

# Gotchas of Everyday systemd

BECAUSE EVEN MAINTAINERS DON'T KNOW EVERYTHING

Anita Zhang  
engineer manager  
“Linux Umbrella” family of teams



# Agenda

01 Background

02 Failures Failing

03 Tricky Timers

04 Miscellaneous Musings

05 Tiny Takeaway

# 01 Background

# Previous SCaLE Presentations

- Solo Talks
  - [\[SCaLE 19x\] The Curious Case of Memory Growth](#)
  - [\[SCaLE 21x\] Adventures of Linux Userspace at Meta](#)
- With [Alvaro Leiva Geisse](#)
  - [\[SCaLE 19x\] Journey into the heart of systemd](#)
  - [\[SCaLE 20x\] Workshop: Guided journey into the heart of systemd](#)
  - [\[SCaLE 21x\] Workshop: A beginners guide into the heart of systemd](#)
  - [\[SCaLE 21x\] Workshop An advanced guide into the heart of systemd](#)

# Linux Umbrella

Linux  
Userspace

OpenBMC

Accelerator  
Userspace

# Before Management...

- Started at Meta 2016 as Software Engineer.
  - Witnessed the CentOS 6 to 7 upgrade.
  - First systemd adoption.

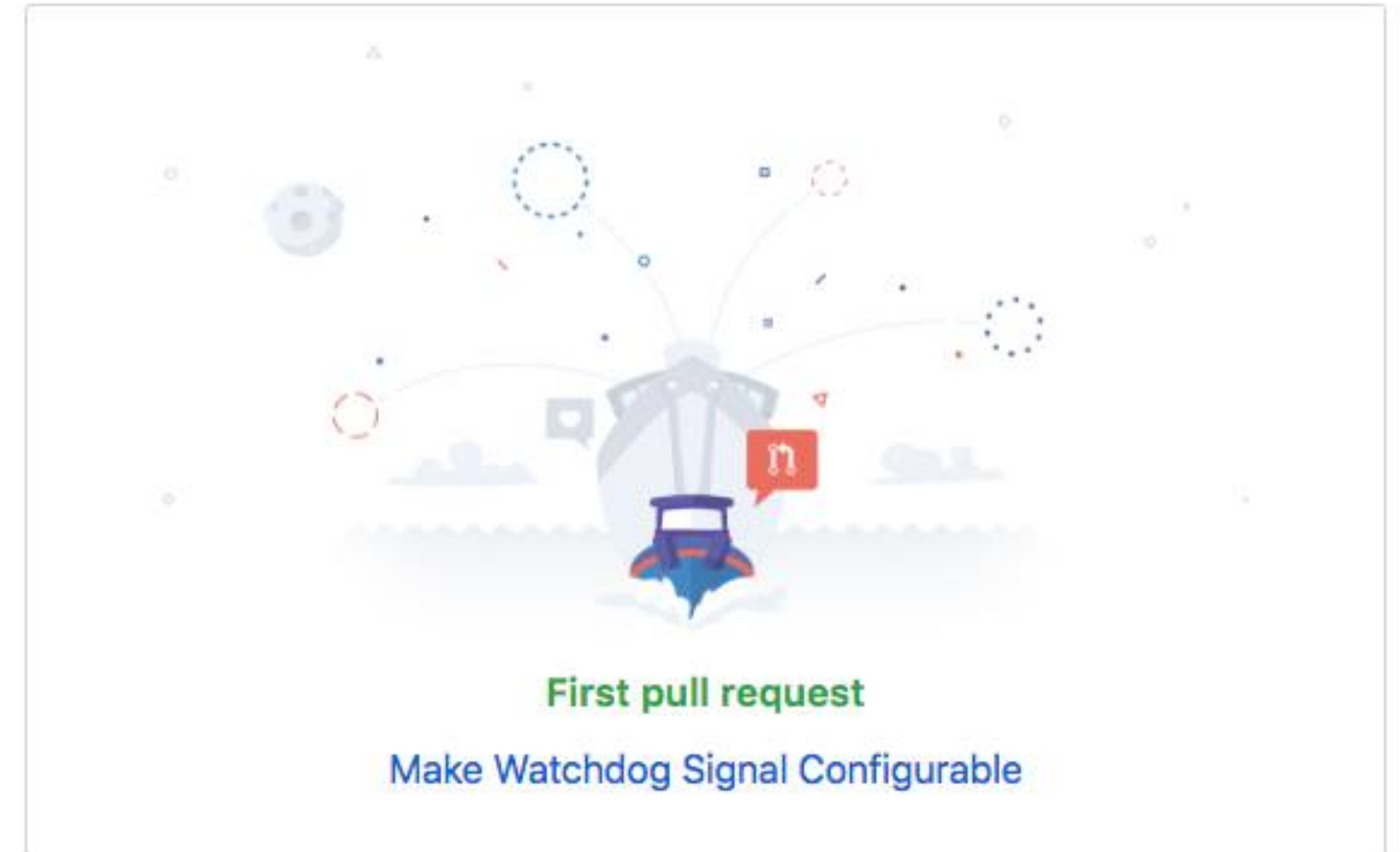
# Before Management...

- First pull request to systemd late 2018.
- Started contributing more in 2019 after a team switch.
  - Involved in every systemd release at Meta since systemd 242 (now releasing 257).
  - Became a systemd upstream maintainer later that year!



