

Если вы видите что-то необычное, просто сообщите мне.

Store the output of cron jobs in systemd's journal

The output from cronjobs is often unceremoniously redirected to `/dev/null` just to keep them quiet. This is a bad practice as you don't get any feedback when potentially important tasks go wrong. (If the task isn't important, why are you running it on a schedule?)

systems provide a simple tool called `systemd-cat` for directing `stdout` and `stderr` to the journal. Below is a complete example of a simple monthly `cron` job that will run certbot once every month and save whatever it outputs to the journal under the arbitrary identifier "certbot-cron". A good identifier will make it easier to look up the output in the journal if you need it later.

```
@monthly systemd-cat -t "certbot-cron" /usr/bin/certbot --renew
```

I'll use the automated TLS certificate renewal program `certbot` from Let's Encrypt as the example command throughout this article. It serves nicely as an example of something you need to know about if it at some point will return an unsuccessful condition.

You can then quickly look at certbot output in the journal using `journalctl` with the same identifier that you gave the `cron` job:

```
journalctl -t "certbot-cron"
```

`systemd-cat` only accepts one executable with any number of arguments. You can't run multiple commands at once or use pipes. It will capture `stdout` and `stderr`, and assign them appropriate properties in the journal.

You can pipe output into `systemd-cat`, but this will only capture stdout. Important and actionable information will often show up in stderr, and you'd have to redirect the flow of `stderr` to `stdout` to capture it. I'll not demonstrate this here, as I believe this is the wrong approach.

If you need to run more complex cron jobs with multiple commands, you should save them to a script, make that executable, and then run that script through `systemd-cat` as a single command. This will capture `stdout` as well as `stderr`, and keep your crontab more readable and easier to maintain.

Revision #1

Created 21 November 2023 19:07:36 by gasick

Updated 21 November 2023 19:10:09 by gasick