Journald

Els registres del sistema (els quals poden representar des de simples missatges informatius fins missatges crítics d'errors i poden ser rebuts directament per part del kernel o bé de processos d'usuari o de dimonis, entre d'altres origens) són recopilats per un demoni central anomenat systemd-journald. Per tant, haurà d'estar encés i habilitat per a què funcioni aquesta recopilació de registres: *systemctl start systemd-journald && systemctl enable systemd-journald*

Basic configuration

The journal stores log data in /run/log/journal/ by default (in binary form). Because the /run/ directory is volatile by nature (it's located on tmpfs, which is, in fact, RAM), log data is lost at reboot. To make the log data persistent, the directory /var/log/journal/ must exist because it's the only place where the systemd-journald service can store its data. Systemd will create the directory for you (and switch to persistent logging) if you edit /etc/systemd/journald.conf file leaving following line like this (the contrary of "persistent" is "volatile"; it is possible to use "no" as a value, too):

Storage=persistent

Journal occupies by default 10% of available partition space, and when it gets it, it rotates the logs like a list (deleting the last one). To find out the disk that journal currently uses, you can execute *journalctl --disk-usage* command. If you want to change the maximum possible usage, you can edit /etc/systemd/journald.conf file to leave following lines like this:

SystemMaxUse=50G

NOTA: Existeix una directiva semblant però que afecta a l'emmagatzematge del Journal en RAM (és a dir, a /run/log/journal) anomenada *RuntimeMaxUse*

NOTA: La directiva *SystemMaxFileSize* (o *RuntimeMaxFileSize*) defineixen el tamany màxim (en K,M,G,T...) que tindrà un fitxer .journal abans de comprimir-se, rotar-se i arxivar-se. Per defecte val 1/8 del tamany indicat a SystemMaxUse, i això vol dir, per tant, que es podran tenir fins a 7 arxius .journal arxivats més l'actiu.

Alternativament, es pot indicar no un tamany màxim sinó un temps màxim, passat el qual s'esborraran els logs més antics. Això es pot fer indicant la directiva següent (on el número per defecte representa segons però es pot convertir en minuts, hores, dies, setmanes, mesos o anys si s'afegeix el sufixe "m","h","day","week","month" o "year", respectivament). Per defecte val 0, que vol dir que no s'aplica:

MaxRetentionSec=nº

NOTA: La directiva *MaxFileSec=nº* ofereix una forma alternativa de rotació de logs que, en comptes de tenir en compte el tamany màxim dels arxius arxivats (com passava amb *SystemMaxFileSize* té en compte el temps màxim que ha de passar per arxivar un arxiu .journal

On the other hand, you can forward the journal to a terminal device to inform you about system messages on a preferred terminal screen, for example /dev/tty12. Change the following options in /etc/systemd/journald.conf to:

ForwardToConsole=yes
TTYPath=/dev/tty12
MaxLevelConsole=info

NOTA: Una manera alternativa de hacer lo mismo sin necesidad de tocar el archivo de configuración general sería creando un archivo de configuración suplementario tal como este, /etc/systemd/journald.conf.d/hola.conf con el siguiente contenido:

[Journal]
ForwardToConsole=yes
TTYPath=/dev/tty12
MaxLevelConsole=info

By default, Journal users without root privileges can only see log files generated by them. Adding an user to "systemd-journal" or "adm" group gives him access to all logs.

