

---

# **pemmican 1.0.3 Documentation**

*Release 1.0.3*

**Dave Jones**

**Mar 08, 2024**



# CONTENTS

<b>1</b>	<b>Installation</b>	<b>1</b>
<b>2</b>	<b>pemmican-cli</b>	<b>3</b>
<b>3</b>	<b>pemmican-reset</b>	<b>5</b>
<b>4</b>	<b>pemmican-mon</b>	<b>7</b>
<b>5</b>	<b>API</b>	<b>9</b>
<b>6</b>	<b>Development</b>	<b>13</b>
<b>7</b>	<b>Changelog</b>	<b>15</b>
<b>8</b>	<b>License</b>	<b>17</b>
	<b>Python Module Index</b>	<b>19</b>
	<b>Index</b>	<b>21</b>



## INSTALLATION

Pemmican should be installed by default on all Ubuntu for Raspberry Pi images from the 24.04 release (“Noble Numbat”) onwards. However, should you wish to install it manually for whatever reason, there are two primary packages which provide different configurations of the application:

- `pemmican-server` installs the MOTD (“Message of the Day”) plugins only, which check for brownout and power supply negotiation states. As the name suggests, this is intended for usage on server installations which lack any kind of desktop notification service.
- `pemmican-desktop` installs two globally enabled user services which attempt to communicate with an implementation of the freedesktop notification service. One service warns about brownout and power supply negotiation issues (the same as the MOTD service), the other is runtime monitor for overcurrent and undervolt issues.

These both rely on the `pemmican-common` package (which is the actual code for both implementations).

### 1.1 From PyPI

You may also choose to install Pemmican from PyPI:

```
$ pip install "pemmican[gui]"
```

The `[gui]` option should be included only if you want to include the dependencies for the graphical `pemmican-reset` and `pemmican-mon` applications.

Please note that, in this case, you will need to add service definitions to launch the applications yourself. For the graphical applications, the following two service definitions are recommended:

Listing 1: `/usr/lib/systemd/user/pemmican-reset.service`

```
[Unit]
Description=Check the Raspberry Pi PMIC for reset notifications
ConditionPathExists=/proc/device-tree/chosen/power
ConditionPathExists=!%E/pemmican/brownout.inhibit
ConditionPathExists=!%E/pemmican/max_current.inhibit
Requires=dbus.socket
After=dbus.socket

[Service]
Slice=session.slice
Type=oneshot
ExecStart=pemmican-reset
Restart=on-failure
RestartSec=1s

[Install]
WantedBy=graphical-session.target
```

Listing 2: /usr/lib/systemd/user/pemmican-monitor.service

```
[Unit]
Description=Check the Raspberry Pi PMIC for power supply issues
ConditionPathExists=/proc/device-tree/chosen/power
ConditionPathExists=!%E/pemmican/overcurrent.inhibit
ConditionPathExists=!%E/pemmican/undervolt.inhibit
Requires=dbus.socket
After=dbus.socket

[Service]
Slice=session.slice
Type=simple
ExecStart=pemmican-mon
Restart=on-failure
RestartSec=1s

[Install]
WantedBy=graphical-session.target
```

As these are user services, they will either need to be enabled on a per-user basis, or globally like so:

```
$ sudo systemctl --global enable pemmican-reset.service
$ sudo systemctl --global enable pemmican-monitor.service
```

For the console application (**pemmican-cli**) the following `update-motd(5)` script is recommended:

Listing 3: /etc/update-motd.d/90-pemmican

```
#!/bin/sh

if [ -x /usr/bin/pemmican-cli ]; then
    exec /usr/bin/pemmican-cli
elif [ -x /usr/local/bin/pemmican-cli ]; then
    exec /usr/local/bin/pemmican-cli
fi
```

## PEMMICAN-CLI

### 2.1 Synopsis

```
usage: pemmican-cli [-h] [--version]
```

### 2.2 Options

**-h, --help**

show the help message and exit

**--version**

show program's version number and exit

### 2.3 Usage

**pemmican-cli** is intended to be a one-shot operation, typically launched by *update-motd(5)*. It first checks whether the last reset occurred due to a brownout (undervolt) condition and, if it was, prints a warning to stdout.

If you wish to suppress this warning for your user, touch the file `~/.config/pemmican/brownout.inhibit`. If you wish to suppress this warning system-wide, touch the file `/etc/xdg/pemmican/brownout.inhibit`.

**Warning:** It is strongly recommended that any such notice is heeded, as brownout is very likely to lead to any manner of other (hard to predict or replicate) issues up to and including data corruption.

