openEuler

操作系统日志管理

课程介绍

- 本课程主要介绍了常见操作系统的日志文件，以及主要日志管理服务：
- 操作系统日志是用于分析操作系统行为最主要的手段，清晰完整的日志记录可以帮助使用者快速查找故障、分析安全事件、保障系统稳定运行。
- 学完本课程后，您将能够：了解openEuler常见的日志文件及其作用；熟悉主流日志记录服务rsyslog；掌握logrotate服务用于完成日志转储。

目录

CONTENT

openEuler

常用系统日志介绍

- dmesg

主要记录系统在开机时内核检测过程所产生的信息,通过执行dmesg命令查看.

- /var/log/wtmp or /var/log/faillog

这两个文件可以记录正确登陆系统者的账户信息(wtmp),与错误登陆时所使用的账户信息,last命令就是读取wtmp文件来获取的.

- /var/log/btmp

记录错误登陆日志,这个文件是二进制的,不能使用cat命令查看,而要使用lastb命令查看.

- /var/run/utmp

记录当前一登陆用户的信息,同样不能使用cat命令查看,而要使用w,who,users命令来查询.

- /var/log/lastlog

记录了系统上面所有账户最近一次登陆系统时的相关信息,lastlog命令就是读取这个文件里的记录来显示的.

- /var/log/secure

只要涉及到需要用户名和密码的操作,那么当登陆系统是(不论正确错误),都会记录到这里.

- /var/log/messages

这个文件非常重要,几乎系统发生的错误信息,或者重要信息都会被记录在这里.

- /var/log/cron

主要记录关于crontab计划任务的相关信息,比如,系统计划任务的错误配置,计划任务的修改等.

- /var/log/malilog or /var/log/mail/*

记录着邮件的往来信息,默认是postfix邮件服务器的一些信息.

rsyslog日志记录服务介绍

rsyslog的全称是rocket-fast system for log，具有较高的性能和安全性及模块化设计。Rsyslog可以接收多种来源的输入，并能将结果输出到不同的目的地。rsyslog每秒可提供超过一百万条消息给目标文件。

当前主流linux操作系统均使用rsyslog服务管理（对应旧版本的syslog服务）系统日志，它可以用于：

1）rsyslog守护进程配置为服务器运行，接收来自各种来源的输入

2）转换过滤格式化输出

3） rsyslog守护进程配置为客户端运行，将结果输出到不同的目的地（本地或者远端日志服务器）。

特点：

1 多线程。输入多线程、输出多线程等
2 可以通过多种协议进行传输。UDP、TCP、RELP、SSL、TLS
3 支持加密协议。ssl, tls, relp
4 强大的过滤器，实现过滤日志信息中任何部分的内容
5 自定义输出格式
6 可将日志写入到数据库

rsyslog日志格式facility介绍

Facility（产生日志的设施，从功能和程序上对日志收集进行分类）在rsyslog中指定了产生日志消息的子系统，包括：

auth	PAM认证相关日志
authpriv	SSH、FTP登录相关日志
cron	任务计划相关日志
daemon	守护进程相关日志
kern	内核相关日志
lpr	打印相关日志
mail	邮件相关日志
mark	标记相关日志
news	新闻相关日志
security	安全相关日志，与auth类似
syslog	rsyslog自己的日志
user	用户相关日志
uucp	UNIX to UNIX cp相关日志
local0 ~ local7	用户自定义使用设置日志
*	代表所有的facility

rsyslog日志格式severity介绍

severity代表日志的严重级别，包括：

Numerical code	Severity
0	Emergency: 会导致系统不可用的严重信息
1	Alert: 必须马上处理的警告信息
2	Critical: 比较严重的信息
3	Error: 错误信息
4	Warning: 警告信息
5	Notice: 不影响正常功能，需要注意的信息
6	Info: 一般信息
7	Debug: 程序或系统调试信息

rsyslog特性(一)

1 属性替代

Rsyslog预定义了一些属性，来代表消息中的信息，可以在定义输出格式、动态文件名的时候使用这些属性。常用的属性有：msg（消息体）、hostname、pri（消息等级和类别）、time（时间相关），属性以$开头的是从本地系统获得的变量、不带$是从消息中获得的变量。