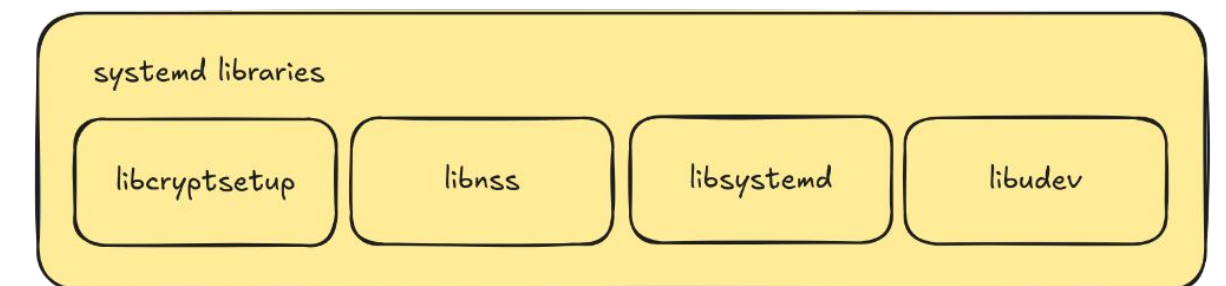
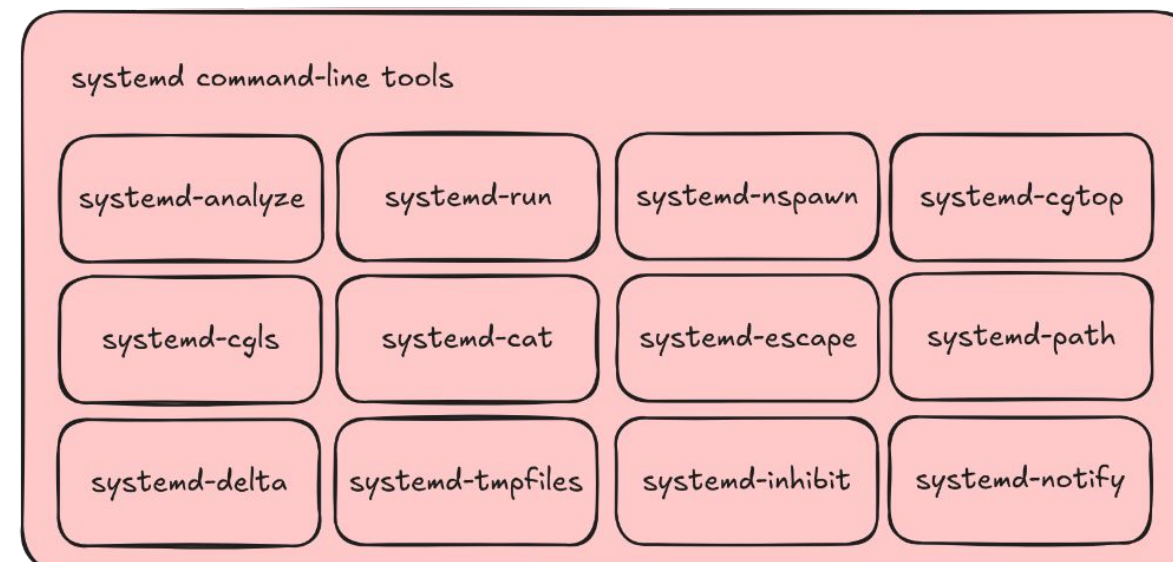
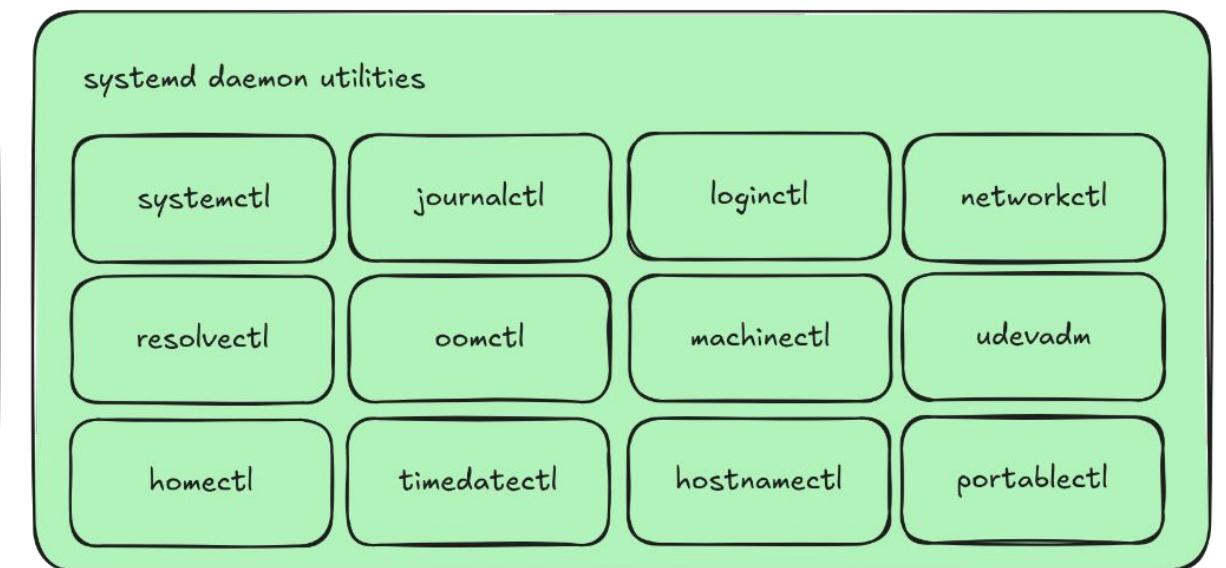
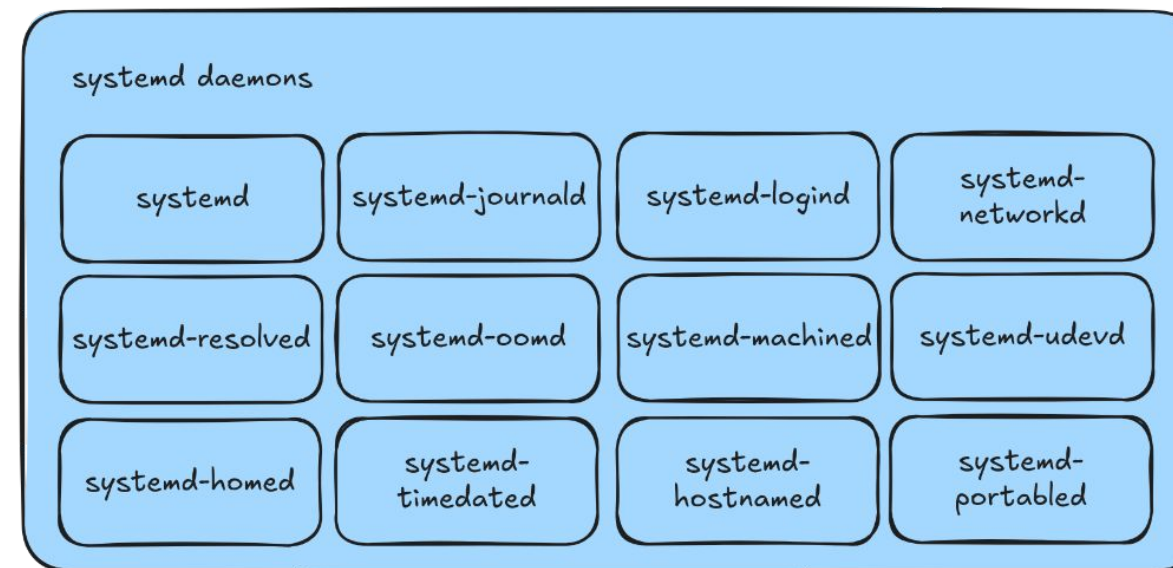
Opened their first pull request on GitHub in systemd/systemd

