第5章 chapter 5

调整操作系统

Linux发行版和Linux内核提供了各种参数和设置，以让Linux管理员调整系统从而得到最大限度的性能。正如本书之前所描述的，没有神奇的调整旋钮可以任意调整应用程序的性能。下面章节中讨论的设置将改善某些硬件配置和应用程序的性能。但是，它们有可能对提高Web服务器和数据库服务器的性能产生不利影响。

本章将介绍调整Linux发行版2.6内核的步骤。目前，Linux发行版2.6内核，从2.6.9到2.6.19有所不同，一些调整选项可能只适用于一个特定的内核版本。描述这些参数的目的是在大多数情况下可以改善性能，并使你对Linux中的技术有基本的了解，这些参数包括：Linux内存管理、系统清理、磁盘子系统调优、使用sysctl对内核调优和网络优化。

本章内容包括以下几个部分：调整原则、安装注意事项、更改内核参数、调整处理器子系统、调整内存子系统、调整磁盘子系统、调整网络子系统和限制资源使用。

5.1 调整原则

调整任何系统应该遵循一些简单的原则，其中最重要的是变更管理。通常，系统调优的第一步应该是分析和评估当前系统配置，确保系统按照硬件制造商所述运行，所有设备在最佳模式下运行，这将为以后的任何调整创建一个坚实的基础。另外，在执行任何特定调整任务之前，系统被设计为最佳性能，最小化运行不必要的任务和子系统。最后，当对特定系统进行调整的时候，应该注意的是，调整通常是让系统适应一个特定的工作负载。因此，系统在预期的负载特征下会有更好的表现，但是在不同的工作模式下它可能表现得更糟。例如，低延迟系统调整，很大可能对磁盘造成不良影响。

虽然变更管理与性能调优不是严格相关，但是变更管理可能是成功进行性能调优的一个重要手段。变更管理注意问题，如表5-1所示。

表5-1 变更管理注意问题

序号	注意问题
1	在调整任何Linux系统之前，实施适当的变更管理过程
2	绝不在生产服务器上开始调整设置

续表

序号	注意问题
3	绝不在调整过程期间改变一个以上的变量
4	重新启动可能提高性能的参数;有时统计会发挥作用
5	记录成功的参数并分析它们,无论你认为它们是多么微不足道。Linux性能可以大大受益于在生产环境中获得的任何结果

5.2 安装注意事项

理想情况下,在服务器系统上对一个特定性能目标的调优,应该从设计和安装阶段开始。适当的安装就是调整系统适应工作量模式,这能为之后的调优节省大量的时间。

5.2.1 安装

在一个完美的世界中,任何指定系统的调优都在一个非常早期的阶段就开始了。理想情况下,一个系统是专门针对应用程序的需要和预期工作量所定制的。我们知道,一般一个管理员由于一个瓶颈需要调整一个已经安装好的系统,但是我们要强调的是,调优在操作系统的初始化安装期间也是有用的。

在开始Linux安装之前,有几个问题需要解决,包括:处理器技术的选择、磁盘技术的选择和应用程序的需求。

这些问题超出了本书的范围,这里不详述。

理想情况下,在开始安装之前先回答以下4个问题。

(1) 我需要什么特点和版本的Linux? 在收集商业需求和应用程序需求之后,决定使用哪个Linux版本。企业通常有合同协议,可以普遍使用一个特定Linux发行版。经济能力和合同利益决定使用的Linux版本。然而,如果在选择Linux发行版本上有完全的自由,则有以下问题需要考虑:

支持企业级Linux发行版或是一个定制发行版。在一些科学的环境中可以接受运行一个不受支持的Linux版本,比如Fedora。考虑到企业工作负载,强烈建议使用完全支持的发行版,比如Red Hat Enterprise Linux或Novell SUSE Enterprise Linux。