Parámetros de journalctl

El escribir más de un parámetro a la vez (si funcionan como filtro) es equivalente a un AND, a no ser que entre ellos se escriba un "+", interpretándose entonces un OR:

-f	Similar a tail -f
-r	Muestra los mensajes al revés (el más moderno primero)
-e	Muestra todos los mensajes y se mueve hasta el último (por defecto se queda en el primero)
-x	Añade información extra respecto las líneas mostradas (enlaces a documentación, contextos de los mensajes, posibles soluciones a errores, etc)
-o { verbose | json | export | short-iso | cat }	Muestra salida en modo verboso (viendo todos los campos de cada registro), en formato JSON, en formato binario (especialmente pensado para exportaciones a través de la red), en formato similar al normal pero añadiendo el año a la fecha mostrada (cosa que por defecto no se hace) o mostrando solo el texto del mensaje propiamente dicho.
--utc	Muestra la fecha de cada mensaje en formato Universal en vez de Local (que es como Systemd las muestra por defecto)
--no-pager	No usa paginador para mostrar los mensajes sino que los "vomita" todos de golpe (útil para entubar la salida a otro comando)
-n nº	Muestra solo los últimos nº mensajes. Si nº no se escribe, por defecto es 10
/ruta/ejecutable	Filtra solo los mensajes relacionados con el ejecutable indicado.También puede /ruta/dev
_PID nº	Filtra solo los mensajes relacionados con el PID idicado
-u nombreUnit	Filtra solo los mensajes relacionados con el "unit" indicado
-k	Filtra solo los mensajes relacionados con el kernel (similar a dmesg)
_UID nº	Filtra solo los mensajes relacionados con el UID indicado
_HOSTNAME nomMaquina	Filtra solo los mensajes relacionados con el host indicado
-t unaTag	Filtra solo los mensajes etiquetados con el valor indicado (ver más abajo *systemd-cat*)
-p nomPrioritat	Filtra solo los mensajes con una prioridad igual o superior a la indicada (también se puede poner -p pri1..pri2). Las prioridades son (en vez del nombre se podría indicar su número): debug (7), info (6), notice (5), warning (4), err (3), crit (2), alert (1), emerg (0)
-F nomCamp	Muestra solo los valores del campo indicado (con -o verbose se pueden consultar cuáles son o a https://www.freedesktop.org/software/systemd/man/systemd.journal-fields.html)
nomCamp=valor	Muestra solo los registros que tengan en el campo indicado el valor indicado
--since=xxxx	Filtra solo los mensajes ocurridos desde la fecha y hora indicadas NOTA:El formato de la fecha ha de ser "YYYY-MM-DD hh:mm:ss"; si se omite el tiempo se asume 00:00:00; si se omiten los segundos se asume :00; si se omite la fecha se asume hoy. También se pueden indicar las expresiones "yesterday", "today" y "tomorrow", que se refieren a la medianoche del día anterior, actual o siguiente. También se puede escribir "now", que se refiere a la hora actual (útil con -f). También se pueden indicar tiempos relativos respecto la hora actual con – (antes) o + (después), así: "-1week", "-1month", "-20day" (ver man systemd.time)
--until=xxx	Filtra solo los mensajes ocurridos hasta la fecha y hora indicadas

-b nº	Filtra solo los mensajes pertenecientes al uso de la máquina tras el arranque nº indicado (donde nº=0 -o no ponerlo- representa el arranque actual, nº=-1 el anterior, y así) NOTA:Para ver todos los inicios posibles hacer *journalctl –list-boots* El segundo campo mostrado por este comando es el "boot ID", el cual sirve para referirse a un arranque concreto y se podría usar, por ejemplo, para filtrar de esta manera: *journalctl _BOOT_ID=valorBootID*. A continuación del "boot ID" aparece la fecha y hora de la primera entrada guardada para cada uno de esos arranques. Finalmente, aparecen la fecha y hora de la última entrada
--vacuum-size = 1{ K | M | G | T }	Borra del disco las entradas necesarias (empezando por las más antiguas) para que el registro ocupe solo 1G en disco. No opera con las entradas activas, solo las archivadas.
--vacuum-time = 1 { s | m | h | days | weeks | months | years }	Borra del disco las entradas más antiguas que el tiempo indicado. Si s'invoca juntament amb --vacuum-size s'aplicarà la restricció que esborri més. Si un fichero .journal incluye entradas más antiguas pero también más nuevas, no será borrado.
--flush	Asks the journal daemon to flush any log data stored in RAM (specifically, in /run/log/journal) into /var/log/journal, if persistent storage is enabled. This call does not return until the operation is complete. Note that the data is only flushed once during system runtime, so this command exits cleanly without executing any operation if this has already happened. There is a service called *systemd-journal-flush* which is normally invoked at startup, responsible of executing this command every boot.
--rotate	Força una rotació dels fitxers .journal
--verify	Comprueba la integridad interna del registro
-D /ruta/journal	Útil cuando se quiere inspeccionar desde un sistema Live el journal (previo montaje de la partición donde reside) de un sistema no arrancado/arrancable