Sep 21

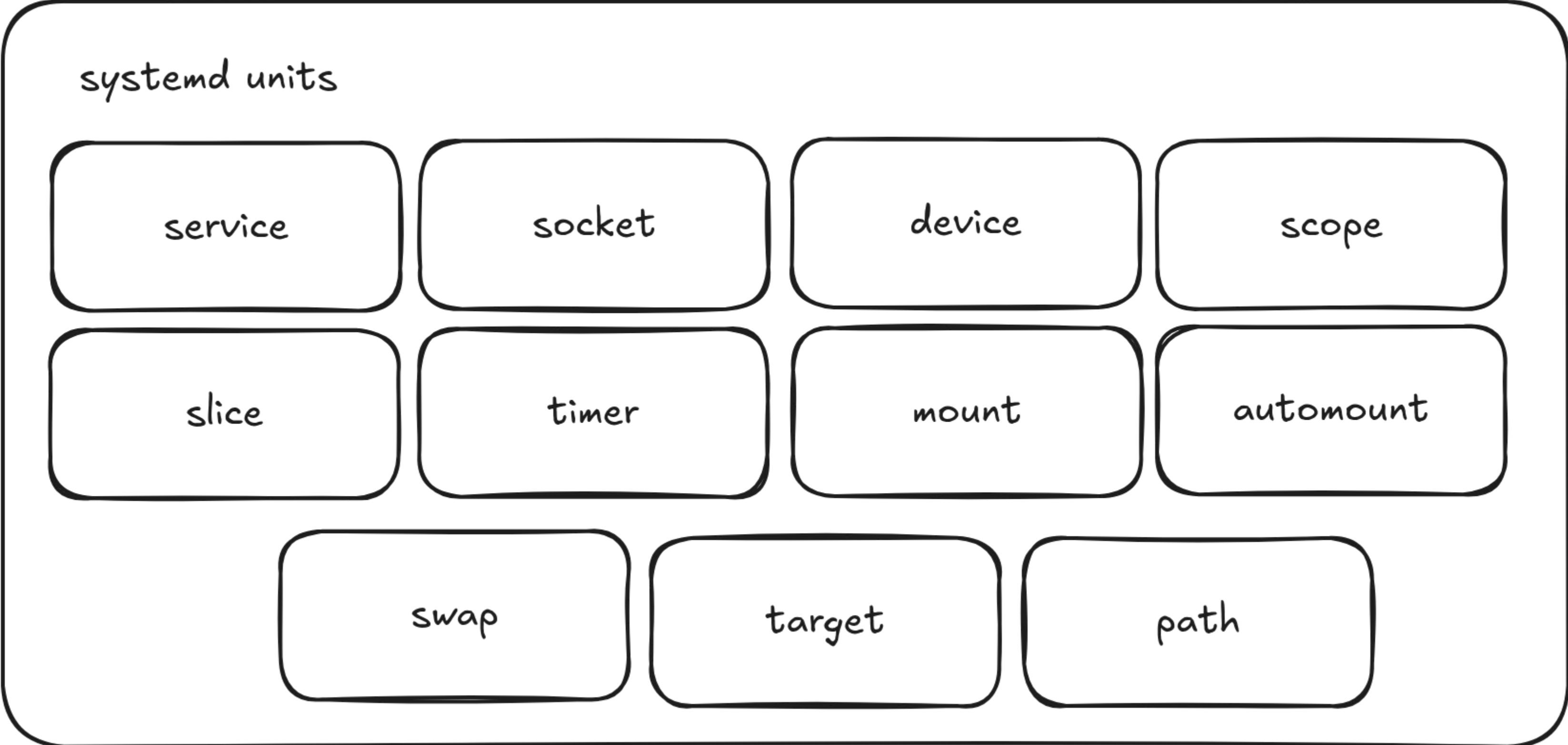


# systemd is...

- PID 1.
- Host-level service manager.
- “Array of system components for Linux”.
  - [Wikipedia](#).
- Core dependency for major Linux distributions.







# example.service Unit File

```
[Unit]
```

```
Description=SCaLE 22x Presentation Example
```

```
[Service]
```

```
ExecStart=/usr/bin/echo Hello
```

```
[Install]
```

```
WantedBy=multi-user.target
```

# systemd is Not Perfect

- Some things are only intuitive because I worked on the code.
- Some things still trip me up.
- Some things still make me facepalm.
- You are not alone!

# 02 Failures Failing

WHAT DO YOU THINK THIS DOES?

[Service]

Type=Simple

TYPE=SIMPLE

If set to **simple** (the default if ExecStart= is specified but neither Type= nor BusName= are), the service manager will consider the unit started immediately after the main service process has been **forked off**.

— [systemd.service man page](#)