属性替代的语法格式：

%propname:fromChar:toChar:options:fieldname%

例：

%msg:2:$%　#选取msg变量中，起始位置为2，终止位置为结尾

%msg:F,32:3% #按照空格分隔，取第三个子串

2 模板template

模板的功能是定义输出格式，或者定义omfile模块的动态路径、动态文件。需用到上面提到的属性替换。

定义输出msg的模板：

$template t_msg, “%msg\n%”

按日期保存输出，需使用动态路径，使用如下模板：

$template f_debug, “/path/logs/%$year%-%$month%-%$day%/debug.log”

rsyslog特性(二)

3 过滤规则

rsyslog可以使用syslog标准的过滤规则，此外还新增了扩展规则。如可以执行输出模板，方法是在规则后面指定template名。

过滤规则与模板组合使用：

```
$template tmp_message, “%msg\n%”
*.info;mail.none;authpriv.none;cron.none        /var/log/messages;tmp_message
```

除了syslog标准规则，rsyslog作者还开发了一个叫做rainerscript的脚本语言，来定义更复杂的过滤规则。如startwith、contains、%（取余）等。如：

```
if ($msg contains 'Time has been changed') then {
:omfile:$sysmonitor;sysmonitor_tmp
stop
}
```

rsyslog模块(一)

rsyslog的消息流从输入模块->预处理模块->主消息队列->过滤模块->动作队列->输出模块。
输入模块主要有imjournal、imuxsock、imudp、imtcp等；
过滤模块主要有jsonparse、normalize等；
输出模块主要有omfile、omfwd（默认会配置，发送到UDP及TCP端口）等。

1 输入模块

1.1 imjournal

systemd journal input module。将systemd journal中的结构化的日志消息输入到rsyslog。使用imjournal模块的性能可能明显低于使用imuxsock模块的性能。Journal给imuxsock模块提供了经典的syslog消息的副本，但是不提供结构化的消息数据。只有在需要结构化数据时，才必须使用imjournal。否则，imjournal可能会被imuxsock替代。

模块参数：

1）状态文件

$imjournalStateFile /run/log/journal/imjournal.state

#imjournal状态文件，imjournal获取journal日志时记录position到状态文件imjournal.state中。如指定了绝对路径，则使用绝对路径名；否则将在工作目录中创建给定的名称的状态文件。

2）限速

$imjournalRatelimitInterval 0

设为0时，关闭限速；

1.2 imudp

提供通过udp接收系统日志消息的能力。

rsyslog模块(二)

2 输出模块

输出模块介绍以omfile为例。Omfile模块提供了将消息写入文件的功能，此模块支持写入以静态名称命名的文件以及基于消息内容命名的文件。

模块参数：

1） 模板

```
$ActionFileDefaultTemplate  RSYSLOG_TraditionalFileFormat
#如果一个action没有配置特定模板的话，则使用默认模板
```

2） 目录创建权限

```
$DirCreateMode  0700
```

3） 文件创建权限

```
$FileCreateMode  0644
action(type="omfile" dirCreateMode="0700" FileCreateMode="0644"
      File="/var/log/messages")
```

rsyslog队列(一)

日志经过rsyslog处理的过程中会经过两个队列，一个是主消息队列（main message queue），另一个是动作队列（action queue）。使用队列的作用，一是加速，二是可靠。

rsyslog作为日志转发工具，主动读取来自journal、文件、本地socket、TCP/UDP端口等的日志作为输入，经过处理后再将日志转发到本地文件、远端服务器、数据库等等目的地。
过滤器会读取rsyslog.conf配置文件中设置的规则，与日志中的内容进行对比，然后发送到对应的action queue，一旦日志进入到action queue之后，会从主消息队列中删除。日志进入action queue以后，动作处理器会从action queue中获取最先进入队列的日志进行处理，根据规则进行日志的输出，如写入文件、录入数据库、发送到远程服务器等。

