第 5 章 集群管理工具

Proxmox VE 集群管理工具 pvecm 用于创建一个由多个物理服务器节点构成的“组”。这样的一组服务器称为一个“集群”。我们使用 Corosync Cluster Engine 来确保集群通信的稳定可靠，目前一个集群最多可拥有 32 个物理节点（也可以更多，关键在于网络时延）。

使用 pvecm 可以创建新的集群，可以向集群新增节点，可以从集群删除节点，可以查看集群状态信息，也可以完成其他各种集群管理操作。Proxmox VE 集群文件系统（pmxcfs）用于确保配置信息透明地发送到集群中所有节点，并保持一致。

以集群方式使用 Proxmox VE 有以下优势：

- 集中的 web 管理
- 多主集群架构：从任何一个节点都可以管理整个集群
- pmxcfs：以数据库驱动的文件系统保存配置文件，并通过 corosync 在确保所有节点的配置信息实时同步。
- 虚拟机和容器可方便地在物理服务器节点之间迁移。
- 快速部署。
- 基于集群的防火墙和 HA 服务。

5.1 部署要求

- 所有节点必须在同一子网，以便各个节点使用 corosync 多播通信（详情可查看 Corosync Cluster Engine）。Corosync 使用 UDP 5404 和 5405 端口进行集群通信。

➢ 注意

有些交换机默认设置关闭了 IP 多播，需要先手工启用多播通信。

- 各节点日期和时间需要保持同步。
- 各节点之间要能够在 TCP 22 端口建立 SSH 通信。
- 如果你需要配置 HA，则最少需要 3 个物理服务器节点，以保证集群多数票机制生效。此外，还需要保证所有节点使用同一版本的 Proxmox VE。
- 我们建议为集群通信分配专用网卡，特别是在配置共享存储的情况下，分配专用网卡能确保集群通信的稳定可靠。

➢ 注意

Proxmox VE 3.x 或更早版本不能和 Proxmox VE 4.x 混合组建集群。

5.2 节点服务器准备

首先需要在物理服务器节点安装 Proxmox VE。确保所有节点的主机名和 IP 地址都配置妥当。加入集群后将不允许再修改主机名和 IP 地址。

目前，创建集群操作必须在命令行控制台下进行，所以你必须通过 ssh 登录节点服务器以便进行操作。

5.3 创建集群

首先通过 ssh 远程登录一个 Proxmox VE 节点。然后为你的集群想一个名字，注意不要和已有的集群重名。一旦建立集群后，将不允许修改集群名称。创建命令如下：

hp1# pvecm create YOUR-CLUSTER-NAME

☒ 警告

集群名称用于计算生成集群多播通信地址。如果在网络里已经有 Proxmox VE 集群在运行，请务必确保集群名称不重复。

创建集群后，可以用如下命令查看集群状态：

hp1# pvecm status

5.4 新增集群节点

通过 ssh 远程登录要加入 Proxmox VE 集群的新节点。执行如下命令。

hp2# pvecm add IP-ADDRESS-CLUSTER

这里 IP-ADDRESS-CLUSTER 可以是已有集群中任意节点的 IP 地址。

☒ 警告

为避免虚拟机 ID 冲突，Proxmox VE 规定新节点加入集群前不能配置有任何虚拟机。此外，新加入节点/etc/pve 目录下的原有配置信息将被集群配置全部覆盖。如果节点上已有虚拟机，可以首先使用 vzdump 将所有虚拟机备份，然后删除节点上的虚拟机，待加入集群后再用新的虚拟机 ID 恢复原有虚拟机。

加入集群后可以查看集群状态：

pvecm status

加入 4 个节点后的集群状态

hp2# pvecm status

Quorum information

~~~~~~~~~~~~~~~~~~~

Date: Mon Apr 20 12:30:13 2015

Quorum provider: corosync_votequorum

Nodes: 4

Node ID: 0x00000001

Ring ID: 1928

Quorate: Yes

Votequorum information
~~~~~~~~~~~~~~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~

