector value>...**]

显示主机的指定数据源。

使用标准选择机制（请参阅[主机选择器](#)）选择将执行此操作的主机。可选参数可以为 [主机参数](#) 部分开头列出的参数中的任意数个。如果未提供选择主机的参数，将在所有主机上执行操作。

### 4.10. 消息命令

用于处理消息的命令。创建消息以通知用户重要事件，并作为警报在 XenCenter 中显示。

#### 4.10.1. 消息参数

参数名称	说明	类型
uuid	消息的唯一标识符/对象引用	只读
name	消息的唯一名称。	只读
priority	消息优先级。数值越大表明优先级越高。	只读
class	消息类型，例如 VM。	只读
obj-uuid	受影响对象的 uuid。	只读
timestamp	生成该消息的时间。	只读
body	消息内容。	只读

#### 4.10.2. message-create

message-create name=<message\_name> body=<message\_text> [[host-uuid=<uuid\_of\_host>] | [sr-uuid=<uuid\_of\_sr>] | [vm-uuid=<uuid\_of\_vm>] | [pool-uuid=<uuid\_of\_pool>]]

创建一个新消息。

#### 4.10.3. message-destroy

message-destroy {uuid=<message\_uuid>}

销毁现有消息。您可以构建一个脚本来销毁所有消息。例如：

```
# Dismiss all alerts \
IFS=","; for m in $(xe message-list params=uuid --minimal); do \
  xe message-destroy uuid=$m \
done
```

#### 4.10.4. message-list

message-list



列出所有消息，或与特定标准可选参数相匹配的消息。

## 4.11. 网络命令

用于处理网络的命令。

网络对象可以使用标准对象列出命令 (xe network-list) 列出，参数可以使用标准参数命令操纵。有关详细信息，请参阅第 3.2 节“低级参数命令”。

### 4.11.1. 网络参数

网络具有下列参数：

参数名称	说明	类型
uuid	网络的唯一标识符/对象引用	只读
name-label	网络名称	读写
name-description	网络的说明文本	读写
VIF-uuids	从 VM 连接到此网络的 VIF (虚拟网络接口) 的唯一标识符列表	只读 set 参数
PIF-uuids	从 XenServer 主机连接到此网络的 PIF (物理网络接口) 的唯一标识符列表	只读 set 参数
bridge	与此网络对应的桥在本地 XenServer 主机上的名称	只读
default-locking-mode	与 VIF 对象一起用于 ARP 过滤的网络对象。设为<已解除锁定>，将删除所有与 VIF 相关联的过滤规则。设为<已禁用>，使 VIF 丢弃所有通信。请参阅第 4.4.10.8 节“使用交换机端口锁定”和 VM 参数。	读写
other-config:static-routes	<subnet>/<netmask>/<gateway> 格式化条目的逗号分隔列表，指定用来路由子网的网关地址。例如，将 other-config:static-routes 设置为 172.16.0.0/15/192.168.0.3,172.18.0.0/16/192.168.0.4 会使 172.16.0.0/15 上的通信通过 192.168.0.3 路由，172.18.0.0/16 上的通信通过 192.168.0.4 路由。	读写
other-config:ethtool-autoneg	设置为 no 会禁用物理接口或桥的自动协商。默认值是 yes。	读写
other-config:ethtool-rx	设置为 on 会启用接收校验和，设置为 off 会禁用接收校验和	读写
other-config:ethtool-tx	设置为 on 会启用发送校验和，设置为 off 会禁用发送校验和	读写

参数名称	说明	类型
other-config:ethtool-sg	设置为 on 会启用分散收集，设置为 off 会禁用分散收集	读写
other-config:ethtool-tso	设置为 on 会启用 tcp 分段卸载，设置为 off 会禁用 tcp 分段卸载	读写
other-config:ethtool-ufo	设置为 on 会启用 UDP 段卸载，设置为 off 会禁用 UDP 段卸载	读写
other-config:ethtool-gso	设置为 on 会启用通用分段卸载，设置为 off 会禁用通用分段卸载	读写
blobs	二进制数据存储	只读

#### 4.11.2. network-create

network-create name-label=<name\_for\_network> [name-description=<descriptive\_text>]

创建新网络。

#### 4.11.3. network-destroy

network-destroy uuid=<network\_uuid>

销毁现有网络。

### 4.12. 修补程序 (更新) 命令

用于处理 XenServer 主机修补程序 (更新) 的命令。这些命令适用于 XenServer 的标准非 OEM 版本，有关与更新 XenServer 的 OEM 版本相关的命令，请参阅第 4.20 节“更新命令”以了解详细信息。

修补程序对象可以使用标准对象列出命令 (xe patch-list) 列出，参数可以使用标准参数命令操纵。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.12.1. 修补程序参数

修补程序具有下列参数：

参数名称	说明	类型
uuid	修补程序的唯一标识符/对象引用	只读
host-uuid	要查询的 XenServer 主机的唯一标识符	只读
name-label	修补程序名称	只读
name-description	修补程序的说明字符串	只读
applied	是否已应用修补程序；true 或 false	只读

参数名称	说明	类型
size	是否已应用修补程序；true 或 false	只读

#### 4.12.2. patch-apply

patch-apply host-uuid=<host\_uuid> uuid=<patch\_file\_uuid>

应用指定的修补程序文件。

#### 4.12.3. patch-pool-clean

patch-pool-clean uuid=<patch\_file\_uuid>

从池中的所有 XenServer 主机删除指定的修补程序文件。可以在通过 XenCenter 或 CLI 应用修补程序后或在 dom0 即将耗尽磁盘空间时使用。

#### 4.12.4. patch-pool-apply

patch-pool-apply uuid=<patch\_uuid>

将指定的修补程序应用于池中的所有 XenServer 主机。

#### 4.12.5. patch-precheck

patch-precheck uuid=<patch\_uuid> host-uuid=<host\_uuid>

对指定 XenServer 主机运行包含在指定修补程序中的预检查。

#### 4.12.6. patch-upload

patch-upload file-name=<patch\_filename>

将指定的修补程序文件上载到 XenServer 主机。这可以为应用修补程序做准备。成功时，会打印出上载的修补程序的 UUID。如果先前已上载过此修补程序，将返回 PATCH\_ALREADY\_EXISTS 错误，而不会再次上载该修补程序。

### 4.13. PBD 命令

用于处理 PBD（物理块设备）的命令。这些命令是软件对象，XenServer 主机通过它们访问存储库 (SR)。

PBD 对象可以使用标准对象列出命令 (xe pbd-list) 列出，参数可以使用标准参数命令操纵。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.13.1. PBD 参数

PBD 具有下列参数：

参数名称	说明	类型
uuid	PBD 的唯一标识符/对象引用。	只读

参数名称	说明	类型
sr-uuid	PBD 指向的存储库	只读
device-config	提供给主机的 SR 后端驱动程序的其他配置信息	只读 map 参数
currently-attached	如果 SR 当前已连接到此主机，则为 True，否则为 False	只读
host-uuid	提供 PBD 的物理机的 UUID	只读
host	不推荐使用 host 字段。请使用 host_uuid。	只读
other-config	其他配置信息。	读取/写入 map 参数

#### 4.13.2. pbd-create

```
pbd-create host-uuid=<uuid_of_host>
sr-uuid=<uuid_of_sr>
[device-config:key=<corresponding_value>...]
```

在 XenServer 主机上创建新 PBD。只读 device-config 参数只能在创建时设置。

若要添加“path”-“/tmp”的映射，命令行应包含参数 device-config:path=/tmp

有关每个 SR 类型上受支持的 device-config 键/值对的完整列表，请参阅[第 5 章 存储](#)。

#### 4.13.3. pbd-destroy

```
pbd-destroy uuid=<uuid_of_pbd>
```

销毁指定的 PBD。

#### 4.13.4. pbd-plug

```
pbd-plug uuid=<uuid_of_pbd>
```

尝试将 PBD 插入 XenServer 主机。如果成功，则引用的 SR（和包含在其中的 VDI）应对 XenServer 主机可见。

#### 4.13.5. pbd-unplug

```
pbd-unplug uuid=<uuid_of_pbd>
```

尝试从 XenServer 主机拔出 PBD。

### 4.14. PIF 命令

用于处理 PIF（代表物理网络接口的对象）的命令。

PIF 对象可以使用标准对象列出命令 (xe pif-list) 列出，参数可以使用标准参数命令操纵。有关详细信息，请参阅[第 3.2 节 “低级参数命令”](#)。

#### 4.14.1. PIF 参数

PIF 具有下列参数：

参数名称	说明	类型
uuid	PIF 的唯一标识符/对象引用	只读
device	接口的计算机可识别的名称（例如 eth0）	只读
MAC	PIF 的 MAC 地址	只读
other-config	其他的 PIF 配置名称：值对。	读取/写入 map 参数
physical	如果为 true，则 PIF 指向实际的物理网络接口	只读
currently-attached	PIF 当前是否连接到此主机？true 或 false	只读
MTU	PIF 的最大传输单位（以字节为单位）。	只读
VLAN	通过此接口的所有通信的 VLAN 标记；-1 表明未分配 VLAN 标记	只读
bond-master-of	此 PIF 作为主对象的绑定的 UUID（如果有）	只读
bond-slave-of	此 PIF 作为从属对象的绑定的 UUID（如果有）	只读
management	是否将此 PIF 指定为控制域的管理接口	只读
network-uuid	此 PIF 连接到的虚拟网络的唯一标识符/对象引用	只读
network-name-label	此 PIF 连接到的虚拟网络的名称	只读
host-uuid	此 PIF 连接到的 XenServer 主机的唯一标识符/对象引用	只读
host-name-label	此 PIF 连接到的 XenServer 主机的名称	只读
IP-configuration-mode	使用的网络地址配置类型；DHCP 或 static	只读
IP	PIF 的 IP 地址，如果 IP 配置模式为 static，则在此处定义；如果 IP 配置模式为 DHCP，则不在此处定义	只读
netmask	PIF 的网络掩码，如果 IP 配置模式为 static，则在此处定义；如果由 DHCP 提供，则不在此处定义	只读
gateway	PIF 的网关地址，如果 IP 配置模式为 static，则在此处定义；如果由 DHCP 提供，则不在此处定义	只读