补充：日志消息被读入rsyslog后，都要先经过Main Message Queue进行缓存，随后过滤器根据过滤规则将消息分发到特定规则对应的Action Queue（每一个规则都对应有一个Action Queue）中，随后Action Processor处理后再将消息输出到目的地。若消息无法送达目的地，且Action Queue不设置主动丢弃日志时，消息会残留在Action Queue中，造成Action Queue堵塞。进而Main Message Queue中对应的日志无法再输出到该Action Queue，造成Main Message Queue堵塞，并且因此影响到其他Action，导致日志转发异常。这是由rsyslog本身的工作原理决定的。

rsyslog队列(二)

1主消息队列

任何消息都要先进入这个队列，直到进入到动作队列之后消息才会从这个队列中删除。

2 动作队列

消息经过主消息队列之后，就被rule processor解析和处理，根据预先配置的规则压入各自的动作队列，动作队列之后消息最终被消费掉，输出到你指定的地方。

如果消息最终不能被消费（输出到指定位置），那么这些消息就会停留在之前的队列中。这就有可能会导致队列被填满，一旦队列填满，后续的输入消息就不能再进入消息队列，最终造成某些服务无法进行日志记录，最坏的结果是导致该服务无法正常提供服务。

3 rsyslog队列的种类

Direct queue、Disk queue、In-memory queue（LinkedList/FixedArray）、Disk-Assisted In-memory queue。

设置队列的语法：

$<Object>QueueType <QueueTypeValue> #Object可以代表MainMsg或Action。

1） direct queue

direct queue是默认的行为，它不是一种队列。通常输出到本地硬盘的时候都是使用这种类型。Direct queue是唯一一个会把执行结果（成功/失败）从消费者（action queue）返回给生产者的队列。action processor正是通过这个返回值提醒action queue，让action queue取回这些处理失败的消息，如此循环知道消息处理成功。

2） In-memory queue

rsyslog队列(三)

队列的设置

1）限制队列的总容量

$<Object>QueueSize <number> 用于设置队列的总容量，即队列可容纳的消息数量。

主队列容量：$MainMsgQueueSize 99999

动作队列容量：$ActionQueueSize 99999

2）丢弃消息

控制这个行为的指令是$<Object>QueueDiscardMark，当队列中的消息达到这个指定的值时，消息就会被丢弃。至于丢弃哪一种消息，则由$<Object>QueueDiscardSeverity指令控制，这个指令接受以文字表示的等级或以数字表示的等级。

3） 队列的终止

用户不能控制队列的终止，只有在系统被关闭的那一刻，队列才会结束。当队列终止的时候，可能会遇到：队列中仍有数据尝试进入。这种情况下rsyslog会试图处理这些数据，如果希望控制这些数据的处理时间，可以使用这个指令：

$<Object>QueueTimeoutShutdown <milliseconds>

当时间超过这个值，队列中的所有数据将被丢弃。

另一种情况是：当超时后，依然希望队列处理完当前正在被处理的数据再关闭，可以使用$<Object>QueueTimeoutActionCompletion指令，它设置了处理当前数据的时间，即除了当前正在被处理的消息外，其他任何的消息都被丢弃。

rsyslog服务安装

Rsyslog服务安装：

步骤 1

默认情况下，Rsyslog守护程序已经安装并在系统中运行。为了验证系统中是否存在rsyslog服务，请执行以下命令：

rpm -q rsyslog

rsyslogd -v

步骤 2

如果未安装rsyslog软件包，在已有yum源的情况下，执行如下命令安装该服务

yum install rsyslog

如果没有配置yum源，则需要使用如下命令安装

#rpm -ivh rsyslog***.rpm

----结束

Rsyslog服务主要组成：

主程序：　　/usr/sbin/rsyslogd

主配置文件：/etc/rsyslog.conf

服务脚本：　/usr/lib/systemd/system/rsyslog.service

rsyslog服务端配置（一）

步骤 1　编辑/etc/rsyslog.conf文件

步骤 2　添加rsyslog服务器接收客户端日志的配置，支持TCP和UDP协议，二选一，通常port设置成514。但如果需要使用TLS加密传输，则必须选择TCP协议。

