**CentOS** 丛书目录 — 系统管理 — 网络服务 — 应用部署

# 日志系统和系统日志

内容提要

1. 理解syslog系统
2. 熟悉syslogd的配置文件及其语法
3. 学会查看系统日志
4. 理解日志滚动的必要性及实现方法

# 日志系统

## 什么是 syslog

日志的主要用途是系统审计、监测追踪和分析统计。

为了保证 Linux 系统正常运行、准确解决遇到的各种各样的系统问题，认真地读取日志文件是管理员的一项非常重要的任务。

Linux 内核由很多子系统组成，包括网络、文件访问、内存管理等。子系统需要给用户传送一些消息，这些消息内容包括消息的来源及其重要性等。所有的子系统都要把消息送到一个可以维护的公用消息区，于是，就有了 syslog。

syslog 是一个综合的日志记录系统。它的主要功能是：方便日志管理和分类存放日志。 syslog 使程序设计者从繁重的、机械的编写日志文件代码的工作中解脱出来，使管理员更好地控制日志的记录过程。在 syslog 出现之前，每个程序都使用自己的日志记录策略。管理员对保存什么信息或是信息存放在哪里没有控制权。

syslog 能设置成根据输出信息的程序或重要程度将信息排序到不同的文件。例如，由于核心信息更重要且需要有规律地阅读以确定问题出在哪里，所以要把核心信息与其他信息分开来，单独定向到一个分离的文件中。

管理员可以通过编辑 /etc/syslog.conf 来配置它们的行为。

## syslogd 的配置文件

syslogd 的配置文件 /etc/syslog.conf 规定了系统中需要监视的事件和相应的日志的保存位置。使用如下命令：

```
cat  /etc/syslog.conf
```

可以查看此文件的内容为：

```
# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                         /dev/console

# 将 info 或更高级别的消息送到 /var/log/messages，
# 除了 mail/news/authpriv/cron 以外。
# 其中*是通配符，代表任何设备；none 表示不对任何级别的信息进行记录。
*.info;mail.none;news.none;authpriv.none;cron.none      /var/log/messages

# 将 authpirv 设备的任何级别的信息记录到 /var/log/secure 文件中，
# 这主要是一些和认证、权限使用相关的信息。
authpriv.*                                      /var/log/secure
```

```
# 将 mail 设备中的任何级别的信息记录到 /var/log/maillog 文件中，
# 这主要是和电子邮件相关的信息。
mail.*                                                  -/var/log/maillog


# 将 cron 设备中的任何级别的信息记录到 /var/log/cron 文件中，
# 这主要是和系统中定期执行的任务相关的信息。
cron.*                                                  /var/log/cron

# 将任何设备的 emerg 级别或更高级别的消息发送给所有正在系统上的用户。
*.emerg                                                 *

# 将 uucp 和 news 设备的 crit 级别或更高级别的消息记录到 /var/log/spooler 文件中。
uucp,news.crit                                          /var/log/spooler

# 将和本地系统启动相关的信息记录到 /var/log/boot.log 文件中。
local7.*                                                /var/log/boot.log

# 将 news 设备的 crit 级别的消息记录到 /var/log/news/news.crit 文件中。
news.=crit                                              /var/log/news/news.crit
# 将 news 设备的 err 级别的消息记录到 /var/log/news/news.err 文件中。
news.=err                                               /var/log/news/news.err
# 将 news 设备的 notice 或更高级别的消息记录到 /var/log/news/news.notice 文件中。
news.notice                                             /var/log/news/news.notice
```

该配置文件的每一行的格式如下：

```
facility.priority          action
设备.级别                       动作
```

其中：

1、设备字段用来指定需要监视的事件。它可取的值如下：

| 设备字段 | 说明 |
|---|---|
| authpriv | 报告认证活动。通常，口令等私有信息不会被记录 |
| cron | 报告与cron和at有关的信息 |
| daemon | 报告与xinetd有关的信息 |
| kern | 报告与内核有关的信息。通常这些信息首先通过klogd传送 |
| lpr | 报告与打印服务有关的信息 |
| mail | 报告与邮件服务有关的信息 |
| mark | 在默认情况下每隔20分钟就会生成一次表示系统还在正常运行的消息。Mark消息很像经常用来确认远程主机是否还在运行的"心跳信号"（Heartbeat）。Mark消息另外的一个用途是用于事后分析，能够帮助系统管理员确定系统死机发生的时间。 |
| news | 报告与网络新闻服务有关的信息 |
| syslog | 由syslog生成的信息 |
| user | 报告由用户程序生成的任何信息，是可编程缺省值 |
| uucp | 由UUCP生成的信息 |
| local0-local7 | 与自定义程序一起使用 |
| * | *代表除了mark之外的所有功能 |