参数名称	说明	类型
DNS	PIF 的 DNS 地址，如果 IP 配置模式为 static，则在此处定义；如果由 DHCP 提供，则不在此处定义	只读
io_read_kbs	设备的平均读取速率（以 kb/s 为单位）	只读
io_write_kbs	设备的平均写入速率（以 kb/s 为单位）	只读
carrier	此设备的链接状态	只读
vendor-id	分配给 NIC 供应商的 ID	只读
vendor-name	NIC 供应商的名称	只读
device-id	供应商分配给此 NIC 型号的 ID	只读
device-name	供应商分配给此 NIC 型号的名称	只读
speed	NIC 的数据传输速率	只读
duplex	NIC 的双工模式；full 或 half	只读
pci-bus-path	PCI 总线路径地址	只读
other-config:ethtool-speed	设置连接速度（以 Mbps 为单位）	读写
other-config:ethtool-autoneg	设置为 no 会禁用物理接口或桥的自动协商。默认值是 yes。	读写
other-config:ethtool-duplex	设置 PIF 的双工功能，full 或 half。	读写
other-config:ethtool-rx	设置为 on 会启用接收校验和，设置为 off 会禁用接收校验和	读写
other-config:ethtool-tx	设置为 on 会启用发送校验和，设置为 off 会禁用发送校验和	读写
other-config:ethtool-sg	设置为 on 会启用分散收集，设置为 off 会禁用分散收集	读写
other-config:ethtool-tso	设置为 on 会启用 tcp 分段卸载，设置为 off 会禁用 tcp 分段卸载	读写
other-config:ethtool-ufo	设置为 on 会启用 udp 段卸载，设置为 off 会禁用 udp 段卸载	读写
other-config:ethtool-gso	设置为 on 会启用通用分段卸载，设置为 off 会禁用通用分段卸载	读写
other-config:domain	用于设置 DNS 搜索路径的逗号分隔列表	读写

参数名称	说明	类型
other-config:bond-miimon	链接活跃性检查之间的间隔（以毫秒为单位）	读写
other-config:bond-downdelay	断开链接后等待的毫秒数，超过此时间即认为链接确实已断开。这允许出现短暂的链接中断现象	读写
other-config:bond-updelay	建立链接后等待的毫秒数，超过此时间即认为链接确实已建立。允许链接出现不稳定的状态。默认值为 31 秒，以留出交换机开始转发通信的时间。	读写
disallow-unplug	如果此 PIF 为专用存储 NIC，则为 true，否则为 false	读取/写入



#### 注意

对 PIF 的 other-config 字段所做的更改仅在重新引导后生效。另外，使用 xe pif-unplug 和 xe pif-plug 命令会导致重写 PIF 配置。

#### 4.14.2. pif-forget

pif-forget uuid=<uuid\_of\_pif>

销毁特定主机上的指定 PIF 对象。

#### 4.14.3. pif-introduce

pif-introduce host-uuid=<UUID of XenServer host> mac=<mac\_address\_for\_pif> device=<machine-readable name of the interface (for example, eth0)>

在指定的 XenServer 主机上创建代表物理接口的新 PIF 对象。

#### 4.14.4. pif-plug

pif-plug uuid=<uuid\_of\_pif>

尝试显示指定的物理接口。

#### 4.14.5. pif-reconfigure-ip

pif-reconfigure-ip uuid=<uuid\_of\_pif> [ mode=<dhcp> | mode=<static> ]  
gateway=<network\_gateway\_address> IP=<static\_ip\_for\_this\_pif>  
netmask=<netmask\_for\_this\_pif> [DNS=<dns\_address>]

修改 PIF 的 IP 地址。对于静态 IP 配置，请将 mode 参数设置为 static，同时将 gateway、IP 和 netmask 参数设置为相应的值。若要使用 DHCP，请将 mode 参数设置为 DHCP 并将静态参数保持为未定义状态。



#### 注意

对于使用跨树协议并且关闭了（或不支持）STP 快速链接的交换机，当物理网络接口与该交换机上的端口相连时，如果在这些物理网络接口上使用静态 IP 地址，则会产生没有网络流量的时段。

#### 4.14.6. pif-scan

pif-scan host-uuid=<UUID of XenServer host>

在 XenServer 主机上扫描新物理接口。

#### 4.14.7. pif-unplug

pif-unplug uuid=<uuid\_of\_pif>

尝试拔出指定的物理接口。

### 4.15. 池命令

用于处理池的命令。池是一个或多个 XenServer 主机的聚合。一个池使用一个或多个共享存储库，以便在池中的一个 XenServer 主机上运行的 VM 可以准实时地（继续运行而无需关闭和恢复运行）迁移到该池中的另一台 XenServer 主机。默认情况下，每台 XenServer 主机实际上是由单个成员组成的一个池。XenServer 主机加入一个池时，它会被指定为一个成员，而它所加入的池会成为池的主服务器。

单例池对象可以使用标准对象列出命令 (xe pool-list) 列出，参数可以使用标准参数命令操纵。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.15.1. 池参数

池具有下列参数：

参数名称	说明	类型
uuid	池的唯一标识符/对象引用	只读
name-label	池的名称	读取/写入
name-description	池的说明字符串	读取/写入
master	分配为池主服务器的 XenServer 主机的唯一标识符/对象引用	只读
default-SR	池的默认 SR 的唯一标识符/对象引用	读取/写入
crash-dump-SR	保存池成员的任何故障转储的 SR 的唯一标识符/对象引用	读取/写入
metadata-vdis	池的所有已知元数据 VDI	只读
suspend-image-SR	保存池成员上挂起 VM 的 SR 的唯一标识符/对象引用	读取/写入
other-config	为池指定其他配置参数的键/值对列表	读取/写入 map 参数
supported-sr-types	可供此池使用的 SR 类型	只读
ha-enabled	如果池启用了高可用性功能，则为 true，否则为 false	只读
ha-configuration	保留以供将来使用。	只读
ha-statefiles	列出高可用性功能使用的 VDI 的 UUID 以确定存储运行状况	只读



参数名称	说明	类型
ha-host-failures-to-tolerate	允许的主机故障数，超过此数量即发送系统警报	读取/写入
ha-plan-exists-for	根据高可用性功能算法的计算，可实际处理的主机故障数	只读
ha-allow-overcommit	如果允许过量使用池，则为 True，否则为 False	读取/写入
ha-overcommitted	如果当前过量使用池，则为 True	只读
blobs	二进制数据存储	只读
pool-designate-new-master		

#### 4.15.2. pool-designate-new-master

pool-designate-new-master host-uuid=<UUID of member XenServer host to become new master>

指示指定的成员 XenServer 主机成为现有池的主服务器。这会将主服务器主机的角色有序移交给资源池中的其他主机。此命令仅在当前主服务器处于联机状态时生效，并且不是下列紧急模式命令的替代项。

#### 4.15.3. pool-dump-database

pool-dump-database file-name=<filename\_to\_dump\_database\_into\_(on\_client)>

下载整个池数据库的副本并将其转储到客户端上的文件。

#### 4.15.4. pool-eject

pool-eject host-uuid=<UUID of XenServer host to eject>

指示指定的 XenServer 主机退出现有池。

#### 4.15.5. pool-emergency-reset-master

pool-emergency-reset-master master-address=<address of the pool's master XenServer host>

指示从属成员 XenServer 主机将其主服务器地址重置为新值，并尝试与其建立连接。不应在主服务器主机上运行此命令。

#### 4.15.6. pool-emergency-transition-to-master

pool-emergency-transition-to-master

指示 XenServer 成员主机成为池主服务器。仅在 XenServer 主机转换到紧急模式后才接受此命令。进入紧急模式意味着该成员主机所在的池中的主服务器已从网络中消失，经过若干次重试仍无法连接。

请注意，如果在主机加入池后修改了主机密码，此命令可能导致该密码重置（请参阅第 4.21 节“用户命令”）

#### 4.15.7. pool-ha-enable

pool-ha-enable heartbeat-sr-uuids=<SR\_UUID\_of\_the\_Heartbeat\_SR>



对资源池启用高可用性，同时使用指定的 SR UUID 作为中央检测信号储存库。

#### 4.15.8. pool-ha-disable

pool-ha-disable

对资源池禁用高可用性功能。

#### 4.15.9. pool-join

pool-join master-address=<address> master-username=<username> master-password=<password>

指示 XenServer 主机加入现有池。

#### 4.15.10. pool-recover-slaves

pool-recover-slaves

指示池主服务器尝试重置当前以紧急模式运行的所有成员的主服务器地址。通常在使用 pool-emergency-transition-to-master 将其中的一个成员设置为主服务器后使用此命令。

#### 4.15.11. pool-restore-database

pool-restore-database file-name=<filename\_to\_restore\_from\_(on\_client)> [dry-run=<true | false>]

向池上载数据库备份（使用 pool-dump-database 创建）。接收上载后，主服务器将自行重新启动并使用新数据库。

该命令中还包含一个 dry run 选项，借助此选项，您可以确定是否可在无需实际执行操作的情况下还原池数据库。默认情况下，dry-run 设置为 false。

#### 4.15.12. pool-sync-database

pool-sync-database

强制池数据库在资源池中的所有主机上进行同步。这在正常操作中没有必要，因为数据库会定期自动复制。但对于确保在执行一系列重要 CLI 操作后快速复制更改，此命令很有用。

### 4.16. 存储管理器命令

用于控制存储管理器插件的命令。

可使用标准对象列出命令 (xe sm-list) 列出存储管理器对象，使用标准参数命令控制其参数。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.16.1. SM 参数

SM 包含下列参数：

参数名称	说明	类型
uuid	SM 插件的唯一标识符/对象引用	只读
name-label	SM 插件的名称	只读
name-description	SM 插件的描述字符串	只读

参数名称	说明	类型
type	此插件连接到的 SR 类型	只读
vendor	创建此插件的供应商的名称	只读
copyright	此 SM 插件的版权声明	只读
required-api-version	XenServer 主机要求的最低 SM API 版本	只读
configuration	设备配置键的名称和描述	只读
capabilities	SM 插件的功能	只读
driver-filename	SR 驱动程序的文件名。	只读

## 4.17. SR 命令

用于控制 SR ( 存储库 ) 的命令。

可使用标准对象列出命令 (xe sr-list) 列出 SR 对象，使用标准参数命令控制其参数。有关详细信息，请参阅第 3.2 节“低级参数命令”。

### 4.17.1. SR 参数

SR 包含下列参数：

参数名称	说明	类型
uuid	SR 的唯一标识符/对象引用	只读
name-label	SR 的名称	读取/写入
name-description	SR 的描述字符串	读取/写入
allowed-operations	在此状态下的 SR 上允许执行的操作的列表	只读 set 参数
current-operations	当前此 SR 上正在进行的操作的列表	只读 set 参数
VDIs	此 SR 中虚拟磁盘的唯一标识符/对象引用	只读 set 参数
PBD	连接到此 SR 的 PBD 的唯一标识符/对象引用	只读 set 参数
physical-utilisation	此 SR 上当前使用的物理空间，以字节为单位。请注意，对于精简置备的磁盘格式，物理利用率可能低于虚拟分配	只读
physical-size	SR 的总物理空间大小，以字节为单位	只读
type	SR 的类型，用于指定要使用的 SR 后端驱动程序	只读
introduced-by	引入 SR 的 drtask ( 如果有 )	只读