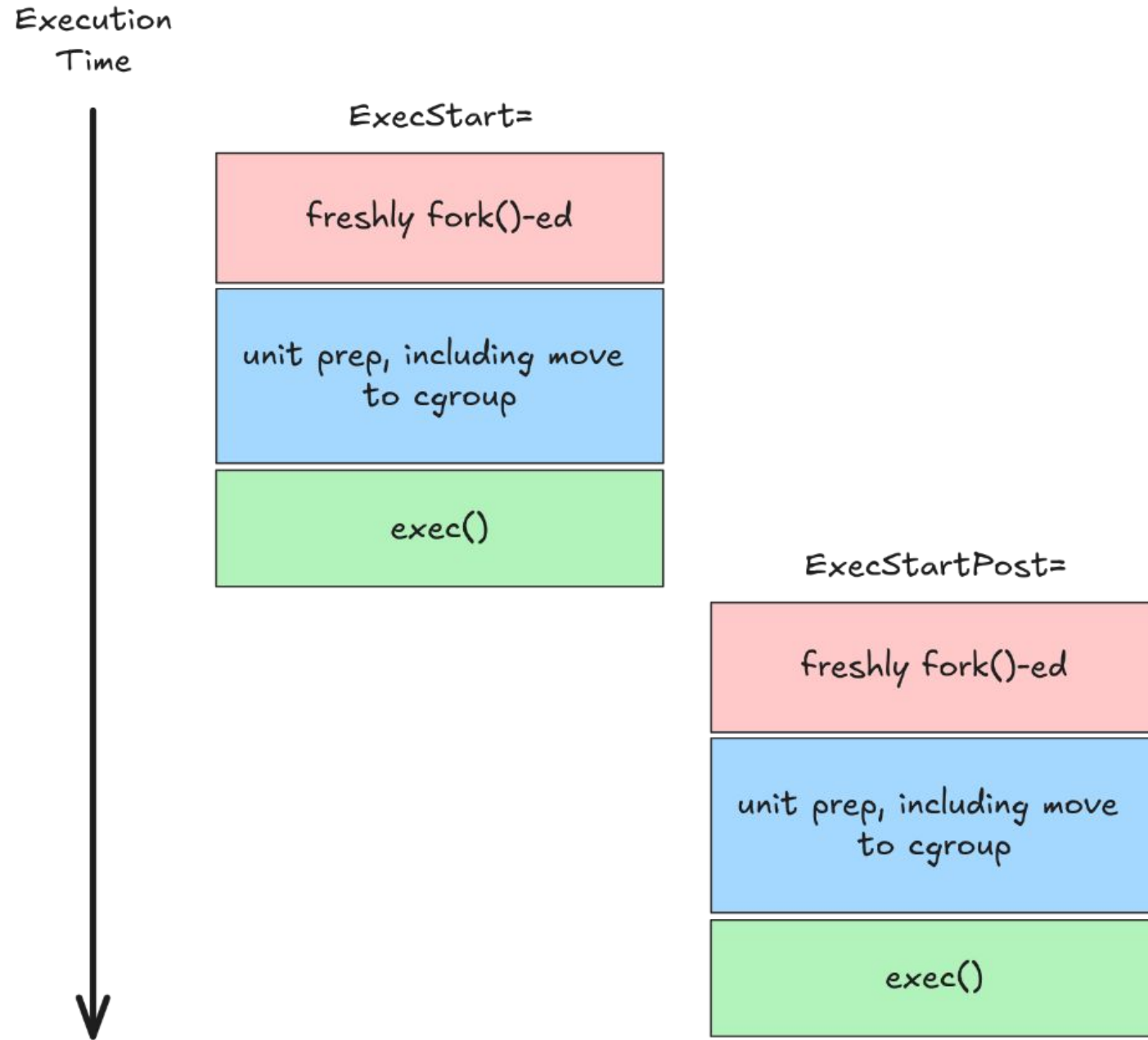
RACY BEHAVIOR

[Service]

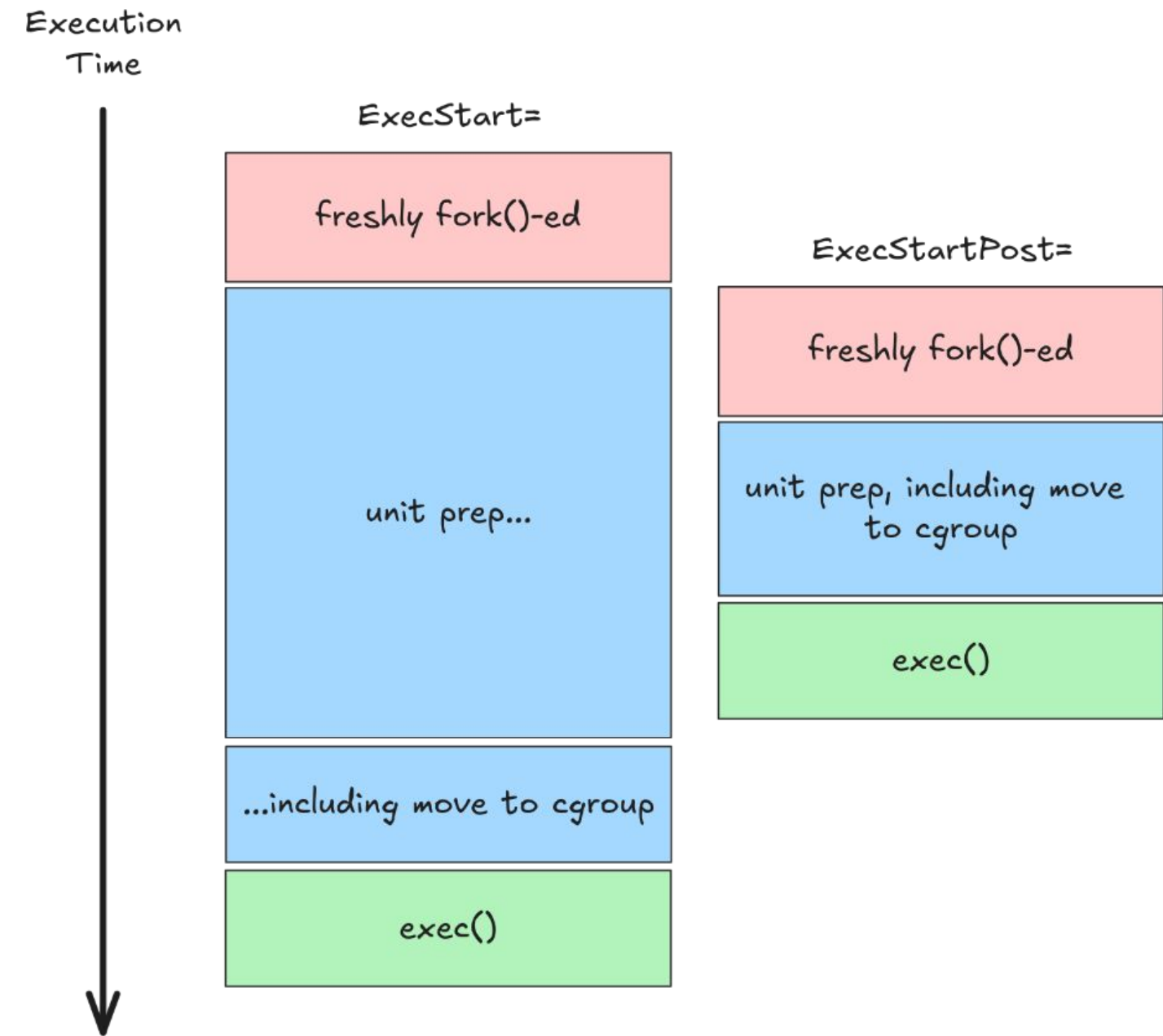
ExecStart=sleep infinity

ExecStartPost=cat /proc/\$MAINPID/cgroup

### Expectation



### Reality





TYPE=SIMPLE

Typically, Type=**exec** is the better choice...The exec type is similar to simple, but the service manager will consider the unit started immediately **after the main service binary has been executed.**

