

**SEAWAY Edge**

**运维指南**

中科海微（北京）科技有限公司

Seaway Technologies Co. Ltd

## **目录**

[目录 1](#_Toc1738187314)

[文档修订目录 1](#_Toc1494136257)

[文档版本 1](#_Toc1421787028)

[Version3.2 1](#_Toc924039427)

[第一章 运维概述 4](#_Toc1876398132)

[第二章 系统资源与性能 4](#_Toc826048329)

[2.1 CPU 4](#_Toc2059971295)

[2.1.1 基本概念 4](#_Toc206198131)

[2.1.2 常用CPU性能分析工具 5](#_Toc1680865106)

[2.2 内存 6](#_Toc152460257)

[2.3 I/O 7](#_Toc451548528)

[第三章 信息收集 8](#_Toc2116385245)

[3.1 查看OS信息 8](#_Toc1315167454)

[3.2 查看硬件信息 8](#_Toc2117704454)

[3.3 查看软件信息 9](#_Toc2012276647)

[3.4 查看OS日志 9](#_Toc1761133173)

[第四章 故障应急处理 10](#_Toc598132010)

[4.1 触发kdump重启 10](#_Toc433740463)

[4.2 强制重启 10](#_Toc1316463723)

[4.3 重启网络 10](#_Toc281777420)

[4.4 修复文件系统 10](#_Toc631656305)

[4.5 手动dropcache 11](#_Toc1235851014)

[4.6 救援模式和单用户模式 11](#_Toc486158514)

[第五章 常用工具 12](#_Toc1838351610)

[5.1 ftrace 12](#_Toc1328279881)

[5.2 strace 14](#_Toc1307449402)

[5.3 kdump 14](#_Toc1249423310)

[第六章 常用技能 16](#_Toc962470804)

[6.1 配置网络 16](#_Toc1399973624)

[6.2 管理RPM包 16](#_Toc1525862036)

[6.3 配置SSH 20](#_Toc2061010225)

# 文档修订目录

## 文档版本

## Version3.2

|  |  |  |  |
| --- | --- | --- | --- |
| 文档版本号 | 修订日期 | 修订内容 | 修订人 |
| V3.1 | 2023/04/29 | 初订 | 陈海鑫 |

**前言**

在使用本手册之前，请您认真阅读以下使用许可协议，只有在同意以下使用许可协议的情况下方能使用本手册中介绍的产品。

**版权声明**

中科海微（北京）科技有限公司版权所有，并保留对本文档及本声明的最终解释权和修改权。本文档中出现的任何文字叙述、文档格式、插图、照片、方法、过程等内容，除另有特别注明外，其著作权或其他相关权利均属于中科海微（北京）科技有限公司。未经中科海微（北京）科技有限公司书面同意，任何人不得以任何方式或形式对本手册内的任何部分进行复制、摘录、备份、修改、传播、翻译成其它语言、将其全部或部分用于商业用途**。**

**免责条款**

本文档依据现有信息制作，其内容如有更改，恕不另行通知。中科海微（北京）科技有限公司在编写该文档的时候已尽最大努力保证其内容准确可靠，但中科海微（北京）科技有限公司不对本文档中的遗漏、不准确、或错误导致的损失和损害承担责任。

**技术支持与信息反馈**

如果您在使用我们的产品时遇到问题,或者您认为我们的产品有某些功能缺陷,请访问我们的官网www.haiwei.tech联系我们的客服,我们将为您解决问题和反馈;或者需要技术支持指导以及有任何宝贵意见，也请您通过官网或者电话联系我们：