De momento no hay la posibilidad de indicar filtros excluyentes del tipo "todo excepto esto" (https://github.com/systemd/systemd/issues/2720)

Para enviar un mensaje manualment al registro existe el comando específico llamado "systemd-cat", que funciona así: *echo "Backup Failed" | systemd-cat -t "nombreSimuladoAplicacion" -p "crit"* (la fecha se añade sola) O también así: *systemd-cat -t "nombreSimuladoAplicacion" -p "crit" echo "Backup Failed"* (lo mismo que el caso anterior pero además registra stderr). Una alternativa a "systemd-cat" es el venerable comando *logger -s -t "nombre" -p " crit"*

Existen programas complementarios que lanzan alertas (ejecución de un determinado programa, envío de mails o similares, etc) cuando se detecta un determinado mensaje en el journal. Ejemplos son:

* https://github.com/jjk-jacky/journal-triggerd (su página web es https://jjacky.com/journal-triggerd)
* https://github.com/The-Compiler/journalwatch (solo sirve para enviar mails)

Accés remot de logs

Si volem veure els missatges de registre d'una màquina remota es pot fer via HTTP gràcies a un "miniservidor web" específic anomenat "systemd-journal-gatewayd", activable mitjançant socket (i que ve inclòs dins del paquet "systemd-journal-remote", el qual caldrà instal.lar prèviament). Si es mira la línia ListenStream del seu arxiu de configuració es pot comprovar que per defecte escolta a totes les interfícies pel port 19531 (procura doncs que aquest port estigui obert per l'eventual tallafocs que hi hagi al sistema!)

> NOTA: Aquest servidor també pot oferir els registres via HTTPS si s'indica a l'executable presenta a la línia ExecStart= del seu arxiu .service un determinat certificat i clau privada amb els paràmetres --cert= i --key=, respectivament

Un cop funcionant el servidor gatewayd (*systemctl start systemd-journal-gatewayd*), per veure els logs en una altra màquina via HTTP podem fer servir qualsevol client com ara un navegador o curl. En aquest darrer cas, això es faria així:

curl -s -H"Accept:application/vnd.fdo.journal" http://ip.Del.Servidor:19531/entries

on el valor de la capçalera de client Accept indica el format de les dades a rebre (altres formats vàlids són "text/plain" (per defecte) o "application/json").

NOTA: També es pot afegir una altra capçalera per indicar un rang d'entrades concretes a obtenir, així: *"Range:entries=cursor[[:nºskip]:nºentrades]"*

Després de la ruta "/entries" es poden indicar diferents paràmetres en forma de "querystring" (és a dir, així /entries?param1&param2&param3...), com ara:

boot : Similar al paràmetre -b de journalctl
follow : Similar al paràmetre -f de journalctl
nomCamp=valor : Per filtrar per un valor concret d'un camp propi del Journal (_UID, _HOSTNAME, etc)

D'altra banda, en substitució de la ruta "/entries" també es poden escriure aquestes altres rutes:

/browse : Especialment pensat per utilitzar un navegador: permet visualitzar els registres de forma interactiva via web
/fields/nomCamp : Mostra els diferents valors trobats pel camp propi del Journal (_UID, _HOSTNAME, etc) indicat