— [systemd.service man page v253](#)

# food.service

```
[Service]
```

```
ExecStart=/bin/false
```

```
Type=exec
```

# dog.service

```
[Unit]
```

```
Requires=food.service
```

```
After=food.service
```

```
[Service]
```

```
ExecStart=echo "I shouldn't run"
```

# food.service

```
[Service]
```

```
ExecStart=/bin/false
```

```
Type=notify
```

# dog.service

```
[Unit]
```

```
Requires=food.service
```

```
After=food.service
```

```
[Service]
```

```
ExecStart=echo "I shouldn't run"
```

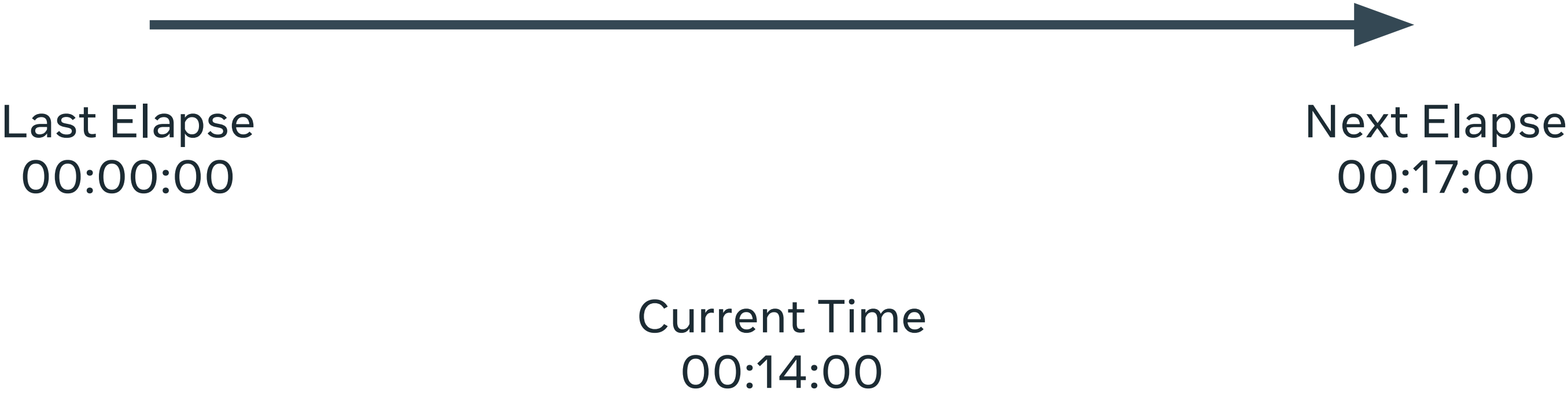
# 03 Tricky Timers

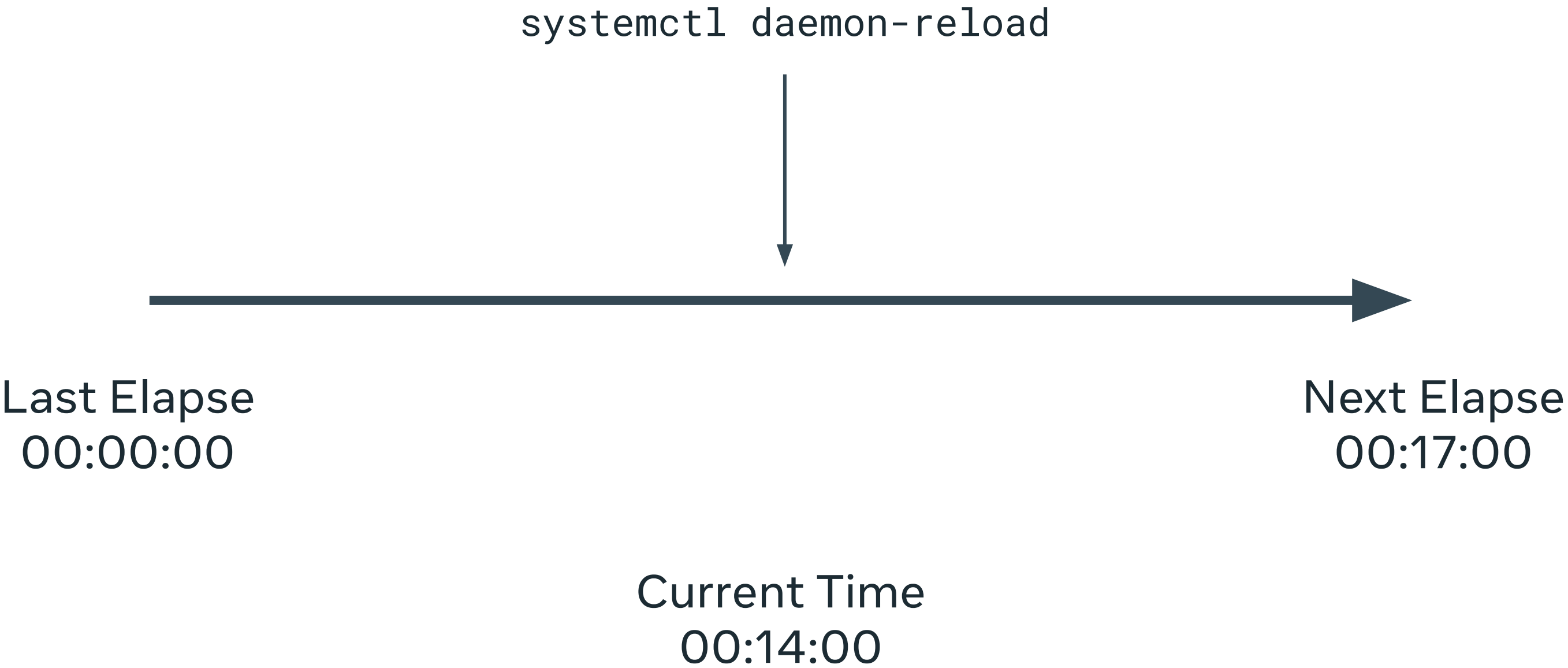
WHAT DO YOU THINK THIS DOES?