**运维指南简介**

本文档提供了SeawayEdge操作系统的运维指南，介绍运维中常用的性能监测、信息收集、故障应急处理以及工具等。

**运维指南**

本文档主要适用于使用SeawayEdge的用户。用户需要具备以下经验和技能：

* 熟悉Linux基本操作
* 对Linux有一定了解

# 运维概述

​ IT运维是指企业IT部门采用技术手段对IT系统进行管理，是一种全面、复杂而又具体的服务。日常的IT运维服务主要包括了软件管理和硬件管理等。在软件管理中，通过操作系统维护设备的稳定性和高效性是IT运维的核心和重点部分。​ 具体来说，通过监测设备中CPU、内存和I/O等性能的动态变化，可以有效预防或定位相关问题。例如，由于各种业务原因导致CPU负载过高，导致服务响应变慢等问题，此时需要对CPU的使用情况进行监测。当内存占用率持续很高时，需要使用内存分析工具针对相关硬件或进程进行监测。进行相关读/写操作效率低时，需要监测I/O数据用来评估I/O性能等。​ 此外，当系统发生崩溃、死锁或者死机等故障时，需要通过操作系统做一些应急处理，用来对故障进行快速排查和修复。例如，通过触发kdump，收集系统内核信息，随后对内核信息进行分析。当需要进行修改系统密码操作时，进入单用户模式，修改root密码。经常强制上下电导致文件系统损坏，当系统无法自动修复成功时，需要手动进行修复，调整drop\_caches内容来手动释放内存等。同时，需要对故障时的现场信息进行收集，如日志文件和设备文件等，以便在后续能够更全面地进行问题根因分析。​ 因此，熟悉操作系统性能分析工具的使用以及故障修复的操作，是实现完善的IT运维管理的关键。

# 第二章 系统资源与性能

## 2.1 CPU

## 2.1.1 基本概念

中央处理器（Central Processing Unit，简称CPU）是计算机的主要设备之一，其功能是解释计算机指令以及处理计算机软件中的数据。

* 物理核：可以真实看到的CPU核，有独立的电路元件以及L1、L2缓存，可以独立地执行指令。一个CPU可以有多个物理核。
* 逻辑核：在同一个物理核内，逻辑层面上存在的核。一般一个物理核对应一个线程，但是如果开启了超线程，当超线程数量为n时，一个物理核可以分成n个逻辑核。 可以通过lscpu命令查看服务器中有多少个CPU，每个CPU中有几个物理核，以及每个CPU有几个逻辑核。

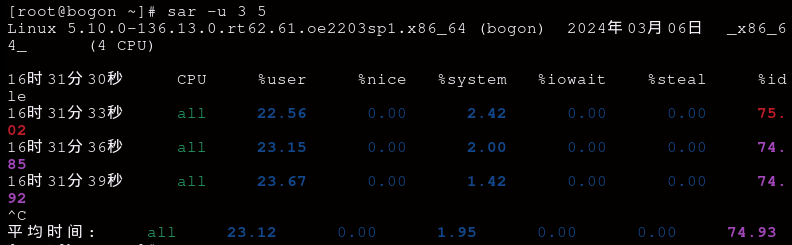
## 2.1.2 常用CPU性能分析工具

1. uptime：可用于打印系统平均负载，通过查看最后三个数字表示系统在过去1分钟、5分钟和15分钟内的平均负载情况，可以判断平均负载的变化趋势。 平均负载大于CPU数量时表示CPU不足以服务线程，部分线程在等待；平均负载小于CPU数量，代表当前还有余量。

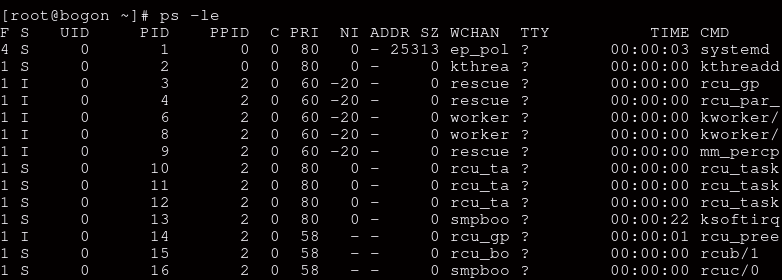


2. vmstat：可以动态地了解系统资源的使用情况，以及查看系统中是哪一个环节最占用系统资源。 通过vmstat -h命令可以查看命令详解参数。 例如：

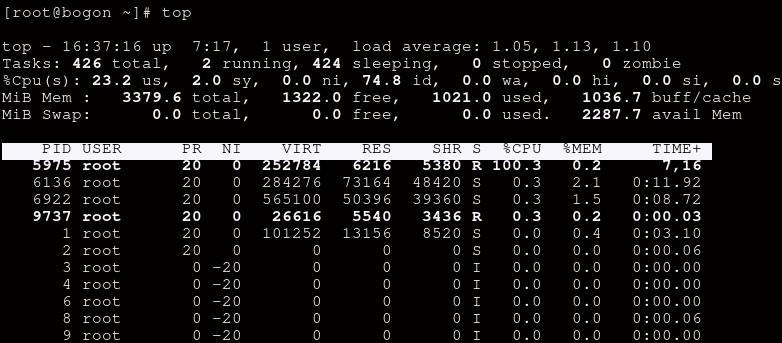
3. sar: 可用于分析系统性能，可以用来观察当前的活动以及配置，用以归档和报告历史统计信息。(需安装sysstat ：yum install -y sysstat)