Expected votes: 4

Highest expected: 4

Total votes: 4

Quorum: 2

Flags: Quorate

Membership information

~~~~~~~~~~~~~~~~~~~~~~~~

Nodeid	Votes	Name
0x00000001	1	192.168.15.91
0x00000002	1	192.168.15.92 (local)
0x00000003	1	192.168.15.93
0x00000004	1	192.168.15.94

如果只需要查看节点列表，可运行如下命令：

pvecm nodes

列出集群节点列表

hp2# pvecm nodes

Membership information

~~~~~~~~~~~~~~~~~~~~~~~~

Nodeid	Votes	Name
1	1	hp1
2	1	hp2 (local)
3	1	hp3
4	1	hp4
~~~~~~~~~~~~~~~~~~~~~~~~

5.4.1 添加位于不同网段的节点

如果要添加一个节点，而该集群网络和该节点在不同网段，你需要使用 ringX_addr 参数来指定节点在集群网络内使用的地址。

pvecm add IP-ADDRESS-CLUSTER -ring0_addr IP-ADDRESS-RING0

如果你要使用冗余环协议，你还需要设置 ring1_addr 参数以传递第二个集群网络地址。

5.5 删除节点

☒ 警告

删除节点前请仔细阅读删除操作步骤，不然很可能会发生你预料不到的情况。

首先将待删除节点上所有虚拟机都迁移到其他节点。确保待删除节点上没有任何你需要保留的数据和备份，或者相关数据已经被妥善备份。

通过 ssh 登录待删除节点。执行 pvecm nodes 命令再次确认节点 ID。

hp1# pvecm status

Quorum information

~~~~~~~~~~~~~~~~~~~

Date:	Mon Apr 20 12:30:13 2015
Quorum provider:	corosync_votequorum
Nodes:	4
Node ID:	0x00000001
Ring ID:	1928
Quorate:	Yes

Votequorum information

~~~~~~~~~~~~~~~~~~~~~~

Expected votes: 4

Highest expected: 4

Total votes: 4

Quorum: 2

Flags: Quorate

Membership information

~~~~~~~~~~~~~~~~~~~~~~~~~

Nodeid	Votes	Name
0x00000001	1	192.168.15.91 (local)
0x00000002	1	192.168.15.92
0x00000003	1	192.168.15.93
0x00000004	1	192.168.15.94

☒ 重要

这个时候，你必须将待删除节点关闭并断电，确保该节点不再启动（在当前集群网络内）。

---

hp1# pvecm nodes

Membership information

~~~~~~~~~~~~~~~~~~~~~~~~~

Nodeid	Votes	Name
1	1	hp1 (local)
2	1	hp2
3	1	hp3
4	1	hp4

通过 ssh 登录集群中其他任何一个节点，执行节点删除命令（这里将删除节点 hp4）：

hp1# pvecm delnode hp4

如果命令执行成功，将直接返回，而且不会有任何输出。可以运行 pvecm nodes 或者 pvecm status 检查删除节点后的集群状态。你会看到类似如下输出：

hp1# pvecm status

Quorum information

~~~~~~~~~~~~~~~~~~~

Date:	Mon Apr 20 12:44:28 2015
Quorum provider:	corosync_votequorum
Nodes:	3
Node ID:	0x00000001
Ring ID:	1992
Quorate:	Yes

Votequorum information

~~~~~~~~~~~~~~~~~~~~~~

Expected votes:	3
Highest expected:	3
Total votes:	3
Quorum:	3
Flags:	Quorate

Membership information