2、级别字段用于指明与每一种功能有关的级别和优先级。它可取的值如下：

| 级别字段 | 说明 |
|---|---|
| emerg | 出现紧急情况使得该系统不可用，有些需广播给所有用户 |
| alert | 需要立即引起注意的情况 |

| | |
|---|---|
| crit | 危险情况的警告 |
| err | 除了emerg、alert、crit的其他错误 |
| warning | 警告信息 |
| notice | 需要引起注意的情况，但不如err、warning重要 |
| info | 值得报告的消息 |
| debug | 由运行于debug模式的程序所产生的消息 |
| none | 用于禁止任何消息 |
| * | 所有级别，除了none |

3、动作字段用于描述对应功能的动作。它可取的值如下：

| 动作字段 | 说明 |
|---|---|
| file | 指定一个绝对路径的日志文件名记录日志信息 |
| username | 发送信息到指定用户，*表示所有用户 |
| device | 将信息发送到指定的设备中，如/dev/console |
| @hostname | 将信息发送到可解析的远程主机hostname，且该主机必须正在运行syslogd并可以识别syslog的配置文件 |

syslog 可以为某一事件指定多个动作，也可以同时指定多个功能和级别，它们之间用分号间隔。

参考

- http://man.cx/syslog.conf [http://www.proxyserve.net/index.php?q=aHR0cDovL21hbi5jeC9zeXNsb2cuY29uZg%3D%3D](5)

# 查看日志文件

## 常见的日志文件

日志文件通常存放在 /var/log 目录下。在该目录下除了包括 syslogd 记录的日志之外，同时还包含所有应用程序的日志。

为了查看日志文件的内容必须要有 root 权限。日志文件中的信息很重要，只能让超级用户有访问这些文件的权限。

管理员可以使用下面的命令

```
ls /var/log/
```

查看系统中使用的日志文件，常用的日志文件如表所示。

| 日志文件 | 说明 |
|---|---|
| audit/ | 存储 auditd 审计守护进程的日志目录 |
| conman/ | 存储 ConMan 串行终端管理守护进程的日志目录 |
| cups/ | 存储 CUPS 打印系统的日志目录 |
| httpd/ | 记录 apache 的访问日志和错误日志目录 |
| mail/ | 存储 mail 日志的目录 |
| news/ | 存储 INN 新闻系统的日志目录 |
| pm/ | 存储电源管理的日志目录 |
| ppp/ | 存储 pppd 的日志目录 |
| prelink/ | prelink 的日志目录 |

| | |
|---|---|
| samba/ | 记录 Samba 的每个用户的日志目录 |
| squid/ | 记录 Squid 的日志目录 |
| vbox/ | ISDN 子系统的日志目录 |
| acpid | 存储 acpid 高级电源管理守护进程的日志 |
| anaconda.* | Red Hat/CentOS 安装程序 anaconda 的日志，参考 redhat 安装程序 anaconda 分析 [http://www.proxyserve.net/index.php?q=aHR0cDovL3d3dy5pYm0uY29tL2RldmVsb3BlcndvcmtzL2NuL2xpbnV4L2wtYW5hY29uZGEvaW5kZXguaHRtbA%3D%3D] |
| boot.log | 记录系统启动日志 |
| btmp | 记录登陆未成功的信息日志 |
| cron | 记录守护进程 crond 的日志 |
| dmesg | 记录系统启动时的消息日志 |
| lastlog | 记录最近几次成功登录的事件和最后一次不成功的登录 |
| maillog | 记录邮件系统的日志 |
| messages | 由 syslogd 记录的 info 或更高级别的消息日志 |
| rpmpkgs | 记录了当前安装的所有 rpm 包 |
| secure | 由 syslogd 记录的认证日志 |
| spooler | 由 syslogd 记录的 uucp 和 news 的日志 |
| vsftpd.log | 记录 vsftpd 的日志 |
| wtmp | 一个用户每次登录进入和退出时间的永久记录 |
| yum.log | 记录 yum 的日志 |

## 查看文本日志文件

绝大多数日志文件是纯文本文件，每一行就是一个消息。只要是在Linux下能够处理纯文本的工具都能用来查看日志文件。可以使用 cat、tac、more、less、tail 和 grep 进行查看。

下面以 /var/log/messages 为例，说明其日志文件的格式。

- 该文件中每一行表示一个消息，而且都由四个域的固定格式组成：
- 时间标签(Timestamp)：表示消息发出的日期和时间。
- 主机名(Hostname)：表示生成消息的计算机的名字。
- 生成消息的子系统的名字：可以是"Kernel"，表示消息来自内核或者是进程的名字，表示发出消息的程序的名字。在方括号里的是进程的PID。
- 消息(Message)，即消息的内容。