(2) 什么版本的企业发行版? 几乎每个企业Linux发行版都有自己的特点,如内核版本、支持的软件包、特征等,最重要的是它们在硬件支持的水平上有所不同。在安装之前,仔细查看支持的硬件配置,这样将不会遗漏任何硬件的功能。

选择正确的内核。企业级Linux发行版提供了几种内核软件包。出于性能考虑,一定要为你的系统选择最合适的内核。然而,大多数情况下,正确的内核是在安装程序中选择的。记住,正确的内核软件包名称由于发行版不同是有差异的。

注意:最新的内核具有被称为SMAP切换的能力,它可在启动时优化自己,详情参

考发行版说明。

(3) 选择什么样的分区布局？磁盘子系统的分区布局通常是由应用程序的需要、系统管理方面考虑、个人喜好所决定的，而不是由性能所决定的。在大多数情况下已经给出分区布局。我们唯一的建议是，如果可能，应使用一个 swap 分区。swap 分区，而不是 swap 文件，对性能是有好处的，因为没有文件系统的开销。swap 分区很简单，还可以扩展额外的 swap 分区，或者如果需要，甚至可以扩展 swap 文件。

(4) 使用的文件系统是什么？不同的文件系统在数据完整性和性能上有不同的特点。一些文件系统可能在有些 Linux 发行版或应用程序上是不被支持的。对于大多数服务器，安装程序建议的默认文件系统将提供足够的性能。如果你有最小延迟或最大吞吐量的具体需求，则建议根据这些需求选择各自的文件系统。

软件包的选择："最小化"或是"所有"？在 Linux 的安装期间，管理员面临着要选择"最小化"还是"所有"的安装方法。有些人喜欢所有安装，因为这样很少需要解决安装软件包的依赖性。

考虑以下几点：虽然没有涉及性能，选择"所有"或"几乎所有"的安装方式对于系统安全威胁有很重要的影响。在生产系统上，开发工具的可用性可能导致明显的安全威胁。安装较少的数据包，浪费的磁盘空间较少，一个具有较多空闲空间的磁盘比一个具有很小空闲空间的磁盘性能要好。如果需要，智能化软件安装程序，比如 Red Hat Packet Manager、rpm 或 yum 将自动解决依赖性。因此，建议考虑最小化软件包选择，只安装那些可以使应用程序成功实现的软件包。

你要决定 Netfilter 防火墙配置是否需要。Netfilter 防火墙通常用来保护系统免受恶意攻击。然而，太过复杂的防火墙规则在高数据流量环境中会降低性能。

在某些 Linux 发行版中，比如 Red Hat Enterprise Linux 5.x，安装程序让你选择是否开启 SELinux。SELinux 会带来明显的性能损失，应仔细地评估是否真需要为一个特定的系统通过 SELinux 提供额外的安全。

运行级别的选择。安装过程最后给出的选择是：系统默认运行级别的选择。除非有特殊的需要，在运行级别 5(图形用户模式)运行系统，否则强烈建议在所有服务器系统上使用运行级别 3。正常情况下，在一个数据中心中驻留的系统是不需要图形界面的，运行级别 5 的开销是相当大的。如果安装程序不提供运行级别的选择，建议在初始化系统配置之后手动选择运行级别 3。

5.2.2 检查当前的配置

如前所述，尝试为系统建立一个坚实的基础很重要。坚实的基础可确保所有子系统按照设计的方式工作，没有异常。一个异常的例子是一个千兆网络接口卡和一个有网络性能瓶颈的服务器。如果网卡自动协商为 100Mb/s 半双工，则调整 Linux 内核的 TCP/IP 实现可能没有多大用处。了解系统理论能力是非常重要的，如表 5-2 所示。

