Versi 1.0

Juni, 2023

PRAKTIKUM SISTEM OPERASI

MODUL 6 - Analyzing and Storing Logs

TIM PENYUSUN: - Mahar Faiqurahman, S.Kom., M.T.
- Syahrul pangestu
- Muh. ridha agam

Presented by: LAB. INFORMATIKA
Universitas Muhammadiyah Malang

SISTEM OPERASI

CAPAIAN PEMBELAJARAN MATA KULIAH

1. Mahasiswa mampu menjelaskan System Log Architecture
2. Mahasiswa mampu mereview Syslog Files
3. Mahasiswa mampu mereview System Journal Entries
4. Mahasiswa mampu memperlakukan System Journal

SUB CAPAIAN PEMBELAJARAN MATA KULIAH

1. Mahasiswa mampu memberikan contoh mengenai system log architecture
2. Mahasiswa mampu mempraktekkan langsung System log files
3. Mahasiswa mampu mempraktekkan system entri
4. Mahasiswa mampu mempraktekkan dan memberikan contoh sistem journal

KEBUTUHAN HARDWARE & SOFTWARE

- Laptop/PC
- Virtual Machine (VMware, Virtual Box, VPS bila ekonomi diatas rata-rata :")
- Sistem Operasi CentOS, download image OVA (Wajib)

MATERI POKOK

System Log Architecture

Proses dan kernel sistem operasi mencatat log dari peristiwa yang terjadi. Log ini digunakan untuk audit sistem dan menyelesaikan masalah.

Banyak sistem mencatat log peristiwa dalam file teks yang disimpan di direktori /var/log. Log ini dapat diperiksa menggunakan utilitas teks normal seperti less dan tail.

Sistem pencatatan standar berbasis protokol Syslog terintegrasi dalam Red Hat Enterprise Linux.

Banyak program menggunakan sistem ini untuk mencatat peristiwa dan mengorganisirnya ke dalam file log. Layanan systemd-journald dan rsyslog menangani pesan syslog dalam Red Hat Enterprise Linux 8.

Layanan systemd-journald berada di inti arsitektur pencatatan peristiwa sistem operasi. Ia mengumpulkan pesan peristiwa dari banyak sumber termasuk kernel, output dari tahap awal proses booting, output standar dan kesalahan standar dari daemon saat mereka mulai dan berjalan, dan peristiwa syslog. Kemudian, ia mengubahnya menjadi format standar, dan menulisnya ke dalam jurnal sistem terstruktur dan terindeks. Secara default, jurnal ini disimpan dalam sistem file yang tidak persisten saat reboot.

Namun, layanan rsyslog membaca pesan syslog yang diterima oleh systemd-journald dari jurnal saat pesan-pesan tersebut tiba. Kemudian, ia memproses peristiwa syslog tersebut, mencatatnya ke dalam file log-nya, atau mengirimkannya ke layanan lain sesuai dengan konfigurasi yang dimilikinya.

Layanan rsyslog menyortir dan menulis pesan syslog ke file log yang persisten saat reboot di direktori /var/log. Layanan rsyslog menyortir pesan log ke file log spesifik berdasarkan jenis program yang mengirimkan setiap pesan, atau fasilitas, dan prioritas setiap pesan syslog.

Selain file pesan syslog, direktori /var/log juga berisi file log dari layanan lain di sistem. Tabel berikut mencantumkan beberapa file yang berguna di direktori /var/log.

File Log Sistem yang Dipilih

Jenis Log	Tipe Pesan yang disimpan
/var/log/messages	Sebagian besar pesan syslog dicatat di sini. Beberapa pengecualian termasuk pesan terkait autentikasi dan pemrosesan email, pelaksanaan pekerjaan terjadwal, dan pesan yang terkait dengan debugging.
/var/log/secure	Pesan syslog terkait dengan keamanan dan peristiwa autentikasi.
/var/log/maillog	Pesan syslog terkait dengan server surel.
/var/log/cron	Pesan syslog terkait dengan pelaksanaan pekerjaan terjadwal.
/var/log/boot.log	Pesan konsol non-syslog terkait dengan proses startup sistem.