4. ps：可用于查看正在运行的进程



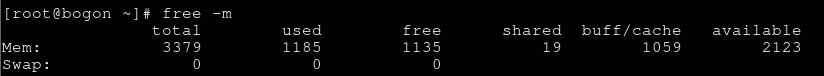
5. top：可以动态地持续监听进程的运行状态，显示最消耗CPU的进程



## 内存

常用内存分析工具/方式

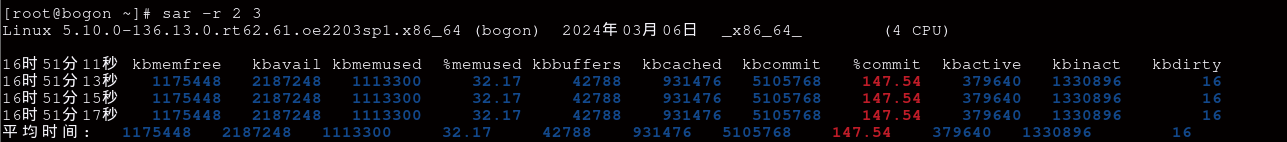
1. free：可用于显示系统内存状态。



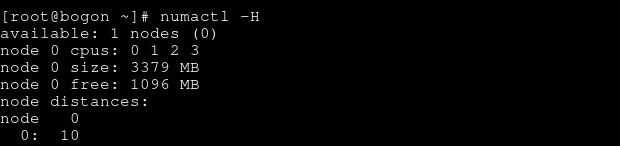
1. vmstat：可以动态地监控系统内存，查看系统内存的使用情况。



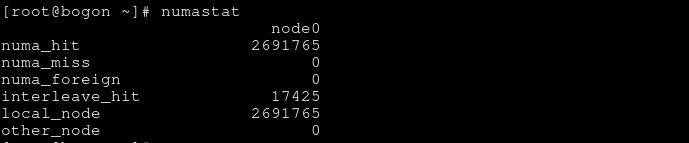
1. sar: 可用于监控系统的内存使用情况。



1. numactl：可用于查看numa节点配置和状态。



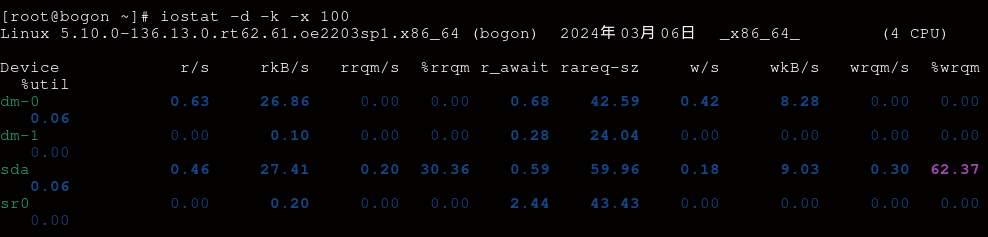
1. numastat：可用于观察各个NUMA节点的状态。



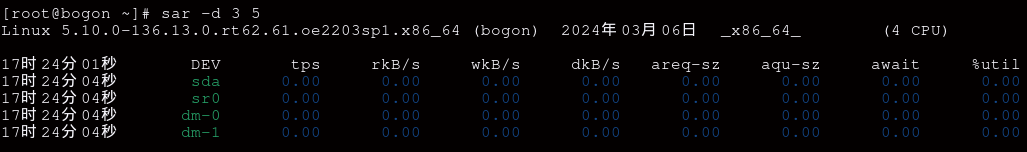
## I/O

常用I/O性能分析工具

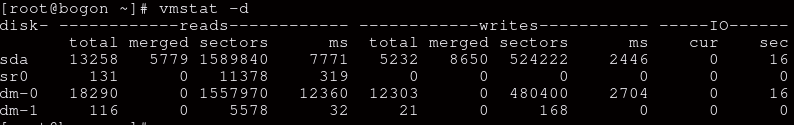
1. iostat：可以汇报所有在线磁盘的统计信息。



1. sar：可用于查看系统磁盘的读写性能。



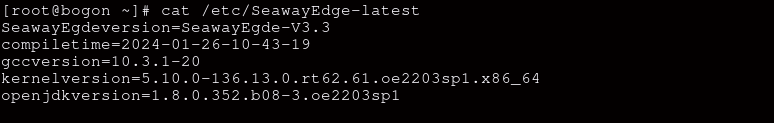
1. vmstat：报告磁盘相关统计信息。



# 第三章 信息收集

## 3.1 查看OS信息

1. 查看OS信息



2. 查看内核版本信息：



## 3.2 查看硬件信息

1. 查看cpu相关参数：



2. 查看系统内存信息：



3. 查看内存硬盘信息：



4. 查看硬盘和分区分布：



5. 查看硬盘和分区的详细信息：



6. 查看网卡硬件信息：



7. 查看所有网络接口：