Put simply, suppressing this warning is probably a very bad idea!

If the last reset was normal (or there was no last reset), the script further checks if the power supply negotiated a full 5A feed. If it did not, this also results in a warning printed to stdout.

The Pi 5 can be reliably operated without a 5A feed, provided the peripherals attached to it are relatively light in their power draw. Depending on circumstance, you may well wish to suppress this warning which can be done for your individual user by touching the file `~/.config/pemmican/max_current.inhibit` or system-wide by touching `/etc/xdg/pemmican/max_current.inhibit`.

## **2.4 See Also**

*pemmican-reset* (page 5), *pemmican-mon* (page 7)

## **2.5 Bugs**

Please report bugs at: <https://github.com/waveform80/pemmican/issues>



## PEMMICAN-RESET

### 3.1 Synopsis

```
usage: pemmican-reset [-h] [--help-all] [--help-gapplication]
                    [--gapplication-service]
```

### 3.2 Options

- h, --help**  
show the help message and exit
- help-all**  
show all available help options and exit
- help-gapplication**  
show help options specific to GLib.Application and exit
- gapplication-service**  
run the application in GApplication service mode (for DBus service files)

### 3.3 Usage

**pemmican-reset** is intended to be a one-shot operation, launched as a user-slice systemd service by the “graphical-session” target. Its operation is in essence identical to **pemmican-cli**, but rather than printing to stdout, it reports warnings via the freedesktop [notification service](#)<sup>1</sup>.

It first checks whether the last reset occurred due to a brownout (undervolt) condition and, if it was, sends a critical notification to the notification service.

If you wish to suppress this warning for your user, touch the file `~/.config/pemmican/brownout.inhibit`. If you wish to suppress this warning system-wide, touch the file `/etc/xdg/pemmican/brownout.inhibit`.

**Warning:** It is strongly recommended that any such notice is heeded, as brownout is very likely to lead to any manner of other (hard to predict or replicate) issues up to and including data corruption.

Put simply, suppressing this warning is probably a very bad idea!

If the last reset was normal (or there was no last reset), the script further checks if the power supply negotiated a full 5A feed. If it did not, this also results in a warning printed to stdout.

---

<sup>1</sup> <https://specifications.freedesktop.org/notification-spec/latest/>

The Pi 5 can be reliably operated without a 5A feed, provided the peripherals attached to it are relatively light in their power draw. Depending on circumstance, you may well wish to suppress this warning which can be done for your individual user by touching the file `~/.config/pemmican/max_current.inhibit` or system-wide by touching `/etc/xdg/pemmican/max_current.inhibit`.

### 3.4 Environment

Because the notifications generated by this application are capable of launching a web-browser (via the “More Information” action), the script will also bail with an error in the event that `DISPLAY` or `WAYLAND_DISPLAY` are not found in the environment at startup.

### 3.5 See Also

*pemmican-cli* (page 3), *pemmican-mon* (page 7)

The freedesktop notifications specification<sup>2</sup>.

### 3.6 Bugs

Please report bugs at: <https://github.com/waveform80/pemmican/issues>

---

<sup>2</sup> <https://specifications.freedesktop.org/notification-spec/latest/>

## PEMMICAN-MON

### 4.1 Synopsis

```
usage: pemmican-mon [-h] [--help-all] [--help-gapplication]
                  [--gapplication-service]
```

### 4.2 Options

**-h, --help**

show the help message and exit

**--help-all**

show all available help options and exit

**--help-gapplication**

show help options specific to GLib.Application and exit

**--gapplication-service**

run the application in GApplication service mode (for DBus service files)

### 4.3 Usage

**pemmican-mon** is intended to be a background service, launched as a user-slice systemd service by the “graphical-session” target. It monitors udev for power supply related events, reporting those of interest via the freedesktop notification service<sup>3</sup>.

Power supply undervolt (typically reported in the event the supply drops below 4.8V) is likely the most important event the application warns about. If you wish to suppress this warning for your user, touch the file `~/.config/pemmican/undervolt.inhibit`. If you wish to suppress this warning system-wide, touch the file `/etc/xdg/pemmican/undervolt.inhibit`.

**Warning:** It is strongly recommended that any such notice is heeded, as brownout is very likely to lead to any manner of other (hard to predict or replicate) issues up to and including data corruption.

Put simply, suppressing this warning is probably a very bad idea!

USB over-current is the other event the application will warn about. This occurs when peripherals attached to the USB ports attempt to draw more current than permitted. If the Pi 5 is being supplied by a PSU that did not negotiate a 5A feed at boot time, the USB ports will be limited in their supply and this warning is more likely.