Reviewing Syslog Files

Logging Events to the System

Banyak program menggunakan protokol syslog untuk mencatat peristiwa ke sistem. Setiap pesan log dikategorikan berdasarkan fasilitas (jenis pesan) dan prioritas (keparahan pesan).

Fasilitas yang tersedia didokumentasikan dalam halaman man rsyslog.conf(5).

Tabel berikut mencantumkan delapan prioritas syslog standar dari tertinggi ke terendah.

Gambaran Prioritas Syslog:

Kode	Prioritas	Keparahan
0	emerg	Sistem tidak dapat digunakan
1	alert	Tindakan harus segera dilakukan
2	crit	Kondisi yang kritis
3	err	Kondisi kesalahan yang tidak kritis
4	warning	Kondisi peringatan
5	notice	Peristiwa normal harus di lihat lihat lagi
6	info	Peristiwa yang memberikan informasi
7	debug	Pesan tingkat debugging

rsyslog menggunakan fasilitas dan prioritas pesan log untuk menentukan cara menanganinya. Ini dikonfigurasi melalui aturan-aturan dalam file /etc/rsyslog.conf dan file apa pun di direktori /etc/rsyslog.d yang memiliki ekstensi nama file .conf. Paket-paket perangkat lunak dapat dengan mudah menambahkan aturan dengan menginstal file yang sesuai di direktori /etc/rsyslog.d.

Setiap aturan yang mengontrol cara menyortir pesan syslog adalah baris dalam salah satu file konfigurasi. Sisi kiri setiap baris menunjukkan fasilitas dan tingkat keparahan pesan syslog yang cocok dengan aturan tersebut. Sisi kanan setiap baris menunjukkan di mana pesan log disimpan (atau tempat lain untuk mengirimkan pesan tersebut). Tanda asterisk (*) adalah wildcard yang cocok dengan semua nilai.

Sebagai contoh, baris berikut akan mencatat pesan yang dikirim ke fasilitas authpriv dengan prioritas apa pun ke file /var/log/secure:

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

Pesan log kadang-kadang cocok dengan lebih dari satu aturan dalam rsyslog.conf. Dalam kasus seperti itu, satu pesan disimpan dalam lebih dari satu file log. Untuk membatasi pesan yang disimpan, kata kunci none dalam bidang prioritas menunjukkan bahwa tidak ada pesan untuk fasilitas yang ditunjukkan harus disimpan dalam file yang diberikan.

Alih-alih mencatat pesan syslog ke file, pesan-pesan tersebut juga dapat dicetak ke terminal semua pengguna yang telah login. File rsyslog.conf memiliki pengaturan untuk mencetak semua pesan syslog dengan prioritas emerg ke terminal semua pengguna yang telah login.

Contoh Aturan Rsyslog

Peraturan:

Jenis	Letak
• Log semua pesan kernel ke konsol. • Logging hal lain akan membuat tampilan layar menjadi berantakan.	/dev/console
• Log apapun (kecuali mail) pada level **info** dan **lebih tinggi** • Tidak dapat log otentikasi privat pesan! • *.info;mail.none;authpriv.none;cron.none	/var/log/messages
• Authpriv file akses restricted • authpriv.*	/var/log/secure
• Log seluruh pesan email dalam satu tempat • mail.*	-/var/log/maillog
• Log cron • cron.*	/var/log/cron
• Seluruh hal yang bersifat **emergency** • *.emerg	:omusrmsg:*
• Menyimpan berita error dari level **crit** keatas dalam sebuah file spesial • uucp,news.crit	/var/log/spooler
• Menyimpan pesan dalam boot.log • local7.*	/var/log/boot.log

Log File Rotation

logrotate memutar file log untuk mencegah mereka menghabiskan terlalu banyak ruang dalam sistem file yang berisi direktori /var/log. Ketika file log diputar, nama file tersebut diganti dengan ekstensi yang menunjukkan tanggal rotasinya. Misalnya, file lama /var/log/messages dapat menjadi /var/log/messages-20190130 jika diputar pada tanggal 2019-01-30. Setelah file log lama diputar, file log baru dibuat dan layanan yang menulis ke dalamnya akan diberitahu.

