注意

有些应用不使用 syslog 管理它们的日志消息，尽管它们通常会将其日志文件放在 /var/log 的某一子目录中。例如，Apache Web Server 将日志消息保存到 **/var/log** 目录的某一子目录下的文件中。

参考文献

`systemd-journald.service`(8)、`rsyslogd`(8) 和 `rsyslog.conf`(5) man page

如需更多信息，请参阅《红帽企业 Linux 8.0 基本系统设置配置指南》中的使用日志文件对问题进行故障排除章节，网址为：
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

小测验

描述系统日志架构

选择以下问题的正确答案：

1. 哪个日志文件存储大多数 syslog 消息，除了与身份验证、邮件、调度作业和调试相关的日志消息以外？

a. `/var/log/maillog`
b. `/var/log/boot.log`
c. `/var/log/messages`
d. `/var/log/secure`

2. 哪个日志文件存储与系统中安全性和身份验证操作相关的 syslog 消息？

a. `/var/log/maillog`
b. `/var/log/boot.log`
c. `/var/log/messages`
d. `/var/log/secure`

3. 哪个服务将 syslog 消息排序并组织到 `/var/log` 的文件中？

a. `rsyslog`
b. `systemd-journald`
c. `auditd`
d. `tuned`

4. 哪个目录存放人类可读的 syslog 文件？

a. `/sys/kernel/debug`
b. `/var/log/journal`
c. `/run/log/journal`
d. `/var/log`

5. 哪个文件存储与邮件服务器相关的 syslog 消息？

a. `/var/log/lastlog`
b. `/var/log/maillog`
c. `/var/log/tallylog`
d. `/var/log/boot.log`

海量视频题库 myitpub.com QQ:556462
www.52myit.com

▶ **6. 哪个文件存储与调度作业相关的 syslog 消息?**

a. `/var/log/cron`

b. `/var/log/tallylog`

c. `/var/log/spooler`

d. `/var/log/secure`

▶ **7. 哪个文件存储与系统启动相关的控制台消息?**

a. `/var/log/messages`

b. `/var/log/cron`

c. `/var/log/boot.log`

d. `/var/log/secure`

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

解决方案

描述系统日志架构

选择以下问题的正确答案：

1. 哪个日志文件存储大多数 syslog 消息，除了与身份验证、邮件、调度作业和调试相关的日志消息以外？

a. /var/log/maillog
b. /var/log/boot.log
c. **/var/log/messages**
d. /var/log/secure

2. 哪个日志文件存储与系统中安全性和身份验证操作相关的 syslog 消息？

a. /var/log/maillog
b. /var/log/boot.log
c. /var/log/messages
d. **/var/log/secure**

3. 哪个服务将 syslog 消息排序并组织到 /var/log 的文件中？

a. rsyslog
b. systemd-journald
c. auditd
d. tuned

4. 哪个目录存放人类可读的 syslog 文件？

a. /sys/kernel/debug
b. /var/log/journal
c. /run/log/journal
d. **/var/log**

5. 哪个文件存储与邮件服务器相关的 syslog 消息？

a. /var/log/lastlog
b. **/var/log/maillog**
c. /var/log/tallylog
d. /var/log/boot.log

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com

▶ **6. 哪个文件存储与调度作业相关的 syslog 消息?**

a. **/var/log/cron**

b. /var/log/tallylog

c. /var/log/spooler

d. /var/log/secure

▶ **7. 哪个文件存储与系统启动相关的控制台消息?**

a. /var/log/messages

b. /var/log/cron

c. **/var/log/boot.log**

d. /var/log/secure

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

查看系统日志文件

培训目标

学完本节后，您应能够解读相关 syslog 日志文件中的事件，以对问题进行故障排除或查看系统状态。

将事件记录到系统

许多程序使用 `syslog` 协议将事件记录到系统。每一日志消息根据设备（消息的类型）和优先级（消息的严重性）分类。`rsyslog.conf`(5) man page 中对可用的功能进行了阐述。

下表从高到低列出了八个标准 syslog 优先级。

Syslog 优先级概述

代码	优先级	严重性
0	emerg	系统不可用
1	alert	必须立即采取措施
2	crit	临界情况
3	err	非严重错误状况
4	warning	警告情况
5	notice	正常但重要的事件
6	info	信息性事件
7	debug	调试级别消息

`rsyslog` 服务使用日志消息的设备和优先级来确定如何进行处理。其配置规则位于 **`/etc/rsyslog.conf`** 文件和 **`/etc/rsyslog.d`** 目录中扩展名为 **`.conf`** 的任何文件。通过在 **`/etc/rsyslog.d`** 目录中安装适当的文件，软件包可以轻松地添加规则。

每个控制着 syslog 消息排序方式的规则都对应了其中一个配置文件中的一行。每行左侧表示与规则匹配的 syslog 消息的设备和严重性。每行右侧表示要将日志消息保存到的文件（或消息所要发送到的其他位置）。星号 (**`*`**) 是一个匹配所有值的通配符。

例如，以下行会将发送给 `authpriv` 设备的任何优先级的消息记录到文件 **`/var/log/secure`** 中：

```
authpriv.*                  /var/log/secure
```

日志消息有时会匹配 **`rsyslog.conf`** 中的多条规则。在这种情况下，一条消息会存储到多个日志文件中。为限制存储的消息，优先级字段中的关键字 **none** 指出不应将指定设备的消息存储在给定的文件中。