<sup>3</sup> <https://specifications.freedesktop.org/notification-spec/latest/>

If you wish to suppress this warning for your individual user, touch the file `~/.config/pemmican/overcurrent.inhibit`. To suppress this warning system-wide, touch `/etc/xdg/pemmican/overcurrent.inhibit`.

## 4.4 Environment

Because the notifications generated by this application are capable of launching a web-browser (via the “More Information” action), the script will also bail with an error in the event that `DISPLAY` or `WAYLAND_DISPLAY` are not found in the environment at startup.

## 4.5 See Also

*pemmican-cli* (page 3), *pemmican-reset* (page 5)

The freedesktop notifications specification<sup>4</sup>.

## 4.6 Bugs

Please report bugs at: <https://github.com/waveform80/pemmican/issues>

---

<sup>4</sup> <https://specifications.freedesktop.org/notification-spec/latest/>

Pemmican has a very basic API, which is probably not much use outside of the scripts provided with the package. Nonetheless, it is documented in the following modules:

## 5.1 Modules

### 5.1.1 `pemmican.power`

`pemmican.power.reset_brownout ()`

Returns `True`<sup>5</sup> if the device-tree reports that a power brownout (undervolt condition) was the cause of the last reset. Raises `OSError`<sup>6</sup> if the reset condition cannot be queried (e.g. if this is executed on a non-Raspberry Pi).

`pemmican.power.psu_max_current ()`

Returns the maximum current negotiated with the PSU by the power supply in mA. Ideally this should be 5000 (indicating a power supply capable of 5V at 5A), but may be 3000 or lower. Raises `OSError`<sup>7</sup> if the maximum current could not be queried (e.g. if this is executed on a non-Raspberry Pi).

### 5.1.2 `pemmican.gui`

**class** `pemmican.gui.NotifierApplication`

Base class for a GLib `GApplication` which needs to talk to the freedesktop notification service<sup>8</sup>. An instance of this class can be called as a “main” function, optionally passing in the command line parameters.

As a `GApplication` with an identifier (see `APP_ID` (page 9)), only one instance is typically permitted to run. Additional instances will exit before activation, but will signal the original instance to activate instead. The XDG directories, particularly those related to configuration (`XDG_CONFIG_HOME` and `XDG_CONFIG_DIRS`) are expected in the environment.

The application will terminate early with a non-zero exit code if `DISPLAY` or `WAYLAND_DISPLAY` are missing from the environment. Finally, if the freedesktop notification service does not show up within 1 minute of the application starting, the application will also terminate with a non-zero exit code.

This is an abstract class; descendents need to implement the `run ()` (page 10) method.

**APP\_ID**

The application’s identifier, in the typical form of a reverse domain-name. This should be overridden at the class-level in each descendent.

---

<sup>5</sup> <https://docs.python.org/3.12/library/constants.html#True>

<sup>6</sup> <https://docs.python.org/3.12/library/exceptions.html#OSError>

<sup>7</sup> <https://docs.python.org/3.12/library/exceptions.html#OSError>

**do\_activate** (*user\_data*)

Application activation. This starts the GLib main loop; any set up which should be performed before entering the main loop should be done here.

The application's main logic (in the abstract `run()` (page 10) method) is ultimately executed as a one-shot idle handler from the GLib main loop, configured here.

**abstract run** ()

This abstract method should be overridden in descendents to provide the main logic of the application.

**class** `pemmican.gui.ResetApplication`

Checks the Raspberry Pi 5's power status and reports, via the freedesktop notification mechanism, if the last reset occurred due to a brownout (undervolt) situation, or if the current power supply failed to negotiate a 5A supply. This script is intended to be run from a systemd user slice as part of the `graphical-session.target`.

**do\_check** ()

This method is the bulk of the `pemmican-reset` application. It runs the checks on the device-tree nodes and, if notifications are required, queries the notification service's capabilities to format the notifications accordingly.

**do\_notification\_action** (*msg\_id, action\_key*)

Callback executed when the user activates an action on one of our pending notifications. This launches the web-browser for the "More information" action, or touches the appropriate file for the "Don't show again" action.

**do\_notification\_closed** (*msg\_id, reason*)

Callback executed when the user dismisses a notification by any mechanism (explicit close, timeout, action activation, etc). As a oneshot application, which can only ever show one notification, we just quit if it's closed.

**run** ()

This abstract method should be overridden in descendents to provide the main logic of the application.