Analyzing a Syslog Entry

Pesan log dimulai dengan pesan tertua di bagian atas dan pesan terbaru di akhir file log. Layanan rsyslog menggunakan format standar saat mencatat entri dalam file log. Contoh berikut menjelaskan anatomi pesan log dalam file log /var/log/secure.

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

❶ The time stamp when the log entry was recorded
❷ The host from which the log message was sent
❸ The program or process name and PID number that sent the log message
❹ The actual message sent

Monitoring Logs

Memantau satu atau lebih file log untuk peristiwa berguna untuk mereproduksi masalah dan isu. Perintah tail -f /path/to/file akan menampilkan 10 baris terakhir dari file yang ditentukan dan terus-menerus menampilkan baris baru dalam file saat ditulis.

Misalnya, untuk memantau percobaan login gagal, jalankan perintah tail dalam satu terminal dan kemudian dalam terminal lain, jalankan perintah ssh sebagai pengguna root saat pengguna mencoba untuk login ke sistem.

Di terminal pertama, jalankan perintah tail berikut ini:

tail -f {letak tempat yang ingin di montor}

Sending Syslog Messages Manually

Perintah logger dapat mengirim pesan ke layanan rsyslog. Secara default, pesan dikirim ke fasilitas pengguna dengan prioritas notice (user.notice) kecuali ditentukan lain dengan opsi -p. Ini berguna untuk menguji setiap perubahan pada konfigurasi layanan rsyslog.

Untuk mengirim pesan ke layanan rsyslog yang direkam dalam file log /var/log/boot.log, jalankan perintah logger berikut ini.

Setelah beberapa putaran tertentu, biasanya setelah empat minggu, file log tertua dihapus untuk membebaskan ruang disk. Pekerjaan terjadwal menjalankan program logrotate setiap hari untuk melihat apakah ada log yang perlu diputar. Sebagian besar file log diputar setiap minggu, tetapi logrotate memutar beberapa log lebih cepat atau lebih lambat, atau ketika mereka mencapai ukuran tertentu.

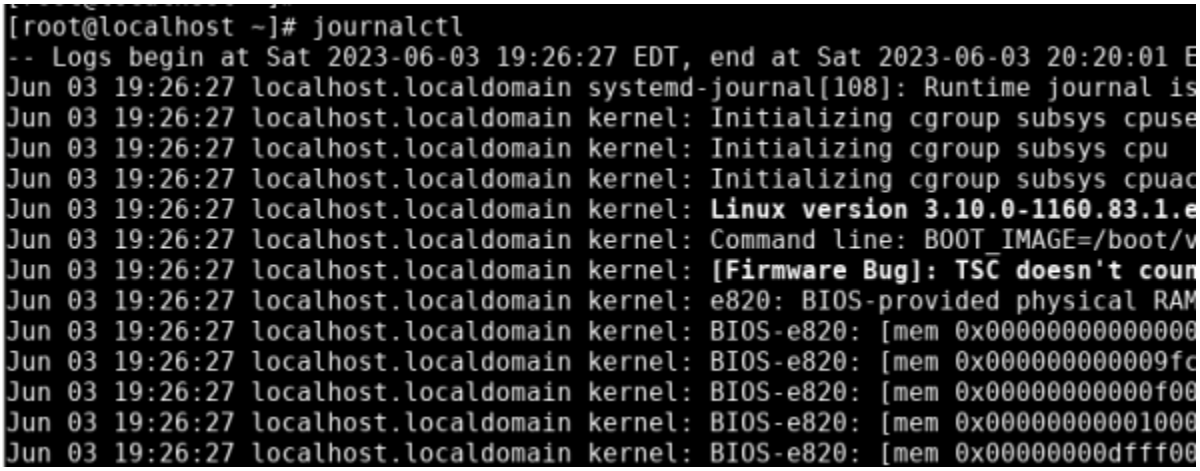
```
[student@localhost ~]$ sudo logger -p local7.notice "Log Entri terbuat"
```

Reviewing System Journal Entries

Finding Events

Layanan systemd-journald menyimpan data logging dalam file biner terstruktur yang diindeks yang disebut jurnal. Data ini mencakup informasi tambahan tentang peristiwa log. Misalnya, untuk peristiwa syslog, ini mencakup fasilitas dan prioritas pesan asli.

