Linux in the Land of LLMs

Greg Kroah-Hartman

Fellow, The Linux Foundation

FINOS Fintech Open Source Foundation

THE LINUX FOUNDATION

Linux in the land of LLMs

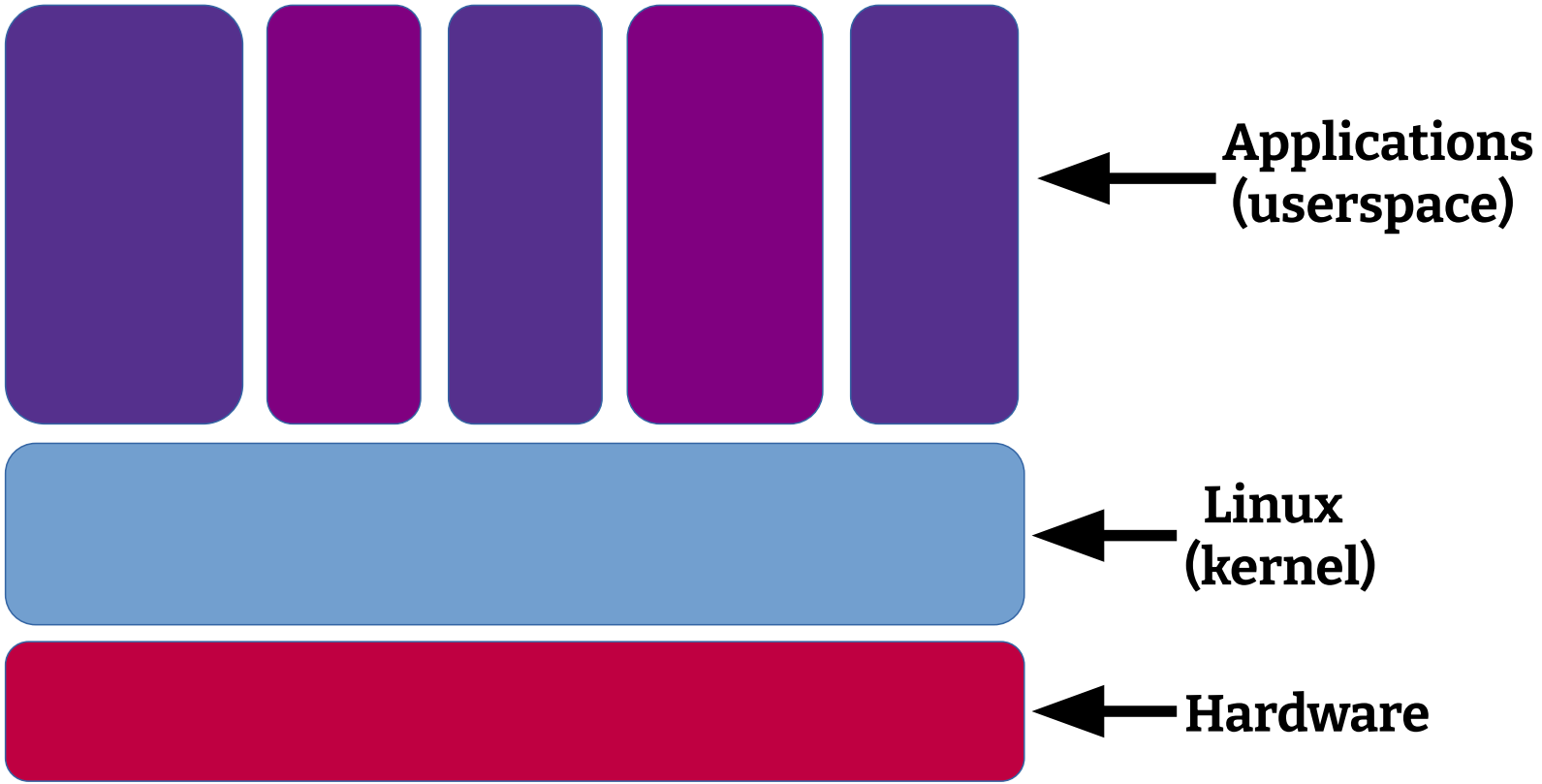
Greg Kroah-Hartman

gregkh@linuxfoundation.org

Disclaimer

All of this is just my personal opinion, based on working as part of the Linux kernel security team since it was created in 2005 and working with a number of different, “modern”, LLM models.

Nothing in here reflects the opinion of the Linux Foundation or any other Linux kernel developer.


Applications
(userspace)
Linux
(kernel)
Hardware

Linux Developer Community

5,221 developers
375+ employers

2025

Linux Developer Community

78,209 changes accepted
8.9 changes per hour
50 changes per day in stable trees
100 CVEs assigned per week
New release every 8-9 weeks

How we do it

90 minute talk, with summary:

search “pragmatic engineer Linux”

Mean time to exploit

2018	63 days
2020	32 days
2024	5 days
2026	-7 days

The traditional discover -> disclose -> patch -> deploy cycle was designed for a slower adversary, that adversary no longer exists.

LLMs are now “good enough”

- Security bug reports / fixes are real
- All open source projects are flooded
- Everyone is reevaluating how to handle this

Network LLMs are “public”

- Any “found” bug will be seen by others.
- Network LLMs want to “please” you.
- Projects are treating them as public

“If AI can find vulnerabilities, it can fix them. The only thing scarce now is collective will.”

– Jim Zemlin

“I’m condemned to use the tools of my enemy to defeat them.”

– Andor

How to fix a problem

```
llm --dangerously-skip-permissions \
    -p "Fix this code"             \
    --verbose                      \
  &> ~/tmp/fix.log
```

Use open models locally

- They are all “good enough”.
- Will find/fix almost all issues.
- Keeps everything private.
- Worry about the next model when you get it.
- Companies wasted $25 billion not doing this.

What you can do today

- Ignore the doom marketing

What you can do today

- Ignore the doom marketing
- Know what software you use/rely on

What you can do today

- Ignore the doom marketing
- Know what software you use/rely on
- Be able to update your systems on a continuous basis

What you can do today

- Ignore the doom marketing
- Know what software you use/rely on
- Be able to update your systems on a continuous basis
- Update your dependancy chains

What you can do today

- Run local models internally

What you can do today

- Run local models internally
- NEVER upload any non-public information

What you can do today

- Run local models internally
- NEVER upload any non-public information
- Fix the bugs you find today

What you can do today

- Run local models internally
- NEVER upload any non-public information
- Fix the bugs you find today
- Send the fix upstream to be merged

What you can do today

- Run local models internally
- NEVER upload any non-public information
- Fix the bugs you find today
- Send the fix upstream to be merged
- Ignore the “model of tomorrow”

What you can do today

- Securing Open Source in the Age of AI

  https://openssf.org/resources/securing-open-source-in-the-age-of-ai-a-practical-guide/

Whose side has more will?

It’s going to be a rocky 18 months...