海量视频题库 itpub.com QQ:556462
www.52nit.com

除了将 syslog 消息记录到文件中外，也可将它们显示到所有已登录用户的终端。**rsyslog.conf** 文件中有一项设置，可将优先级为 **emerg** 的所有 syslog 消息显示到所有已登录用户的终端。

RSYSLOG 规则示例

```
#### RULES ####

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

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none                /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                              /var/log/secure

# Log all the mail messages in one place.
mail.*                                                  -/var/log/maillog


# Log cron stuff
cron.*                                                  /var/log/cron

# Everybody gets emergency messages
*.emerg                                                 :omusrmsg:*

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                          /var/log/spooler

# Save boot messages also to boot.log
local7.*                                                /var/log/boot.log
```

注意

syslog 子系统还有许多功能，但已超出本课程的讨论范畴。如果想要进一步探索，可以参考 rsyslog-doc 软件包中所含的 **rsyslog.conf**(5) man page 及 **/usr/share/doc/rsyslog/html/index.html** 详细 HTML 文档，它们可从红帽企业 Linux 8 的 AppStream 存储库中获得。

日志文件轮转

logrotate 工具会轮转日志文件，以防止它们在含有 **/var/log** 目录的文件系统中占用太多空间。轮转日志文件时，使用指示轮转日期的扩展名对其重命名。例如，如果在 2019-01-30 轮转，旧的 **/var/log/messages** 文件可能会变成 **/var/log/messages-20190130**。轮转原日志文件之后，会创建新日志文件，并通知对它执行写操作的服务。

轮转若干次之后（通常在四周之后），丢弃最旧的日志文件以释放磁盘空间。调度作业每日运行一次 **logrotate** 程序，以查看是否有任何日志需要轮转。大多数日志文件每周轮转一次，但是 **logrotate** 轮转文件的速度有时较快，有时较慢，或在文件达到特定大小时进行轮转。

本课程不阐述 **logrotate** 的配置。如需更多信息，请参阅 **logrotate**(8) man page。

海量视频题库 mypub.com QQ:5565462 www.52nwt.com

分析 SYSLOG 条目

日志消息在文件的开头显示最旧的消息，在文件的末尾显示最新的消息。rsyslog 服务在日志文件中记录条目时采用一种标准的格式。下例将深入介绍 **/var/log/secure** 日志文件中的日志消息：

```
❶Feb 11 20:11:48 ❷localhost ❸sshd[1433]: ❹Failed password for student from
 172.25.0.10 port 59344 ssh2
```

❶ 记录该日志条目的时间戳
❷ 发送该日志消息的主机
❸ 发送该日志消息的程序或进程名称和 PID 编号
❹ 发送的实际消息

监控日志

监控一个或多个日志文件中的事件有助于重现问题。**tail -f /path/to/file** 命令输出指定文件的最后 10 行，并在新行写入到文件中时继续输出它们。

例如，要监控失败的登录尝试，在一个终端中运行 **tail** 命令，然后在用户尝试登录系统时在另一个终端中以 root 用户身份运行 **ssh** 命令。

在第一个终端中，运行以下 **tail** 命令：

```
[root@host ~]# tail -f /var/log/secure
```

在第二个终端中，运行以下 **ssh** 命令：

```
[root@host ~]# ssh root@localhost
root@localhost's password: redhat
...output omitted...
[root@host ~]#
```

返回第一个终端并查看日志。

```
...output omitted...
Feb 10 09:01:13 host sshd[2712]: Accepted password for root from 172.25.254.254
 port 56801 ssh2
Feb 10 09:01:13 host sshd[2712]: pam_unix(sshd:session): session opened for user
 root by (uid=0)
```

手动发送 SYSLOG 消息

logger 命令可以发送消息到 **rsyslog** 服务。默认情况下，它将优先级为 **notice** (**user.notice**) 的消息发送给 **user** 设备，除非通过 **-p** 选项另有指定。这对测试 rsyslog 服务配置更改很有用。

若要向 **rsyslog** 服务发送消息并记录在 **/var/log/boot.log** 日志文件中，可以执行以下 **logger** 命令：

```
[root@host ~]# logger -p local7.notice "Log entry created on host"
```

海量视频题库 myitpub.com QQ:556462
www.52myit.com

参考文献

`logger`(1)、`tail`(1)、`rsyslog.conf`(5) 和 `logrotate`(8) man page

`rsyslog` 手册

· **`/usr/share/doc/rsyslog/html/index.html`**，由 rsyslog-doc 软件包提供

如需更多信息，请参阅《红帽企业 Linux 8.0 基本系统设置配置指南》中的使用日志文件对问题进行故障排除章节，网址为：
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

指导练习

查看系统日志文件

在本练习中，您将重新配置 rsyslog，以便将特定的日志消息写入到新的文件中。

成果

您应能够配置 rsyslog 服务，将优先级为 **debug** 的所有日志消息写入到 **/var/log/messages-debug** 日志文件。

在你开始之前

以 student 用户身份并使用 student 作为密码登录 workstation。

在 workstation 上，运行 **lab log-configure start** 脚本来开始本练习。此脚本会确保环境设置正确无误。

```
[student@workstation ~]$ lab log-configure start
```