Untuk mengambil pesan log dari jurnal, gunakan perintah journalctl. Anda dapat menggunakan perintah ini untuk melihat semua pesan dalam jurnal, atau untuk mencari peristiwa tertentu berdasarkan berbagai pilihan dan kriteria. Jika Anda menjalankan perintah sebagai root, Anda memiliki akses penuh ke jurnal. Pengguna biasa juga dapat menggunakan perintah ini, tetapi mungkin dibatasi untuk melihat pesan tertentu.



(Melihat controller dan melihat seluruh pesan)

Perintah journalctl menyoroti pesan log penting: pesan dengan **prioritas notice** atau **warning** ditampilkan dalam teks tebal sedangkan pesan dengan **prioritas error** atau lebih tinggi ditampilkan dalam teks merah.

Kunci untuk berhasil menggunakan jurnal untuk pemecahan masalah dan audit adalah membatasi pencarian jurnal agar hanya menampilkan keluaran yang relevan.

Secara default, journalctl -n menampilkan 10 entri log terakhir. Anda dapat mengubah ini dengan argumen opsional yang menentukan berapa banyak entri log yang akan ditampilkan. Untuk lima entri log terakhir, jalankan perintah journalctl berikut ini.

```
[root@localhost ~]# journalctl -n 5
-- Logs begin at Sat 2023-06-03 19:26:27 EDT, end at Sat 2023-06-03 20:24:31 E
Jun 03 20:24:27 localhost.localdomain NetworkManager[680]: <info>  [1685838267
Jun 03 20:24:27 localhost.localdomain NetworkManager[680]: <info>  [1685838267
Jun 03 20:24:27 localhost.localdomain avahi-daemon[556]: Registering new addre
Jun 03 20:24:27 localhost.localdomain dhclient[3545]: DHCPDISCOVER on enp0s3 t
Jun 03 20:24:31 localhost.localdomain dhclient[3545]: DHCPDISCOVER on enp0s3 t
...skipping...
```

(Membatasi jumlah pesan yang ingin dilihat)

```
Jun 03 19:26:27 localhost.localdomain kernel: Detected CPU family 17h model 96
Jun 03 19:26:27 localhost.localdomain kernel: Warning: AMD Processor - this ha
```

(Jika ada pesan error akan muncul warna merah)

Mirip dengan perintah tail -f, perintah journalctl -f akan menampilkan 10 baris terakhir dari jurnal sistem dan terus-menerus menampilkan entri jurnal baru saat ditulis ke jurnal. Untuk keluar dari proses journalctl -f, gunakan kombinasi tombol Ctrl+C.

```
[root@localhost ~]# journalctl -f
-- Logs begin at Sat 2023-06-03 19:26:27 EDT. --
Jun 03 20:26:30 localhost.localdomain dhclient[3575]: DHCPDISCOVER on enp0s3 t
o 255.255.255.255 port 67 interval 21 (xid=0x285bbc76)
Jun 03 20:26:44 localhost.localdomain NetworkManager[680]: <warn>  [1685838404
.2765] dhcp4 (enp0s3): request timed out
Jun 03 20:26:44 localhost.localdomain NetworkManager[680]: <info>  [1685838404
.2767] dhcp4 (enp0s3): state changed unknown -> timeout
Jun 03 20:26:44 localhost.localdomain NetworkManager[680]: <info>  [1685838404
.3012] dhcp4 (enp0s3): canceled DHCP transaction, DHCP client pid 3575
Jun 03 20:26:44 localhost.localdomain NetworkManager[680]: <info>  [1685838404
.3013] dhcp4 (enp0s3): state changed timeout -> done
Jun 03 20:26:44 localhost.localdomain NetworkManager[680]: <info>  [1685838404
```

(Melihat 10 pesan terakhir dan melanjutkan terus menerus kecuali di CTRL + C)

