

KVM 虚拟化进阶与提高

作者：李杰

前言

该内容为个人学习 KVM 虚拟化进阶与提高过程中所记述的笔记，内容主要包括虚拟网络高级特性、虚拟机迁移、KVM 群集、嵌套虚拟化、性能监视与优化、P2V、V2V 迁移、备份与恢复、oVirt (RHEV) 安装和基本管理等方面的内容。

虚拟网络高级特性包括：多物理网卡的绑定、配置 VLAN、网络过滤。**虚拟机迁移**：静态迁移和动态迁移。**KVM 群集三种方式**：NFS、GFS2、OCFS2。**嵌套虚拟化**：KVM 虚拟机中运行 KVM 虚拟机。**性能监视与优化**：CPU、内存、IO、网络等性能监视与优化。**P2V**：物理机转虚拟机。**V2V**：虚拟机转虚拟机。**备份与恢复**：快照和备份。**oVirt (RHEV)** 安装和基本管理。

李杰

2025 年 5 月 19 日

术 语 表

KVM 基于内核的虚拟机 (Kernel-based Virtual Machine), 是 Linux 内核中的一个模块, 允许在 Linux 系统上运行虚拟机。

VLAN 虚拟局域网 (Virtual Local Area Network), 是一种在单一物理网络上创建多个逻辑网络的技术。

oVirt 红帽企业虚拟化管理工具, 提供了一个集中管理虚拟化基础设施的平台。

P2V 物理机转虚拟机 (Physical to Virtual), 将物理服务器转换为虚拟机的过程。

V2V 虚拟机转虚拟机 (Virtual to Virtual), 将一种虚拟化平台上的虚拟机转换到另一种虚拟化平台的过程。

目 录

第 1 章	虚拟网络的高级特性	1
1.1	多物理网卡的绑定	1
1.1.1	配置多网卡绑定的 KVM 桥接模式	1



第 1 章 虚拟网络的高级特性

1.1 多物理网卡的绑定

如果一台服务器只有一块物理网卡，那么虚拟机只能使用这块网卡，无法使用其他的网卡。为了提高网络性能（提供负载平衡与冗余），可以将多块物理网卡绑定在一起，形成一个虚拟网卡，这样虚拟机就可以使用这个虚拟网卡。

1.1.1 配置多网卡绑定的 KVM 桥接模式

接下来将介绍如何配置多网卡绑定的 KVM 桥接模式。假设有两块物理网卡 `eth0` 和 `eth1`，分别连接到不同的交换机上。我们将这两块网卡绑定在一起，形成一个虚拟网卡 `bond0`，然后将 `bond0` 配置为 KVM 桥接模式。

【实验 1.1】（配置多网卡绑定的 KVM 桥接模式）

1. 绑定网卡

- (a) 启用 bonding 模块
- (b) 配置物理网卡
- (c) 配置绑定接口
- (d) 重新启动服务
- (e) 测试

2. 配置桥接

首先使用 `ip link`、`ifconfig` 指令查看当前的网络配置，确认与 `eth0` 和 `eth1` 相关的信息。接在使用 `virtsh` 或者 `virt-manager` 新建网络接口，选择桥接模式，网卡选择 `bond0`。接着使用 `ifconfig bond0` 指令查看 `bond0` 的状态，确认 `bond0` 已经创建成功。接着使用 `modprobe bonding` 指令启用 bonding 模块。使用 `lsmod grep bonding` 指令查看当前加载的模块，确认 bonding 模块已经加载成功。在主网卡 `eth0` 和 `eth1` 的配置文件中添加以下内容（提前做好备份）：

1.1 多物理网卡的绑定

代码 1.1: 网卡配置文件

```
1  # ifcfg-eth0
2  {
3      Type=Ethernet
4      BOOTPROTO=none
5      NAME=eth0
6      DEVICE=eth0
7      ONBOOT=yes
8      MASTER=bond0 # 绑定到bond0
9      SLAVE=yes
10     NM_CONTROLLED=no # 不受NetworkManager控制
11     USERCTL=no # 不允许用户控制
12 }
13 # ifcfg-eth1
14 {
15     Type=Ethernet
16     BOOTPROTO=none
17     NAME=eth1
18     DEVICE=eth1
19     ONBOOT=yes
20     MASTER=bond0
21     SLAVE=yes
22     NM_CONTROLLED=no
23     USERCTL=no
24 }
25 # ifcfg-bond0
26 {
27     DEVICE=bond0
28     ONBOOT=yes
29     NM_CONTROLLED=no
30     USERCTL=no
31     BONDING_OPTS="mode=1 miimon=100"
32     BOOTPROTO=static
33     IPADDR=192.168.200.11
34     NETMASK=255.255.255.0
35 }
```

以上只有 TYPE、BOOTPROTO、DEVICE、ONBOOT、MASTER、SLAVE、NM_CONTROLLED、

USERCTL、BONDING_OPTS、BOOTPROTO、IPADDR、NETMASK 等参数是必须的，其他参数可以根据需要添加。接着使用 `ifup bond0` 指令启用 bond0 接口，使用 `ifconfig bond0` 指令查看 bond0 的状态，确认 bond0 已经创建成功。

【思考 1.1.1】对于 mode 有七种选项，分别是

- **mode=0 (balance-rr)**: 轮询模式，数据包按顺序在每个接口上发送，提供负载均衡和容错能力。
- **mode=1 (active-backup)**: 主备模式，只有一个接口处于活动状态，其他接口作为备份，提供高可用性。
- **mode=2 (balance-xor)**: 基于源 MAC 地址和目标 MAC 地址的哈希算法选择接口，提供负载均衡和容错能力。
- **mode=3 (broadcast)**: 广播模式，所有数据包在所有接口上发送，提供容错能力。
- **mode=4 (802.3ad)**: IEEE 802.3ad 动态链路聚合模式，需要交换机支持，提供负载均衡和高性能。
- **mode=5 (balance-tlb)**: 适配器传输负载均衡，根据负载动态分配流量，接收流量由当前接口处理。
- **mode=6 (balance-alb)**: 适配器负载均衡，扩展了 mode=5，支持接收和发送流量的负载均衡。