1. 从 workstation，以 student 用户身份打开连接 servera 的 SSH 会话。

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

2. 在 servera 上，通过添加 rsyslog 配置文件 **/etc/rsyslog.d/debug.conf** 来配置 rsyslog，从而将任何服务的优先级为 **debug** 或以上的所有消息记录到新的 **/var/log/messages-debug** 日志文件中。

2.1. 使用 **sudo -i** 命令，切换为 root 用户。如果在运行 **sudo -i** 命令时要求提供密码，可指定 student 作为 student 用户的密码。

```
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```

2.2. 创建 **/etc/rsyslog.d/debug.conf** 文件并使其包含必要的条目，将优先级为 **debug** 的所有日志消息重定向到 **/var/log/messages-debug**。您可以使用 **vim /etc/rsyslog.d/debug.conf** 命令来创建包含以下内容的文件。

```
*.debug /var/log/messages-debug
```

此配置行会捕获具有任何设备并且优先级为 **debug** 及以上的 syslog 消息。rsyslog 服务将这些 syslog 消息写入到 **/var/log/messages-debug** 文件中。配置行的 **facility** 字段或 **priority** 字段中的通配符 (*****) 表示任何设备或优先级。

2.3. 重新启动 rsyslog 服务。

海量视频题库 QQ:556462 www.52mvpt.com

```
[root@servera ~]# systemctl restart rsyslog
```

3. 验证优先级为 **debug** 的所有日志消息是否都出现在 **/var/log/messages-debug** 文件中。

   3.1. 使用 **logger** 命令及 **-p** 选项，以生成设备为 **user** 且优先级为 **debug** 的日志消息。

```
[root@servera ~]# logger -p user.debug "Debug Message Test"
```

   3.2. 使用 **tail** 命令来查看 **/var/log/messages-debug** 文件的最后十条日志消息，并确认您看到了 **Debug Message Test** 消息及其他日志消息。

```
[root@servera ~]# tail /var/log/messages-debug
Feb 13 18:22:38 servera systemd[1]: Stopping System Logging Service...
Feb 13 18:22:38 servera rsyslogd[25176]: [origin software="rsyslogd"
 swVersion="8.37.0-9.el8" x-pid="25176" x-info="http://www.rsyslog.com"] exiting
 on signal 15.
Feb 13 18:22:38 servera systemd[1]: Stopped System Logging Service.
Feb 13 18:22:38 servera systemd[1]: Starting System Logging Service...
Feb 13 18:22:38 servera rsyslogd[25410]: environment variable TZ is not set, auto
 correcting this to TZ=/etc/localtime  [v8.37.0-9.el8 try http://www.rsyslog.com/
e/2442 ]
Feb 13 18:22:38 servera systemd[1]: Started System Logging Service.
Feb 13 18:22:38 servera rsyslogd[25410]: [origin software="rsyslogd"
 swVersion="8.37.0-9.el8" x-pid="25410" x-info="http://www.rsyslog.com"] start
Feb 13 18:27:58 servera student[25416]: Debug Message Test
```

   3.3. 退出 `servera` 上的 `root` 和 `student` 用户的 shell，以返回到 `workstation` 上 `student` 用户的 shell。

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

完成

在 `workstation` 上，运行 **lab log-configure finish** 来完成本练习。此脚本会确保环境重新恢复到清理干净的状态。

```
[student@workstation ~]$ lab log-configure finish
```

本引导式练习到此结束。

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

查看系统日志条目

培训目标

学完本节后，您应能够查找和解读系统日志中的条目，以对问题进行故障排除或查看系统状态。

查找事件

`systemd-journald` 服务将日志数据存储在带有索引的结构化二进制文件中，该文件称为日志。此数据包含与日志事件相关的额外信息。例如，对于 syslog 事件，这可包含原始消息的设备和优先级。

> **重要**
> 在红帽企业 Linux 8 中，默认情况下 **/run/log** 目录用于存储系统日志。**/run/log** 的内容在系统重启后将被清除。您可以更改此设置，具体操作方式将在本章的后续部分讨论。

要从日志中检索日志消息，请使用 **journalctl** 命令。您可以使用此命令来查看日志中的所有消息，或根据各种选项和标准来搜索特定事件。如果以 `root` 身份运行该命令，则对日志具有完全访问权限。普通用户也可以使用此命令，但可能会被限制查看某些消息。

```
[root@host ~]# journalctl
...output omitted...
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Stopped target Sockets.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Closed D-Bus User Message Bus
 Socket.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Closed Multimedia System.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Reached target Shutdown.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Starting Exit the Session...
Feb 21 17:46:25 host.lab.example.com systemd[24268]: pam_unix(systemd-
user:session): session c>
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopped User Manager for UID
 1001.
Feb 21 17:46:25 host.lab.example.com systemd[1]: user-runtime-dir@1001.service:
 Unit not neede>
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopping /run/user/1001 mount
 wrapper...
Feb 21 17:46:25 host.lab.example.com systemd[1]: Removed slice User Slice of UID
 1001.
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopped /run/user/1001 mount
 wrapper.
Feb 21 17:46:36 host.lab.example.com sshd[24434]: Accepted publickey for root from
 172.25.250.>
Feb 21 17:46:37 host.lab.example.com systemd[1]: Started Session 20 of user root.
Feb 21 17:46:37 host.lab.example.com systemd-logind[708]: New session 20 of user
 root.
Feb 21 17:46:37 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
 opened for u>
```

海量视频题库 myicpub.com QQ:556462
www.52myit.com

