

Ask ChatGPT

* * * * */usr/local/bin/system_spike_logger.sh

for every 2 minutes

Your current cron expression:

ruby

* * * * * /usr/local/bin/system_spike_logger.sh

 Copy code

👉 Runs every 1 minute, not every 2 minutes.

Correct cron for every 2 minutes

Use this instead:

perl

*/2 * * * * /usr/local/bin/system_spike_logger.sh

 Copy code

Breakdown

Field	Meaning
*/2	Every 2 minutes
*	Every hour
*	Every day
*	Every month
*	Every day of week

Quick verify

After adding to crontab:

bash

 Copy code

```
crontab -l
```

Check logs:

bash

 Copy code

```
grep CRON /var/log/syslog
```

ChatGPT 5.2 ▾

 Share ...

perl

 Copy code

```
*/2 * * * * /usr/local/bin/system_spike_logger.sh >> /var/log/system-spike-cron.log
```

Ensure script is executable:

bash

 Copy code

```
chmod +x /usr/local/bin/system_spike_logger.sh
```

If you want:

- ⌚ Every 5 / 10 / 15 minutes
- 🕒 Specific hours only
- ✉️ Email on failure
- 🔒 Lock to avoid overlapping runs

tell me 

     ...

⌚ Every 5 / 10 / 15 / 25 / 30 / 35 / 40 / 50 / 55 / 60 minutes

Here are **exact cron expressions** for each interval you asked for 
(ready to copy-paste)

⌚ Cron schedules by interval

Every 5 minutes

perl

 Copy code

```
*/5 * * * * /usr/local/bin/system_spike_logger.sh
```

Every 10 minutes

perl

 Copy code

```
*/10 * * * * /usr/local/bin/system_spike_logger.sh
```

Every 15 minutes

perl

 Copy code

```
*/15 * * * * /usr/local/bin/system_spike_logger.sh
```

Non-standard minute intervals (must be explicit)

Cron only supports 0–59 minutes, so for 25/35/50/55 you must list minutes.

Every 25 minutes

ruby

 Copy code

```
0,25,50 * * * * /usr/local/bin/system_spike_logger.sh
```

Every 35 minutes

ruby

 Copy code

```
0,35 * * * * /usr/local/bin/system_spike_logger.sh
```

Every 50 minutes

ruby

 Copy code

```
0,50 * * * * /usr/local/bin/system_spike_logger.sh
```

Every 55 minutes

ruby

 Copy code

```
0,55 * * * * /usr/local/bin/system_spike_logger.sh
```

⌚ Every 40 minutes

ruby

 Copy code

```
0,40 * * * * /usr/local/bin/system_spike_logger.sh
```

⌚ Every 60 minutes (hourly)

ruby

 Copy code

```
0 * * * * /usr/local/bin/system_spike_logger.sh
```

⭐ Recommended (safe cron with logging + lock)

Avoid overlapping executions:

ruby

 Copy code

```
*/5 * * * * flock -n /tmp/system_spike.lock /usr/local/bin/system_spike_logger.sh
```

🧪 Verify

bash

 Copy code

```
crontab -l
```

🔥 Want next?

- ➡ Skip if CPU already high
- ✉ Email only on spike
- 📊 Push metrics to CloudWatch / Grafana
- 📁 Auto-rotate logs

Just say the word 