表 5-2 检查当前的配置

配置	问 题
CPU	CPU(S)模式是什么？处理器的架构是哪一个？使用什么主板芯片？
	有多少个 socket？每个 CPU 有多少个核心？如果核心支持超线程，那么每个核心上有多少个线程？每个 socket 是如何连接的，数据速率是多少(QPI，HyperTransport，FSB)？
	CPU 每个层级的 cache 有多大？cache 是核心私有的，还是与 socket 上的其他核心共享的，还是与一个系统中的其他 socket 共享的？cache 是如何被组织和访问的？
	处理器支持哪些特征？处理器支持硬件虚拟化吗？处理器的架构是 32 位还是 64 位？有其他特殊功能或指令吗？
内存	系统有多少内存？硬件允许的最大数量的内存是多少？
	系统使用的内存技术是什么？它具有什么带宽和延迟？
	系统的内存是如何被连接的，通过一个 Northbridge 和前端总线，或者直接到 CPU；是 SMP 或 NUMA 吗？如果是 NUMA，NUMA 节点使用什么技术，当访问不同区域的内存时会有什么影响？
存储	附加在系统上的存储设备是什么类型的？实际设备是什么模型？它们是基于旋转磁盘片的传统硬盘，还是固态硬盘(solid-state disk，SSD)驱动器？
	传统的硬盘，它们有没有寻址时间和旋转延迟？你期待的最大化带宽和延迟是多少？访问驱动器不同分区的速度变化是多少？
	SSD 驱动器，它们使用的闪存是 MLC 还是 SLC？它们支持 TRIM 吗？如何有效地对它们的耐久性和垃圾收集系统进行测量？
	磁盘阵列，已使用的硬件 RAID 级别是什么？在一个阵列里有多少个存储设备？RAID 级别是条带，在阵列中每个条带的大小是多少？
	直连式存储使用什么(SATA、SAS 或其他)进行互连？它是怎么连接主板到系统的其他部分呢？是否有充足的带宽让所有的设备支持通信的连接？
	SAN 设备，使用 Fibre Channel 或 iSCSI 越过以太网？SAN 设备的带宽和延迟是多少？多路径 SAN 设备，基于可用路径的性能变化如何？
网络	系统上可用的网络接口卡是什么？使用的网络技术是什么(如以太网、无限带宽技术等)？
	对机器来说，什么网络是可用的或需要的？什么是 VLAN？如何大量利用这些网络？它们有特殊用途吗(如存储网络、带外管理等)？它们运转的速度是多少？
	我们需要特别的能力吗(例如，I/O 虚拟化可以将一个网卡划分成多种 vNIC(虚拟网卡)作为系统上的 guest 虚拟机)？

物理访问一台机器可以较容易地回答这些问题。通常，管理员在一个远程数据中心管理主机。下面将寻找能够帮助系统管理员远程收集系统上使用的硬件的信息的方法和工具。

1. 查看内核消息

一个系统上有关硬件最简单、最详细、最直接的信息源常常是 Linux 内核。因为内核是访问所有硬件的主要介质。内核通过/proc 文件系统、/sys 文件系统和内核消息公开此信息。

可以从内核消息中找到可用的信息。为了能在几乎任何环境中生成消息，设计机制为：将内核消息写入一个预分配的环形缓冲区，像是已知的 dmesg 缓冲区。想读取消息，必须以某种方式将它们放入用户空间。

在 RHEL 中，这些消息被 klogd 进程监听，并将消息转发到 syslog（或 rsyslog）日志库，记录的内核消息默认放在/var/log/messages 下。另一种方法是使用 dmesg 命令导出当前 dmesg 缓冲区中的内容。