参数名称	说明	类型
content-type	SR 内容的类型。用于将 ISO 库与其他 SR 区分开来。对于存储 ISO 库的存储库，content-type 必须设置为 iso。对于其他情况，Citrix 建议将此参数设置为空或字符串 user。	只读
shared	如果可以在多个 XenServer 主机之间共享此 SR，则为 True；否则为 False。	读取/写入
other-config	键/值对列表，用于指定 SR 的其他配置参数。	读取/写入 map 参数
host	存储库主机名称	只读
virtual-allocation	此存储库中的所有 VDI 的总虚拟大小值，以字节为单位	只读
sm-config	SM 依赖数据	只读 map 参数
blobs	二进制数据存储	只读

#### 4.17.2. sr-create

```
sr-create name-label=<name> physical-size=<size> type=<type>
content-type=<content_type> device-config:<config_name>=<value>
[host-uuid=<XenServer host UUID>] [shared=<true | false>]
```

在磁盘上创建一个 SR，将其引入数据库，然后创建一个将该 SR 连接到 XenServer 主机的 PBD。如果将 shared 设置为 true，则为池中的每个 XenServer 主机创建 PBD；如果未指定 shared 或将其设置为 false，则仅为使用 host-uuid 指定的 XenServer 主机创建 PBD。

具体的 device-config 参数因设备 type 的不同而异。有关不同存储后端上这些参数的详细信息，请参阅第 5 章 存储。

#### 4.17.3. sr-destroy

```
sr-destroy uuid=<sr_uuid>
```

销毁 XenServer 主机上指定的 SR。

#### 4.17.4. sr-enable-database-replication

```
sr-enable-database-replication uuid=<sr_uuid>
```

启用以指定（共享）SR 为目标的 xapi 数据库复制。例如：

```
xe sr-enable-database-replication uuid=<sr-uuid>
```

#### 4.17.5. sr-disable-database-replication

```
sr-disable-database-replication uuid=<sr_uuid>
```

禁用以指定 SR 为目标的 xapi 数据库复制。例如：

```
xe sr-enable-database-replication uuid=<sr-uuid>
```

#### 4.17.6. sr-forget

```
sr-forget uuid=<sr_uuid>
```

xapi 代理忽略了 XenServer 主机上某个指定的 SR，这意味着该 SR 被分离，您无法访问其上的 VDI，但该 SR 在源介质中保持原样（数据未丢失）。

#### 4.17.7. sr-introduce

```
sr-introduce name-label=<name>
physical-size=<physical_size>
type=<type>
content-type=<content_type>
uuid=<sr_uuid>
```

即将 SR 记录添加到数据库中。device-config 参数由 device-config:<parameter\_key>=<parameter\_value> 指定，例如：

```
xe sr-introduce device-config:<device>=</dev/sdb1>
```



#### 注意

此命令不用于正常操作中。它是一个高级操作，对于在创建 SR 后需要将 SR 重新配置为共享这种情况可能很有用，或者有助于在出现各种故障情形后恢复 SR。

#### 4.17.8. sr-probe

```
sr-probe type=<type> [host-uuid=<uuid_of_host>] [device-config:<config_name>=<value>]
```

使用提供的 device-config 键执行特定于后端的扫描。如果已为 SR 后端完成 device-config，则将返回设备上的 SR 的列表（如果有）。如果仅完成部分 device-config 参数，则将执行特定于后端的扫描，返回结果将有助于您改进其余 device-config 参数。扫描结果以特定于后端的 XML 形式返回，显示在 CLI 上。

具体的 device-config 参数因设备 type 的不同而异。有关不同存储后端上这些参数的详细信息，请参阅第 5 章 存储。

#### 4.17.9. sr-scan

```
sr-scan uuid=<sr_uuid>
```

强制执行 SR 扫描，同时将 xapi 数据库与基础存储基底中的 VDI 同步。

### 4.18. 任务命令

用于长时间运行的异步任务的命令。此类任务包括启动、停止和挂起虚拟机等，通常由一组共同完成请求的操作的其他原子任务组成。

可使用标准对象列出命令 (xe task-list) 列出任务对象，使用标准参数命令控制其参数。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.18.1. 任务参数

任务包含下列参数：

参数名称	说明	类型
uuid	任务的唯一标识符/对象引用	只读
name-label	任务的名称	只读

参数名称	说明	类型
name-description	任务的描述字符串	只读
resident-on	运行任务的主机的唯一标识符/对象引用	只读
status	任务的当前状态	只读
progress	如果任务尚未完成，则此字段包含估算的百分比（介于 0 到 1 之间）。如果任务已完成，无论成功与否，此字段都应该是 1。	只读
type	如果任务已成功完成，则此参数包含编码结果的类型（即，其引用在结果字段中的类的名称）；否则，此参数的值处于未定义状态	只读
result	如果任务已成功完成，此字段包含结果值（为空或对象引用）；否则，此参数的值处于未定义状态	只读
error_info	如果任务已失败，则此参数包含一组相关的错误字符串；否则，此参数的值处于未定义状态	只读
allowed_operations	此状态下允许的操作的列表	只读
created	创建任务的时间	只读
finished	任务完成（成功或失败）时间。如果任务状态是未完成，则此字段的值没有任何意义	只读
subtask_of	包含以此任务为子任务的任务的 UUID	只读
subtasks	包含此任务的所有子任务的 UUID	只读

#### 4.18.2. task-cancel

task-cancel [uuid=<task\_uuid>]

指示取消指定的任务并返回。

#### 4.19. 模板命令

用于 VM 模板的命令。

模板实质上是 is-a-template 参数设置为 true 的 VM。模板是一种“黄金映像”，包含用于实例化特定 VM 的所有配置设置。XenServer 随附一套基础模板，它们是通用的“原始”VM，能够引导操作系统厂商提供的安装 C（例如，RHEL、CentOS、SLES、Windows）。通过 XenServer，您可以创建 VM，按照您的特定需要以标准形式配置它们，将它们的副本保存为模板供将来在 VM 部署中使用。

可使用标准对象列出命令 (xe template-list) 列出模板对象，使用标准参数命令控制其参数。有关详细信息，请参阅第 3.2 节“低级参数命令”。

## 4.19.1. VM 模板参数

模板包含下列参数：

参数名称	说明	类型
uuid	模板的唯一标识符/对象引用	只读
name-label	模板的名称	读取/写入
name-description	模板的描述字符串	读取/写入
user-version	描述 VM 和模板的创建者的字符串，用于输入版本信息	读取/写入
is-a-template	如果是模板，则为 true。模板 VM 无法启动，它们仅用于克隆其他 VM	读取/写入
is-control-domain	如果是控制域（域 0 或驱动程序域），则为 true	只读
power-state	当前电源状态；对于模板，始终为 halted	只读
memory-dynamic-max	最大动态内存，以字节为单位。当前未使用，但如果更改则必须遵守以下限制： memory_static_max >= memory_dynamic_max >= memory_dynamic_min >= memory_static_min.	读取/写入
memory-dynamic-min	最小动态内存，以字节为单位。当前未使用，但如果更改，则必须遵守与针对 memory-dynamic-max 的限制相同的限制。	读取/写入
memory-static-max	最大静态设置（绝对）内存，以字节为单位。这是用于确定分配给 VM 的内存量的主要值。	读取/写入
memory-static-min	最小静态设置（绝对）内存，以字节为单位。这代表绝对最小内存，memory-static-min 必须小于 memory-static-max。此值当前未用于正常操作，但是必须遵守前面提到的限制。	读取/写入
suspend-VDI-uuid	存储挂起映像的 VDI（对于模板没有意义）	只读

参数名称	说明	类型
VCPUs-params	<p>所选 VCPU 策略的配置参数。</p> <p>可以使用以下命令调整 VCPU 分配</p> <pre>xe template-param-set \   uuid=&lt;template_uuid&gt; \   VCPUs-params:mask=1,2,3</pre> <p>从此模板创建的 VM 稍后将仅在物理 CPU 1、CPU 2 和 CPU 3 上运行。</p> <p>还可以使用 cap 和 weight 参数调整 VCPU 优先级 (xen 调度) ; 例如</p> <pre>xe template-param-set \   uuid=&lt;template_uuid&gt; \   VCPUs-params:weight=512 xe template-param-set \   uuid=&lt;template_uuid&gt; \   VCPUs-params:cap=100</pre> <p>基于此模板的权重为 512 的 VM 所获得的 CPU 将是竞争 XenServer 主机上权重为 256 的域的两倍。合法的权重范围为 1 到 65535, 默认值为 256。</p> <p>上限可选地规定基于此模板的 VM 可以占用的最大 CPU 量, 即使 XenServer 主机具有空闲 CPU 周期也应用此上限。此上限以一个物理 CPU 的百分比表示: 100 是 1 个物理 CPU, 50 是半个物理 CPU, 400 是 4 个物理 CPU。默认值为 0, 表示无上限。</p>	读取/写入 map 参数
VCPUs-max	VCPU 的最大数目	读取/写入
VCPUs-at-startup	VCPU 的引导编号	读取/写入
actions-after-crash	当基于此模板的 VM 崩溃时要执行的操作	读取/写入
console-uuids	虚拟控制台设备	只读 set 参数



参数名称	说明	类型
platform	<p>特定于平台的配置</p> <p>禁用 HVM 来宾系统（例如，Windows 来宾系统）的并行端口模拟：</p> <pre>xe vm-param-set \   uuid=&lt;vm_uuid&gt; \   platform:parallel=none</pre> <p>禁用 HVM 来宾系统的串行端口模拟：</p> <pre>xe vm-param-set \   uuid=&lt;vm_uuid&gt; \   platform:hvm_serial=none</pre> <p>禁用 HVM 来宾系统的 USB 控制器和 USB 平板电脑设备模拟：</p> <pre>xe vm-param-set \   uuid=&lt;vm_uuid&gt; \   platform:usb=false</pre> <pre>xe vm-param-set \   uuid=&lt;vm_uuid&gt; \   platform:usb_tablet=false</pre>	读取/写入 map 参数
allowed-operations	此状态下允许的操作的列表	只读 set 参数
current-operations	当前在此模板上进行的操作的列表	只读 set 参数
allowed-VBD-devices	可供使用的 VBD 标识符的列表，用 0 到 15 之间的整数表示。此列表仅供参考，可以使用未在此列表中列出的其他设备（但可能无法正常运行）。	只读 set 参数
allowed-VIF-devices	可供使用的 VIF 标识符的列表，用 0 到 15 之间的整数表示。此列表仅供参考，可以使用未在此列表中列出的其他设备（但可能无法正常运行）。	只读 set 参数
HVM-boot-policy	HVM 来宾系统的引导策略。值为 BIOS Order 或空字符串。	读取/写入
HVM-boot-params	order 关键字用于控制 HVM 来宾系统引导顺序，以字符串表示，其中每个字符代表一种引导方法：d 代表 CD/DVD，c 代表根磁盘，n 代表网络 PXE 引导。默认值为 dc。	读取/写入 map 参数
PV-kernel	内核路径	读取/写入

