Linux Kernel Release Model

(and security stuff)

Greg Kroah-Hartman
gregkh@linuxfoundation.org

github.com/gregkh/presentation-release-model

60,000 files
24,767,000 lines

Kernel release 4.13.0

4,316 developers
519 companies

Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017

10,000 lines added
2,500 lines removed
2,100 lines modified

Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017

10,000 lines added
2,500 lines removed
2,100 lines modified

Every day

Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017

8.5 changes per hour

Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017

9.7 changes per hour

4.9 & 4.12 release

4.12 release July 7th?

2nd largest release

Old release model

2.2 – January 1999

2.4 – January 2001

2.6 – December 2003

“New” release model

Release every 2-3 months
All releases are stable

“Cambridge Promise”

Will not break userspace.

“Cambridge Promise”

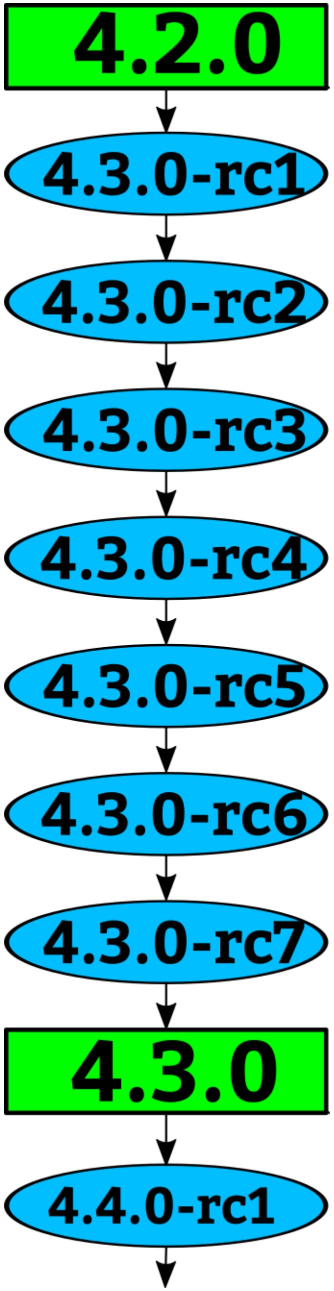
Will not break userspace, knowingly.