```
Feb 21 18:01:01 host.lab.example.com CROND[24468]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 21 18:01:01 host.lab.example.com run-parts[24471]: (/etc/cron.hourly) starting
 0anacron
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
 0anacron
lines 1464-1487/1487 (END) q
```

journalctl 命令突出显示重要的日志消息：优先级为 **notice** 或 **warning** 的消息显示为粗体文本，而优先级为 **error** 或以上的消息则显示为红色文本。

成功利用日志进行故障排除和审核的关键在于，将日志搜索限制为仅显示相关的输出。

默认情况下，**journalctl -n** 显示最后 10 个日志条目。您可以借助可选参数对此进行调整，它可以指定要显示的日志条目数。如果是最后五个日志条目，可运行以下 **journalctl** 命令：

```
[root@host ~]# journalctl -n 5
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:01:01 +07.
 --
...output omitted...
Feb 21 17:46:37 host.lab.example.com systemd-logind[708]: New session 20 of user
 root.
Feb 21 17:46:37 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
 opened for u>
Feb 21 18:01:01 host.lab.example.com CROND[24468]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 21 18:01:01 host.lab.example.com run-parts[24471]: (/etc/cron.hourly) starting
 0anacron
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
 0anacron
lines 1-6/6 (END) q
```

与 **tail -f** 命令相似，**journalctl -f** 命令输出系统日志的最后 10 行，并在新日志条目写入到日志中时继续输出它们。要退出 **journalctl -f** 进程，可使用 **Ctrl+C** 组合键。

```
[root@host ~]# journalctl -f
-- Logs begin at Wed 2019-02-20 16:01:17 +07. --
...output omitted...
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
 0anacron
Feb 21 18:22:42 host.lab.example.com sshd[24437]: Received disconnect from
 172.25.250.250 port 48710:11: disconnected by user
Feb 21 18:22:42 host.lab.example.com sshd[24437]: Disconnected from user root
 172.25.250.250 port 48710
Feb 21 18:22:42 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
 closed for user root
Feb 21 18:22:42 host.lab.example.com systemd-logind[708]: Session 20 logged out.
 Waiting for processes to exit.
Feb 21 18:22:42 host.lab.example.com systemd-logind[708]: Removed session 20.
Feb 21 18:22:43 host.lab.example.com sshd[24499]: Accepted
 publickey for root from 172.25.250.250 port 48714 ssh2: RSA
 SHA256:1UGybTe52L2jzEJa1HLVKn9QUCKrTv3ZzxnMJol1Fro
```

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

```
Feb 21 18:22:44 host.lab.example.com systemd-logind[708]: New session 21 of user
 root.
Feb 21 18:22:44 host.lab.example.com systemd[1]: Started Session 21 of user root.
Feb 21 18:22:44 host.lab.example.com sshd[24499]: pam_unix(sshd:session): session
 opened for user root by (uid=0)
^C
[root@host ~]#
```

为了帮助对问题进行故障排除，可能需要根据日志条目的优先级过滤日志输出。**journalctl -p** 可以接受优先级的名称或编号作为参数，并显示该优先级别及以上的日志条目。**journalctl** 命令理解 **debug**、**info**、**notice**、**warning**、**err**、**crit**、**alert** 和 **emerg** 优先级。

运行以下 **journalctl** 命令，可列出优先级为 **err** 或以上的日志条目：

```
[root@host ~]# journalctl -p err
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:01:01 +07.
 --
..output omitted...
Feb 20 16:01:17 host.lab.example.com kernel: Detected CPU family 6 model 13
 stepping 3
Feb 20 16:01:17 host.lab.example.com kernel: Warning: Intel Processor - this
 hardware has not undergone testing by Red Hat and might not be certif>
Feb 20 16:01:20 host.lab.example.com smartd[669]: DEVICESCAN failed: glob(3)
 aborted matching pattern /dev/discs/disc*
Feb 20 16:01:20 host.lab.example.com smartd[669]: In the system's table of devices
 NO devices found to scan
lines 1-5/5 (END) q
```

查找具体的事件时，可以将输出限制为特定的时间段。**journalctl** 命令有两个选项，分别是 **--since** 和 **--until** 选项，它们可以将输出限制为特定的时间范围。这两个选项都采用格式为 "YYYY-MM-DDhh:mm:ss" 的时间参数（必须使用双引号，以保留选项中的空格）。如果省略日期，则命令会假定日期为当天；如果省略时间，则命令假定为自 00:00:00 起的一整天。除了日期和时间字段外，这两个选项还接受 **yesterday**、**today** 和 **tomorrow** 作为有效的参数。

运行以下 **journalctl** 命令，可列出今天记录中的所有日志条目。