~~~~~~~~~~~~~~~~~~~~~~

Nodeid	Votes	Name
0x00000001	1	192.168.15.90 (local)
0x00000002	1	192.168.15.91
0x00000003	1	192.168.15.92
~~~~~~~~~~~~~~~~~~~~~~

☒ 重要

如前所述，必须在执行删除命令前先关闭待删除节点，并且确保被删除点不再启动（在原集群网络中）。这是非常非常重要的！

如果你在原集群网络中重新启动被删除的节点，你的集群会因此而崩溃，并且很难恢复到一个干净的状态。

如果出于某种原因，你需要将被删除节点重新加入原集群，需要按如下步骤操作：

- 格式化被删除节点，并重新安装 Proxmox VE。
- 如前一节所述步骤，将该节点重新加入集群。

5.5.1 隔离节点

☒ 重要

我们不推荐使用隔离节点操作，按此方法操作时请务必小心。如果你对操作结果存有疑虑，建议使用删除节点的方法。

你可以将一个节点从集群中隔离出去，而无需格式化并重装该节点。但将节点从集群中隔离出去后，被隔离的节点仍然能够访问原 Proxmox VE 集群配置给它的共享存储。你必须在将节点隔离出去之前解决这个问题。由于不能确保避免发生虚拟机 ID 冲突，所以 Proxmox VE 集群之间不能共享同一个存储设备。

建议为待隔离节点专门创建一个独享的新存储服务。例如，可以为待隔离节点分配一个新的 NFS 服务或者 Ceph 存储池。必须确保该存储服务是独享的。在分配存储之后，可以将该节点的虚拟机迁移到新存储服务，接下来就可以开始进行隔离节点的操作。

☒ 警告

必须确保所有的资源都被已经彻底被隔离。否则将可能发生冲突或其他问题。

首先在待隔离节点上停止 pve-cluster 服务：

systemctl stop pve-cluster

systemctl stop corosync

然后将待隔离节点的集群文件系统设置为本地模式：

pmxcfs –l

接下来删除 corosync 配置文件：

rm /etc/pve/corosync.conf

rm /etc/corosync/*

最后重新启动集群文件系统服务：

killall pmxcfs

systemctl start pve-cluster

到此，该节点已经从集群中被隔离出去了。你可以在原集群中任意节点上执行删除命令：

pvecm delnode oldnode

如果因前面的隔离操作，原集群中剩余节点已经不满足多数票，节点删除命令就会失败。你可以将期望的多数票数量设置为 1，如下：

pvecm expected 1

然后重复节点删除命令即可。

接下来你可以重新登录被隔离出去的节点，删除原集群遗留下的各个配置文件。删除完成后，该节点即可重新加入任意其他集群。

rm /var/lib/corosync/*

被隔离节点的集群文件系统中仍然残留有和原集群其他节点有关的配置文件，这些也是需要删除的。你可以递归删除/etc/pve/nodes/NODENAME 目录清除这些文件。但在执行删除操作前请再三检查，确保删除操作无误。

☒ 警告

原集群中其他节点的 SSH 公钥仍会保留在 authorized_key 文件中。这意味着被隔离节点和原集群节点之间仍然可以用 SSH 公钥互相访问。为避免出现意外情况，可以删除/etc/pve/priv/authorized_keys 文件中的对应公钥。

5.6 多数票

Proxmox VE 采用了基于多数票（quorum）的机制确保集群节点状态一致。

多数票是指在一个分布式系统内一个分布式交易获准执行所必须得到的最低票数。

——Wikipedia 多数票（分布式计算）

在网络可能分裂为多个区域的情况下，修改集群状态需要得到大多数节点在线。如果集群内节点数量不足以构成多数票，集群将自动转为只读状态。

➢ 注意

默认情况下，Proxmox VE 集群内每个节点都有一票的投票权。

5.7 集群网络

集群网络是 Proxmox VE 集群的核心。集群网络必须确保可靠地将集群通信数据包按顺序送达所有节点。Proxmox VE 使用 corosync 来实现集群网络通信，确保集群网络通信的高性能，低延时，高可用。我们的分布式集群文件系统（pmxcfs）就基于此构建。

5.7.1 集群网络配置要求

Proxmox VE 集群网络只有在网络延时低于 2ms 时（局域网内）才可以正常工作。尽管 corosync 支持节点间使用单播方式通信，但我们强烈建议使用多播方式进行集群通信。集群网络内不应有其他大流量通信。理想情况下，corosync 最好能拥有专用网络。注意，一定不要在同一个网络同时运行 Proxmox VE 集群和存储服务。

最佳实践是在创建集群前先检测网络质量，确保网络能满足集群通信要求。

- 确认所有的节点都在同一网段。并且要确保网络中只连接了用于集群通信（corosync）的网卡。
- 确保节点彼此之间的网络都连接正常。可以使用 ping 命令测试。
- 确保多播网络通信工作正常并能达到很高的数据包传输速度。可以使用 omping 命令测试。正常情况下，丢包率应小于 1%。

  omping -c 10000 -i 0.001 -F -q NODE1-IP NODE2-IP ...
- 确保多播通信能在要求的时间段内可靠工作。这主要是为了避免物理交换机启用 IGMP 但未配置多播查询器（multicast querier）。该项测试至少需要持续 10 分钟。

  omping -c 600 -i 1 -q NODE1-IP NODE2-IP ...

如以上测试有任何一项未能通过，则你的网络不适合用于组建 Proxmox VE 集群。此时你需要检查网络配置。一般情况下，或者是交换机未启用多播通信，或者是交换机配置了 IGMP 但未启用 multicast querier。

如果你的集群节点数量很少，在实在无法使用多播通信的情况下也可以考虑使用单播方式。

5.7.2 独立集群网络

默认情况下，不带任何参数创建集群时，Proxmox VE 集群会和 Web GUI 以及虚拟机共享使用同一个网络。如果你配置不当，存储网络通信流量也有可能会通过集群网络传输。我们建议避免和其他应用共享使用集群网络，因为 corosync 是一个对时延非常敏感的实时应用。

准备一个新的网络

首先，你需要准备一个新的网络端口，该端口应连接在一个独立物理网络上。其次，需要确保这个网络满足集群网络配置要求。

创建集群时配置独立网络

可以用带 ring0_addr 和 bindnet0_addr 参数的 pvecm 命令创建拥有独立网络的 Proxmox VE 集群。

如果你想配置独立网卡用于集群通讯，而该网卡又配置了静态 IP 地址 10.10.10.1/25，那么可以使用如下命令：

pvecm create test --ring0_addr 10.10.10.1 --bindnet0_addr 10.10.10.0

然后可以使用如下命令检查集群通信是否正常：

systemctl status corosync

创建集群后配置独立网络

即使在你创建集群后，你也可以配置集群改用其他独立网络进行通信，而无须重建整个集群。修改集群通信网络，各节点的 corosync 服务需要逐个重启，以便使用新网络通信，这可能会导致集群短时间处于丧失多数票的状态。

首先确认你了解编辑 corosync.conf 文件的方法。然后打开 corosync.conf 文件。配置文件 corosync.conf 的内容示例如下：