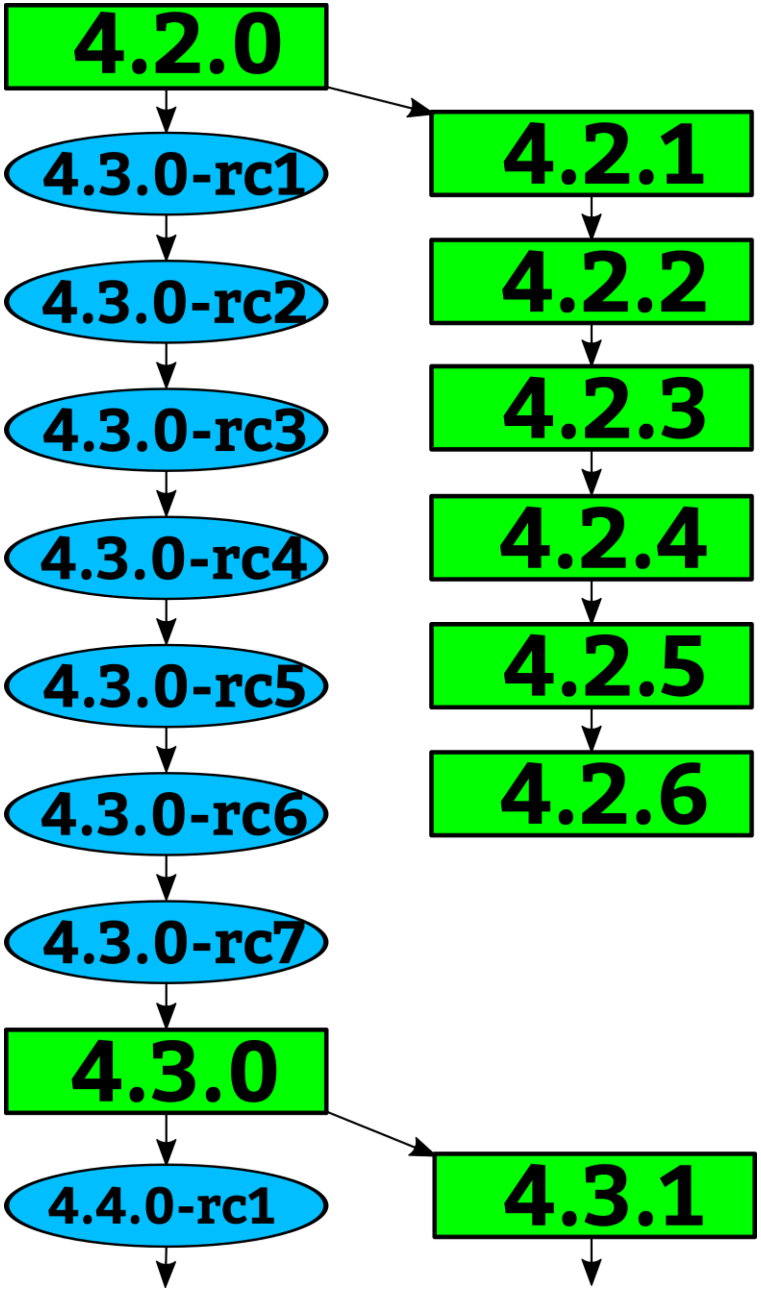
– July 2007

Version numbers mean nothing

2.6.x → 3.x 2011

3.x → 4.x 2015


4.2.0
4.3.0-rc1
4.3.0-rc2
4.3.0-rc3
4.3.0-rc4
4.3.0-rc5
4.3.0-rc6
4.3.0-rc7
4.3.0
4.4.0-rc1


4.2.0
4.3.0-rc1
4.2.1
4.3.0-rc2
4.2.2
4.3.0-rc3
4.2.3
4.3.0-rc4
4.2.4
4.3.0-rc5
4.2.5
4.3.0-rc6
4.2.6
4.3.0-rc7
4.3.0
4.4.0-rc1
4.3.1

Stable rules

- Bugfix
- Less than 100 lines
- New ids or quirks
- Must be in Linus’s tree

https://www.kernel.org/doc/html/latest/process/stable-kernel-rules.html

Stable kernels

- Bugfix
- Less than 100 lines
- New ids or quirks
- Must be in Linus's tree

“Longterm kernels”

One picked per year
Maintained for at least 2 years

4.4 4.9 4.14

Every release is stable

Decade old guarantee

Always update your kernel

Can’t update your kernel?

Blame your SoC provider…

Kernel Security

Kernel Security

Almost all bugs can be a “security” issue.

Kernel Security

Almost all bugs can be a “security” issue.

Fix them as soon as possible.

Kernel Security

Averaging 13 fixes per day.

> “If you are not using a stable / longterm kernel, your machine is insecure”
>
> – me

“The kernel needs airbags”

– Konstantin Ryabitsev

slides.com/mricon/giant-bags-of-mostly-water#/

> “We will always have bugs,
> we must stop their exploitation”
>
> – Kees Cook

outflux.net/slides/2015/ks/security.pdf

Kernel Hardening

kernsec.org/wiki/index.php/Kernel_Self_Protection_Project

Core Infrastructure Initiative

> “Ceaseless change is the only constant thing in Nature.”
>
> – John Candee Dean