```
[root@host ~]# journalctl --since today
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:31:14 +07.
 --
...output omitted...
Feb 21 18:22:44 host.lab.example.com systemd-logind[708]: New session 21 of user
 root.
Feb 21 18:22:44 host.lab.example.com systemd[1]: Started Session 21 of user root.
Feb 21 18:22:44 host.lab.example.com sshd[24499]: pam_unix(sshd:session): session
 opened for user root by (uid=0)
Feb 21 18:31:13 host.lab.example.com systemd[1]: Starting dnf makecache...
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Red Hat Enterprise Linux 8.0
 AppStream (dvd)    637 kB/s | 2.8 kB     00:00
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Red Hat Enterprise Linux 8.0
 BaseOS (dvd)       795 kB/s | 2.7 kB     00:00
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Metadata cache created.
Feb 21 18:31:14 host.lab.example.com systemd[1]: Started dnf makecache.
lines 533-569/569 (END) q
```

海量视频题库 myitpub.com QQ:556462
www.52myit.com

运行以下 **journalctl** 命令，可列出范围从 **2019-02-10 20:30:00** 到 **2019-02-13 12:00:00** 的所有日志条目。

```
[root@host ~]# journalctl --since "2019-02-10 20:30:00" \
--until "2019-02-13 12:00:00"
...output omitted...
```

您也可以指定相对于当前的某个时间以后的所有条目。例如，要指定上一小时的所有条目，可以使用以下命令：

```
[root@host ~]# journalctl --since "-1 hour"
...output omitted...
```

注意

您可以通过 **--since** 和 **--until** 选项指定其他更复杂的时间要求。部分示例可参阅 systemd.time(7) man page。

除了日志的可见内容外，日志条目中还附带了只有在打开详细输出时才可看到的字段。任何显示的额外字段都可用于过滤日志查询的输出。这可用于减少查找日志中特定事件的复杂搜索的输出。

```
[root@host ~]# journalctl -o verbose
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:31:14 +07.
 --
...output omitted...
Thu 2019-02-21 18:31:14.509128 +07...
    PRIORITY=6
    _BOOT_ID=4409bbf54680496d94e090de9e4a9e23
    _MACHINE_ID=73ab164e278e48be9bf80e80714a8cd5
    SYSLOG_FACILITY=3
    SYSLOG_IDENTIFIER=systemd
    _UID=0
    _GID=0
    CODE_FILE=../src/core/job.c
    CODE_LINE=826
    CODE_FUNC=job_log_status_message
    JOB_TYPE=start
    JOB_RESULT=done
    MESSAGE_ID=39f53479d3a045ac8e11786248231fbf
    _TRANSPORT=journal
    _PID=1
    _COMM=systemd
    _EXE=/usr/lib/systemd/systemd
    _CMDLINE=/usr/lib/systemd/systemd --switched-root --system --deserialize 16
    _CAP_EFFECTIVE=3ffffffffff
    _SELINUX_CONTEXT=system_u:system_r:init_t:s0
    _SYSTEMD_CGROUP=/init.scope
    _SYSTEMD_UNIT=init.scope
    _SYSTEMD_SLICE=-.slice
    UNIT=dnf-makecache.service
    MESSAGE=Started dnf makecache.
    _HOSTNAME=host.lab.example.com
```

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

```
    INVOCATION_ID=d6f90184663f4309835a3e8ab647cb0e
    _SOURCE_REALTIME_TIMESTAMP=1550748674509128
lines 32239-32275/32275 (END) q
```

以下列表提供了系统日志的常用字段，可用于搜索与特定进程或事件相关的行。

- _COMM 是命令的名称
- _EXE 是进程的可执行文件的路径
- _PID 是进程的 PID
- _UID 是运行该进程的用户的 UID
- _SYSTEMD_UNIT 是启动该进程的 systemd 单元

可以组合多个系统日志字段，以便在运行 **journalctl** 命令时形成精细的搜索查询。例如，以下 **journalctl** 命令将显示与 PID 为 1182 的 **sshd.service** systemd 进程单元相关的所有日志条目。

```
[root@host ~]# journalctl _SYSTEMD_UNIT=sshd.service _PID=1182
Apr 03 19:34:27 host.lab.example.com sshd[1182]: Accepted password for root
 from ::1 port 52778 ssh2
Apr 03 19:34:28 host.lab.example.com sshd[1182]: pam_unix(sshd:session): session
 opened for user root by (uid=0)
...output omitted...
```

注意

如需常用日志字段的列表，请参见 `systemd.journal-fields`(7) man page。

参考文献

`journalctl`(1)、`systemd.journal-fields`(7) 和 `systemd.time`(7) man page

如需更多信息，请参阅《红帽企业 Linux 8.0 基本系统设置配置指南》中的使用日志文件对问题进行故障排除章节，网址为：
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com

指导练习

查看系统日志条目

在本练习中，您将在系统日志中搜索记录符合特定条件的事件的条目。

成果

您应能够在系统日志中搜索基于不同条件记录事件的条目。

在你开始之前

以 `student` 用户身份并使用 `student` 作为密码登录 `workstation`。

在 `workstation` 上，运行 **`lab log-query start`** 脚本来开始本练习。此脚本会确保环境设置正确无误。

```
[student@workstation ~]$ lab log-query start
```

1. 从 `workstation`，以 `student` 用户身份打开连接 `servera` 的 SSH 会话。

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

2. 将 **`_PID=1`** 匹配项与 **`journalctl`** 命令搭配，以仅显示源自 `servera` 上使用进程标识符 1 运行的 `systemd` 进程的日志事件。要退出 **`journalctl`**，请按 **`q`** 键。