参数名称	说明	类型
PV-ramdisk	initrd 路径	读取/写入
PV-args	内核命令行参数字符串	读取/写入
PV-legacy-args	使旧 VM 基于此模板引导的参数字符串	读取/写入
PV-bootloader	引导加载程序的名称或路径	读取/写入
PV-bootloader-args	引导加载程序杂项参数的字符串	读取/写入
last-boot-CPU-flags	描述上次引导基于此模板的 VM 时所在的 CPU 的标志；不为模板填充此参数	只读
resident-on	基于此模板的 VM 当前所属的 XenServer 主机；对于模板，显示为 <not in database>	只读
affinity	基于此模板的 VM 首选在其上运行的 XenServer 主机；xe vm-start 命令使用此参数决定在何处运行 VM	读取/写入
other-config	键/值对的列表，用于为模板指定其他配置参数	读取/写入 map 参数
start-time	读取基于此模板的 VM 的指标的日期和时间的戳，以 <code>yyyymmddThh:mm:ss z</code> 的形式表示，其中 <code>z</code> 是单字母军用时区指示器（例如， <code>Z</code> 代表 UTC (GMT)）；对于模板，设置为 <code>1 Jan 1970 Z</code> （Unix/POSIX 时期的起始日期）	只读
install-time	读取基于此模板的 VM 的指标的日期和时间的戳，以 <code>yyyymmddThh:mm:ss z</code> 的形式表示，其中 <code>z</code> 是单字母军用时区指示器（例如， <code>Z</code> 代表 UTC (GMT)）；对于模板，设置为 <code>1 Jan 1970 Z</code> （Unix/POSIX 时期的起始日期）	只读
memory-actual	基于此模板的 VM 正使用的实际内存；对于模板，该值为 0	只读
VCPUs-number	分配给基于此模板的 VM 的虚拟 CPU 数目；对于模板，该值为 0	只读
VCPUs-Utilization	虚拟 CPU 及其权重的列表	只读 map 参数
os-version	基于此模板的 VM 的操作系统版本；对于模板，显示为 <not in database>	只读 map 参数

参数名称	说明	类型
PV-drivers-version	基于此模板的 VM 的半虚拟化驱动程序版本；对于模板，显示为 <not in database>	只读 map 参数
PV-drivers-detected	基于此模板的 VM 的半虚拟化驱动程序的最新版本的标志；对于模板，显示为 <not in database>	只读
memory	基于此模板的 VM 上的代理报告的内存指标；对于模板，显示为 <not in database>	只读 map 参数
disks	基于此模板的 VM 上的代理报告的磁盘指标；对于模板，显示为 <not in database>	只读 map 参数
networks	基于此模板的 VM 上的代理报告的网络指标；对于模板，显示为 <not in database>	只读 map 参数
other	基于此模板的 VM 上的代理报告的其他指标；对于模板，显示为 <not in database>	只读 map 参数
guest-metrics-last-updated	来宾代理上次写入这些字段的时间戳，以 yyyyymmddThh:mm:ss z 的形式表示，其中 z 是单字母军用时区指示器，例如，Z 表示 UTC (GMT)	只读
actions-after-shutdown	将在 VM 关闭后执行的操作	读取/写入
actions-after-reboot	将在 VM 重新引导后执行的操作	读取/写入
possible-hosts	可以承载 VM 的潜在主机列表	只读
HVM-shadow-multiplier	应用于被设置为可供来宾系统使用的重影量的系数	读取/写入
dom-id	域 ID (如果可用，否则为 -1)	只读
recommendations	此 VM 的建议值和属性范围的 XML 规范	只读
xenstore-data	在创建 VM 后将插入到 xenstore 树 (/local/domain/<domid>/vm-data) 的数据。	读取/写入 map 参数
is-a-snapshot	如果此模板为 VM 快照，则为 true	只读
snapshot_of	以此模板为快照的 VM 的 UUID	只读

参数名称	说明	类型
snapshots	根据此模板生成的任何快照的 UUID	只读
snapshot_time	最近生成的 VM 快照的时间戳	只读
memory-target	为此模板设置的目标内存量	只读
blocked-operations	列出无法在此模板上执行的操作	读取/写入 map 参数
last-boot-record	此模板的上次引导参数的记录，使用 XML 格式	只读
ha-always-run	如果此模板的实例在其所在的主机发生故障后总在其他主机上重新启动，则为 True	读取/写入
ha-restart-priority	1、2、3 或 best-effort。1 是最高重新启动优先级	读取/写入
blobs	二进制数据存储	只读
live	仅与正在运行的 VM 相关。	只读

#### 4.19.2. template-export

template-export template-uuid=<uuid\_of\_existing\_template> filename=<filename\_for\_new\_template>

将指定模板的副本导出到文件，使用指定的新文件名。

#### 4.20. 更新命令

用于更新 XenServer OEM 版本的命令。有关与更新 XenServer 标准非 OEM 版本相关的命令的详细信息，请参阅第 4.12 节“修补程序（更新）命令”。

##### 4.20.1. update-upload

update-upload file-name=<name\_of\_upload\_file>

将新软件映像发送到 OEM 版 XenServer 主机。要使此操作生效，必须稍后重新启动主机。

#### 4.21. 用户命令

##### 4.21.1. user-password-change

user-password-change old=<old\_password> new=<new\_password>

更改用户登录密码。未选中旧密码字段，因为执行此调用需要主管特权。

#### 4.22. VBD 命令

用于 VBD（虚拟块设备）的命令。

VBD 是一个软件对象，用于将 VM 连接到 VDI（代表虚拟磁盘的内容）。VBD 中包含可将 VDI 与 VM 相联系的属性（可引导性、读/写指标等），而 VDI 中包含虚拟磁盘的物理属性信息（SR 类型、磁盘是否可共享、介质是可读写介质还是只读介质等）。

可使用标准对象列举命令 (xe vbd-list) 列出 VBD 对象，使用标准参数命令控制其参数。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.22.1. VBD 参数

VBD 包含下列参数：

参数名称	说明	类型
uuid	VBD 的唯一标识符/对象引用	只读
vm-uuid	此 VBD 连接到的 VM 的唯一标识符/对象引用	只读
vm-name-label	此 VBD 连接到的 VM 的名称	只读
vdi-uuid	此 VBD 映射到的 VDI 的唯一标识符/对象引用	只读
vdi-name-label	此 VBD 映射到的 VDI 的名称	只读
empty	如果为 true，则表示空驱动器	只读
device	可对来宾系统显示的设备，例如 hda	只读
userdevice	vbd-create 期间通过 device 参数指定的设备号，例如，0 表示 hda，1 表示 hdb，等等	读取/写入
bootable	如果此 VBD 可引导，则为 true	读取/写入
mode	装载 VBD 应使用的模式	读取/写入
type	VBD 对 VM 的显示方式，例如磁盘或 CD	读取/写入
currently-attached	如果 VBD 当前连接到此主机，则为 True，否则，为 false	只读
storage-lock	如果获得了存储级别锁，则为 True	只读
status-code	与上次连接操作相关联的错误/成功代码	只读
status-detail	与上次连接操作状态相关联的错误/成功信息	只读
qos_algorithm_type	要使用的 QoS 算法	读取/写入

参数名称	说明	类型
qos_algorithm_params	所选 QoS 算法的参数	读取/写入 map 参数
qos_supported_algorithms	此 VBD 支持 QoS 算法	只读 set 参数
io_read_kbs	此 VBD 的平均读取速率，以 kbps 为单位	只读
io_write_kbs	此 VBD 的平均写入速率，以 kbps 为单位	只读
allowed-operations	此种状态下允许执行的操作列表。此列表只是建议性的，并且客户端读取此字段时服务器状态可能已更改	只读 set 参数
current-operations	将每个使用此对象（通过引用）的正在运行的任务链接到描述此任务本质的 current_operation 枚举。	只读 set 参数
unpluggable	如果此 VBD 支持热拔出，则为 true	读取/写入
attachable	如果设备可以连接，则为 True	只读
other-config	其他配置	读取/写入 map 参数

#### 4.22.2. vbd-create

```
vbd-create vm-uuid=<uuid_of_the_vm> device=<device_value>
vdi-uuid=<uuid_of_the_vdi_the_vbd_will_connect_to> [bootable=true] [type=<Disk | CD>] [mode=<RW | RO>]
```

在 VM 上新建 VBD。

device 字段允许使用的值为介于 0 到 15 之间的整数，并且每个 VM 的数值必须唯一。可以在指定 VM 上的 allowed-VBD-devices 参数中查看当前允许使用的值。这将被视为 vbd 参数中的 userdevice。

如果 type 为 Disk，则必须指定 vdi-uuid。对于磁盘，模式可以为 RO 或 RW。

如果 type 为 CD，则可选择是否指定 vdi-uuid；如果不指定 VDI，将为 CD 创建空 VBD。对于 CD，模式必须为 RO。

#### 4.22.3. vbd-destroy

```
vbd-destroy uuid=<uuid_of_vbd>
```

销毁指定的 VBD。

如果 VBD 将其 other-config:owner 参数设置为 true，则其关联的 VDI 也将被销毁。

#### 4.22.4. vbd-eject

```
vbd-eject uuid=<uuid_of_vbd>
```



从 VBD 代表的驱动器中删除介质。仅当介质是可移动类型 ( 物理 CD 或 ISO ) 时，此命令才有效；否则，将返回错误消息 VBD\_NOT\_REMOVABLE\_MEDIA。

#### 4.22.5. vbd-insert

vbd-insert uuid=<uuid\_of\_vbd> vdi-uuid=<uuid\_of\_vdi\_containing\_media>

将新介质插入到 VBD 代表的驱动器中。仅当介质是可移动类型 ( 物理 CD 或 ISO ) 时，此命令才有效；否则，将返回错误消息 VBD\_NOT\_REMOVABLE\_MEDIA。