github.com/gregkh/kernel-development

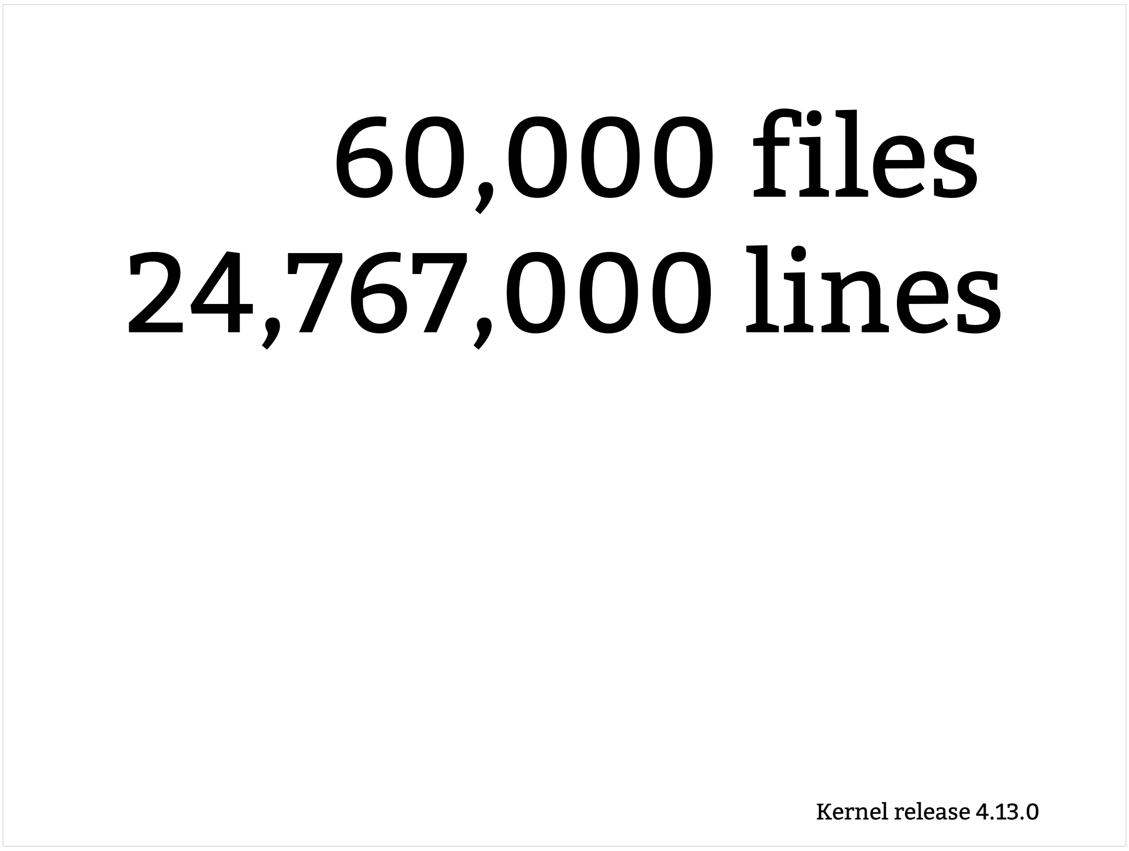
Linux Kernel Release Model

(and security stuff)

Greg Kroah-Hartman
gregkh@linuxfoundation.org

github.com/gregkh/presentation-release-model

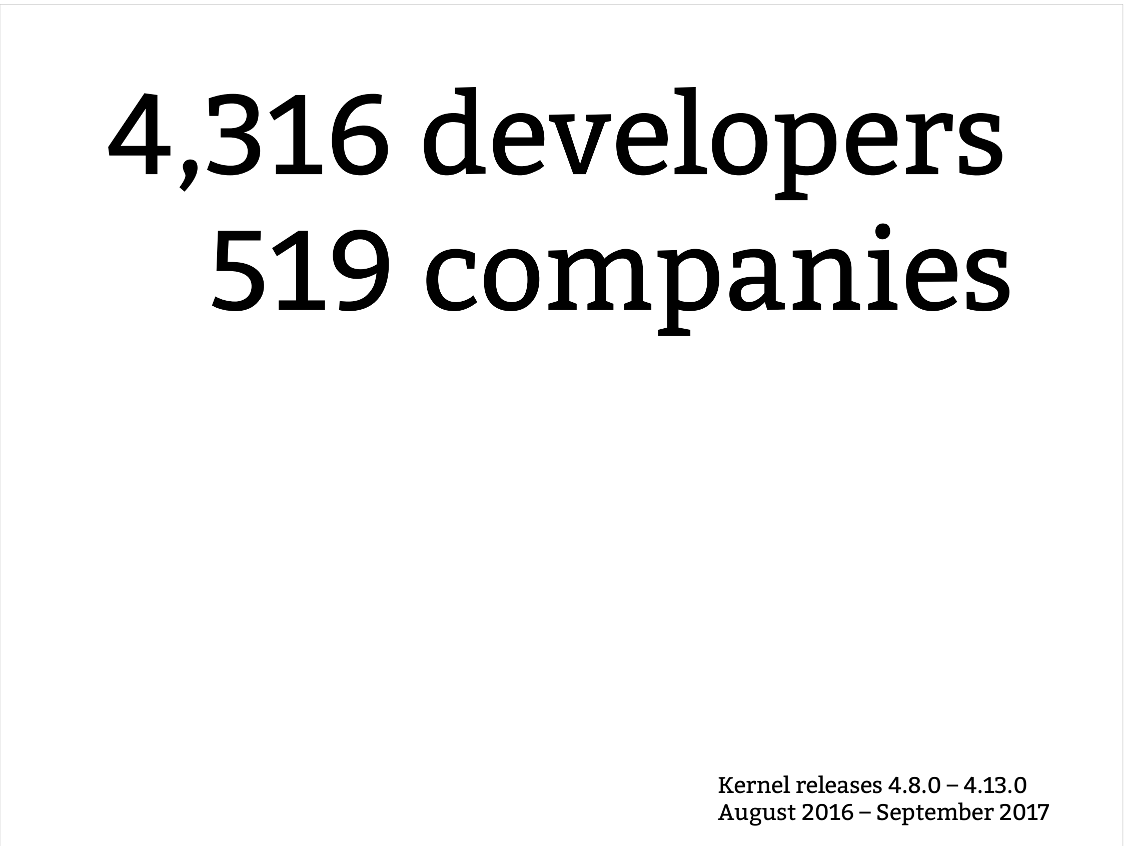
I'm going to discuss the how fast the kernel is moving, how we do it all, and how you can get involved.