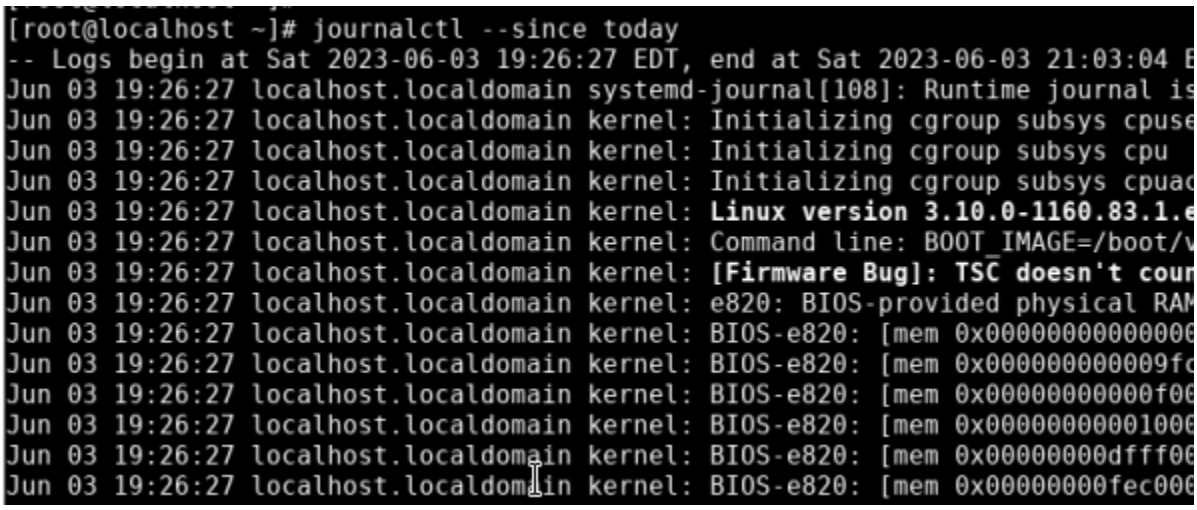
Untuk membantu memecahkan masalah, Kita dapat menyaring keluaran jurnal berdasarkan prioritas entri jurnal. Perintah journalctl -p menerima nama atau nomor tingkat prioritas dan menampilkan entri jurnal untuk entri dengan prioritas tersebut dan di atasnya. Perintah journalctl memahami tingkat prioritas debug, info, notice, warning, err, crit, alert, dan emerg.

```
[root@localhost ~]# journalctl -p err
-- Logs begin at Sat 2023-06-03 19:26:27 EDT, end at Sat 2023-06-03 20:34:44 E
Jun 03 19:26:27 localhost.localdomain kernel: Detected CPU family 17h model 96
Jun 03 19:26:27 localhost.localdomain kernel: Warning: AMD Processor - this ha
Jun 03 19:26:36 localhost.localdomain kernel: [drm:vmw_host_log [vmwgfx]] *ERR
Jun 03 19:26:36 localhost.localdomain kernel: [drm:vmw_host_log [vmwgfx]] *ERR
Jun 03 19:27:20 localhost.localdomain mcelog[566]: ERROR: AMD Processor family
Jun 03 19:27:21 localhost.localdomain mcelog[613]: ERROR: AMD Processor family
Jun 03 19:27:21 localhost.localdomain systemd[1]: Failed to start Machine Chec
Jun 03 19:27:54 localhost.localdomain systemd[1]: Failed to start Network Mana
```

(Melihat seluruh error yang terjadi dalam sistem anda!)

Ketika mencari peristiwa tertentu, Anda dapat membatasi keluaran jurnal ke rentang waktu tertentu. Perintah journalctl memiliki dua opsi untuk membatasi keluaran ke rentang waktu tertentu, yaitu opsi --since dan --until. Kedua opsi tersebut menerima argumen waktu dalam format "YYYY-MM-DD hh:mm:ss" (tanda kutip ganda diperlukan untuk mempertahankan spasi dalam opsi tersebut). Jika tanggal dihilangkan, perintah tersebut akan mengasumsikan hari saat ini, dan jika waktu dihilangkan, perintah tersebut akan mengasumsikan seluruh hari mulai dari 00:00:00. Kedua opsi tersebut juga menerima argumen yesterday, today, dan tomorrow sebagai argumen valid selain tanggal dan waktu.

Jalankan perintah journalctl berikut ini untuk menampilkan semua entri jurnal dari catatan hari ini.