例如：

```
# syslog 发出的消息，说明了守护进程已经在 Dec 16，03:32:41 重新启动了。
Dec 16 03:32:41 cnetos5 syslogd 1.4.1: restart.
# 在 Dec 19，00:20:56 启动了内核日志 klogd
Dec 19 00:20:56 cnetos5 kernel: klogd 1.4.1, log source = /proc/kmsg started.
# 在 Dec 19，00:21:01 启动了xinetd
Dec 19 00:21:01 cnetos5 xinetd[2418]: xinetd Version 2.3.14 started with libwrap
 loadavg labeled-networking options compiled in.
```

可以看出，实际上在 /var/log/message 文件中的消息都不是特别重要或紧急的。

## 查看非文本日志文件

也有一些日志文件是二进制文件，需要使用相应的命令进行读取。

### lastlog

使用 lastlog 命令来检查某特定用户上次登录的时间，并格式化输出上次登录日志 /var/log/lastlog 的内容。例如：

```
# lastlog
Username         Port     From             Latest
root             pts/0    192.168.0.77     Wed Dec 19 02:11:14 +0800 2007
bin                                        **Never logged in**
……
osmond           pts/0    192.168.0.77     Wed Dec 19 07:37:34 +0800 2007
```

### last

last 命令往回搜索 /var/log/wtmp 来显示自从文件第一次创建以来登录过的用户。例如：

```
# last
osmond   pts/0        192.168.0.77     Wed Dec 19 07:37   still logged in
osmond   pts/0        192.168.0.77     Wed Dec 19 02:19 - 07:14  (04:54)
root     pts/0        192.168.0.77     Wed Dec 19 02:11 - 02:17  (00:06)
osmond   pts/0        192.168.0.77     Wed Dec 19 00:43 - 02:11  (01:27)
reboot   system boot  2.6.18-53.el5    Wed Dec 19 00:20         (32+16:26)
root     tty1                          Fri Dec 14 20:33 - down   (15:11)
reboot   system boot  2.6.18-53.el5    Sun Dec  9 01:08          (00:05)

wtmp begins Sun Dec  9 01:08:00 2007
```

### lastb

lastb 命令搜索 /var/log/btmp 来显示登录未成功的信息。例如：

```
# lastb
osmond   ssh:notty    192.168.0.77     Sat Dec 15 17:24 - 17:24  (00:00)
rroot    tty1                          Tue Dec 11 06:40 - 06:40  (00:00)

btmp begins Tue Dec 11 06:40:57 2007
```

### who

who 命令查询 wtmp 文件并报告当前登录的每个用户。who 命令的缺省输出包括用户名、终端类型、登录日期及远程主机。例如：

```
$ who
root     tty1         2007-12-20 16:49
osmond   pts/0        2007-12-19 07:37 (192.168.0.77)
```

## 日志滚动

## 为什么使用日志滚动

所有的日志文件都会随着时间的推移和访问次数的增加而迅速增长，因此必须对日志文件进行定期清理以免造成磁盘空间的不必要的浪费。同时也加快了管理员查看日志所用的时间，因为打开小文件的速度比打开大文件的速度要快。

## logrotate

Linux 下有一个专门的日志滚动处理程序 logrotate [http://www.proxyserve.net/index.php?q=aHR0cDovL21hbi5jeC9sb2dyb3RhdGU%3D]，能够自动完成日志的压缩、备份、删除、和日志

邮寄等工作。每个日志文件都可被设置成每日，每周或每月处理，也能在文件太大时立即处理。一般把 logrotate 加入到系统每天执行的计划任务中，这样就省得管理员自己去处理了。

其命令格式为：

```
logrotate [选项] <configfile>
```

选项说明如下：

- -d：详细显示指令执行过程，便于排错或了解程序执行的情况。
- -f：强行启动记录文件维护操作，即使 logrotate 指令认为无需要亦然。
- -m command：指定发送邮件的程序，默认为 /usr/bin/mail。
- -s statefile：使用指定的状态文件。
- -v：在执行日志滚动时显示详细信息。
- -?：显示命令帮助。
- --usage：显示使用摘要信息。

<configfile> 是 logrotate 命令的配置文件的路径。

## logrotate 的配置文件

管理员可以在 logrotate 的配置文件中设置日志的滚动周期，日志的备份数目，以及如何备份日志等等。

- logrotate 默认的主配置文件是 /etc/logrotate.conf
- /etc/logrotate.d 的目录下的文件，这些文件被 include 到主配置文件 /etc/logrotate.conf 中