```
logging {
        debug: off
        to_syslog: yes
}

nodelist {
        node {
                name: due
                nodeid: 2
```

```
                quorum_votes: 1
                ring0_addr: due
        }

        node {
                name: tre
                nodeid: 3
                quorum_votes: 1
                ring0_addr: tre
        }

        node {
                name: uno
                nodeid: 1
                quorum_votes: 1
                ring0_addr: uno
        }
}

quorum {
        provider: corosync_votequorum
}

totem {
        cluster_name: thomas-testcluster
        config_version: 3
        ip_version: ipv4
        secauth: on
        version: 2
        interface {
```

```
        bindnetaddr: 192.168.30.50
        ringnumber: 0
    }
}
```

首先，如果 node 对象中缺少 name 属性，你需要手工增添该属性。注意 name 属性值必须和节点主机名一致。

然后，你需要将 ring0_addr 属性的值修改为节点在新集群网络内的地址。你可以使用 IP 地址或主机名设置 ring0_addr 属性。如果你使用主机名，必须确保所有的节点都能顺利解析该主机名。

在这里，我们计划将集群通信网络改为 10.10.10.1/25，所以需要相应修改所有的 ring0_addr 属性。此外，还需要将 totem 一节中的 bindnetaddr 属性值修改为新网络中的地址。该地址可以配置为当前节点连接到新集群网络网卡的 IP 地址。

最后，你需要将 config_version 参数值增加 1。修改后的配置文件内容示例如下：