```
[student@servera ~]$ journalctl _PID=1
...output omitted...
Feb 13 13:21:08 localhost systemd[1]: Found device /dev/disk/by-uuid/
cdf61ded-534c-4bd6-b458-cab18b1a72ea.
Feb 13 13:21:08 localhost systemd[1]: Started dracut initqueue hook.
Feb 13 13:21:08 localhost systemd[1]: Found device /dev/disk/by-
uuid/44330f15-2f9d-4745-ae2e-20844f22762d.
Feb 13 13:21:08 localhost systemd[1]: Reached target Initrd Root Device.
lines 1-5/5 (END) q
[student@servera ~]$
```

> **注意**
> **`journalctl`** 命令可能会在您的系统上生成不同的输出。

3. 将 **`_UID=81`** 匹配项与 **`journalctl`** 命令搭配，以显示源自 `servera` 上通过用户标识符 81 启动的系统服务的所有日志事件。要退出 **`journalctl`**，请按 **`q`** 键。

海量视频题库 my1tpub.com QQ:5565462 www.52my1t.com

```
[student@servera ~]$ journalctl _UID=81
...output omitted...
Feb 22 01:29:09 servera.lab.example.com dbus-daemon[672]: [system] Activating via
 systemd: service name='org.freedesktop.nm_dispatcher'>
Feb 22 01:29:09 servera.lab.example.com dbus-daemon[672]: [system] Successfully
 activated service 'org.freedesktop.nm_dispatcher'
lines 1-5/5 (END) q
[student@servera ~]$
```

4. 将 **-p warning** 选项与 **journalctl** 命令搭配，以显示 servera 上优先级为 **warning** 及以上的日志事件。要退出 **journalctl**，请按 **q** 键。

```
[student@servera ~]$ journalctl -p warning
...output omitted...
Feb 13 13:21:07 localhost kernel: Detected CPU family 6 model 13 stepping 3
Feb 13 13:21:07 localhost kernel: Warning: Intel Processor - this hardware has not
 undergone testing by Red Hat and might not >
Feb 13 13:21:07 localhost kernel: acpi PNP0A03:00: fail to add MMCONFIG
 information, can't access extended PCI configuration s>
Feb 13 13:21:07 localhost rpc.statd[288]: Running as root.  chown /var/lib/nfs/
statd to choose different user
Feb 13 13:21:07 localhost rpc.idmapd[293]: Setting log level to 0
...output omitted...
Feb 13 13:21:13 servera.lab.example.com rsyslogd[1172]: environment variable TZ is
 not set, auto correcting this to TZ=/etc/lo>
Feb 13 14:51:42 servera.lab.example.com systemd[1]: cgroup compatibility
 translation between legacy and unified hierarchy sett>
Feb 13 17:15:37 servera.lab.example.com rsyslogd[25176]: environment variable TZ
 is not set, auto correcting this to TZ=/etc/l>
Feb 13 18:22:38 servera.lab.example.com rsyslogd[25410]: environment variable TZ
 is not set, auto correcting this to TZ=/etc/l>
Feb 13 18:47:55 servera.lab.example.com rsyslogd[25731]: environment variable TZ
 is not set, auto correcting this to TZ=/etc/l>
lines 1-17/17 (END) q
[student@servera ~]$
```

5. 显示 servera 上自当前时间起的过去 10 分钟内记录的所有日志事件。

5.1. 将 **--since** 选项与 **journalctl** 命令搭配使用，显示 servera 上过去 10 分钟内记录的所有日志事件。要退出 **journalctl**，请按 **q** 键。

```
[student@servera ~]$ journalctl --since "-10min"
...output omitted...
Feb 13 22:31:01 servera.lab.example.com CROND[25890]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 13 22:31:01 servera.lab.example.com run-parts[25893]: (/etc/cron.hourly)
 starting 0anacron
Feb 13 22:31:01 servera.lab.example.com run-parts[25899]: (/etc/cron.hourly)
 finished 0anacron
Feb 13 22:31:41 servera.lab.example.com sshd[25901]: Bad protocol version
 identification 'brain' from 172.25.250.254 port 37450
```

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

```
Feb 13 22:31:42 servera.lab.example.com sshd[25902]: Accepted publickey for root
 from 172.25.250.254 port 37452 ssh2: RSA SHA2>
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Started /run/user/0 mount
 wrapper.
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Created slice User Slice of
 UID 0.
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Starting User Manager for UID
 0...
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Started Session 118 of user
 root.
Feb 13 22:31:42 servera.lab.example.com systemd-logind[712]: New session 118 of
 user root.
Feb 13 22:31:42 servera.lab.example.com systemd[25906]: pam_unix(systemd-
user:session): session opened for user root by (uid=0)
...output omitted...
lines 1-32/84 39% q
[student@servera ~]$
```

6. 将 **--since** 选项及 **_SYSTEMD_UNIT="sshd.service"** 匹配项与 **journalctl** 命令搭配，以显示 servera 上源自从今天上午 **09 :00:00** 起记录的 sshd 服务的所有日志事件。要退出 **journalctl**，请按 **q** 键。