8. 查看某个网络接口的详细信息：



9. 查看pci信息：



10. 查看设备树：



11. 查看bios信息：



## 3.3 查看软件信息

因此加载uacce.ko之后需要加载hisi\_qm.ko，然后加载ZIP、HPRE、SEC模块的驱动。

1. 查看软件包的详细信息



1. 查看软件包提供的模块



1. 查看所有已安装软件包



1. 查看软件包文件列表



## 3.4 查看OS日志

1. 查看系统启动后的信息和错误日志：



2. 查看安全相关的日志信息：



3. 查看邮件相关的日志信息：



4. 查看定时任务相关的日志信息：



5. 查看UUC和news设备相关的日志信息：

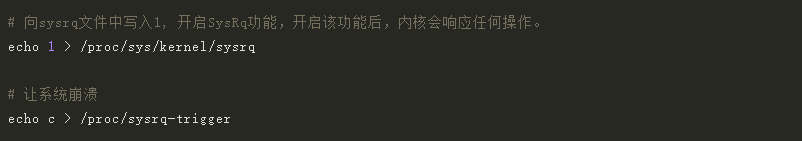


6. 查看守护进程启动和停止相关的日志信息：



# 第四章 故障应急处理

## 4.1 触发kdump重启



## 4.2 强制重启

强制重启：（手动输入以下指令，或者通过iBMC强制上下电）



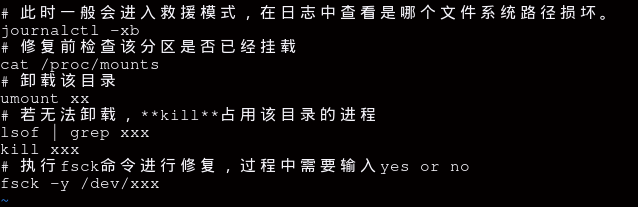
## 4.3 重启网络

SeawayEdge使用NetworkManager来管理网络，执行下面命令即可重启网络。

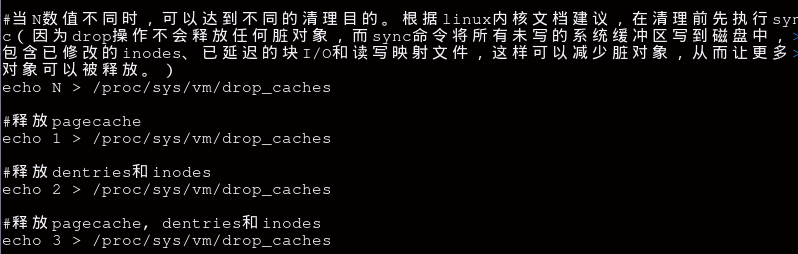


## 4.4 修复文件系统

当系统强行上下电重启后，文件系统可能受到损坏，系统启动时会自动检查并修复文件系统，当文件系统没有自动修复成功时，便需要手动使用fsck进行扫描和修复。



## 4.5 手动dropcache



## 4.6 救援模式和单用户模式

* 救援模式

选择Troubleshooting。选择Rescue a SeawayEdge system。按提示操作进行。1)Continue2)Read-only mount3)Skip to shell4)Quit(Reboot)