60,000 files
24,767,000 lines

Kernel release 4.13.0

This was for the 4.11 kernel release, which happened April 30, 2017.



4,316 developers
519 companies

Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017

This makes the Linux kernel the largest contributed body of software out there that we know of.

This is just the number of companies that we know about, there are more that we do not, and as the responses to our inquiries come in, this number will go up.

Have surpassed 400 companies for 4 years now.

10,000 lines added

2,500 lines removed

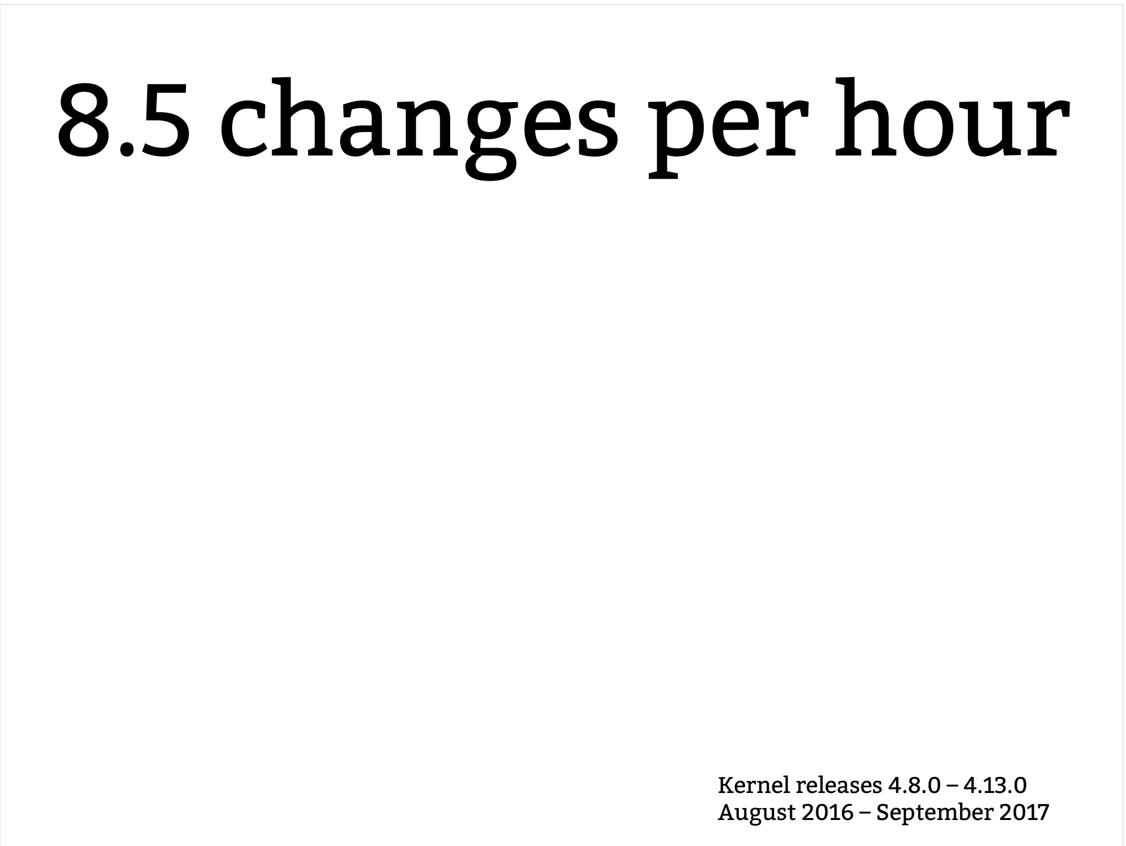
2,100 lines modified

Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017

10,000 lines added
2,500 lines removed
2,100 lines modified

Every day

Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017



8.5 changes per hour

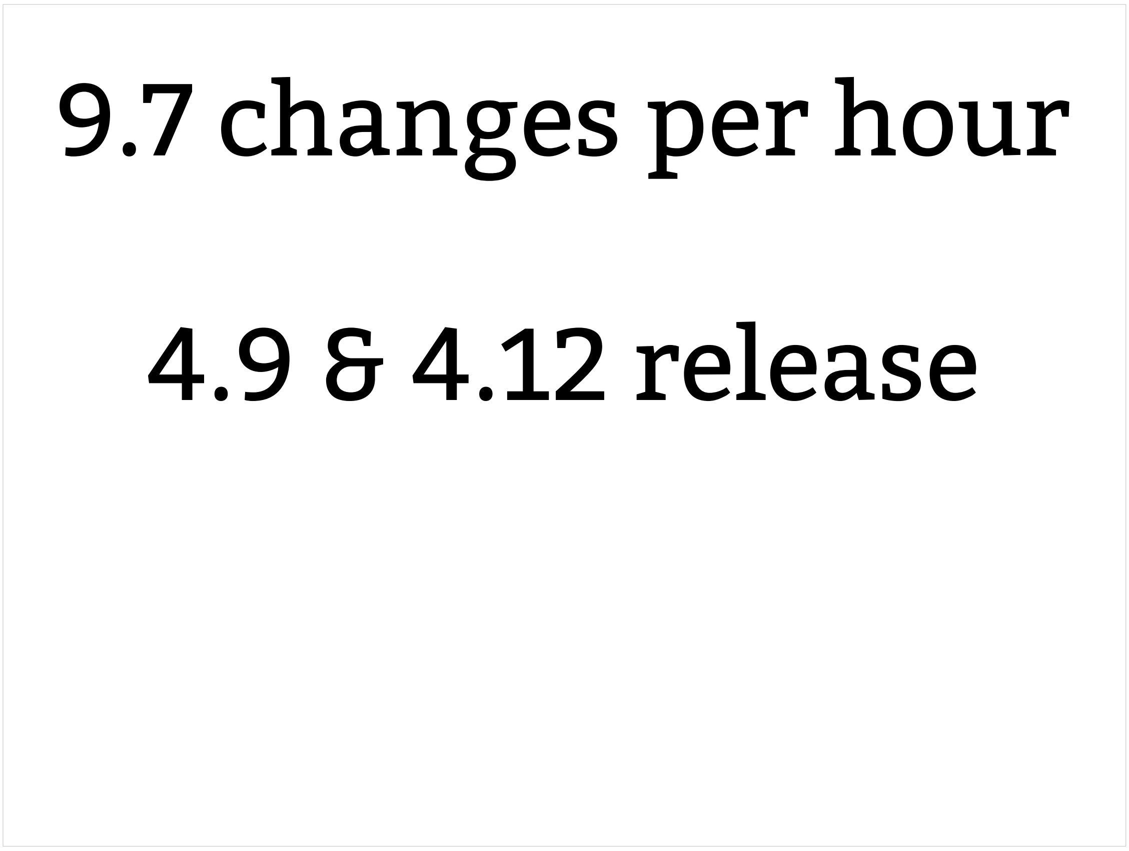
Kernel releases 4.8.0 – 4.13.0
August 2016 – September 2017

This is 24 hours a day, 7 days a week, for a full year.