通常，审查 dmesg 缓冲区有 3 个动机，如表 5-3 所示。

表 5-3 审查 dmesg 缓冲区的动机

序 号	内 容
动机 1	回顾一下启动时的硬件检测
动机 2	观察显示出的硬件连接或检测的信息
动机 3	观察显示出的警告或错误情况发生的信息

在启动之后，dmesg 缓冲区将包含大量信息，dmesg 的部分输出如图 5-1 所示。

```
[root@zgh ~]# dmesg
[    0.000000] Linux version 4.18.0-147.el8.x86_64 (mockbuild@x86-vm-09.build.en
g.bos.redhat.com) (gcc version 8.3.1 20190507 (Red Hat 8.3.1-4) (GCC)) #1 SMP Th
u Sep 26 15:52:44 UTC 2019
[    0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-4.18.0-147.el8.x86_
64 root=/dev/mapper/rhel-root ro crashkernel=auto resume=/dev/mapper/rhel-swap r
d.lvm.lv=rhel/root rd.lvm.lv=rhel/swap rhgb quiet
[    0.000000] Disabled fast string operations
[    0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point regi
sters'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x008: 'MPX bounds registers'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x010: 'MPX CSR'
[    0.000000] x86/fpu: xstate_offset[2]:  576, xstate_sizes[2]:  256
[    0.000000] x86/fpu: xstate_offset[3]:  832, xstate_sizes[3]:   64
[    0.000000] x86/fpu: xstate_offset[4]:  896, xstate_sizes[4]:   64
[    0.000000] x86/fpu: Enabled xstate features 0xlf, context size is 960 bytes,
 using 'compacted' format.
[    0.000000] BIOS-provided physical RAM map:
[    0.000000] BIOS-e820: [mem 0x0000000000000000-0x0000000000097bff] usable
[    0.000000] BIOS-e820: [mem 0x0000000000097c00-0x000000000009ffff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000000ce000-0x00000000000cffff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000000dc000-0x00000000000fffff] reserved
[    0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000bfecffff] usable
```

图 5-1 dmesg 的部分输出

然而，当前的消息将填充缓冲区，并随着时间的推移最终循环并覆盖原来的信息。因此，每次启动后，Red Hat Enterprise Linux 存储当前 dmesg 缓冲区的内容到/var/log/dmesg。花费一些时间查阅内核启动信息，能够得到关于一个平台的硬件的综合情况。

2. 查看 CPU 信息

现代系统中通常有多个 CPU，每个 socket 会有多个核心，每个核心又可能有多个超线程，有不同级别的本地和共享 cache。lscpu 命令可以提供一个本地配置的快速概要。

下面是 lscpu 的输出和相关信息的说明，如图 5-2 所示。

```
[root@zgh ~]#lscpu
```

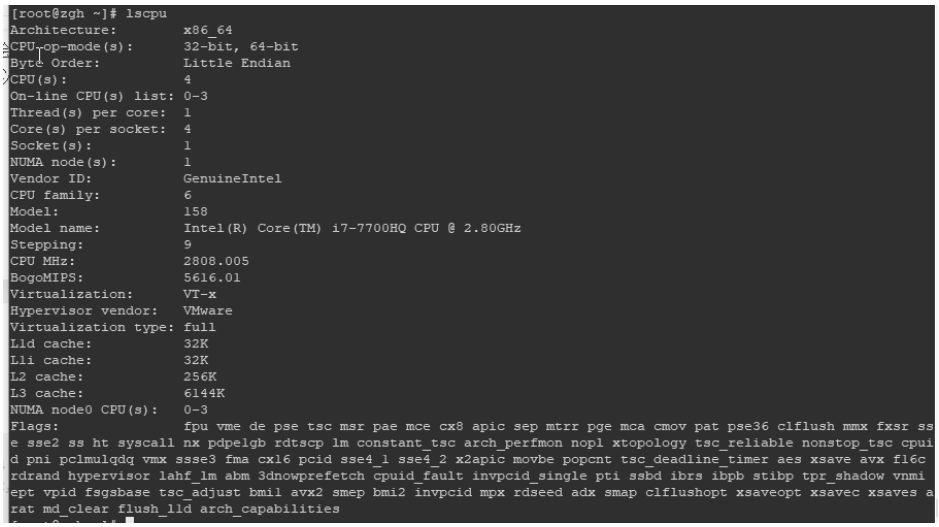

图 5-2　使用 lscpu 查看本地配置

lscpu 列出了多个 CPU cache 的大小，它不会告诉我们逻辑 CPU 是如何共享 cache 的。使用 lscpu -p，可输出 cache 共享信息，如图 5-3 所示。

```
[root@zgh ~]# lscpu -p
```