```
[student@servera ~]$ journalctl --since 9:00:00 _SYSTEMD_UNIT="sshd.service"
...output omitted...
Feb 13 13:21:12 servera.lab.example.com sshd[727]: Server listening on 0.0.0.0
 port 22.
Feb 13 13:21:12 servera.lab.example.com sshd[727]: Server listening on :: port 22.
Feb 13 13:22:07 servera.lab.example.com sshd[1238]: Accepted publickey for student
 from 172.25.250.250 port 50590 ssh2: RSA SH>
Feb 13 13:22:07 servera.lab.example.com sshd[1238]: pam_unix(sshd:session):
 session opened for user student by (uid=0)
Feb 13 13:22:08 servera.lab.example.com sshd[1238]: pam_unix(sshd:session):
 session closed for user student
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: Accepted publickey for root
 from 172.25.250.254 port 37194 ssh2: RSA SHA25>
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: pam_unix(sshd:session):
 session opened for user root by (uid=0)
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: pam_unix(sshd:session):
 session closed for user root
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: Accepted publickey for root
 from 172.25.250.254 port 37196 ssh2: RSA SHA25>
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: pam_unix(sshd:session):
 session opened for user root by (uid=0)
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: pam_unix(sshd:session):
 session closed for user root
Feb 13 13:26:07 servera.lab.example.com sshd[1355]: Accepted publickey for student
 from 172.25.250.254 port 37198 ssh2: RSA SH>
Feb 13 13:26:07 servera.lab.example.com sshd[1355]: pam_unix(sshd:session):
 session opened for user student by (uid=0)
Feb 13 13:52:28 servera.lab.example.com sshd[1473]: Accepted publickey for root
 from 172.25.250.254 port 37218 ssh2: RSA SHA25>
```

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

```
Feb 13 13:52:28 servera.lab.example.com sshd[1473]: pam_unix(sshd:session):
 session opened for user root by (uid=0)
...output omitted...
lines 1-32 q
[student@servera ~]$
```

7. 从 servera 注销。

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

完成

在 workstation 上，运行 **lab log-query finish** 来完成本练习。此脚本会确保环境重新恢复到清理干净的状态。

```
[student@workstation ~]$ lab log-query finish
```

本引导式练习到此结束。

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

保留系统日志

培训目标

学完本节后，您应能够配置系统日志，以在服务器重启时保留事件记录。

永久存储系统日志

默认情况下，系统日志保存在 **/run/log/journal** 目录中，这意味着系统重启时这些日志会被清除。您可以在 **/etc/systemd/journald.conf** 文件中更改 systemd-journald 服务的配置设置，使日志在系统重启后保留下来。

/etc/systemd/journald.conf 文件中的 Storage 参数决定系统日志以易失性方式存储，还是在系统重启后持久保留。按照如下所示，将该参数设为 **persistent**、**volatile** 或 **auto**：

- **persistent**：将日志存储在 **/var/log/journal** 目录中，这可在系统重启后持久保留。

  如果 **/var/log/journal** 目录不存在， systemd-journald 服务会创建它。

- **volatile**：将日志存储在易失性 **/run/log/journal** 目录中。

  因为 **/run** 文件系统是临时的，仅存在于运行时内存中，存储在其中的数据（包括系统日志）不会在系统启后持久保留。

- **auto**：rsyslog 决定要使用持久存储还是易失性存储。如果 **/var/log/journal** 目录存在，那么 rsyslog 会使用持久存储，否则使用易失性存储。

  如果未设置 Storage 参数，此为默认操作。

持久系统日志的优点是系统启动后就可立即利用历史数据。然而，即便是持久日志，并非所有数据都将永久保留。该日志具有一个内置日志轮转机制，会在每个月触发。此外，默认情况下，日志的大小不能超过所处文件系统的 10%，也不能造成文件系统的可用空间低于 15%。可以在 **/etc/systemd/journald.conf** 中为运行时和持久日志调整这些值。当 systemd-journald 进程启动时，会记录当前的日志大小限额。以下命令输出显示了反映当前大小限额的日志条目：

```
[user@host ~]$ journalctl | grep -E 'Runtime|System journal'
Feb 25 13:01:46 localhost systemd-journald[147]: Runtime journal (/run/log/
journal/ae06db7da89142138408d77efea9229c) is 8.0M, max 91.4M, 83.4M free.
Feb 25 13:01:48 remotehost.lab.example.com systemd-journald[548]: Runtime journal
 (/run/log/journal/73ab164e278e48be9bf80e80714a8cd5) is 8.0M, max 91.4M, 83.4M
 free.
Feb 25 13:01:48 remotehost.lab.example.com systemd-journald[548]: System journal
 (/var/log/journal/73ab164e278e48be9bf80e80714a8cd5) is 8.0M, max 3.7G, 3.7G free.
Feb 25 13:01:48 remotehost.lab.example.com systemd[1]: Starting Tell Plymouth To
 Write Out Runtime Data...
Feb 25 13:01:48 remotehost.lab.example.com systemd[1]: Started Tell Plymouth To
 Write Out Runtime Data.
```

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

> **注意**
> 在上面的 **grep** 中，竖线 (**|**) 符号充当 or 指示符。也就是说，**grep** 将匹配 **journalctl** 输出中任何含有 **Runtime** 字符串或 **System** 字符串的行。这将获取易失性 (**Runtime**) 日志存储和持久性 (**System**) 日志存储的当前大小限额。

配置持久系统日志

要将 systemd-journald 服务配置为在系统重启后持久保留系统日志，请在 **/etc/systemd/journald.conf** 文件中将 Storage 设为 **persistent**。以超级用户身份运行您选择的文本编辑器，以编辑 **/etc/systemd/journald.conf** 文件。