**class** `pemmican.gui.MonitorApplication`

Monitors the Raspberry Pi 5's power supply for reports of undervolt (deficient power supply), or overcurrent (excessive draw by USB peripherals). Issues are reported via the freedesktop notification mechanism. This script is intended to be run from a systemd user slice as part of the `graphical-session.target`.

**do\_hwmon\_device** (*observer, device*)

Callback registered for hardware monitoring events. This performs further filtering to determine if this is actually an undervolt event, and dispatches a notification if it is.

**do\_notification\_action** (*msg\_id, action\_key*)

Callback executed when the user activates an action on one of our pending notifications. This launches the web-browser for the "More information" action, or touches the appropriate file for the "Don't show again" action.

**do\_notification\_closed** (*msg\_id, reason*)

Callback executed when the user dismisses a notification by any mechanism (explicit close, timeout, action activation, etc).

**do\_usb\_device** (*observer, device*)

Callback registered for USB device events. This method performs further filtering to determine if this is actually an overcurrent event, and dispatches a notification if it is.

**notify** (*key, msg, \*, replaces\_id=0*)

This method is called by the monitoring callbacks (`do_usb_device()` (page 10) and `do_hwmon_device()` (page 10)) to format and dispatch a notification according to the capabilities of the system's notification mechanism.

---

<sup>8</sup> <https://specifications.freedesktop.org/notification-spec/>

**run** ()

This abstract method should be overridden in descendents to provide the main logic of the application.

### 5.1.3 pemmican.cli

`pemmican.cli.main` (*args=None*)

The entry-point for the **pemmican-cli** application. Takes the command line *args* as its only parameter and returns the exit code of the application.





## DEVELOPMENT

The main GitHub repository for the project can be found at:

<https://github.com/waveform80/pemmican>

### 6.1 Development installation

If you wish to develop pemmican, obtain the source by cloning the GitHub repository and then use the “develop” target of the Makefile which will install the package as a link to the cloned repository allowing in-place development. The following example demonstrates this method within a virtual Python environment:

```
$ sudo apt install build-essential git virtualenvwrapper
$ sudo apt build-dep python3-gi
$ sudo apt build-dep python3-dbus
$ sudo apt build-dep python3-pyudev
```

After installing `virtualenvwrapper` you’ll need to restart your shell before commands like `mkvirtualenv` will operate correctly. Once you’ve restarted your shell, continue:

```
$ cd
$ mkvirtualenv pemmican
$ workon pemmican
(pemmican) $ git clone https://github.com/waveform80/pemmican.git
(pemmican) $ cd pemmican
(pemmican) $ make develop
```

To pull the latest changes from git into your clone and update your installation:

```
$ workon pemmican
(pemmican) $ cd ~/pemmican
(pemmican) $ git pull
(pemmican) $ make develop
```

To remove your installation, destroy the sandbox and the clone:

```
(pemmican) $ deactivate
$ rmvirtualenv pemmican
$ rm -rf ~/pemmican
```

## 6.2 Building the docs

If you wish to build the docs, you'll need a few more dependencies. Inkscape is used for conversion of SVGs to other formats, Graphviz is used for rendering certain charts, and TeX Live is required for building PDF output. The following command should install all required dependencies:

```
$ sudo apt install texlive-latex-recommended texlive-latex-extra \
    texlive-fonts-recommended texlive-xetex graphviz inkscape \
    python3-sphinx python3-sphinx-rtd-theme latexmk xindy
```

Once these are installed, you can use the “doc” target to build the documentation in all supported formats (HTML, ePub, and PDF):

```
$ workon pemmican
(pemmican) $ cd ~/pemmican
(pemmican) $ make doc
```

However, the easiest way to develop the documentation is with the “preview” target which will build the HTML version of the docs, and start a web-server to preview the output. The web-server will then watch for source changes (in both the documentation source, and the application’s source) and rebuild the HTML automatically as required:

```
$ workon pemmican
(pemmican) $ cd ~/pemmican
(pemmican) $ make preview
```

The HTML output is written to `build/html` while the PDF output goes to `build/latex`.

## 6.3 Test suite

If you wish to run the pemmican test suite, follow the instructions in *Development installation* (page 13) above and then make the “test” target within the sandbox:

```
$ workon pemmican
(pemmican) $ cd ~/pemmican
(pemmican) $ make test
```

The test suite is also setup for usage with the `tox` utility, in which case it will attempt to execute the test suite with all supported versions of Python. If you are developing under Ubuntu you may wish to look into the [Dead Snakes PPA<sup>9</sup>](#) in order to install old/new versions of Python; the `tox` setup *should* work with the version of `tox` shipped with Ubuntu Focal, but more features (like parallel test execution) are available with later versions.