* 单用户模式

在登录界面，输入字母e，进入grub界面，在linux行加入init=/bin/sh，按ctrl+x进入界面。1)执行mount -o remount,rw /。2)执行修改密码等操作。3)exit退出。

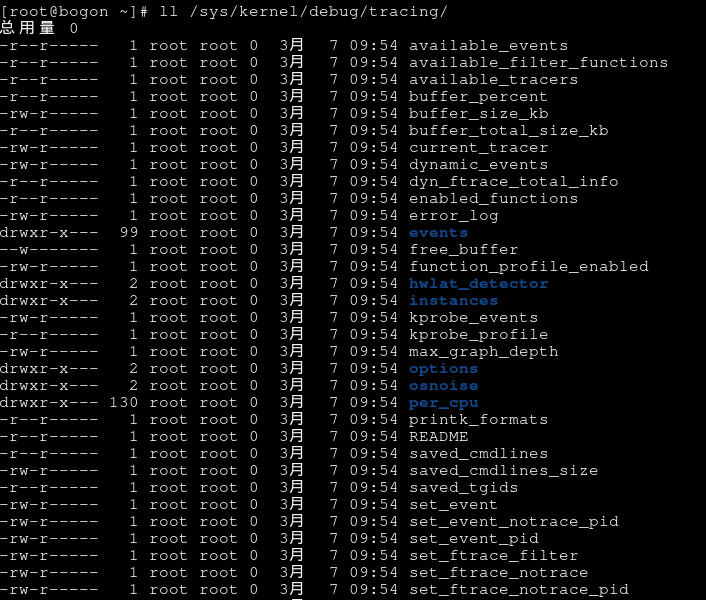
# 第五章 常用工具

## 5.1 ftrace

ftrace：是一个针对linux kernel内核空间的debug工具，内核中会提供trace events供用户追踪。而ftrace则可以将events抓取出来，让用户能够直观地看到这些事件，同时也可以追踪内核的函数。ftrace的配置和使用：要使用ftrace，需要将其相关的依赖编译进内核，SeawayEdge已经默认编译了ftrace选项，如果没开启可以在menuconfig里选择Kernel hacking -> Tracers -> Trace syscalls开启，同时还要编译debugfs，选择Kernel hacking -> Generic Kernel Debugging Instruments -> Debug Filesystem。

* ftrace的功能配置

ftrace通过debugfs向用户空间提供访问接口，内核配置debugfs后，会创建/sys/kernel/debug目录，debugfs文件系统就是挂载到该目录。如果内核支持ftrace相关的配置项后，会在debugfs下创建一个tracing目录，debugfs文件系统会挂载到该目录，该目录内容如下图所示：



* ftrace debugfs接口介绍

如下对常用的文件做简单说明：

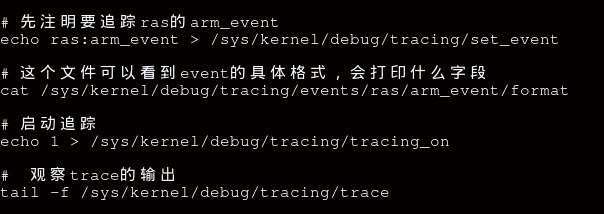
available\_tracers：可用的跟踪程序current\_tracer：正在运行的跟踪程序available\_events：列举了系统所有可用的trace eventsevents：该目录对events按模块做了区分。set\_event：列举当前要追踪的eventstracing\_on：用于控制跟踪打开或停止，echo 0 > tracing\_on表示关闭，1表示打开trace：查看跟踪数据

* 可用的跟踪程序



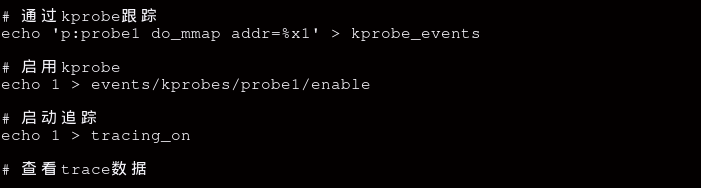
function：一个无需参数的函数调用跟踪程序function\_graph：一个使用子调用的函数调用跟踪程序

