

NAME

periodic - run periodic system functions

SYNOPSIS

periodic **daily|weekly|monthly|security|directory** ...

DESCRIPTION

The **periodic** utility is intended to be called by cron(8) to execute shell scripts located in the specified directory.

One or more of the following arguments must be specified:

daily Perform the standard daily periodic executable run. This usually occurs early in the morning (local time).

weekly Perform the standard weekly periodic executable run. This usually occurs very early on Saturday mornings.

monthly Perform the standard monthly periodic executable run. This usually occurs on the first day of the month.

security Perform the standard daily security checks. This is usually spawned by the **daily** run.

directory An arbitrary directory containing a set of executables to be run.

If an argument is an absolute directory name it is used as is, otherwise it is searched for under */etc/periodic* and any other directories specified by the *local_periodic* setting in *periodic.conf(5)* (see below).

The **periodic** utility will run each executable file in the directory or directories specified. If a file does not have the executable bit set, it is silently ignored.

Each script is required to exit with one of the following values:

- 0 The script has produced nothing notable in its output. The *<basedir>_show_success* variable controls the masking of this output.
- 1 The script has produced some notable information in its output. The *<basedir>_show_info* variable controls the masking of this output.

2 The script has produced some warnings due to invalid configuration settings. The `<basedir>_show_badconfig` variable controls the masking of this output.

>2 The script has produced output that must not be masked.

If the relevant variable (where `<basedir>` is the base directory in which the script resides) is set to "NO" in `periodic.conf`, **periodic** will mask the script output. If the variable is not set to either "YES" or "NO", it will be given a default value as described in `periodic.conf(5)`.

All remaining script output is delivered based on the value of the `<basedir>_output` setting.

If this is set to a path name (beginning with a `'` character), output is simply logged to that file. `newsyslog(8)` knows about the files `/var/log/daily.log`, `/var/log/weekly.log` and `/var/log/monthly.log`, and if they exist, it will rotate them at the appropriate times. These are therefore good values if you wish to log **periodic** output.

If the `<basedir>_output` value does not begin with a `'` and is not empty, it is assumed to contain a list of email addresses, and the output is mailed to them. If `<basedir>_show_empty_output` is set to "NO", then no mail will be sent if the output was empty.

If `<basedir>_output` is not set or is empty, output is sent to standard output.

ENVIRONMENT

The **periodic** utility sets the PATH environment to include all standard system directories, but no additional directories, such as `/usr/local/bin`. If executables are added which depend upon other path components, each executable must be responsible for configuring its own appropriate environment.

FILES

`/etc/crontab` the **periodic** utility is typically called via entries in the system default `cron(8)` table

`/etc/periodic` the top level directory containing `daily`, `weekly`, `monthly`, and `security` subdirectories which contain standard system periodic executables

`/etc/defaults/periodic.conf` the `periodic.conf` system registry contains variables that control the behaviour of **periodic** and the standard `daily`, `weekly`, `monthly`, and `security` scripts

`/etc/periodic.conf`, `${LOCALBASE}/etc/periodic.conf`
this file contains local overrides for the default **periodic** configuration

EXIT STATUS

Exit status is 0 on success and 1 if the command fails.

EXAMPLES

The system crontab should have entries for **periodic** similar to the following example:

```
# do daily/weekly/monthly maintenance
0 2 * * * root periodic daily
0 3 * * 6 root periodic weekly
0 5 1 * * root periodic monthly
```

The `/etc/defaults/periodic.conf` system registry will typically have a `local_periodic` variable reading:

```
local_periodic="${_localbase}/etc/periodic"
```

where `$_localbase` is being set from within `/usr/sbin/periodic`.

To log **periodic** output instead of receiving it as email, add the following lines to `/etc/periodic.conf`:

```
daily_output=/var/log/daily.log
weekly_output=/var/log/weekly.log
monthly_output=/var/log/monthly.log
```

To only see important information from daily periodic jobs, add the following lines to `/etc/periodic.conf`:

```
daily_show_success=NO
daily_show_info=NO
daily_show_badconfig=NO
```

DIAGNOSTICS

The command may fail for one of the following reasons:

usage: periodic <directory of files to execute> No directory path argument was passed to **periodic** to specify where the script fragments reside.

<directory> not found Self explanatory.

SEE ALSO

sh(1), crontab(5), periodic.conf(5), cron(8), newsyslog(8)

HISTORY

The **periodic** utility first appeared in FreeBSD 3.0.

AUTHORS

Paul Traina <*pst@FreeBSD.org*>

Brian Somers <*brian@Awfulhak.org*>

BUGS

Since one specifies information about a directory using shell variables containing the string, *<basedir>*, *<basedir>* must only contain characters that are valid within a sh(1) variable name, alphanumeric and underscores, and the first character may not be numeric.