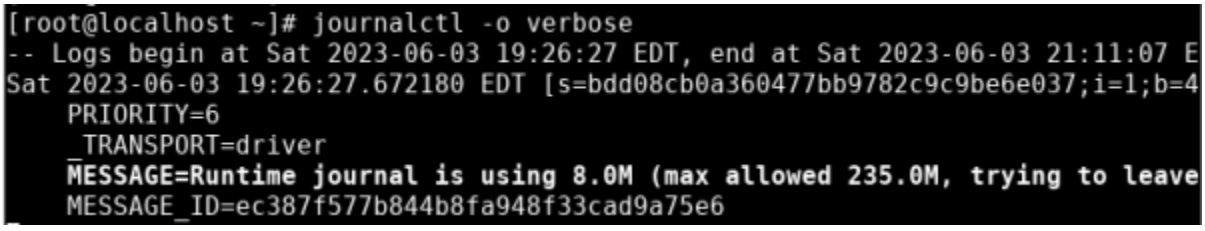


(Melihat isi journal hari ini)

Kita juga dapat mengunakan **journalctl –since “2023-06-4 08:00”** atau **journalctl –since “-1 hour”**

Selain konten yang terlihat pada jurnal, terdapat juga field yang terlampir pada entri log yang hanya dapat dilihat ketika output verbose diaktifkan. Setiap field tambahan yang ditampilkan dapat digunakan untuk menyaring output dari kueri jurnal. Hal ini berguna untuk mengurangi output dari pencarian kompleks untuk peristiwa tertentu dalam jurnal.

journalctl -o verbose



(Melihat verbose dari messages)

Berikut adalah daftar field umum dari jurnal sistem yang dapat digunakan untuk mencari baris yang relevan dengan proses atau peristiwa tertentu:

- _COMM adalah nama perintah
- _EXE adalah path menuju eksekutor untuk proses tersebut
- _PID adalah PID dari proses
- _UID adalah UID dari pengguna yang menjalankan proses
- _SYSTEMD_UNIT adalah unit systemd yang memulai proses tersebut

Lebih dari satu field jurnal sistem dapat digabungkan untuk membentuk kueri pencarian yang lebih terperinci dengan perintah journalctl. Sebagai contoh, perintah journalctl berikut ini menampilkan semua entri jurnal terkait dengan unit systemd sshd.service dari sebuah proses dengan PID 1182.

Command: **journalctl _SYSTEMD_UNIT=sshd.service _PID={PID Process}**

Preserving the System Journal

Storing the System Journal Permanently

Secara default, jurnal sistem disimpan di direktori /run/log/journal, yang berarti jurnal akan dihapus saat sistem melakukan reboot. Anda dapat mengubah pengaturan konfigurasi layanan systemd-journald di file /etc/systemd/journald.conf untuk membuat jurnal persisten setelah reboot.

Parameter Storage dalam file /etc/systemd/journald.conf mendefinisikan apakah jurnal sistem disimpan secara volatil atau persisten setelah reboot. Atur parameter ini sebagai persistent, volatile, auto, atau none sebagai berikut:

persistent: menyimpan jurnal di direktori /var/log/journal yang persisten setelah reboot. Jika direktori /var/log/journal belum ada, layanan systemd-journald akan membuatnya.

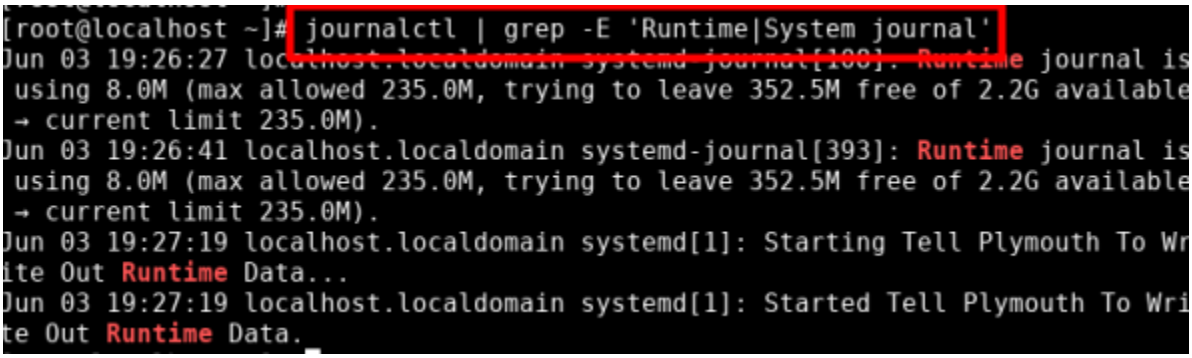
volatile: menyimpan jurnal di direktori volatil /run/log/journal. Karena sistem file /run bersifat sementara dan hanya ada dalam memori saat runtime, data yang disimpan di dalamnya, termasuk jurnal sistem, tidak akan persisten setelah reboot.