```
logging {
    debug: off
    to_syslog: yes
}

nodelist {

    node {
        name: due
        nodeid: 2
        quorum_votes: 1
        ring0_addr: 10.10.10.2
    }

    node {
        name: tre
```

```
        nodeid: 3
        quorum_votes: 1
        ring0_addr: 10.10.10.3
    }

    node {
        name: uno
        nodeid: 1
        quorum_votes: 1
        ring0_addr: 10.10.10.1
    }
}

quorum {
    provider: corosync_votequorum
}

totem {
    cluster_name: thomas-testcluster
    config_version: 4
    ip_version: ipv4
    secauth: on
    version: 2
    interface {
        bindnetaddr: 10.10.10.1
        ringnumber: 0
    }
}
```

最后你需要再次检查配置修改是否正确，然后可以根据编辑 corosync.conf 文件一节的内容，启用新的配置。

由于修改后的配置不能实时在线生效，所以必须重启 corosync 服务。

在一个节点上执行：

systemctl restart corosync

然后检查集群通信是否正常

systemctl status corosync

如果在所有节点上 corosync 服务都能顺利重启并正常运行，那么所有的节点都将逐个改接入新的集群网络。

5.7.3 冗余环协议

为避免网络设备单点故障导致集群通信中断的风险，你可以使用冗余网络技术。比如，你可以利用硬件和操作系统的支持，使用多网卡绑定技术。

Corosync 本身支持配置冗余的通信网络，该技术称为冗余环协议（RRP）。通过该协议可以在另一个网络上同时运行第二个 totem 环。而为了切实增强网络可用性，该网络应该和原集群网络物理隔离，避免共用网络设备。

5.7.4 创建集群时配置 RRP

集群管理工具 pvecm 提供了参数项 bindnetX_addr，ringX_addr 和 rrp_mode，可用于在创建集群时配置使用 RRP。

➢ 注意

可以查看 Corosync 参数说明，以了解每个参数的含义。

如果你已经有两个网络，地址分别为 10.10.10.1/24 和 10.10.20.1/24，可以运行如下命令创建 RRP 集群：

pvecm create CLUSTERNAME -bindnet0_addr 10.10.10.1 -ring0_addr 10.10.10.1 \

-bindnet1_addr 10.10.20.1 -ring1_addr 10.10.20.1

5.7.5 创建集群后配置 RRP

创建集群后再配置 RRP 和创建集群后配置独立网络的操作非常类似，区别只在于配置 RRP 是增加一个新的 totem 环。

首先需要在 corosync.conf 配置文件中的 totem 节新增一个 interface 对象，并设置 ringnumber 属性为 1，bindnetaddr 值为当前节点在新增网络中的地址。另外还需要设置 rrp_mode 属性值为 passive，这是唯一的稳定模式。

然后在 nodelist 一节中，为每个 node 对象添加 ring1_addr 属性，并将其值设置为对应节点在新增网络中的地址。

假定你有两个网络，其中原集群网络地址为 10.10.10.1/24，新增网络地址为 10.10.20.1/24，最终修改后的配置文件内容示例如下：

```
totem {
        cluster_name: tweak
        config_version: 9
        ip_version: ipv4
        rrp_mode: passive
        secauth: on
        version: 2
        interface {
                bindnetaddr: 10.10.10.1
                ringnumber: 0
        }
        interface {
                bindnetaddr: 10.10.20.1
                ringnumber: 1
        }
}

nodelist {
        node {
```

```
        name: pvecm1
        nodeid: 1
        quorum_votes: 1
        ring0_addr: 10.10.10.1
        ring1_addr: 10.10.20.1
    }

    node {
        name: pvecm2
        nodeid: 2
        quorum_votes: 1
        ring0_addr: 10.10.10.2
        ring1_addr: 10.10.20.2
    }
    [...] # other cluster nodes here
}
[...] # other remaining config sections here
```

最后，按照编辑 corosync.conf 文件一节的内容启用新配置。

新配置不能实时在线生效，必须重启 corosync 服务才可以生效。建议最好重启所有节点的操作系统。

如果你不能重启所有节点的操作系统，可以在确认没有启用 HA 的情况下，先停止所有节点的 corosync 服务，然后再逐个启动每个节点的 corosync 服务。

5.8 配置 Corosync

/etc/pve/corosync.conf 文件是 Proxmox VE 集群的核心配置文件。该文件控制着集群成员构成和网络通信。可以查看 corosync.conf 的 man 手册，以获取更多信息。