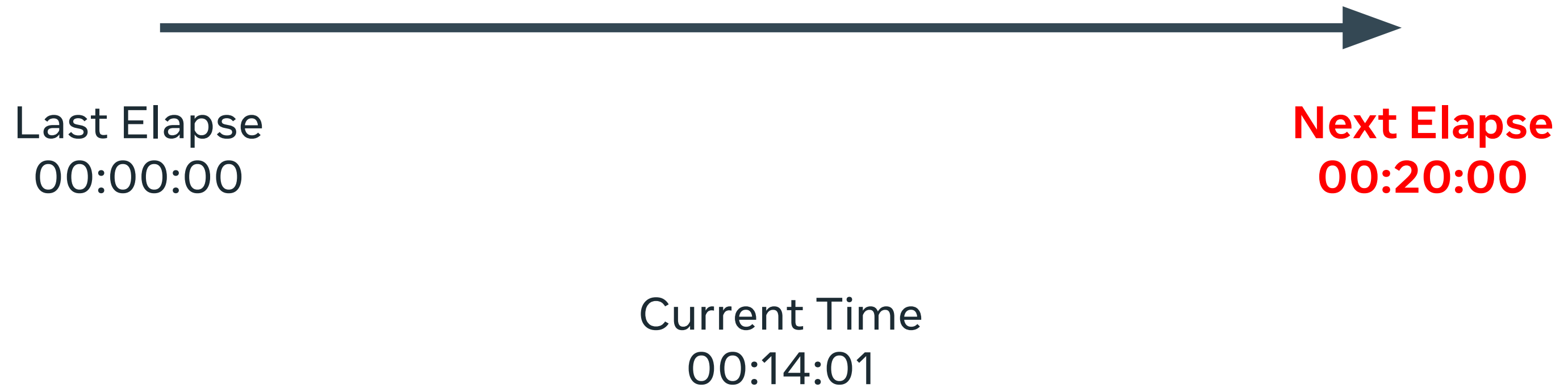
[Timer]

