

NAME

apmd - Advanced Power Management monitor daemon

SYNOPSIS

apmd [-d] [-f *file*] [-s] [-v]

DESCRIPTION

The **apmd** utility monitors the occurrence of the specified Advanced Power Management (APM) events and, if one of the events occurs, it executes the sequence of commands corresponding to the event. Only the events specified in the configuration file are notified to **apmd**; all other events are ignored. For each event posted by the APM BIOS, **apmd** invokes the sequence of commands specified in the configuration file. When **apmd** is running with monitoring suspend/standby requests, the kernel will not process those requests. Therefore, if you wish action to be taken when these events occur, you need to explicitly configure the appropriate commands or built-in functions in the configuration file.

The **apmd** utility recognizes the following runtime options:

-d Starts in debug mode. This causes **apmd** to execute in the foreground instead of in daemon mode.

-f *file*

Specifies a different configuration file *file* to be used in place of the default */etc/apmd.conf*.

-s Causes **apmd** to simulate a POWERSTATECHANGE event when a power state change is detected (AC_POWER_STATE) but the bios of the laptop does not report it. This enables you to do things like dimming the LCD backlight when you unplug the power cord.

-v Verbose mode.

When **apmd** starts, it reads the configuration file (*/etc/apmd.conf* as default) and notifies the set of events to be monitored to the APM device driver. When it terminates, the APM device driver automatically cancels monitored events.

If the **apmd** process receives a SIGHUP, it will reread its configuration file and notify the APM device driver of any changes to its configuration.

The **apmd** utility uses the device */dev/apmctl* to issue ioctl(2) requests for monitoring events and for controlling the APM system. This device file is opened exclusively, so only a single **apmd** process can be running at any time.

When **apmd** receives an APM event, it forks a child process to execute the commands specified in the

configuration file and then continues listening for more events. The child process executes the commands specified, one at a time and in the order that they are listed.

While **apmd** is processing the command list for SUSPEND/STANDBY requests, the APM kernel device driver issues notifications to APM BIOS once per second so that the BIOS knows that there are still some commands pending, and that it should not complete the request just yet.

The **apmd** utility creates the file */var/run/apmd.pid*, and stores its process id there. This can be used to kill or reconfigure **apmd**.

CONFIGURATION FILE

The structure of the **apmd** configuration file is quite simple. For example:

```
apm_event SUSPENDREQ {
    exec "sync && sync && sync";
    exec "sleep 1";
    exec "zzz";
}
```

will cause **apmd** to receive the APM event 'SUSPENDREQ' (which may be posted by an LCD close), run the 'sync' command 3 times and wait for a while, then execute **zzz** (**apm -z**) to put the system in the suspend state.

• The apm_event keyword

'apm_event' is the keyword which indicates the start of configuration for each event.

• APM events

If you wish to execute the same commands for different events, the event names should be delimited by a comma. The following are valid event names:

- Events ignored by the kernel if **apmd** is running:

STANDBYREQ	
USERSTANDBYREQ	
SUSPENDREQ	should include sync in the command list,
USERSUSPENDREQ	should include sync in the command list,
BATTERYLOW	only <i>zzz</i> should be specified in the command list.

- Events passed to **apmd** after kernel handling:

```
NORMRESUME
CRITRESUME
STANDBYRESUME
POWERSTATECHANGE
UPDATETIME
CAPABILITIESCHANGE
```

Other events will not be sent to **apmd**.

• command line syntax

In the example above, the three lines beginning with 'exec' are commands for the event. Each line should be terminated with a semicolon. The command list for the event should be enclosed by '{' and '}'. The **apmd** utility uses */bin/sh* for double-quotation enclosed command execution, just as with `system(3)`. Each command is executed in order until the end of the list is reached or a command finishes with a non-zero status code. The **apmd** utility will report any failed command's status code via `syslog(3)` and will then reject the request event posted by the APM BIOS.

• Built-in functions

You can also specify **apmd** built-in functions instead of command lines. A built-in function name should be terminated with a semicolon, just as with a command line. The following built-in functions are currently supported:

- reject Reject last request posted by APM BIOS. This can be used to reject a SUSPEND request when the LCD is closed and put the system in a STANDBY state instead.

FILES

```
/etc/apmd.conf
/dev/apmctl
/var/run/apmd.pid
```

EXAMPLES

Sample configuration commands include:

```
apm_event SUSPENDREQ {
```

```
    exec "/etc/rc.suspend apm suspend";
}

apm_event USERSUSPENDREQ {
    exec "sync && sync && sync";
    exec "sleep 1";
    exec "apm -z";
}

apm_event NORMRESUME {
    exec "/etc/rc.resume apm suspend";
}

apm_event STANDBYRESUME {
    exec "/etc/rc.resume apm standby";
}

# resume event configuration for serial mouse users by
# reinitializing a moused(8) connected to a serial port.
#
#apm_event NORMRESUME {
#    exec "kill -HUP `cat /var/run/moused.pid`";
#}
#
# suspend request event configuration for ATA HDD users:
# execute standby instead of suspend.
#
#apm_event SUSPENDREQ {
#    reject;
#    exec "sync && sync && sync";
#    exec "sleep 1";
#    exec "apm -Z";
#}
```

SEE ALSO

apm(4), apm(8)

HISTORY

The **apmd** utility appeared in FreeBSD 3.3.

AUTHORS

Mitsuru IWASAKI <*iwasaki@FreeBSD.org*>

KOIE Hidetaka <*koie@suri.co.jp*>

Some contributions made by Warner Losh <*imp@FreeBSD.org*>, Hiroshi Yamashita <*bluemoon@msj.biglobe.ne.jp*>, Yoshihiko SARUMARU <*mistral@imasy.or.jp*>, Norihiro Kumagai <*kuma@nk.rim.or.jp*>, NAKAGAWA Yoshihisa <*nakagawa@jp.FreeBSD.org*>, and Nick Hilliard <*nick@foobar.org*>.