Log Monitoring Tools Cheatsheet

A quick reference guide to effectively monitoring logs using `tail` and `journalctl`, essential tools for system administrators and developers. This cheatsheet provides practical commands and options to streamline log analysis and troubleshooting.

Tail Command Basics

Basic Usage

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

Following Multiple Files

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

Other useful options

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

Journalctl Command Basics

Basic Usage

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

Filtering by Unit

- `journalctl -u servicename.service` - Shows entries for a specific systemd service.
- `journalctl -u servicename.service -f` - Follows logs for a specific service.

Filtering by Priority

- `journalctl -p err` - Shows error messages.
- `journalctl -p warning` - Shows warning messages.
- `journalctl -p crit` - Shows critical messages.
- `journalctl -p alert` - Shows alert messages.
- `journalctl -p emerg` - Shows emergency messages.

Advanced Journalctl Usage

Filtering by PID and UID

- `journalctl _PID=1234` - Shows entries for a specific process ID.
- `journalctl _UID=1000` - Shows entries for a specific user ID.

Filtering by Kernel Messages

- `journalctl -k` - Shows kernel messages.
- `journalctl -k -f` - Follows kernel messages in real-time.

Disk Usage and Cleanup

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

Output Formatting

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

Combining Tail and Journalctl

Real-time Monitoring

- Use `tail -f` for specific application logs and `journalctl -f` for system-level logs simultaneously to get a comprehensive view.

Troubleshooting Example

- If an application (e.g., `myapp.service`) is failing, use `tail -f /var/log/myapp.log` to check its log file and `journalctl -u myapp.service -f` 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 `tail` to monitor application-specific logs that are not managed by systemd.
- Use `journalctl` for system-level debugging, especially for services managed by systemd.
- Combine both for comprehensive application and system debugging.