OnCalendar=\*:0/15:0

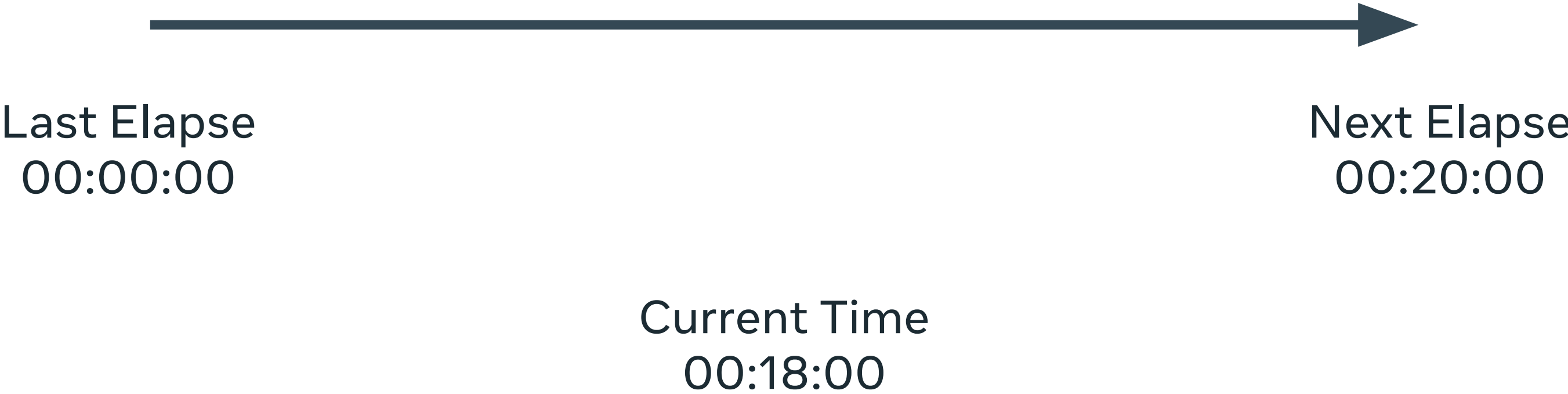
RandomizedDelaySec=870

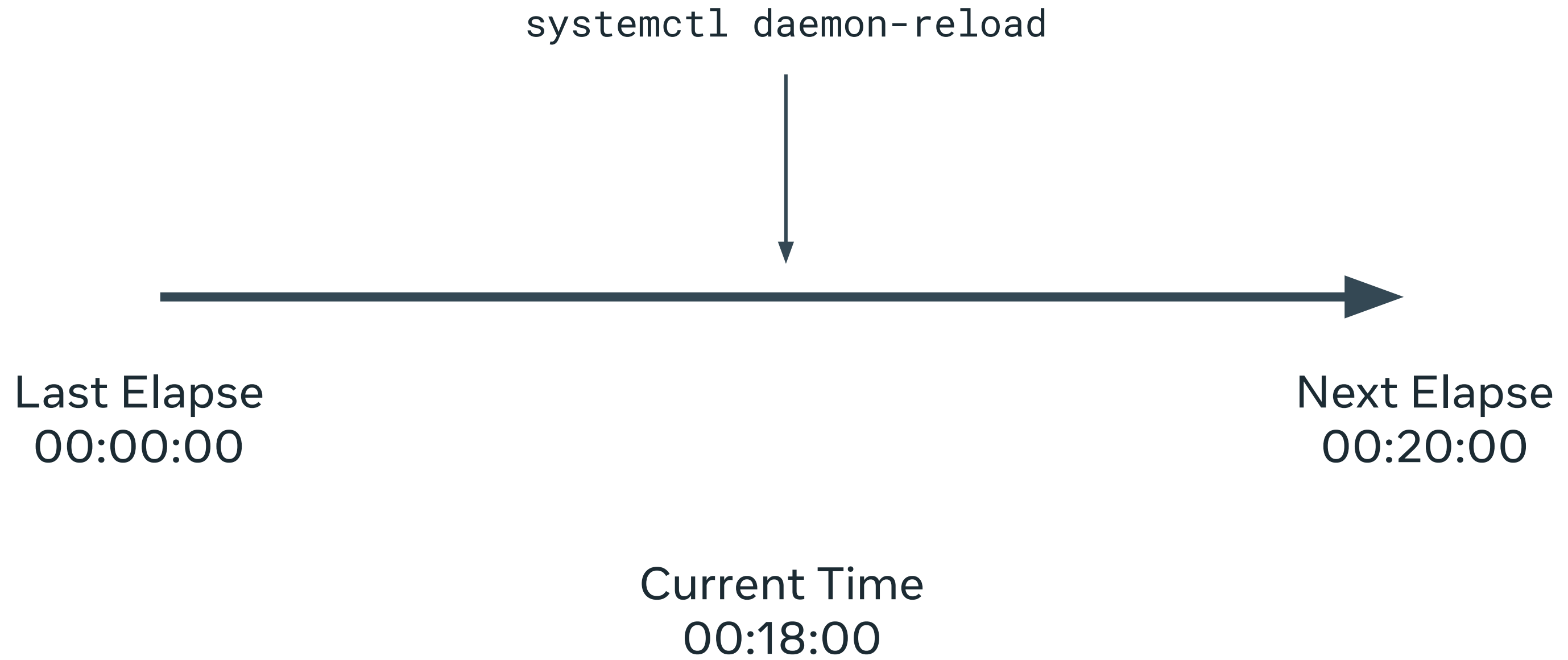


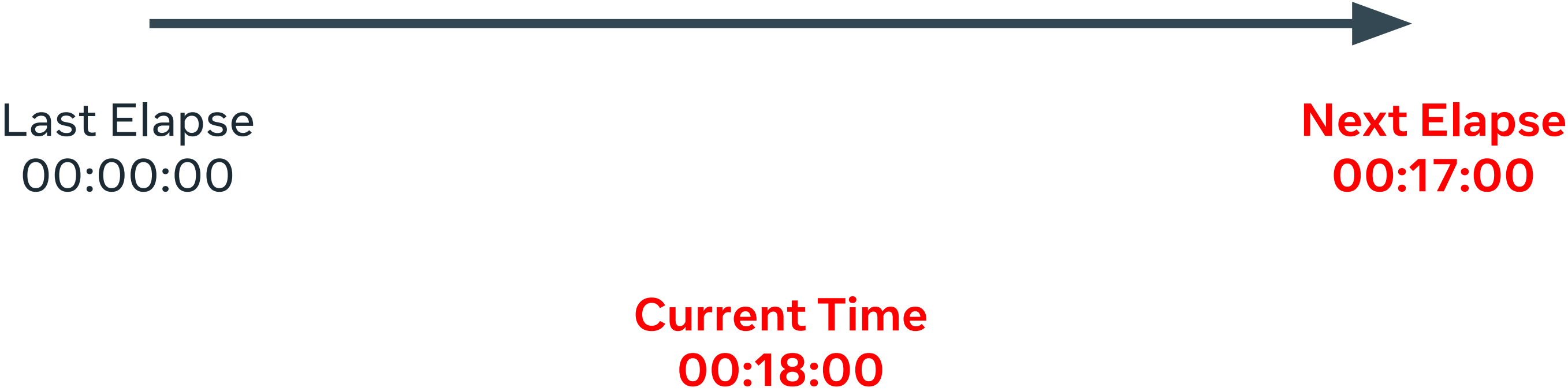




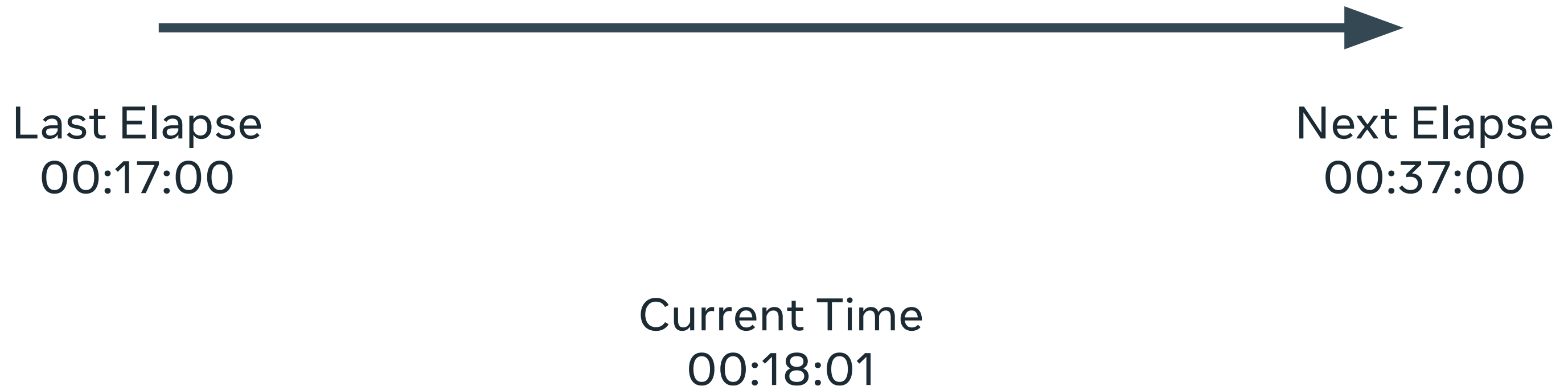








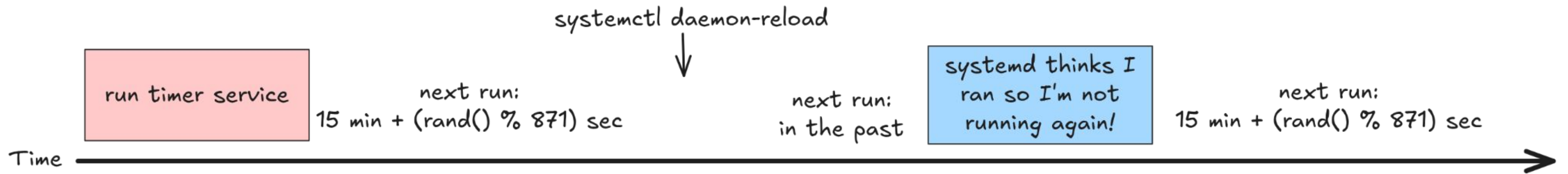
**Elapse times are updated,  
but timer never ran!**



### Expectation



### Reality



DETERMINISTIC SPLAY (ADDED IN SYSTEMD 247)

[Timer]

OnCalendar=\*:0/15:0

RandomizedDelaySec=870

FixedRandomDelay=true

WHAT'S THE CATCH?