如选择TCP协议配置，将如下2行内容打开：

```
# Provides TCP syslog reception
#提供TCP的514端口来接收UDP协议发送过来的数据
module(load="imtcp")
input(type="imtcp" port="514")
```

如选择UDP协议配置如下：

```
# Provides UDP syslog reception
#提供UDP的514端口来接收UDP协议发送过来的数据
module(load="imudp")
input(type="imudp" port="514“
```

步骤 3　编辑/etc/rsyslog.conf文件，定义客户端传送过来的日志保存路径。

local5. /home/client_local5.log*

rsyslog通过Facility概念定义日志消息的来源，其中local0~local7为本地使用预留的Facility，客户端的业务日志需要定义到local0~local7中。

此处表示服务端接收到客户端定义的*local5*所有日志保存到 */home/client_local5.log*中。

rsyslog服务端配置（二）

步骤 4　重启rsyslog服务

systemctl restart rsyslog.service

步骤 5　检查rsyslog服务器端是否正常监听

若使用的是UDP协议，执行如下命令：若能查看到配置中的服务器信息，则说明syslog服务器端配置正常。

netstat -lnutp

Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address　Foreign Address　State　PID/Program name

udp　0　0 10.90.184.186:514　0.0.0.0:　42047/rsyslogd*

若使用的是TCP协议，执行如下命令：

netstat -lntp

Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address　Foreign Address　State　PID/Program name

tcp　0　0 0.0.0.0:22　0.0.0.0:*　LISTEN　13784/sshd

tcp　0　0 10.90.183.186:514　0.0.0.0:　LISTEN　51914/syslog-ng*

tcp　0　0 :::22　:::*　LISTEN　13784/sshd

如果能查看到配置中的服务器信息，则说明rsyslog服务器端配置正常。

----结束

rsyslog客户端配置（一）

基本概念：rsyslog通过Facility概念定义日志消息的来源，以方便对日志进行分类，常用的facility有以下几种：

Facility	详细
kern	内核消息
user	用户级消息
mail	邮件系统消息
daemon	系统服务消息
auth	认证系统消息
syslog	日志系统自身消息
lpr	打印系统消息
authpriv	权限系统消息
cron	定时任务消息
news	新闻系统消息
uucp	uucp系统消息
ftp	ftp服务消息
local0~local7	本地使用预留的Facility

通过定义不同的日志等级，可以方便做到日志过滤，做到日志按需记录

序号	日志级别	描述
0	EMEGR(紧急) emergencies	会导致主机系统不可用的情况
1	ALERT(警告)	必须马上采取措施解决的问题
2	CRIT(严重) critical	比较严重的情况
3	ERR(错误) error	运行出现错误
4	WARNING(提醒)	可能会影响系统功能的事件
5	NOTICE(注意)	不会影响系统但是值得注意
6	INFO(信息)	一般信息
7	DEBUG(调试)	程序或系统调试信息
8	none	没有优先级，不记录任何日志消息

Rsyslog对于Facility的处理规则配置在配置文件/etc/rsyslog.conf中，示例：

```
# 记录所有日志类型的info级别以及大于info级别的信息到/var/log/messages，mail邮件信息，authpriv验证方面的信息和cron时间任务相关的信息除外
*.info;mail.none;authpriv.none;cron.none                /var/log/messages
# 邮件的所有信息存放在/var/log/maillog; 这里有一个“-”符号, 表示是使用异步的方式记录, 因为日志一般会比较大
mail.
*                                                       -/var/log/maillog
```

rsyslog客户端配置（二）

步骤 1　编辑/etc/rsyslog.conf文件。

步骤 2　将客户端的所有日志发送给远端服务器的的Rsyslog服务接收端口，通常是514（注意，包含local0~local7的日志）。

```
# 将本地服务器的所有日志以tcp协议发送至远端服务器8.4.186.48的514端口，远端服务器在接收到日志后，根据它对不同的Facility定义的规则保存到不同的日志中。
*.* action(type="omfwd"
target="8.4.186.48" port="514" protocol="tcp"
action.resumeRetryCount="100"
queue.type="linkedList" queue.size="10000")
```