* 追踪event

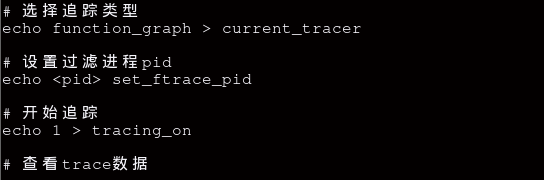


* 追踪内核函数入参

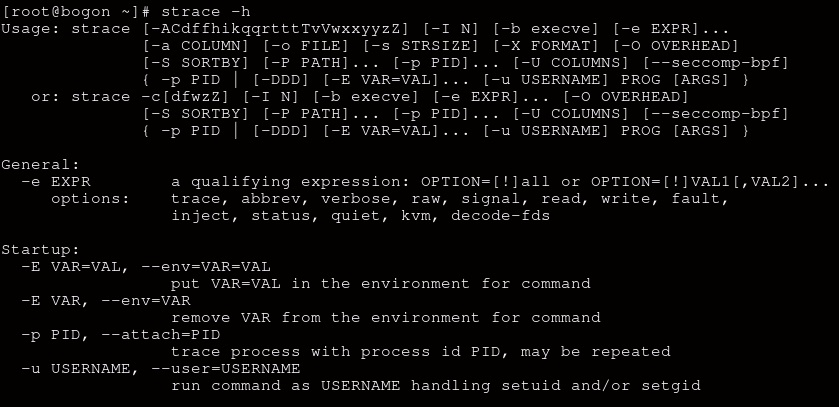
选择跟踪mmap，其对应系统调用do\_mmap，选择输出addr入参。



* 追踪函数调用



## 5.2 strace

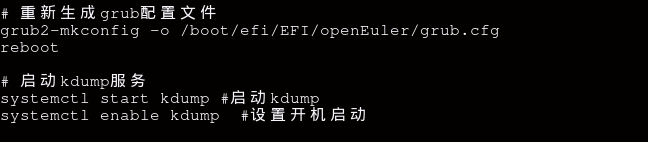
strace命令是一个诊断、调试工具，可以通过使用strace对应用的系统调用及信号传递来进行分析，从而达到解决问题或了解应用执行过程的目的。可以通过strace -h来查看strace提供了哪些功能。

最常用的使用方式（追踪xx命令，跟踪forks，打印时间，结果输出到output文件中）。

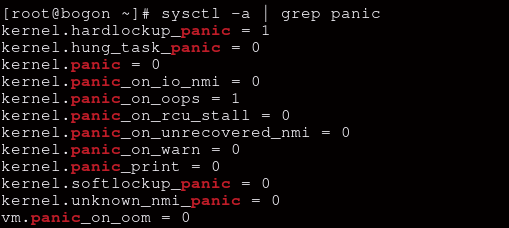


## 5.3 kdump

kdump是系统运行在某个时间点的内存状态的快照，便于运维人员debug分析系统挂掉的原因，常用于系统宕机和panic。



crash触发：



crash分析：



crash调试命令格式为 command args，command为需要执行的命令，args为部分调试命令需要的参数。

| 命令 | 用途 |
| --- | --- |
| help | 打印命令的help信息，可展示支持命令，也可查看具体命令的help信息，如help bt。 |
| bt | 打印函数调用栈信息。 |
| log | 打印系统消息缓冲区，可追加参数，如log。 |
| ps | 显示进程的状态，>表示进程为活跃状态。 |
| dis | 对给定函数或者地址进行反汇编，如dis -l [func]。 |
| mount | 展示当前的文件系统信息 |