#### 4.22.6. vbd-plug

vbd-plug uuid=<uuid\_of\_vbd>

尝试在 VM 处于运行状态时连接 VBD。

#### 4.22.7. vbd-unplug

vbd-unplug uuid=<uuid\_of\_vbd>

尝试在 VM 处于运行状态时将 VBD 从 VM 分离。

### 4.23. VDI 命令

用于 VDI ( 虚拟磁盘映像 ) 的命令。

VDI 是一个软件对象，表示可对 VM 显示的虚拟磁盘的内容。而 VBD 则是一个将 VM 连接到 VDI 的连接器对象。VDI 中包含虚拟磁盘的物理属性信息 ( SR 类型、磁盘是否可共享、虚拟磁盘、介质是可读写介质还是只读介质等 )，而 VBD 中包含将 VDI 与 VM 相联系的属性 ( 可引导性、读/写指标等 )。

可使用标准对象列举命令 (xe vdi-list) 列出 VDI 对象，使用标准参数命令控制其参数。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.23.1. VDI 参数

VDI 包含下列参数：

参数名称	说明	类型
uuid	VDI 的唯一标识符/对象引用	只读
name-label	VDI 的名称	读取/写入
name-description	VDI 的描述字符串	读取/写入
allowed-operations	在此状态下允许执行的操作的列表	只读 set 参数
current-operations	此 VDI 上当前正在进行的操作列表	只读 set 参数
sr-uuid	VDI 所在的 SR	只读
vbd-uuids	引用此 VDI 的 VBD 的列表	只读 set 参数
crashdump-uuids	引用此 VDI 的故障转储的列表	只读 set 参数

参数名称	说明	类型
virtual-size	向 VM 提供的磁盘大小，以字节为单位。请注意，此大小可能会不准确，具体取决于存储后端的类型	只读
physical-utilisation	VDI 当前在 SR 上占用的物理空间量，以字节为单位	只读
type	VDI 的类型，例如系统或用户	只读
sharable	如果此 VDI 可以共享，则为 true	只读
read-only	如果此 VDI 仅可以装载为只读，则为 true	只读
storage-lock	如果此 VDI 锁定在存储级别，则为 true	只读
parent	如果此 VDI 是链的一部分，则引用父 VDI	只读
missing	如果 SR 扫描操作报告此 VDI 不存在，则为 true	只读
other-config	此 VDI 的其他配置信息	读取/写入 map 参数
sr-name-label	所包含的存储库的名称	只读
location	位置信息	只读
managed	如果 VDI 是托管的，则为 true	只读
xenstore-data	在连接 VDI 后将插入到 xenstore 树 (/local/domain/0/backend/vbd/<domid>/<device-id>/sm-data) 的数据。这通常由 vdi_attach 上的 SM 后端设置。	只读 map 参数
sm-config	SM 依赖数据	只读 map 参数
is-a-snapshot	如果此 VDI 是 VM 存储快照，则为 true	只读
snapshot_of	以此 VDI 为快照的存储的 UUID	只读
snapshots	此 VDI 的所有快照的 UUID	只读
snapshot_time	创建此 VDI 的快照操作的时间戳	只读
metadata-of-pool	创建此元数据 VDI 的池的 UUID	只读
metadata-latest	指示 VDI 中是否包含此池的最新已知元数据的标志	只读

#### 4.23.2. vdi-clone

vdi-clone uuid=<uuid\_of\_the\_vdi> [driver-params:<key=value>]

创建指定 VDI 的新可写副本，此副本可直接使用。它是 vdi-copy 的变体，可公开其所在的映像高速克隆工具。



可使用可选 `driver-params` 映射参数将额外的供应商特定的配置信息传递到 VDI 所基于的后端存储驱动程序。有关详细信息，请参阅存储供应商驱动程序文档。

#### 4.23.3. vdi-copy

```
vdi-copy uuid=<uuid_of_the_vdi> sr-uuid=<uuid_of_the_destination_sr>
```

将 VDI 复制到指定的 SR。

#### 4.23.4. vdi-create

```
vdi-create sr-uuid=<uuid_of_the_sr_where_you_want_to_create_the_vdi>
name-label=<name_for_the_vdi>
type=<system | user | suspend | crashdump>
virtual-size=<size_of_virtual_disk>
sm-config-*=<storage_specific_configuration_data>
```

创建 VDI。

`virtual-size` 参数可以以字节为单位指定，或者使用 IEC 标准后缀 KiB ( $2^{10}$  字节)、MiB ( $2^{20}$  字节)、GiB ( $2^{30}$  字节) 和 TiB ( $2^{40}$  字节) 进行指定。



#### 注意

支持磁盘的精简置备的 SR 类型（例如本地 VHD 和 NFS）不强制执行磁盘的虚拟分配。因此用户在 SR 上过量分配虚拟磁盘空间时应十分小心。当过量分配的 SR 变满时，SR 目标基底上必须提供可用磁盘空间，或通过删除 SR 中未使用的 VDI 来获得磁盘空间。



#### 注意

某些 SR 类型可能会对 `virtual-size` 值进行舍入处理，以使此值可被已配置块的大小整除。

#### 4.23.5. vdi-destroy

```
vdi-destroy uuid=<uuid_of_vdi>
```

销毁指定的 VDI。



#### 注意

对于本地 VHD 和 NFS SR 类型，使用 `vdi-destroy` 不能立即释放磁盘空间，而是在存储库扫描操作期间定期释放磁盘空间。如果用户需要强制使删除所得的磁盘空间可用，应当手动调用 `sr-scan`。

#### 4.23.6. vdi-forget

```
vdi-forget uuid=<uuid_of_vdi>
```

从数据库无条件删除 VDI 记录，不涉及存储后端。在正常操作中，您应当使用 `vdi-destroy`。

#### 4.23.7. vdi-import

```
vdi-import uuid=<uuid_of_vdi> filename=<filename_of_raw_vdi>
```

导入原始 VDI。

#### 4.23.8. vdi-introduce

```
vdi-introduce uuid=<uuid_of_vdi>
sr-uuid=<uuid_of_sr_to_import_into>
name-label=<name_of_the_new_vdi>
type=<system | user | suspend | crashdump>
location=<device_location_(varies_by_storage_type)>
[name-description=<description_of_vdi>]
[sharable=<yes | no>]
[read-only=<yes | no>]
[other-config=<map_to_store_misc_user_specific_data>]
[xenstore-data=<map_to_of_additional_xenstore_keys>]
[sm-config<storage_specific_configuration_data>]
```

创建代表现有存储设备的 VDI 对象，无需实际修改或创建任何存储。此命令主要在内部使用，用来自动引入热插入存储设备。

#### 4.23.9. vdi-pool-migrate

```
vdi-pool-migrate <uuid>=<VDI_uuid> <sr-uuid>=<destination-sr-uuid>
```

将 VDI 迁移到指定的 SR，同时将 VDI 连接到正在运行的来宾系统。(Storage XenMotion)

请参阅《XenServer 虚拟机用户指南》中的“VM 迁移”章节。

#### 4.23.10. vdi-resize

```
vdi-resize uuid=<vdi_uuid> disk-size=<new_size_for_disk>
```

调整由 UUID 指定的 VDI 的大小。

#### 4.23.11. vdi-snapshot

```
vdi-snapshot uuid=<uuid_of_the_vdi> [driver-params=<params>]
```

生成 VDI 的读写版本，该版本可用作进行备份和/或创建模板时的参考。除在 VM 内部安装和运行备份软件之外，您还可以通过快照执行备份。当外部备份软件将快照内容发送到备份介质时，VM 可以继续运行。同样，快照可用作“黄金映像”，可以基于此映像创建模板。可以使用任何 VDI 创建模板。

可使用可选 driver-params 映射参数将额外的供应商特定的配置信息传递到 VDI 所基于的后端存储驱动程序。有关详细信息，请参阅存储供应商驱动程序文档。

克隆快照应始终生成可写 VDI。

#### 4.23.12. vdi-unlock

```
vdi-unlock uuid=<uuid_of_vdi_to_unlock> [force=true]
```

尝试取消对指定 VDI 的锁定。如果将 force=true 传递给此命令，它将强制执行取消锁定操作。

### 4.24. VIF 命令

用于 VIF (虚拟网络接口) 的命令。

可使用标准对象列举命令 (xe vif-list) 列出 VIF 对象，使用标准参数命令控制其参数。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.24.1. VIF 参数

VIF 包含下列参数：

参数名称	说明	类型
uuid	VIF 的唯一标识符/对象引用	只读
vm-uuid	此 VIF 所驻留的 VM 的唯一标识符/对象引用	只读
vm-name-label	此 VIF 所驻留的 VM 的名称	只读
allowed-operations	在此状态下允许执行的操作的列表	只读 set 参数
current-operations	此 VIF 上当前正在进行的操作列表	只读 set 参数
device	此 VIF 的整数标签，指示 VIF 后端的创建顺序	只读
MAC	对 VM 公开的 VIF 的 MAC 地址	只读
MTU	<p>VIF 的最大传输单位（以字节为单位）。此参数为只读参数，但是您可以使用 other-config map 参数利用 mtu 键覆盖 MTU 设置。例如，在虚拟 NIC 上重置 MTU 以使用巨型帧：</p> <pre>xe vif-param-set \   uuid=&lt;vif_uuid&gt; \   other-config:mtu=9000</pre>	只读
currently-attached	如果当前已连接了设备，则为 true	只读
qos_algorithm_type	要使用的 QoS 算法	读取/写入
qos_algorithm_params	所选 QoS 算法的参数	读取/写入 map 参数
qos_supported_algorithm	此 VIF 支持 QoS 算法	只读 set 参数
MAC-autogenerated	如果自动生成 VIF 的 MAC 地址，则为 True	只读
other-config	其他配置键/值对	读取/写入 map 参数
other-config:ethtool-rx	设置为 on 会启用接收校验和，设置为 off 会禁用接收校验和	读写
other-config:ethtool-tx	设置为 on 会启用发送校验和，设置为 off 会禁用发送校验和	读写
other-config:ethtool-sg	设置为 on 会启用分散收集，设置为 off 会禁用分散收集	读写
other-config:ethtool-tso	设置为 on 会启用 tcp 分段卸载，设置为 off 会禁用 tcp 分段卸载	读写