For example, to execute the test suite under `tox`:

```
$ tox
```

To execute the test suite under all installed interpreter versions in parallel, using as many parallel tasks as there are CPUs, then displaying a combined report of coverage from all environments:

```
$ tox -p auto
$ coverage combine
$ coverage report
```

---

<sup>9</sup> <https://launchpad.net/~deadsnakes/%2Barchive/ubuntu/ppa>

## CHANGELOG

### 7.1 Release 1.0.3 (2024-03-08)

- Moved translation stuff into the standard locations

### 7.2 Release 1.0.2 (2024-02-23)

- Re-license under GPL3 only (not later), according to policy

### 7.3 Release 1.0.1 (2024-02-20)

- Mostly documentation changes (PyPI installation, development installation, and minor corrections)

### 7.4 Release 1.0 (2024-02-19)

- Initial release



**LICENSE**

This file is part of pemmican.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License version 3, as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but **WITHOUT ANY WARRANTY**; without even the implied warranty of **MERCHANTABILITY** or **FITNESS FOR A PARTICULAR PURPOSE**. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.



## PYTHON MODULE INDEX

### p

pemmican.cli, 11  
pemmican.gui, 9  
pemmican.power, 9





## Symbols

- gapplication-service
    - pemmican-mon command line option, 7
    - pemmican-reset command line option, 5
  - help
    - pemmican-cli command line option, 3
    - pemmican-mon command line option, 7
    - pemmican-reset command line option, 5
  - help-all
    - pemmican-mon command line option, 7
    - pemmican-reset command line option, 5
  - help-gapplication
    - pemmican-mon command line option, 7
    - pemmican-reset command line option, 5
  - version
    - pemmican-cli command line option, 3
  - h
    - pemmican-cli command line option, 3
    - pemmican-mon command line option, 7
    - pemmican-reset command line option, 5
- A**
- APP\_ID (*pemmican.gui.NotifierApplication* attribute), 9
- D**
- DISPLAY, 9
  - do\_activate() (*pemmican.gui.NotifierApplication* method), 9
  - do\_check() (*pemmican.gui.ResetApplication* method), 10
  - do\_hwmon\_device() (*pemmican.gui.MonitorApplication* method), 10
  - do\_notification\_action() (*pemmican.gui.MonitorApplication* method), 10
  - do\_notification\_action() (*pemmican.gui.ResetApplication* method), 10
  - do\_notification\_closed() (*pemmican.gui.MonitorApplication* method), 10
  - do\_notification\_closed() (*pemmican.gui.ResetApplication* method), 10
- E**
- do\_usb\_device() (*pemmican.gui.MonitorApplication* method), 10
  - environment variable
    - DISPLAY, 9
    - WAYLAND\_DISPLAY, 9
    - XDG\_CONFIG\_DIRS, 9
    - XDG\_CONFIG\_HOME, 9
- M**
- main() (*in module pemmican.cli*), 11
  - module
    - pemmican.cli, 11
    - pemmican.gui, 9
    - pemmican.power, 9
  - MonitorApplication (*class in pemmican.gui*), 10
- N**
- NotifierApplication (*class in pemmican.gui*), 9
  - notify() (*pemmican.gui.MonitorApplication* method), 10
- P**
- pemmican.cli
    - module, 11
  - pemmican.gui
    - module, 9
  - pemmican.power
    - module, 9
  - pemmican-cli command line option
    - help, 3
    - version, 3
    - h, 3
  - pemmican-mon command line option
    - gapplication-service, 7
    - help, 7
    - help-all, 7
    - help-gapplication, 7
    - h, 7
  - pemmican-reset command line option
    - gapplication-service, 5
    - help, 5
    - help-all, 5
    - help-gapplication, 5
    - h, 5

`psu_max_current()` (*in module `pemmican.power`*),  
9

## R

`reset_brownout()` (*in module `pemmican.power`*), 9

`ResetApplication` (*class in `pemmican.gui`*), 10

`run()` (*`pemmican.gui.MonitorApplication` method*), 10

`run()` (*`pemmican.gui.NotifierApplication` method*), 10

`run()` (*`pemmican.gui.ResetApplication` method*), 10

## W

`WAYLAND_DISPLAY`, 9

## X

`XDG_CONFIG_DIRS`, 9

`XDG_CONFIG_HOME`, 9