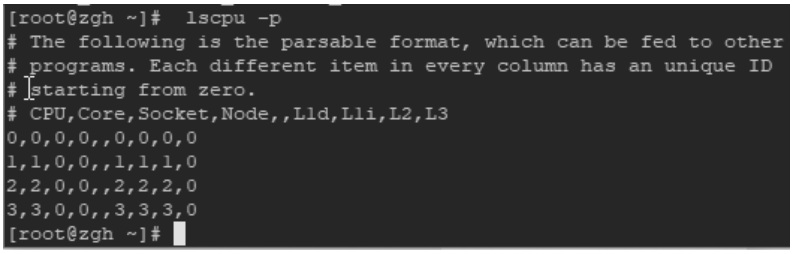

图 5-3　使用 lscpu -p 输出 cache 共享信息

注意最后 4 个字段，L1 和 L2 cache 每对超线程是不同的，L3 cache 是被各自 socket 上的所有核心共享的，附加一个单独的内存地址总线。

lscpu 命令是在 Red Hat Enterprise Linux 6 中引入的。在之前的 Red Hat Enterprise Linux 版本中，使用 x86info 命令提供类似的信息，其可在 x86info 软件包中找到。下面显示的是 x86info 输出的部分信息，如图 5-4 所示。

```
[root@zgh ~]# x86info
```

3. 确定 SMBIOS/DMI 信息

dmidecode 工具（由 dmidecode 软件包提供）可以探测本地 System Management BIOS(SMBIOS)和 Desktop Management Interface(DMI)，并提供大量有关本地硬件的信息。

首先看到 BIOS 的信息，如图 5-5 和图 5-6 所示。

```
[root@zgh ~]# x86info
x86info v1.30.  Dave Jones 2001-2011
Feedback to <davej@redhat.com>.

 (but found 128d CPUs in MPTable!)Extended Family: 0 Extended Model: 9 Family: 6 Model: 158 Stepping: 9
Type: 0 (Original OEM)
CPU Model (x86info's best guess): Unknown model.
Processor name string (BIOS programmed): Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz

Total processor threads: 1
This system has 1 processor running at an estimated 2.80GHz
[root@zgh ~]#
```

图 5-4 x86info 输出的部分信息

[root@zgh ~]#dmidecode

```
root@zgh:~
        Serial Number: Not Specified
        Asset Tag: Not Specified
        Part Number: Not Specified
        Rank: Unknown
        Configured Memory Speed: Unknown

Handle 0x01E7, DMI type 17, 34 bytes
Memory Device
        Array Handle: 0x0025
        Error Information Handle: No Error
        Total Width: 32 bits
        Data Width: 32 bits
        Size: No Module Installed
        Form Factor: DIMM
        Set: None
        Locator: NVD #4
        Bank Locator: NVD #4
        Type: Other
        Type Detail: Unknown
        Speed: Unknown
        Manufacturer: Not Specified
        Serial Number: Not Specified
        Asset Tag: Not Specified
        Part Number: Not Specified
        Rank: Unknown
        Configured Memory Speed: Unknown
```

图 5-5 dmidecode 工具探测 SMBIOS/DMI 信息(1)

```
Handle 0x0267, DMI type 30, 6 bytes
Out-of-band Remote Access
        Manufacturer Name: Intel
        Inbound Connection: Enabled
        Outbound Connection: Disabled

Handle 0x0268, DMI type 32, 20 bytes
System Boot Information
        Status: No errors detected

Handle 0x0269, DMI type 33, 31 bytes
64-bit Memory Error Information
        Type: OK
        Granularity: Unknown
        Operation: Unknown
        Vendor Syndrome: Unknown
        Memory Array Address: Unknown
        Device Address: Unknown
        Resolution: Unknown

Handle 0x026A, DMI type 126, 4 bytes
Inactive

Handle 0x026B, DMI type 127, 4 bytes
End Of Table

[root@zgh ~]#
```