```
man corosync.conf
```

在查看节点成员时，你可以用 pvecm 命令。但在手工改变集群配置时，你可以直接编辑该 corosync.conf 配置文件。以下是编辑该文件时的一些最佳实践和提示。

5.8.1 编辑 corosync.conf

编辑配置文件 corosync.conf 并非看起来那么简单。Proxmox VE 服务器上一共有两个 corosync.conf 配置文件，一个是集群文件系统中的/etc/pve/corosync.conf，另一个是本地文件系统中的/etc/corosync/corosync.conf。对集群文件系统中的 corosync.conf 进行编辑，配置变更会自动同步到本地文件系统中的 corosync.conf，但反过来编辑本地文件系统中的 corosync.conf，配置变更不会同步到集群文件系统中的副本。

配置文件 corosync.conf 的内容变化会自动更新到服务配置中。这意味着，配置文件内容的变化会实时生效，并影响 corosync 服务的运行。为了避免编辑过程对 corosync 服务产生意想不到的影响，建议你先复制一个 corosync.conf 的副本，修改该副本而不是直接修改 corosync.conf。

cp /etc/pve/corosync.conf /etc/pve/corosync.conf.new

可以使用你喜欢的编辑器对 corosync.conf 副本进行修改。例如可以使用 Proxmox VE 预装的 nano 或者 vim.tiny。

➢ 注意

编辑完成后，请记住增加 config_version 的值，否则可能会导致错误。

当编辑完成后，在启用新的配置之前，最好先备份当前配置文件，以便新配置文件启用失败或出现错误时能够回退到原配置。可执行如下命令进行备份：

cp /etc/pve/corosync.conf /etc/pve/corosync.conf.bak

现在可以用新的配置文件覆盖原配置文件，如下：

mv /etc/pve/corosync.conf.new /etc/pve/corosync.conf

启用新配置后，可以用以下命令查看服务状态及配置是否生效

systemctl status corosync

journalctl -b -u corosync

如果新配置未能自动生效，可以尝试重启 corosync 服务，如下：

systemctl restart corosync

如果发现错误，可以尝试按下一节介绍的方法进行排查。

5.8.2 故障排查

错误现象：quorum.expected_votes must be configured

如果 corosync 服务停止运行，而系统日志文件中有如下日志信息时：

[...]

corosync[1647]: [QUORUM] Quorum provider: corosync_votequorum failed to initialize.

corosync[1647]: [SERV] Service engine 'corosync_quorum' failed to load for reason ' configuration error: nodelist or quorum.expected_votes must be configured!'

[...]

原因是你在配置文件中设置的 ringX_addr 主机名不能正常解析。

失去多数票的状态下修改配置文件

如果你需要在一个失去多数票的节点上修改/etc/pve/corosync.conf 文件，你可以执行如下命令：

pvecm expected 1

将期望的多数票数目设置为 1，可以让当前节点重新满足多数票的要求，这样你就可以修改配置，或者是将配置恢复到的正常状态。

在 corosync 服务无法启动时，你还需要采取进一步措施以恢复集群。最好是修改本地文件系统中的配置文件/etc/corosync/corosync.conf，先确保 corosync 服务能正常启动。你需要特别注意，确保所有节点的 corosync.conf 内容都完全一致，避免集群出现脑裂的情况。如果你最终还是无法确认故障原因，可以向 Proxmox 社区寻求帮助。

5.8.3 Corosync 参数说明

ringX_addr

用于设置各个节点在不同的 totem 环内的地址，以用于进行 corosync 集群通信。

bindnetaddr

定义当前节点当前 totem 环所要绑定的网卡端口。可以设置为网卡端口上配置的集群网络内的任意地址。但一般情况下，我们建议在网卡端口上仅绑定一个地址。

rrp_mode