auto: jika direktori /var/log/journal ada, maka systemd-journald menggunakan penyimpanan persisten; jika tidak, maka menggunakan penyimpanan volatil. Ini adalah tindakan default jika parameter Storage tidak diatur.

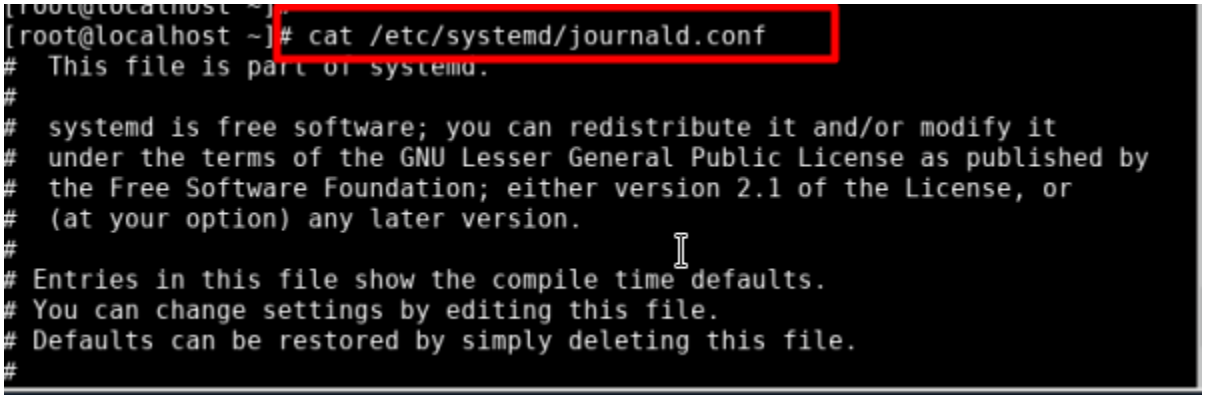
none: tidak menggunakan penyimpanan apa pun. Semua log akan dihapus, tetapi pengiriman log masih berfungsi seperti yang diharapkan.

Keuntungan dari jurnal sistem yang persisten adalah data historisnya tersedia segera saat boot. Namun, bahkan dengan jurnal persisten, tidak semua data disimpan selamanya. Jurnal memiliki mekanisme rotasi log bawaan yang dijalankan secara bulanan. Selain itu, secara default, jurnal tidak diizinkan melebihi 10% dari sistem file di mana jurnal berada, atau meninggalkan ruang kosong kurang dari 15%

pada sistem file tersebut. Nilai-nilai ini dapat disesuaikan untuk jurnal runtime dan persisten di file /etc/systemd/journald.conf. Batasan ukuran jurnal saat ini dicatat saat proses systemd-journald dimulai. Keluaran perintah berikut menampilkan entri jurnal yang mencerminkan batasan ukuran saat ini.



Configuring Persistent System Journals



Bisa di nano saja buat edit.

Setelah mengedit file konfigurasi, restart layanan systemd-journald untuk menerapkan perubahan konfigurasi.

Anda dapat melakukannya dengan menjalankan perintah berikut:

systemctl restart systemd-journald

Jika layanan systemd-journald berhasil di-restart, Anda akan melihat bahwa direktori /var/log/journal dibuat dan berisi satu atau lebih subdirektori. Subdirektori ini memiliki nama panjang dengan karakter heksadesimal dan berisi file *.journal. File *.journal ini adalah file biner yang menyimpan entri jurnal yang terstruktur dan diindeks.

