NAME

ratrun — periodic reminder runner

SYNOPSIS

/usr/libexec/ratrun [**-n**]
/usr/libexec/ratrun -a

DESCRIPTION

Sends reminders for **Events** to the calling user for files in `~/.ratrun/`, then moves expired events to `old`. With **-n**, doesn't, and lists what would be sent to the standard output stream.

With **-a**, runs **ratrun** for all configured users: those limited to `RATRUN_GROUPS`, plus the individually-named `RATRUN_USERS`, who have a `.ratrun` directory in their home directory. If any errors resulted, users are additionally mailed with a summary;

No user-specified code is ever run, and no root mail is generated (unless, of course, root schedules a reminder): **ratrun** is essentially just a way for users to schedule mail delivery to themselves.

Events

Have a very simple format:

- the first line is the event date in **date -r** format,
- the second line is the per-event reminder override (optional, and taken as part of the body if doesn't start with a digit),
- the remainder of the file is the event body, sent verbatim in the reminder.

An event is said to be expired if it doesn't have any more **Reminders** left.

Reminders

Start with a digit, are decimal integers, and optionally end with a recursively-expanded multiplicative suffix:

mo = 4**wk**
wk = 7**d**
yr = 365**d**
d = 24**h**
h = 60**m**
m = 60
s = 1

in this order, i.e. 10**h** = 600**m** = 36000 = 36000**s**.

They correspond to the minimal time, in seconds, to send a reminder before the event time — i.e. for reminders "3600 60 0" (equiv. "1**h** 1**m** 0"), a mail will be sent no sooner than an hour, a minute, and the time of the event.

Notification format

For each reminder that expires, users will receive mail with a subject of

`ratrun: in {`*`latest-expired reminder`*`} / on {`*`HH:MM`*`} ({`*`raw event date`*`}):`
`{`*`event filename`*`}`

and body of the rest of the event. The reminder time is folded, in reverse, per the suffix table above.

Additionally, users may receive messages with a subject of

`Errors for your ratrun at ...`

from root. These are produced by **-a** runs, and contain the standard error and output streams from the user's run.

ENVIRONMENT

`RATRUN_REMINDERS` System-wide default list of reminders. Overriden per-user with `.reminders` and per-event with the second line.

`RATRUN_GROUPS`	If non-empty, in **`-a`** mode, only check `.ratrun` presence for the specified groups. Field-split.
`RATRUN_USERS`	In **`-a`** mode, only check `.ratrun` presence for the specified users, or add them to the result from `RATRUN_GROUPS`. Empty means no limit. Field-split.

FILES

`/etc/default/ratrun`	Sourced at the top.
`~/.ratrun/`	Contains **Events**. All directories and hidden files are ignored.
`~/.ratrun/old/`	Expired events are moved here. If one already existed, its new name is appended with its date.
`~/.ratrun/.reminders`	Overrides `RATRUN_REMINDERS` per-user, if present. One-line. Field-split.
`~/.ratrun/.tz`	If present, value exported as `TZ` per-user (cf. `tzset(3)`). One-line.
`~/.ratrun/.prefix`	Changes the mail subject prefix from "`ratrun:`"; "🐇" and empty are popular choices. One-line.
`~/.ratrun/.expcnt/`	Contains counts for expired reminders for each event. If you want to be re-notified about something on the next run, and before the next reminder, remove the file corresponding to the event from this directory,

EXAMPLES

```
$ cat > .ratrun/call-robert
12:00
1h 0
bring up sales for q4
^D

$ echo 18:30 > .ratrun/'meet henry in 201'
$ { echo 2022-12-12T23:11; date; uptime; } > .ratrun/1yr-uptime
```

Assuming a default reminder time of "30**m**", the first reminders from each of these will, respectively, produce the following messages:

```
Date: Mon, 31 Oct 2022 11:00:42 +0100
Subject: ratrun: in 1h / on 12:00 (12:00): call-robert

bring up sales for q4
```

and

```
Date: Mon, 31 Oct 2022 18:00:21 +0100
Subject: ratrun: in 30m / on 18:30 (18:30): meet henry in 201
```

and

```
Date: Mon, 12 Dec 2022 22:41:12 +0100
Subject: ratrun: in 30m / on 23:11 (2022-12-12T23:11): 1yr-uptime

Mon 31 Oct 21:00:54 CET 2022
 21:00:54 up 322 days, 12:09,  4 users,  load average: 1.28, 0.69, 0.52
```

SEE ALSO

`date(1)`, `rat(1)`

It is safe to run multiple instances of **`ratrun`** for any given user at any given time, for example via system and per-user `crontab(5)`s.

🐇