图 5-6 dmidecode 工具探测 SMBIOS/DMI 信息(2)

相关服务器的系统信息如图 5-7 所示。

```
Handle 0x0100, DMI type 1, 27 bytes
System Information
        Manufacturer: Dell Inc.
        Product Name: PowerEdge R710
        Version: Not Specified
        Serial Number: C5V8D3X
        UUID: 6C6C6566-0035-5610-8038-C3C06F663358
        Wake-up Type: Power Switch
        SKU Number: Not Specified
        Family: Not Specified
```

图 5-7　相关服务器的系统信息

相关服务器上(部分)CPU 的信息,列出了主板上多种设备,如网络接口卡、显卡、RAID 控制器,如图 5-8 所示。

```
Handle 0x0A00, DMI type 10, 16 bytes
On Board Device 1 Information
        Type: Video
        Status: Enabled
        Description: Embedded Matrox G200 Video
On Board Device 2 Information
        Type: Ethernet
        Status: Enabled
        Description: Embedded Broadcom 5709C NIC 1
On Board Device 3 Information
        Type: Ethernet
        Status: Enabled
        Description: Embedded Broadcom 5709C NIC 2
```

图 5-8　主板设备相关信息

还有已安装的(部分)RAM 信息,如图 5-9 所示。

```
Handle 0x1000, DMI type 16, 15 bytes
Physical Memory Array
        Location: System Board Or Motherboard
        Use: System Memory
        Error Correction Type: Multi-bit ECC
        Maximum Capacity: 288 GB
        Error Information Handle: Not Provided
        Number Of Devices: 18

Handle 0x1101, DMI type 17, 28 bytes
Memory Device
        Array Handle: 0x1000
        Error Information Handle: Not Provided
        Total Width: 72 bits
        Data Width: 64 bits
        Size: 4096 MB
        Form Factor: DIMM
        Set: 1
        Locator: DIMM_A2
        Bank Locator: Not Specified
        Type: DDR3
```

图 5-9　RAM 信息

注意，这些信息是由内核中的 sysfs 文件系统提供的，在/sys/class/dmi/id 目录下，如图 5-10 所示。

```
[root@zgh ~]#ls /sys/class/dmi/id/
```

```
[root@zgh ~]#  ls /sys/class/dmi/id/
bios_date     board_asset_tag  board_vendor       chassis_serial  chassis_version  product_name    product_version  uevent
bios_vendor   board_name       board_version      chassis_type    modalias         product_serial  subsystem
bios_version  board_serial     chassis_asset_tag  chassis_vendor  power            product_uuid    sys_vendor
```

图 5-10 sysfs 文件系统

5.2.3 最小化资源使用

以最高性能水平设计的系统必须最小化任何资源的浪费。赛车不能提供如同普通轿车的舒适，但是驾驶的目的是尽可能地比普通轿车快，并且舒适的座椅是资源的浪费。同样的道理也适用于服务器系统。运行消耗内存的 GUI 和大量不必要的守护进程，将会降低系统的整体性能。本节将介绍系统资源消耗的优化。

1. 守护进程

在 Linux 发行版默认安装之后，有些不必要的服务和守护进程可能会被启用。要禁用不需要的守护进程，减少整个系统的内存占用量，减少运行进程的数量和上下文切换，更重要的是，减少暴露的各种安全威胁。禁用不需要的守护进程也可以降低服务器的启动时间。

默认情况下，在大多数系统上，有些已经启动的守护进程可以被安全地停止和禁用。表 5-4 列出了在各种 Linux 发行版中已经启动的守护进程。如果适用，应考虑在你的环境中禁用它们。注意，表 5-4 中列出了一些商用 Linux 发行版各自的守护进程。特定的 Linux 安装在运行守护进程的确切数量上可能有所不同。这些守护进程更详细的说明，参考下面的 system-config-services 的显示。