参数名称	说明	类型
other-config:ethtool-ufo	设置为 on 会启用 udp 段卸载，设置为 off 会禁用 udp 段卸载	读写
other-config:ethtool-gso	设置为 on 会启用通用分段卸载，设置为 off 会禁用通用分段卸载	读写
other-config:promiscuous	为 true 时，VIF 在桥上为混杂模式，以便监测通过该桥的所有通信。对于在 VM 中运行入侵检测系统 (IDS) 或类似系统会有所帮助。	读写
network-uuid	此 VIF 连接到的虚拟网络的唯一标识符/对象引用	只读
network-name-label	此 VIF 连接到的虚拟网络的描述性名称	只读
io_read_kbs	此 VIF 的平均读取速率 (以 kb/s 为单位)	只读
io_write_kbs	此 VIF 的平均写入速率 (以 kb/s 为单位)	只读
locking_mode	影响 VIF 过滤到/来自 MAC 和 IP 地址列表的通信。需要其他参数。	读取/写入
locking_mode:default	因 VIF 网络的默认锁定模式而异。如果 default-locking-mode 设为 <b>&lt;禁用&gt;</b> ，则 XenServer 应用过滤规则，使 VIF 无法发送或接收通信。如果 default-locking-mode 设为 <b>&lt;已解除锁定&gt;</b> ，XenServer 将删除所有与 VIF 相关联的过滤规则。另请参阅第 4.11 节“网络命令”。	读取/写入
locking_mode:locked	仅允许通过 VIF 传输发送到指定 MAC 和 IP 地址的流量或从这些地址发送的流量。如果不指定 IP 地址，则不允许任何流量。	读取/写入
locking_mode:unlocke	不对任何传入或传出 VIF 的流量应用任何过滤器。	读取/写入
locking_mode:disablec	XenServer 会应用一个过滤规则，以使 VIF 丢弃所有通信。	读取/写入

#### 4.24.2. vif-create

vif-create vm-uuid=**<uuid\_of\_the\_vm>** device=**<see below>**  
network-uuid=**<uuid\_of\_the\_network\_the\_vif\_will\_connect\_to>** [mac=**<mac\_address>**]

在 VM 上新建 VIF。

device 字段中相应的值列在指定 VM 上的 allowed-VIF-devices 参数中。如果 VM 中尚不存在 VIF，则允许的值为从 0 到 15 之间的整数。

mac 参数为标准 MAC 地址，格式为 aa:bb:cc:dd:ee:ff。如果不指定此地址，则会创建一个适当的随机 MAC 地址。您还可以通过指定 mac=random 显式设置一个随机 MAC 地址。

#### 4.24.3. vif-destroy

vif-destroy uuid=**<uuid\_of\_vif>**

销毁 VIF。

#### 4.24.4. vif-plug

```
vif-plug uuid=<uuid_of_vif>
```

尝试在 VM 处于运行状态时连接 VIF。

#### 4.24.5. vif-unplug

```
vif-unplug uuid=<uuid_of_vif>
```

当 VM 处于运行状态时尝试将 VIF 从中分离出来。

### 4.25. VLAN 命令

与 VLAN ( 虚拟网络 ) 一起使用的命令。要列出和编辑虚拟接口，请参阅 PIF 命令，这些命令包含 VLAN 参数以指示虚拟接口具有关联虚拟网络 ( 请参阅第 4.14 节 “PIF 命令” )。例如，要列出 VLAN，需要使用 `xe pif-list`。

#### 4.25.1. vlan-create

```
vlan-create pif-uuid=<uuid_of_pif> vlan=<vlan_number> network-uuid=<uuid_of_network>
```

在 XenServer 主机上创建新 VLAN。

#### 4.25.2. pool-vlan-create

```
vlan-create pif-uuid=<uuid_of_pif> vlan=<vlan_number> network-uuid=<uuid_of_network>
```

通过确定特定网络所在的每个主机的接口 ( 例如 `eth0` )，并在每个相应的主机上创建和插入一个新的 PIF 对象，为池中的所有主机创建一个新 VLAN。

#### 4.25.3. vlan-destroy

```
vlan-destroy uuid=<uuid_of_pif_mapped_to_vlan>
```

销毁 VLAN。需要使用表示 VLAN 的 PIF 的 UUID。

### 4.26. VM 命令

控制 VM 及其属性的命令。

#### 4.26.1. VM 选择器

此处列出的几个命令具有一个共同的机制，即可选择一个或多个要对其执行操作的 VM。最简单的方法是提供参数 `vm=<name_or_uuid>`。例如，获取实际 VM 的 `uuid` 的一种简单的方法是执行 `xe vm-list power-state=running`。( 可匹配字段的完整列表可以通过命令 `xe vm-list params-all` 获取。 ) 例如，指定 `power-`

state=halted 将选择 power-state 参数的值为 halted 的所有 VM。如果有多个匹配的 VM，则必须指定选项 --multiple 才能执行操作。本部分开头介绍了可匹配的参数的完整列表，该列表可通过命令 `xe vm-list params=all` 获得。

可使用标准对象列表命令 (`xe vm-list`) 以及使用标准参数命令操纵的参数列出 VM 对象。有关详细信息，请参阅第 3.2 节“低级参数命令”。

#### 4.26.2. VM 参数

VM 具有下列参数：



##### 注意

当 VM 运行时，可以更改所有可写入的 VM 参数值，但不会动态应用新参数，并且重新引导 VM 之后才会应用新参数。

参数名称	说明	类型
appliance	VM 所属的设备/vApp	读取/写入
uuid	VM 的唯一标识符/对象引用	只读
name-label	VM 的名称	读取/写入
name-description	VM 的描述字符串	读取/写入
order	vApp 启动/关闭功能的启动顺序，以及进行高可用性故障转移之后的启动顺序	读取/写入
version	此 VM 已经恢复的次数，如果用户希望用旧版 VM 覆盖新 VM，则必须调用 vm-recover	只读
user-version	描述 VM 和模板的创建者的字符串，用于输入版本信息	读取/写入
is-a-template	除非为模板，否则为 false；始终无法启动模板 VM，它们只能用于克隆其他 VM	读取/写入
is-control-domain	如果是控制域（域 0 或驱动程序域），则为 true	只读
power-state	当前电源状态	只读
start delay	在用来启动 VM 的调用返回之前等待的时间	读取/写入
shutdown-delay	在用来关闭 VM 的调用返回之前等待的时间	读取/写入
memory-dynamic-max	最大动态内存，以字节为单位	读取/写入
memory-dynamic-min	最小动态内存，以字节为单位	读取/写入

参数名称	说明	类型
memory-static-max	静态集（绝对）最大值，以字节为单位。  如果要更改此值，必须关闭 VM。	读取/写入
memory-static-min	静态集（绝对）最小值，以字节为单位。如果要更改此值，必须关闭 VM。	读取/写入
suspend-VDI-uuid	存储挂起映像的 VDI	只读
VCPUs-params	<p>所选 VCPU 策略的配置参数。</p> <p>可以使用以下命令调整 VCPU 分配</p> <pre>xe vm-param-set \ uuid=&lt;vm_uuid&gt; \ VCPUs-params:mask=1,2,3</pre> <p>然后，所选 VM 仅在物理 CPU 1、2 和 3 上运行。</p> <p>还可以使用 cap 和 weight 参数调整 VCPU 优先级（xen 调度）；例如</p> <pre>xe vm-param-set \ uuid=&lt;vm_uuid&gt; \ VCPUs-params:weight=512 xe vm-param-set \ uuid=&lt;vm_uuid&gt; \ VCPUs-params:cap=100</pre> <p>在竞争的 XenServer 主机上，权重为 512 的 VM 获得的 CPU 将是权重为 256 的域的两倍。合法的权重范围为 1 到 65535，默认值为 256。</p> <p>即使 XenServer 主机具有空闲 CPU 周期，该上限也可以选择性地确定 VM 能够占用的最大 CPU 量。此上限以一个物理 CPU 的百分比表示：100 是 1 个物理 CPU，50 是半个物理 CPU，400 是 4 个物理 CPU。默认值为 0，表示无上限。</p>	读取/写入 map 参数
VCPUs-max	虚拟 CPU 的最大数量	读取/写入
VCPUs-at-startup	虚拟 CPU 的引导次数	读取/写入

参数名称	说明	类型
actions-after-crash	VM 崩溃后采取的措施。preserve ( 仅适用于分析 )、coredump_and_restart ( 记录内核转储并重新引导 VM )、coredump_and_destroy ( 记录内核转储并使 VM 保持为停止状态 )、restart ( 不记录内核转储，但重新引导 VM ) 以及 destroy ( 不记录内核转储，但使 VM 保持为停止状态 )。	读取/写入
console-uuids	虚拟控制台设备	只读 set 参数
platform	特定于平台的配置	读取/写入 map 参数
allowed-operations	此状态下允许的操作的列表	只读 set 参数
current-operations	VM 上当前正在执行的操作的列表	只读 set 参数
allowed-VBD-devices	可供使用的 VBD 标识符的列表，用 0 到 15 之间的整数表示。此列表仅供参考，可以使用未在此列表中列出的其他设备 ( 但可能无法正常运行 )。	只读 set 参数
allowed-VIF-devices	可供使用的 VIF 标识符的列表，用 0 到 15 之间的整数表示。此列表仅供参考，可以使用未在此列表中列出的其他设备 ( 但可能无法正常运行 )。	只读 set 参数
HVM-boot-policy	HVM 来宾系统的引导策略。值为 BIOS Order 或空字符串。	读取/写入
HVM-boot-params	order 关键字用于控制 HVM 来宾系统引导顺序，以字符串表示，其中每个字符代表一种引导方法：d 代表 CD/DVD，c 代表根磁盘，n 代表网络 PXE 引导。默认值为 dc。	读取/写入 map 参数
HVM-shadow-multiplier	浮点值，用于控制授予 VM 的影子内存开销量。默认值为 1.0 ( 最小值 )，只有高级用户才能更改此值。	读取/写入
PV-kernel	内核路径	读取/写入
PV-ramdisk	initrd 路径	读取/写入



参数名称	说明	类型
PV-args	内核命令行参数字符串	读取/写入
PV-legacy-args	引导传统 VM 的参数字符串	读取/写入
PV-bootloader	引导加载程序的名称或路径	读取/写入
PV-bootloader-args	引导加载程序杂项参数的字符串	读取/写入
last-boot-CPU-flags	描述上次引导 VM 时所在的 CPU 的标志	只读
resident-on	VM 当前所驻留的 XenServer 主机	只读
affinity	VM 要优先在其上运行的 XenServer 主机；xe vm-start 命令使用该参数确定运行 VM 的位置	读取/写入
other-config	为 VM 指定其他配置参数的键/值对列表  例如，如果 other-config 参数包括键/值对 auto_poweron: true，则引导主机后将自动启动 VM	读取/写入 map 参数
start-time	读取 VM 指标的日期和时间的戳，以 yyyyymmddThh:mm:ss z 的形式表示，其中 z 是单字母军用时区指示器，例如，Z 表示 UTC (GMT)	只读
install-time	读取 VM 指标的日期和时间的戳，以 yyyyymmddThh:mm:ss z 的形式表示，其中 z 是单字母军用时区指示器，例如，Z 表示 UTC (GMT)	只读
memory-actual	VM 使用的实际内存	只读