[Timer]

OnCalendar=\*:0/15:0

FIXING DAYLIGHT SAVINGS (ADDED IN SYSTEMD 228)

[Timer]

OnCalendar=\*:0/15:0 UTC



# 04 Miscellaneous Musings

WHAT DO YOU THINK THIS DOES?

[Path]

PathExists=/etc/testfile

# PathExists= Intuition

- If a path exists, run the corresponding service.
- Don't run the service again until the path disappears and reappears.

# PathExists= Reality

- While the path exists, keep running the service over and over.
  - Behavior as of systemd v246.

# Can I Get the Original Behavior?

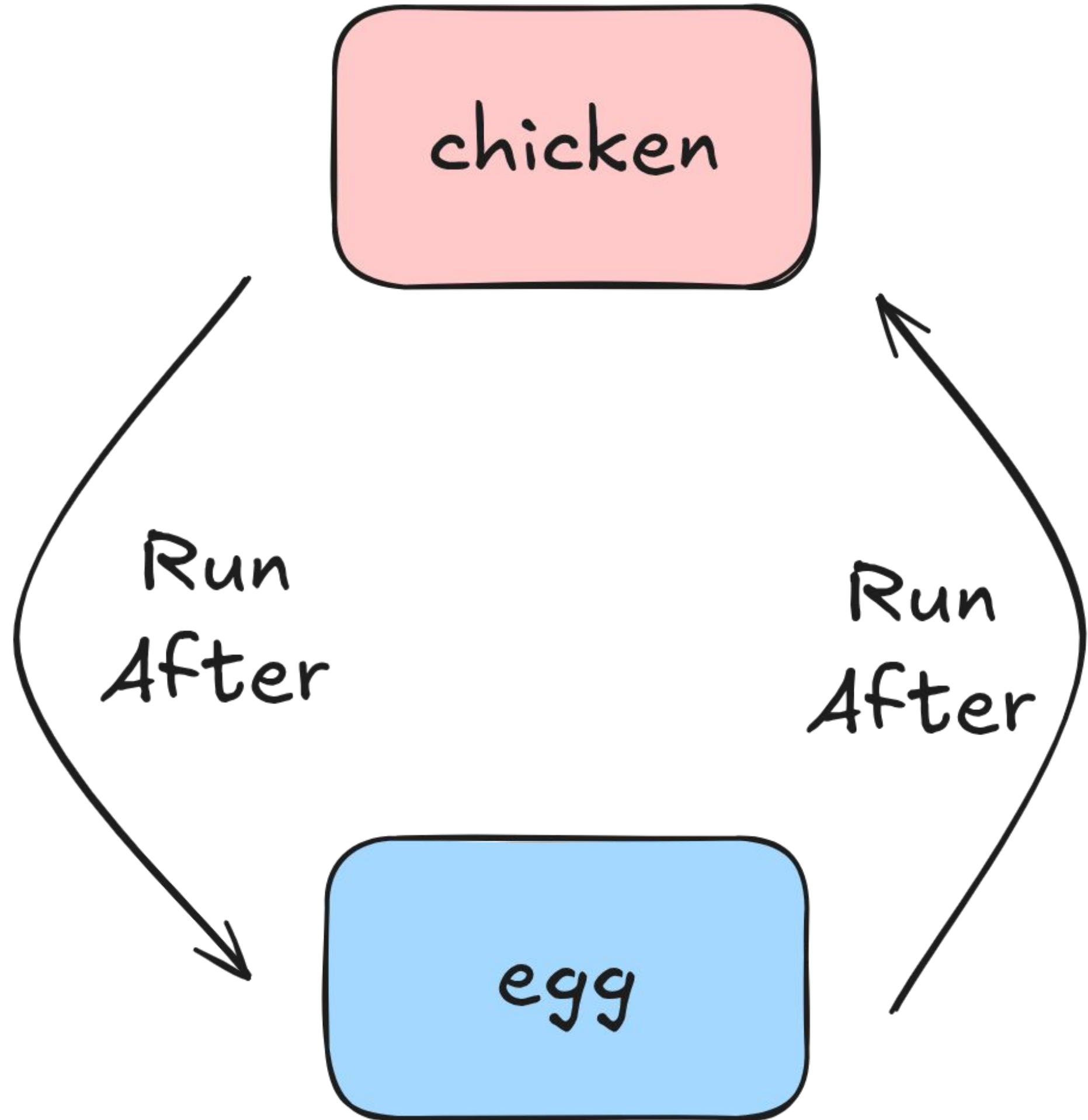
- PathChanged=
- RemainAfterExit=yes

**KillMode=**  
[systemd.kill man page](#)

# “KillMode=mixed considered harmful”

- When main process exits, KillMode=mixed will SIGKILL all children spawned.
  - No chance to clean up!
  - Example: FUSE mountpoints leaked.
  - “Please think twice before using KillMode=mixed. And then think a third time.”
    - [KillMode=mixed Github query.](#)
- KillMode=control-group is almost always the better choice (and it is the default).

What's a  
Cycle?





# Detecting Cycles

- `systemd-analyze verify <unit>` to detect cycles.
  - Example: `systemd-analyze verify multi-user.target`
- Caveat:
  - Does not perfectly mimic boot.
  - Does not detect some cycles.

## chicken.service

```
[Unit]
After=egg.service

[Service]
ExecStart=echo "I'm a chicken"

[Install]
WantedBy=multi-user.target
```

## egg.service

```
[Unit]
After=chicken.service

[Service]
ExecStart=echo "I'm an egg"

[Install]
WantedBy=multi-user.target
```

# No Cycle Detected

- `systemd-analyze verify chicken.service`
- `systemd-analyze verify egg.service`

# Yes Cycle Detected

- `systemd-analyze verify chicken.service egg.service`
- `systemd-analyze verify multi-user.target`

# 05 Tiny Takeaway

# You're Not Alone

# You're Not Alone

- Misconfigurations happen.
- Documentation isn't always clear.
  - Users don't always read the documentation.
- New improvements landing all the time.
- systemd isn't perfect:
  - Could something be better?
  - Struggling with configuration?
  - Let us know!
    - [systemd GitHub](#) is very active and maintainers read every issue / pull request.

# Questions?

THANK YOU FOR YOUR TIME

[github.com/anitazha](https://github.com/anitazha)

[twitter.com/the\\_anitazha](https://twitter.com/the_anitazha)

@anitazha@fosstodon.org

@anitazha:matrix.org

[ ● ◀ ] **systemd**