```
[Journal]
Storage=persistent
...output omitted...
```

编辑配置文件后，重新启动 systemd-journald 服务使配置更改生效。

```
[root@host ~]# systemctl restart systemd-journald
```

如果 systemd-journald 服务成功重新启动，您可以看到 **/var/log/journal** 目录已创建好，并且包含一个或多个子目录。这些子目录的长名称中包含十六进制字符，含有 ***.journal** 文件。***.journal** 文件是存储带有索引的结构化日志条目的二进制文件。

```
[root@host ~]# ls /var/log/journal
73ab164e278e48be9bf80e80714a8cd5
[root@host ~]# ls /var/log/journal/73ab164e278e48be9bf80e80714a8cd5
system.journal  user-1000.journal
```

由于系统日志在系统重启后保留，您在 **journalctl** 命令的输出中得到了大量的条目，其包括来自当前系统引导以及之前系统引导的条目。要将输出限制为特定的系统启动，请将 **-b** 选项用于 **journalctl** 命令。以下 **journalctl** 命令仅检索第一次系统启动的条目：

```
[root@host ~]# journalctl -b 1
...output omitted...
```

以下 **journalctl** 命令仅检索第二次系统启动的条目。只有当系统重启两次以上时，以下参数才有意义：

```
[root@host ~]# journalctl -b 2
```

以下 **journalctl** 命令仅检索当前系统启动的条目：

```
[root@host ~]# journalctl -b
```

海量视频题库 myitpub.com QQ:5565462
www.52myit.com

注意

利用永久日志调试系统崩溃时，通常需要将日志查询限制为崩溃发生之前的重新启动。可以给 **-b** 选项附上一个负数值，指示输出中应包含过去多少次系统引导。例如，**journalctl -b -1** 将输出限制为上一次启动。

参考文献

`systemd-journald.conf`(5)、`systemd-journald`(8) man page

如需更多信息，请参阅《红帽企业 Linux 8.0 基本系统设置配置指南》中的使用日志文件对问题进行故障排除章节，网址为：
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com

指导练习

保留系统日志

在本练习中，您将配置系统日志以在系统重启后保留其数据。

成果

您应能够配置系统日志以在系统重启后保留其数据。

在你开始之前

以 `student` 用户身份并使用 `student` 作为密码登录 `workstation`。

在 `workstation` 上，运行 **`lab log-preserve start`** 脚本来开始本练习。此脚本会确保环境设置正确无误。

```
[student@workstation ~]$ lab log-preserve start
```

1. 从 `workstation`，以 `student` 用户身份打开连接 `servera` 的 SSH 会话。

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

2. 以超级用户身份，确认 **`/var/log/journal`** 目录不存在。使用 **`ls`** 命令列出 **`/var/log/journal`** 目录的内容。使用 **`sudo`** 提升 `student` 用户的特权。如果系统要求，使用 `student` 作为密码。

```
[student@servera ~]$ sudo ls /var/log/journal
[sudo] password for student: student
ls: cannot access '/var/log/journal': No such file or directory
```

由于 **`/var/log/journal`** 目录不存在，`systemd-journald` 服务没有保留其日志。

3. 将 `servera` 上的 `systemd-journald` 服务配置为在系统重启后保留日志。

3.1. 在 **`/etc/systemd/journald.conf`** 文件中，取消注释 **`Storage=auto`** 行并将 `Storage` 设为 **`persistent`**。您可以使用 **`sudo vim /etc/systemd/journald.conf`** 命令来编辑配置文件。以 **`vim`** 命令模式键入 **`/ Storage=auto`**，搜索 **`Storage=auto`** 行。

```
...output omitted...
[Journal]
Storage=persistent
...output omitted...
```

3.2. 使用 **`systemctl`** 命令重新启动 `systemd-journald` 服务以使配置更改生效。

海量视频题库 myitpub.com QQ:556546 2
www.52myit.com

```
[student@servera ~]$ sudo systemctl restart systemd-journald.service
```

4. 确认 servera 上的 systemd-journald 服务保留其日志，以便这些日志在系统重启后持久保留。

4.1. 使用 **systemctl reboot** 命令来重启 servera。

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

请注意，一旦您重启 servera 系统，SSH 连接便会被终止。

4.2. 重新打开连接到 servera 的 SSH 会话。

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

4.3. 使用 **ls** 命令确认 **/var/log/journal** 目录存在。**/var/log/journal** 目录中包含一个带有十六进制长名称的子目录。日志文件就位于该目录中。您的系统上的子目录名称会有所不同。

```
[student@servera ~]$ sudo ls /var/log/journal
[sudo] password for student: student
73ab164e278e48be9bf80e80714a8cd5
[student@servera ~]$ sudo ls /var/log/journal/73ab164e278e48be9bf80e80714a8cd5
system.journal  user-1000.journal
```

4.4. 从 servera 注销。

```
[student@servera ~]$ exit
logout
Connection to servera closed.
```

完成

在 workstation 上，运行 **lab log-preserve finish** 来完成本练习。此脚本会确保环境重新恢复到清理干净的状态。

```
[student@workstation ~]$ lab log-preserve finish
```

本引导式练习到此结束。

海量视频题库 myitpub.com QQ:556462
www.52myit.com