参数名称	说明	类型
VCPUs-number	<p>分配给 VM 的虚拟 CPU 的数量</p> <p>对于半虚拟化的 Linux VM，此数量可能与 VCPUS-max 不同。使用 vm-vcpu-hotplug 命令就可以对此数量进行更改，而不必重新引导 VM。请参阅第 4.26.34 节 “vm-vcpu-hotplug”。Windows VM 始终在 vCPU 数量设置为 VCPUs-max 的情况下运行并且必须重新引导 Windows VM 才可以更改此值。</p> <p>请注意，如果为 VCPUs-number 设置的值大于 XenServer 主机上的物理 CPU 数量，性能将急剧下降。</p>	只读
VCPUs-Utilization	虚拟 CPU 及其权重的列表	只读 map 参数
os-version	VM 的操作系统版本	只读 map 参数
PV-drivers-version	VM 的半虚拟化驱动程序版本	只读 map 参数
PV-drivers-detected	VM 的半虚拟化驱动程序最新版本的标志	只读
memory	VM 上代理报告的内存指标	只读 map 参数
disks	VM 上代理报告的磁盘指标	只读 map 参数
networks	VM 上代理报告的网络指标	只读 map 参数
other	VM 上代理报告的其他指标	只读 map 参数
guest-metrics-last-updated	来宾代理上次写入这些字段的时间戳，以 yyyyymmddThh:mm:ss z 的形式表示，其中 z 是单字母军用时区指示器，例如，Z 表示 UTC (GMT)	只读
actions-after-shutdown	将在 VM 关闭后执行的操作	读取/写入
actions-after-reboot	将在 VM 重新引导后执行的操作	读取/写入
possible-hosts	可能驻留此 VM 的主机	只读
dom-id	域 ID (如果可用，否则为 -1)	只读

参数名称	说明	类型
recommendations	此 VM 的建议值和属性范围的 XML 规范	只读
xenstore-data	在创建 VM 后将插入到 xenstore 树 (/local/domain/<domid>/vm-data) 的数据。	读取/写入 map 参数
is-a-snapshot	如果此 VM 是快照，则为 true	只读
snapshot_of	此快照针对的 VM 的 UUID	只读
snapshots	此 VM 的所有快照的 UUID	只读
snapshot_time	创建此 VM 快照的快照操作的时间戳	只读
memory-target	为此 VM 设置的目标内存量	只读
blocked-operations	列出无法在此 VM 上执行的操作	读取/写入 map 参数
last-boot-record	此模板的上次引导参数的记录，使用 XML 格式	只读
ha-always-run	如果当此 VM 所在的主机发生故障时，它始终在另一台主机上重新启动，则为 true	读取/写入
ha-restart-priority	1、2、3 或 best-effort。1 是最高重新启动优先级	读取/写入
blobs	二进制数据存储	只读
live	如果 VM 正在运行，则为 true；如果高可用性功能猜测 VM 可能没有运行，则为 false。	只读

#### 4.26.3. vm-assert-can-be-recovered

vm-assert-can-be-recovered <uuid> [<database>] <vdi-uuid>

测试存储是否可用于恢复此 VM。

#### 4.26.4. vm-cd-add

vm-cd-add cd-name=<name\_of\_new\_cd> device=<integer\_value\_of\_an\_available\_vbd> [<vm-selector>=<vm\_selector\_value>...]

将新虚拟 CD 添加到所选的 VM。应从 VM 的 allowed-VBD-devices 参数值中选择 device 参数。

将使用标准选择机制来选择应在其上执行此操作的 VM (请参阅 [VM 选择器](#))。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.5. vm-cd-eject

```
vm-cd-eject [<vm-selector>=<vm_selector_value>...]
```

从虚拟 CD 驱动器弹出 CD。仅当有且只有一个 CD 连接到 VM 时，此命令才会起作用。如果有两个或更多 CD，请使用命令 `xe vbd-eject`，并指定 VBD 的 UUID。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.6. vm-cd-insert

```
vm-cd-insert cd-name=<name_of_cd> [<vm-selector>=<vm_selector_value>...]
```

将 CD 插入到虚拟 CD 驱动器。当有且仅有一个空 CD 设备连接到 VM 时，此命令才会起作用。如果有两个或多个空 CD 设备，请使用命令 `xe vbd-insert`，并指定要插入的 VBD 和 VDI 的 UUID。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.7. vm-cd-list

```
vm-cd-list [vbd-params] [vdi-params] [<vm-selector>=<vm_selector_value>...]
```

列出连接到指定 VM 的 CD。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

您还可以选择要列出哪些 VBD 和 VDI 参数。

## 4.26.8. vm-cd-remove

```
vm-cd-remove cd-name=<name_of_cd> [<vm-selector>=<vm_selector_value>...]
```

从指定 VM 中移除虚拟 CD。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.9. vm-clone

```
vm-clone new-name-label=<name_for_clone>  
[new-name-description=<description_for_clone>] [<vm-selector>=<vm_selector_value>...]
```

如果可用，请使用存储级别的快速磁盘克隆操作克隆现有的 VM。使用 `new-name-label` 和 `new-name-description` 参数为生成的克隆 VM 指定名称和可选说明。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.10. vm-compute-maximum-memory

```
vm-compute-maximum-memory total=<amount_of_available_physical_ram_in_bytes>
```

```
[approximate=<add overhead memory for additional vCPU? true | false>]
[<vm_selector>=<vm_selector_value>...]
```

通过将物理 RAM 的总量用作上限来计算可以分配给现有 VM 的最大静态内存量。可选参数 `approximate` 在计算过程中保留足够的额外内存，以便以后将额外的 vCPU 添加到 VM。

例如：

```
xe vm-compute-maximum-memory vm=testvm total=`xe host-list params=memory-free --minimal`
```

此命令使用 `xe host-list` 命令返回的 `memory-free` 参数的值设置名为 `testvm` 的 VM 的最大内存。

使用标准选择机制选择要在其上执行操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

#### 4.26.11. vm-copy

```
vm-copy new-name-label=<name_for_copy> [new-name-description=<description_for_copy>]
[sr-uuid=<uuid_of_sr>] [<vm-selector>=<vm_selector_value>...]
```

复制现有 VM，但不使用存储级别的快速磁盘克隆操作 ( 即使该操作可用 )。保证 VM 副本的磁盘映像为“完整映像”，即该磁盘映像不是写入时复制 (CoW) 链的一部分。

使用 `new-name-label` 和 `new-name-description` 参数为生成的 VM 副本指定名称和可选说明。

使用 `sr-uuid` 为生成的 VM 副本指定目标 SR。如果未指定此参数，则目标 SR 与原始 VM 所在的 SR 相同。

将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

#### 4.26.12. vm-crashdump-list

```
vm-crashdump-list [<vm-selector>=<vm selector value>...]
```

列出与指定 VM 关联的故障转储。

如果使用可选参数 `params`，则 `params` 的值是一个字符串，其中包含您希望显示的此对象的参数列表。或者，可以使用关键字 `all` 显示所有参数。如果不使用 `params`，则返回的列表将显示所有可用参数的默认子集。

将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

#### 4.26.13. vm-data-source-list

```
vm-data-source-list [<vm-selector>=<vm selector value>...]
```

列出可为 VM 记录的数据源。

使用标准选择机制 ( 请参阅 [VM 选择器](#) ) 选择将执行此操作的 VM。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。如果未给定用于选择主机的参数，将在所有 VM 上执行操作。

数据源具有两个参数 — `standard` 和 `enabled`，如此命令的输出所示。如果数据源的 `enabled` 设置为 `true`，则指标当前记录到性能数据库中。如果数据源的 `standard` 设置为 `true`，则默认情况下指标会记录到性能数据库中 ( 因此，该数据源的 `enabled` 也将设置为 `true` )。如果数据源的 `standard` 设置为 `false`，则默认情况下指标不会记录到性能数据库中 ( 因此，该数据源的 `enabled` 也将设置为 `false` )。



要开始将数据源指标记录到性能数据库，请运行 `vm-data-source-record` 命令。这会将 `enabled` 设置为 `true`。要停止记录，请运行 `vm-data-source-forget`。这会将 `enabled` 设置为 `false`。

#### 4.26.14. vm-data-source-record

```
vm-data-source-record data-source=<name_description_of_data-source> [<vm-selector>=<vm selector value>...]
```

记录 VM 的指定数据源。

此操作会将数据源中的信息写入指定 VM 的静态性能指标数据库中。由于性能的原因，此数据库不同于普通的代理数据库。

使用标准选择机制（请参阅 [VM 选择器](#)）选择将执行此操作的 VM。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。如果未给定用于选择主机的参数，将在所有 VM 上执行操作。

#### 4.26.15. vm-data-source-forget

```
vm-data-source-forget data-source=<name_description_of_data-source> [<vm-selector>=<vm selector value>...]
```

停止为 VM 记录指定数据源并忽略所有已记录的数据。

使用标准选择机制（请参阅 [VM 选择器](#)）选择将执行此操作的 VM。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。如果未给定用于选择主机的参数，将在所有 VM 上执行操作。

#### 4.26.16. vm-data-source-query

```
vm-data-source-query data-source=<name_description_of_data-source> [<vm-selector>=<vm selector value>...]
```

显示 VM 的指定数据源。

使用标准选择机制（请参阅 [VM 选择器](#)）选择将执行此操作的 VM。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。如果未给定用于选择主机的参数，将在所有 VM 上执行操作。

#### 4.26.17. vm-destroy

```
vm-destroy uuid=<uuid_of_vm>
```

销毁指定的 VM。这会使与 VM 关联的存储保留不变。要同时删除存储，请使用 `xe vm-uninstall`。

#### 4.26.18. vm-disk-add

```
vm-disk-add disk-size=<size_of_disk_to_add> device=<uuid_of_device> [<vm-selector>=<vm_selector_value>...]
```

将新磁盘添加到指定 VM。从 VM 的 `allowed-VBD-devices` 参数值中选择 `device` 参数。

`disk-size` 参数可以使用字节来指定，也可以使用 IEC 标准后缀 KiB ( $2^{10}$  字节)、MiB ( $2^{20}$  字节)、GiB ( $2^{30}$  字节) 以及 TiB ( $2^{40}$  字节) 来指定。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.19. vm-disk-list

```
vm-disk-list [vbd-params] [vdi-params] [<vm-selector>=<vm_selector_value>...]
```

列出连接到指定 VM 的磁盘。vbd-params 和 vdi-params 参数控制要输出的各个对象的字段，应以逗号分隔的列表形式给出，或使用特殊键 all 表示给出完整列表。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.20. vm-disk-remove

```
vm-disk-remove device=<integer_label_of_disk> [<vm-selector>=<vm_selector_value>...]
```

从指定 VM 移除磁盘并将其销毁。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.21. vm-export

```
vm-export filename=<export_filename>
[metadata=<true | false>]
[<vm-selector>=<vm_selector_value>...]
```