表 5-4 默认安装的可调整启动的守护进程

进　　程	注　　释
Network Manager	如果希望手工管理网络可以禁用，这是一个简单管理网络连接的工具
abrt-ccpp abrt-oops abrtd	abrt(automatic bug report tool)服务监控应用程序崩溃并收集崩溃时数据，稍后汇报给 Bugzilla 以方便开发者修复。可以禁用
acpi	acpi(advanced configuration and power interface)，为替代传统的 APM 电源管理标准而推出的新型电源管理标准。建议笔记本用户开启它。服务器一般不需要 acpi
atd	在特定的时间运行 at 命令来调度要运行的命令。如果不使用，则该进程可以禁用
autofs	自动挂载需求的文件系统(例如，自动挂载 CD-ROM)。在服务器系统上，文件系统很少有自动挂载的情况
bluetooth	当发现蓝牙设备时，触发 bluetoothd 启动。如果不连接蓝牙设备，则该进程可以禁用

续表

进　　程	注　　释
cgcongif	创建和设置 control group 文件系统。如果不使用 cgroup，则该进程可以禁用
cgred	cgroup 规则引擎用来自动对进程进行分类。如果不使用，则该进程可以禁用
cpuspeed	守护进程将自动调整 CPU 的频率，在服务器环境中，建议关闭这个守护进程
cpus	通用 UNIX 打印系统，如果不打算在服务器上使用打印服务，则该进程可以禁用
dnsmasq	如果你的服务器不是用作 DNS 缓存服务器，则该进程可以禁用
ebtables	与 iptables 类似，只不过是对数据链路层的 MAC 做过滤，可以禁用
firstboot	在第一次安装后引导用户对系统进行配置，可以禁用
hacacheclean	将 mod_disk_cache 的存储缓冲区所占用的空间限定在一个合理的水平，可以禁用
Ip6tables	IPv6 防火墙，服务器一般不启用，可以禁用
iptables	IPv4 防火墙，服务器一般不启用，可以禁用
kdump	在系统崩溃、死锁或者死机的时候用来收集内存运行的状态和数据信息，可以禁用
mdmonitor	检查系统上的所有软件磁盘阵列(RAID)的状态，可以禁用
netcf-transaction	保存当前网络配置的状态，并在稍后恢复配置或提交新的配置，可以禁用
netconsole	用于将本地主机的日志信息打印到远程主机上，如果不使用远程日志，则该进程可以禁用
nfs	如果不是使用 NFS 服务，则该进程可以禁用
ntpd	NTP(网络时间协议)用来与时间服务器进行时间同步。如果不需要进行时间同步，则该进程可以禁用
ntpdate	用通过轮询 NTP 服务器得到的时间设置本地日期和时间。如果不需要进行时间同步，则该进程可以禁用
numad	动态监视系统的运行和内存的使用，并动态平衡 CPU 和内存的负载。如果不使用 MUMA 架构的服务器，则该进程可以禁用
oddyobd	oddjobd 服务提供支持，当无特权的应用程序需要执行一组特权操作时，代表它们执行，可以禁用
postfix	系统中的 MTA，如果服务器不需要邮件服务，则该进程可以禁用
psacct	审计系统账号连接时间，用户操作，可以禁用
quota_nld	监听 netlink socket 并通过内核产生磁盘配额记录警告，并将其传递给系统 D-Bus 或终端用户，可以禁用
radvd	定期发送 IPv6 路由通告消息和回应路由请求消息，可以禁用
rdisc	用来在本地子网中发现路由器，可以禁用
restorecond	使用 inotify 寻找在/etc/selinux/restorecond.conf 文件中指定的新文件，并恢复正确的安全上下文。如果不使用 SELinux，则该进程可以禁用
rhnsd	负责定期连接到红帽网络服务器来检查更新、通知。如果没有购买红帽的服务，则该进程可以禁用