



Tail Command Basics

Basic Usage

<code>tail filename</code> - Displays the last 10 lines of a file.
<code>tail -n N filename</code> - Displays the last N lines of a file.
<code>tail -f filename</code> - Follows the file in real-time, displaying new lines as they are added.
<code>tail -F filename</code> - Similar to <code>-f</code> , but also monitors for file rotation.
<code>tail -q filename</code> - Suppresses printing the headers identifying the files being followed.

Following Multiple Files

<code>tail -f file1 file2 file3</code> - Follows multiple files simultaneously.
<code>tail -q -f file1 file2</code> - Follows multiple files without headers.
<code>tail 'foo.*'</code> - Follows all files matching glob pattern <code>foo.*</code> .

Other useful options

<code>tail --retry -f filename</code> - Keep trying to open a file even if it is inaccessible.
<code>tail -s N</code> - With <code>-f</code> , sleep for approximately N seconds between iterations.

Journalctl Command Basics

Basic Usage

<code>journalctl</code> - Displays all log entries.
<code>journalctl -n N</code> - Displays the last N log entries.
<code>journalctl -f</code> - Follows the journal in real-time.
<code>journalctl --since "yesterday"</code> - Shows entries from yesterday.
<code>journalctl --until "today"</code> - Shows entries until today.
<code>journalctl --since "2024-01-01" --until "2024-01-02"</code> - Shows entries between specific dates.

Filtering by Unit

<code>journalctl -u servicename.service</code> - Shows entries for a specific systemd service.
<code>journalctl -u servicename.service -f</code> - Follows logs for a specific service.

Filtering by Priority

<code>journalctl -p err</code> - Shows error messages.
<code>journalctl -p warning</code> - Shows warning messages.
<code>journalctl -p crit</code> - Shows critical messages.
<code>journalctl -p alert</code> - Shows alert messages.
<code>journalctl -p emerg</code> - Shows emergency messages.

Advanced Journalctl Usage

Filtering by PID and UID

<code>journalctl _PID=1234</code> - Shows entries for a specific process ID.
<code>journalctl _UID=1000</code> - Shows entries for a specific user ID.

Filtering by Kernel Messages

<code>journalctl -k</code> - Shows kernel messages.
<code>journalctl -k -f</code> - Follows kernel messages in real-time.

Disk Usage and Cleanup

<code>journalctl --disk-usage</code> - Shows disk space used by journal logs.
<code>journalctl --vacuum-size=1G</code> - Reduces disk usage by keeping only 1GB of logs.
<code>journalctl --vacuum-time=2weeks</code> - Removes logs older than 2 weeks.

Output Formatting

<code>journalctl -o verbose</code> - Show all available fields, including the message.
<code>journalctl -o cat</code> - Show only the message field.
<code>journalctl -o json</code> - Show output in JSON format.

Combining Tail and Journalctl

Real-time Monitoring

Use <code>tail -f</code> for specific application logs and <code>journalctl -f</code> for system-level logs simultaneously to get a comprehensive view.

Troubleshooting Example

If an application (e.g., <code>myapp.service</code>) is failing, use <code>tail -f /var/log/myapp.log</code> to check its log file and <code>journalctl -u myapp.service -f</code> to check systemd logs for related errors.
Example: Application log shows connection timeout; systemd log shows network service failure around the same time. This helps correlate issues.

Use Cases

Use <code>tail</code> to monitor application-specific logs that are not managed by systemd.
Use <code>journalctl</code> for system-level debugging, especially for services managed by systemd.
Combine both for comprehensive application and system debugging.