将指定 VM（包括磁盘映像）导出到本地计算机上的文件中。使用 filename 参数指定将 VM 导出到的文件的文件名。按照惯例，此文件名的扩展名应该为 .xva。

如果 metadata 参数为 true，则不会导出磁盘，而仅将 VM 元数据写入到输出文件中。这适用于基础存储通过其他机制进行传输的情况，允许重新创建 VM 信息（请参阅 [第 4.26.22 节“vm-import”](#)）。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

## 4.26.22. vm-import

```
vm-import filename=<export_filename>
[metadata=<true | false>]
[preserve=<true | false>]
[sr-uuid=<destination_sr_uuid>]
```

从之前导出的文件导入 VM。如果将 preserve 设置为 true，则将保留原始 VM 的 MAC 地址。sr-uuid 确定将 VM 导入到的目标 SR；如果未指定，则导入到默认 SR。

filename 参数还可以指向 XVA 格式的 VM，该格式是自 XenServer 3.2 以来的传统导出格式，一些第三方供应商使用 XVA 格式来提供虚拟设备。该格式使用目录存储 VM 数据，因此将 filename 设置为 XVA 导出的根目录而不是实际的文件。导入的传统来宾系统的后续导出内容将自动升级到新的基于文件名的格式，这种格式可以存储更多关于 VM 配置的数据。



## 注意

早期的基于目录的 XVA 格式不能完整保留所有 VM 属性。尤其是默认情况下导入的 VM 不会连接任何虚拟网络接口。如果需要网络，则使用 vif-create 和 vif-plug 创建一个网络。

如果 metadata 为 true，则可以导入之前导出的一组元数据，而不会导入与之关联的磁盘块。如果找不到 VDI（按照 SR 和 VDI.location 命名），只导入元数据的操作将会失败，除非指定 --



force 选项，这时无论遇到什么情况导入都会继续。如果可以对磁盘进行镜像或将其移动到带外，则元数据导入/导出是一种在无连接的池之间移动 VM 的快捷方法（例如，作为灾难恢复方案的一部分）。



#### 注意

以串行方式导入多个 VM 比使用并行方式更快。

### 4.26.23. vm-install

```
vm-install new-name-label=<name>
[ template-uuid=<uuid_of_desired_template> | [template=<uuid_or_name_of_desired_template>]]
[ sr-uuid=<sr_uuid> | sr-name-label=<name_of_sr> ]
[ copy-bios-strings-from=<uuid of host> ]
```

基于模板安装或克隆 VM。使用 template-uuid 或 template 参数指定模板名称。使用 sr-uuid 或 sr-name-label 参数指定 SR。指定使用 copy-bios-strings-from 参数安装锁定了 BIOS 的介质。



#### 注意

默认情况下，在基于包含现有磁盘的模板进行安装时，将在现有磁盘所在的同一个 SR 中创建新磁盘。如果该 SR 支持基于模板的安装，将进行快速复制。如果在命令行上指定了其他 SR，则将在该 SR 中创建新磁盘。在这种情况下，无法实现快速复制，磁盘将进行完整复制。

基于不含现有磁盘的模板进行安装时，将在指定的 SR 中创建任何新磁盘；如果没有指定 SR，则将在池的默认 SR 中创建新磁盘。

### 4.26.24. vm-memory-shadow-multiplier-set

```
vm-memory-shadow-multiplier-set [<vm-selector>=<vm_selector_value>...]
[multiplier=<float_memory_multiplier>]
```

设置指定 VM 的影子内存系数。

这是用于修改分配给硬件辅助 VM 的影子内存量的高级选项。在 Citrix XenApp 等某些专用应用程序工作负载中，需要具有额外的影子内存才可以获得最佳性能。

此内存被认为是一种系统开销。在计算 VM 内存时，此种内存与普通内存分开计算。调用此命令后，可用 XenServer 主机内存量将减少（减少量视系数而定），并且 HVM\_shadow\_multiplier 字段将使用 Xen 分配给 VM 的实际值进行更新。如果没有足够的可用 XenServer 主机内存，将返回错误。

应在其上执行此操作的 VM 使用标准选择机制进行选择（有关详细信息，请参阅 [VM 选择器](#)）。

### 4.26.25. vm-migrate

```
vm-migrate    [<copy>=<true|false>]    [host-uuid=<destination_host_UUID>]    [host=<name    or
UUID    of    destination    host>]    [<force>=<true|false>]    [<live>=<true|false>]    [<vm-
selector>=<vm_selector_value>...]    [<remote-master>=<destination_pool_master_uuid>]    [<remote-
username>=<destination_pool_username>]    [<remote-password>=<destination_pool_password>]
[<remote-network>=<destination_pool_network_uuid>] [<vif>=<vif_uuid>] [<vdi>=<vdi_uuid>]
```

此命令在物理主机之间迁移指定的 VM。host 参数可以是 XenServer 主机的名称或 UUID。例如，要将 VM 迁移到池中的其他主机，其中，VM 磁盘位于两个主机共享的存储上（称为 XenMotion），请执行以下操作：

```
xe vm-migrate uuid=<vm_uuid> remote-master=12.34.56.78 \
remote-username=<username> remote-password=<password> \
host-uuid=<desination_host_uuid>
```



要在同一个池中不共享存储的主机之间迁移 VM (Storage XenMotion) :

```
xe vm-migrate uuid=<vm_uuid> remote-master=12.34.56.78 \
remote-username=<username> remote-password=<password> \
host-uuid=<desination_host_uuid> vdi=<vdi_uuid>
```

您可以选择存储每个 VDI 的 SR :

```
xe vm-migrate uuid=<vm_uuid> host-uuid=<destination_host_uuid> \
vdi1:<vdi_1_uuid>=<destination_sr_uuid> \
vdi2:<vdi_2_uuid>=<destination_sr2_uuid> \
vdi3:<vdi_3_uuid>=<destination_sr3_uuid>
```

此外，您还可以选择迁移后要连接 VM 的网络：

```
xe vm-migrate uuid=<vm_uuid> \
vdi1:<vdi_1_uuid>=<destination_sr_uuid> \
vdi2:<vdi_2_uuid>=<destination_sr2_uuid> \
vdi3:<vdi_3_uuid>=<destination_sr3_uuid> \
vif:<vif_uuid>=<network_uuid>
```

对于跨池迁移：

```
xe vm-migrate uuid=<vm_uuid> remote-master=12.34.56.78
remote-username=<username> remote-password=<password> \
host-uuid=<desination_host_uuid> vdi=<vdi_uuid>
```

有关 Storage XenMotion、XenMotion 和实时 VDI 迁移的详细信息，请参阅《VM 用户指南》。

默认情况下，VM 将挂起、迁移并在另一台主机上恢复。live 参数激活 XenMotion，并使 VM 在执行迁移时继续运行，从而使 VM 停机时间缩短至少于一秒。在某些情况下（如 VM 中的内存工作负载繁重），XenMotion 将自动恢复到默认模式，并在完成内存传输前将 VM 挂起一小段时间。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

#### 4.26.26. vm-reboot

```
vm-reboot [<vm-selector>=<vm_selector_value>...] [force=<true>]
```

重新引导指定的 VM。

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

使用 force 参数将导致非正常关机，与拔掉物理服务器上的插头相似。

#### 4.26.27. vm-recover

```
vm-recover <vm-uuid> [<database>] [<vdi-uuid>] [<force>]
```

从提供的 VDI 中包含的数据库恢复 VM。

#### 4.26.28. vm-reset-powerstate

```
vm-reset-powerstate [<vm-selector>=<vm_selector_value>...] {force=true}
```

将使用标准选择机制来选择应在其上执行此操作的 VM（请参阅 [VM 选择器](#)）。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。



这是一个高级命令，仅在池中的成员主机出现故障时使用。您可以使用此命令强制池主服务器将 VM 的电源状态重置为 halted。实际上这将强制锁定 VM 及其磁盘，以便随后可在另一台池主机上启动此 VM。此调用要求指定 force 标志，如果命令行中不存在 force 标志，则调用将失败。

#### 4.26.29. vm-resume

```
vm-resume [<vm-selector>=<vm_selector_value>...] [force=<true | false>] [on=<XenServer host UUID>]
```

恢复指定的 VM。

将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

如果 VM 位于主机池内的共享 SR 上，则使用 on 参数指定要在其上启动 VM 的主机。默认情况下，系统将确定一台适合的主机 ( 可以是池中的任何成员 )。

#### 4.26.30. vm-shutdown

```
vm-shutdown [<vm-selector>=<vm_selector_value>...] [force=<true | false>]
```

关闭指定的 VM。

将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

使用 force 参数将导致非正常关机，与拔掉物理服务器上的插头相似。

#### 4.26.31. vm-start

```
vm-start [<vm-selector>=<vm_selector_value>...] [force=<true | false>] [on=<XenServer host UUID>] [--multiple]
```

启动指定的 VM。

将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

如果 VM 位于主机池中的共享 SR 上，则使用 on 参数指定池中在其上启动 VM 的主机。默认情况下，系统将确定一台适合的主机 ( 可以是池中的任何成员 )。

#### 4.26.32. vm-suspend

```
vm-suspend [<vm-selector>=<vm_selector_value>...]
```

挂起指定的 VM。

将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

#### 4.26.33. vm-uninstall

```
vm-uninstall [<vm-selector>=<vm_selector_value>...] [force=<true | false>]
```

卸载 VM — 销毁其磁盘 ( 标记了 RW 且仅连接到此 VM 的 VDI ) 及其元数据记录。要仅销毁 VM 元数据，请使用 xe vm-destroy。



将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

#### 4.26.34. vm-vcpu-hotplug

```
vm-vcpu-hotplug new-vcpus=<new_vcpu_count> [<vm-selector>=<vm_selector_value>...]
```

在参数 `VCPUs-max` 限定的数量范围内，动态调整正在运行的半虚拟化 Linux VM 可用的 VCPU 数量。Windows VM 始终在 vCPU 数量设置为 `VCPUs-max` 的情况下运行并且必须重新引导 Windows VM 才可以更改此值。

应在其上执行此操作的半虚拟化 Linux VM 使用标准选择机制进行选择 ( 请参阅 [VM 选择器](#) )。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。

#### 4.26.35. vm-vif-list

```
vm-vif-list [<vm-selector>=<vm_selector_value>...]
```

列出指定 VM 中的 VIF。

将使用标准选择机制来选择应在其上执行此操作的 VM ( 请参阅 [VM 选择器](#) )。请注意，过滤时选择器对 VM 记录执行操作，而不对 VIF 值执行操作。可选参数可以为 [VM 参数](#) 部分开头列出的参数中的任意数个。