# 第六章 常用技能

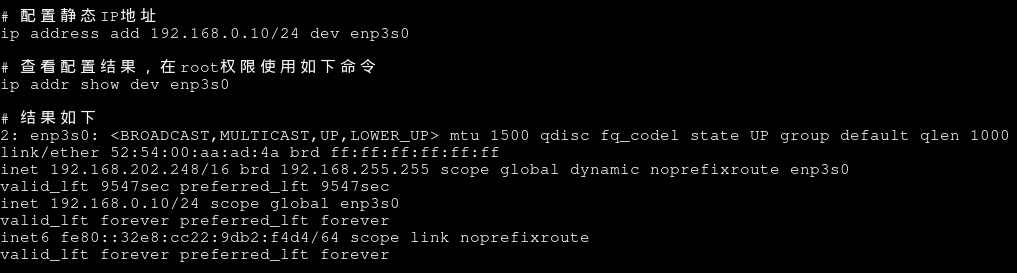
## 6.1 配置网络

1. 配置IP地址

用IP命令为接口配置地址，interface-name为网卡名称。



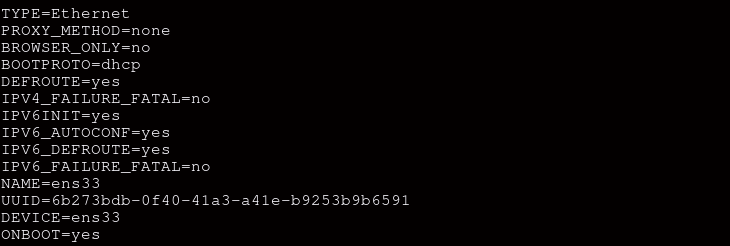
2. 配置静态地址



3. 配置静态路由



4. 通过ifcfg文件配置网络



## 6.2 管理RPM包

RPM以数据库记录的方式将需要的软件安装到Linux主机的一套管理程序，特点是将要安装的软件先编译并打包，通过包装好的软件中默认的数据库记录，记录这个软件在安装的时候需要的依赖属性模块，在用户的Linux主机安装时，RPM会先根据软件里的记录数据，查询Linux主机的依赖属性软件是否满足：

* 若满足则予以安装。
* 若不满足则不安装。

安装时将该软件的信息全部写入RPM的数据库中以便后续查询、验证与卸载。1. 安装RPM包



2. 查询已安装的RPM包：



3. 查询所有已安装的RPM包：



4. 查询已安装的RPM包详细信息：



5. 查看未安装的RPM包所有文件：



6. 查看未安装的RPM包依赖项：



7. 验证所有已安装的RPM包：



8. 查看特定文件的RPM包：



9. 查看已安装RPM包中的文件：



10. 查看已安装RPM包文档：



11. 升级已安装的RPM包：



12. 移除已安装的RPM包：



13. 重建损坏的RPM数据库：



* Dnf命令

dnf命令及其相关概述：

| 命令 | 概述 |
| --- | --- |
| repolist | 显示已配置的软件repo源。 |
| install | Linux上安装单个或多个软件包。 |
| upgrade | 升级Linux上的一个或多个软件包。 |
| list | 列出一个或一组软件包。 |
| info | 显示关于软件包或软件包组的详细信息。 |
| updateinfo | 显示关于包的公告信息。 |
| search | 在软件包详细信息中搜索指定字符串。 |
| check-update | 检查是否有软件包升级。 |
| remove | 从系统中移除一个或多个软件包。 |
| reinstall | 重装一个包。 |
| downgrade | 降级软件包。 |
| autoremove | 删除所有原先因为依赖关系安装的不需要的软件包。 |
| distro-sync | 同步已经安装的软件包到最新可用版本。 |
| makecache | 创建元数据缓存。 |
| repository-package | 对指定仓库中的所有软件包运行命令。 |
| provides | 查找提供指定内容的软件包。 |
| group | 显示或使用组信息。 |
| history | 显示或使用事务历史。 |
| clean | 删除已缓存的数据。 |

1. 显示已配置的软件repo：



1. 安装单个或多个软件包：



1. 重新安装软件包：



1. 升级软件包：



1. 降级软件包：



1. 列出一个或一组软件包：



1. 查看软件包详细信息：



1. 搜索软件包：



1. 卸载一个或多个软件包：

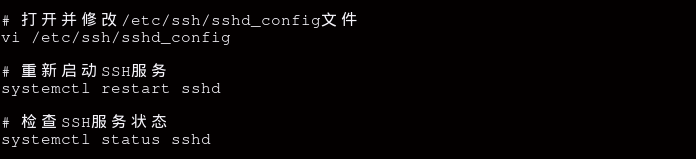


1. 自动删除因为依赖关系安装的软件包：



## 6.3 配置SSH

* 配置SSH服务：



* SSH服务配置文件主要选项：