Setelah restart, Anda dapat memeriksa direktori /var/log/journal untuk melihat subdirektori dan file *.journal yang berisi jurnal sistem yang disimpan secara persisten.

command 1 : **ls /var/log/journal** command 2: **ls /var/log/journal/{Binary indexed journal entries}**

Meskipun jurnal sistem persisten setelah reboot, Anda akan mendapatkan jumlah entri yang sangat banyak dalam output perintah journalctl yang mencakup entri dari boot sistem saat ini serta boot sebelumnya. Untuk membatasi output hanya pada boot sistem tertentu, gunakan opsi -b dengan perintah journalctl. Perintah journalctl berikut ini mengambil entri yang terbatas pada boot sistem pertama.

Bisa menggunakan **journalctl -b 1** atau **journalctl -b 2** untuk melihat entries limited terhadap sistem boot tersebut. Atau bahkan bisa langsung **journalctl -b** untuk melihat system boot sementara

LEMBAR KERJA

KEGIATAN 1

Reviewing Syslog Files

1. Konfigurasi rsyslog dengan prioritas debug dan lebih tinggi pada service **/var/log/messages-debug** dan log file dengan menambahkan konfigurasi file pada **/etc/rsyslog.d/debug.conf**
2. Buat /etc/rsyslog.d/debug.conf dan **redirect** file dengan kebutuhannya yang diperlukan dengan cara **debug** prioritas.
3. Restart servicenya (bukan komputer/laptop kamu yaa 🙂)
4. Verifikasi bahwa seluruh pesan log dengan **debug prioritas** telah muncul pada log yang telah dibuat
5. Gunakan command pada kegiatan 1 yang memunculkan **log message** dengan **user** dan **debug priority**
6. Gunakan command untuk monitoring pada **/var/log/messages-debug** exit dan buatlah cmd baru untuk lanjut kegiatan 2

KEGIATAN 2

1. Gunakan _PID yang menunjukkan **log events** mengenai **systemd** process running (Pilih angka PID yang tepat!!)
2. Keluar dari journalctl tersebut (bukan tutup cmdnya ya)
3. Gunakan UID yang dapat memperlihatkan **log events** yang berasal dari **sistem service** yang dimulai dari **user identifier**. (cluenya diatas 80 dibawah 85 _uidnya)
4. Gunakan opsi -p warning dengan perintah journalctl untuk menampilkan peristiwa log dengan prioritas warning dan yang lebih tinggi.
5. Perlihatkan keseluruhan log yang terekam pada **20 menit** terakhir
6. Gunakan –since dan perlihatkan seluruh **log events** pada **sshd service** sejak **09:00** pagi
7. Jelaskan apa itu **conf, systemd, mem, pid, cron, tail**
8. Selesai buka cmd baru untuk kegiatan 3

KEGIATAN 3

1. Sebagai SuperUser periksalah apakah **/var/log/journal tidak ada** maka lanjut. jika **masih ada maka hapus** terlebih dahulu lognya agar kembali tidak terkonfigurasi
2. **Restart services** pada **systemd journald** agar konfigurasi tersebut jalan (bukan reboot vb kalian yaaa dah 3 kali ni)
3. Gunakan command yang bisa melihat isi **folder** log **journal** tersebut (bukan cat)
4. Konfigutasi agar **systemd-journald** pada **configure** nya dan set agar storagenya menjadi persistent (silahkan cari dimana letak konfigurasi journald)
5. Perlihatkan **isi dalam log**, **jelaskan** isinya apa saja
6. Jelaskan kenapa **butuh sistem ini**? apa gunanya? berikan **contoh kasus** penggunaannya!
7. Selamatt selesai sudah **modul 6**.. tinggal nunggu nilai akhir nih! sama ndatau ya uap ada nda he.

CATATAN:
Sebelum mulai praktikum silahkan setting kembali untuk **kegiatan 1,2,3**

RUBRIK PENILAIAN

Aspek Penilaian	Bobot Penilaian
Ketepatan menjawab semua kegiatan	65%
Pemahaman setiap aspek materi yang dibahas	20%
Quiz pertemuan materi minggu pertama	15%
Total	**100%**