在这些文件中可以使用如下的配置语句。

| 配置语句 | 功能 |
|---|---|
| compress | 对滚动的旧日志文件使用 gzip 压缩。 |
| nocompress | 不压缩滚动的旧日志文件。 |
| copytruncate | 用在处于打开状态的日志文件，将当前日志备份并截断。 |
| nocopytruncate | 备份日志文件但是不截断。 |
| create mode owner group | 滚动日志时使用指定的文件模式创建新的日志文件。 |
| nocreate | 不创建新的日志文件。 |
| delaycompress | 和 compress 一起使用，转储的日志文件到下一次滚动时才压缩。 |
| nodelaycompress | 覆盖 delaycompress 选项，转储同时压缩。 |
| ifempty | 即使是空文件也滚动日志，这个是 logrotate 的缺省选项。 |
| notifempty | 如果是空文件的话，不滚动日志 |
| errors address | 将滚动日志时的错误信息发送到指定的 Email 地址。 |
| mail address | 把转储的日志文件发送到指定的 E-mail 地址。 |
| nomail | 转储时不发送日志文件。 |
| olddir directory | 将滚动的旧日志文件存储到指定的目录，必须和当前日志文件在同一个文件系统。 |
| noolddir | 将滚动的旧日志文件和当前日志文件放在同一个目录下。 |
| prerotate/endscript | 在滚动日志以前需要执行的命令可以放入此语句括号内，这两个关键字必须单独成行。 |
| postrotate/endscript | 在滚动日志以后需要执行的命令可以放入此语句括号内，这两个关键字必须单独成行。 |
| daily | 指定日志滚动周期为每天。 |
| weekly | 指定日志滚动周期为每周。 |

| | |
|---|---|
| monthly | 指定日志滚动周期为每月。 |
| rotate n | 指定日志文件删除之前日志滚动的备份次数，0 指没有备份，5 指保留 5 个备份 |
| tabootext [+] list | 让 logrotate 不转储指定扩展名的文件，缺省的扩展名是：.rpmorig, .rpmsave, dpkg-dist, .dpkg-old, .dpkg-new, .disabled, ,v, .swp, .rpmnew, 和 ~。 |
| size n | 当日志文件到达指定的大小时才进行日志滚动，n 可以指定 bytes(缺省)，或使用 G/M/k 后缀单位。 |

1. 在 /etc/logrotate.conf 中可以使用以上的配置语句设置全局值
2. 在 /etc/logrotate.conf 中使用 include 语句包含的配置文件中也可以使用上述的配置语句，被 include 的配置文件中的语句会覆盖 /etc/logrotate.conf 中的配置
3. 为指定的文件配置日志滚动使用如下的语法

```
# 注释

/full/path/to/logfile {

    配置语句1
    ……
    配置语句n

}
```

## CnetOS 默认的 /etc/logrotate.conf

```
$ cat /etc/logrotate.conf
```

```
# see "man logrotate" for details
# rotate log files weekly
# 每周清理一次日志文件
weekly

# 保存过去四周的日志文件
rotate 4

# 清除旧日志文件的同时，创建新的空日志文件
create

# 若使用压缩的日志文件，请删除下面行的注释符
#compress

# 包含 /etc/logrotate.d 目录下的所有配置文件
include /etc/logrotate.d

# 设置 /var/log/wtmp 的日志滚动
/var/log/wtmp {
    monthly
    create 0664 root utmp
    rotate 1
}

# system-specific logs may be also be configured here.
```

可以使用 ls 命令显示 /etc/logrotate.d 目录：

```
$ ls /etc/logrotate.d
acpid   cups   mgetty  ppp     rpm    sa-update  syslog  vsftpd.log
conman  httpd  named   psacct  samba  squid      tux     yum
```

其中，每个文件的基本格式均相同，下面以 syslog 文件为例进行说明：

```
$ cat  /etc/logrotate.d/syslog
```

```
# 对日志文件 /var/log/messages、/var/log/secure、/var/log/maillog、
# /var/log/spooler、/var/log/boot.log 和 /var/log/cron 进行日志滚动
/var/log/messages /var/log/secure /var/log/maillog /var/log/spooler
/var/log/boot.log /var/log/cron {
  # 调用日志滚动通用函数
  sharedscripts
  # 在日志滚动之后执行语句括号 postrotate 和 endscript 之间的命令
  postrotate
    # 重新启动 syslogd
    /bin/kill -HUP `cat /var/run/syslogd.pid 2> /dev/null` 2> /dev/null || true
  endscript
}
```

另外，logrotate 是由 crond 运行的，在默认配置中，可以发现在 /etc/cron.daily 目录中有一个名为logrotate 的文件，该文件内容如下：

```
$ cat /etc/cron.daily/logrotate
```

```
#!/bin/sh

/usr/sbin/logrotate /etc/logrotate.conf
EXITVALUE=$?
if [ $EXITVALUE != 0 ]; then
    /usr/bin/logger -t logrotate "ALERT exited abnormally with [$EXITVALUE]"
fi
exit 0
```

表示由 crond 每天执行一次 logrotate，执行时读取其配置文件 /etc/logrotate.conf。

- 显示源文件
- 登录