用于设置冗余环协议的运行模式。目前有三种参数值可选，分别是 passive，active 和 none。其中 active 仍处于试验阶段，官方不推荐使用。选用 passive 较为合适，不仅可以将集群网络通信带宽提高一倍，而且可以实现集群网络的高可用性。

5.9 集群冷启动

很显然，当所有节点都断线时，集群是无法达到多数票要求的。例如，机房意外断电后，集群往往就处于这样的状态。

➢ 注意

使用不间断电源（UPS，也称为“后备电池电源”）是防止断电导致集群失去多数票的一个好办法，特别是在你需要实现 HA 效果的时候。

当节点重启启动时，pve-manager 服务会等待该节点加入集群并获得多数票状态。一旦获得多数票，该服务会启动所有设置了 onboot 标识的虚拟机。

因此，当你启动节点时，或者是意外断电后恢复供电时，你会发现一些节点启动速度会比其他节点快。另外要注意的是，在你的集群获得多数票之间，任何虚拟机都无法启动。

5.10 虚拟机迁移

能够把虚拟机从一个节点迁移到其他节点是集群的重要特性。Proxmox VE 提供了一些方法以便你控制虚拟机迁移过程。首先是 datacenter.cfg 提供了一些配置参数，其次是迁移命令行和 API 接口提供了相关控制参数。

5.10.1 迁移类型

迁移类型是指迁移过程采用加密（secure）或不加密（insecure）通道传输虚拟机数据。将迁移类型设置为 insecure 后，在迁移过程中虚拟机内存数据将以明文方式传输，这有可能导致虚拟机上的敏感数据泄露（例如口令、密钥）。

因此，我们强烈建议使用安全通道迁移虚拟机，特别在你无法控制整个网络链路并无法保证网络不受窃听时。

- 注意

虚拟机磁盘迁移不受该配置影响。目前，虚拟机磁盘总是通过安全通道迁移。

由于数据加密会耗费大量计算资源，所以该虚拟机迁移时经常会选用“不安全”的传输方式，以节约计算资源。较新型的系统采用了硬件方式进行 AES 加密，受此影响较小。但在 10Gb 或更高速的网络中，该参数设置对于性能的影响会十分明显。

5.10.2 专用迁移网络

默认情况下，Proxmox VE 通过集群网络传输虚拟机迁移数据。但这样虚拟机迁移流量很可能会干扰到敏感的集群网络通信，并且集群网络也未必就是服务器节点所能利用的最大带宽网络。

通过设置迁移网络参数，可以选择一个专用网络进行虚拟机迁移。除传输内存数据，还可以在离线迁移时传输虚拟机磁盘镜像数据。

迁移网络参数值为 CIDR 格式的网络地址。这样的好处是，你无须分别设置迁移的源节点和目标节点地址。Proxmox VE 能够根据 CIDR 格式的网络地址自动决定目标节点应使用的 IP 地址。为确保该机制的正常工作，每个节点在迁移网络中都必须被分配一个且仅分配一个 IP 地址。

迁移网络配置示例

假定有一个 3 节点集群，节点之间通过 3 个不同的网络连接。其中一个是公共的互联网，一个是集群通信网络，一个是高带宽的快速网络。这里我们将选择该快速网络作为虚拟机迁移网络。

该集群的网络配置示例如下：

```
iface eth0 inet manual

# public network
auto vmbr0
iface vmbr0 inet static
        address 192.X.Y.57
        netmask 255.255.250.0
        gateway 192.X.Y.1
        bridge_ports eth0
        bridge_stp off
        bridge_fd 0

# cluster network
auto eth1
iface eth1 inet static
        address 10.1.1.1
        netmask 255.255.255.0

# fast network
auto eth2
iface eth2 inet static
        address 10.1.2.1
        netmask 255.255.255.0
```

这里，我们将选择快速网络 10.1.2.0/24 作为虚拟机迁移网络。对于命令行方式的迁移操作，可以使用 migration_network 设置迁移网络：

```
# qm migrate 106 tre --online --migration_network 10.1.2.0/24
```

如需将该快速网络设置为默认的虚拟机迁移网络，可利用/etc/pve/datacenter.cfg 中的 migration 属性进行设置：

```
# use dedicated migration network
migration: secure,network=10.1.2.0/24
```

- 注意

在/etc/pve/datacenter.cfg 中设置虚拟机迁移网络时，必须同时设置迁移类型。