步骤 3　重启rsyslog服务。

```
# systemctl restart rsyslog.service
```

步骤 4　检查message日志是否写入服务器。

```
# logger “hello world~”
```

步骤 5　登录远端服务器，查看/var/log/messages日志是否存在这条记录。

为什么是到远端服务器的/var/log/messages查看“hello world~”这行日志呢？因为远端服务器/etc/rsyslog.conf对于*.info日志定义的规则是保存到/var/log/messages中。

----结束

logrotate日志转储服务介绍

系统使用过程中，日志持续增长会使存储日志的磁盘空间被占满，从而导致日志无法继续打印，或者影响到系统部分业务功能（进程异常或一些操作无法进行）；另外如果系统出现了问题，因为缺少日志也会影响问题定位效率，甚至导致问题无法定位。

用linux中一般使用logrotate服务把旧文件删除或压缩备份，并创建新的日志文件，达到日志转储的目的。

logrotate日志转储服务配置方法

文件配置方法

在/etc/logrotate.d目录下，添加配置文件（新增配置文件的权限建议不大于640），一个配置文件中可同时配置多个需要切割的日志。

如：/etc/logrotate.d/example文件中配置。

```
/var/log/logexample
/var/log/logexample1
{
maxage 365
rotate 30
notifempty
compress
copytruncate
missingok
size +4096k
}
```

该配置会对/var/log/logexample、/var/log/logexample1进行切割。

- maxage 365：当前系统时间下，只存储最近365天的切割出来的日志文件，超过365天则删除。
- rotate 30：指定日志文件删除之前切割的次数，此处保留30个备份。
- notifempty：表示日志为空则不处理。
- compress：通过gzip压缩转储以后的日志。
- copytruncate：用于还在打开中的日志文件，把当前日志备份并截断。
- missingok：如果日志文件丢失，不报错继续执行下一个。
- size +4096k：表示日志超过4096k大小才分割，size默认单位是KB，可使用k、M和G来指定KB、MB和GB。

logrotate日志转储服务配置说明

配置项	功能
compress	通过gzip压缩转储以后的日志。
missingok	找不到日志时，跳过。
nomissingok	找不到日志时，报错。
nocompress	不需要压缩时，用这个参数。
copytruncate	用于还在打开中的日志文件，把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	转储文件，使用指定的文件模式创建新的日志文件。
nocreate	不建立新的日志文件。
prerotate/endscript	在转储以前需要执行的命令可以放入这个对，这两个关键字必须单独成行。
postrotate/endscript	在转储以后需要执行的命令可以放入这个对，这两个关键字必须单独成行。
daily	指定转储周期为每天。
weekly	指定转储周期为每周。
monthly	指定转储周期为每月。
rotate count	指定日志文件删除之前转储的次数，0默认不保留备份，5指保留5个备份。
size	当日志文件到达指定的大小时才转储，size可以指定bytes（缺省）以及KB、MB或者GB。

说明

1. nocreate与配置文件中的copytruncate是互斥的，不能同时配置，否则nocreate不生效。
2. create mode owner group（例如：create 0600 root root）与配置文件中的copytruncate是互斥的，否则create配置不生效。
3. 时间频度（daily，weekly，monthly，yearly）和日志大小（size）这两项参数同时配置的时候，以最后的配置项为日志切分条件。

随堂测

- 1. dmesg命令可以查看系统开机过程中产生的信息？（判断题）

- 2. openEuler操作系统中，常用于日志转储的服务是？ （单选题）

  A. rsyslog

  B. logrotate

  C. audit

  D. cron

openEuler

随堂测

- 3. rsyslog服务器接收客户端日志的配置，通常port设置成？（单选题）
  - A. 22
  - B. 67
  - C. 123
  - D. 514

- 4. rsyslog服务器接收客户端日志的配置，可以支持（）协议。（多选题）
  - A. TCP
  - B. UDP
  - C. ICMP
  - D. SMTP

Thank you

openEuler