We went this fast the year before this as well, this is an amazing rate of change.

Interesting note, all of these changes are all through the whole kernel.

For example, the core kernel is only 5% of the code, and 5% of the change was to the core kernel. Drivers are 55%, and 55% was done to them, it's completely proportional all across the whole kernel.

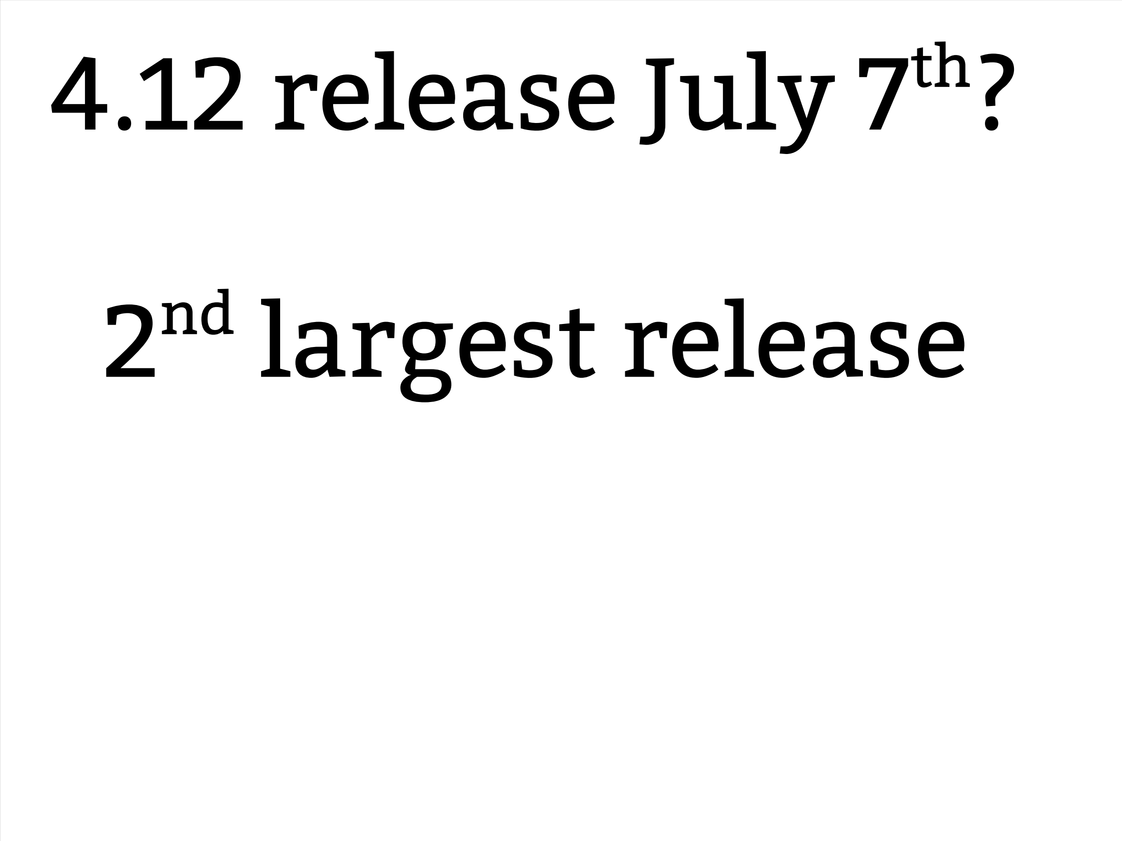


9.7 changes per hour

4.9 & 4.12 release

4.9 was the “largest” in number of changes that we have ever accepted. After 4.9, things went down a bit for 4.10 and 4.11, but 4.12 is getting very big.

Now this is just the patches we accepted, not all of the patches that have been submitted, lots of patches are rejected, as anyone who has ever tried to submit a patch can attest to.



4.12 release July 7th?

2nd largest release

4.12 should be released on July 7th and is on track to be the 2nd largest release by number of changes we have ever done.

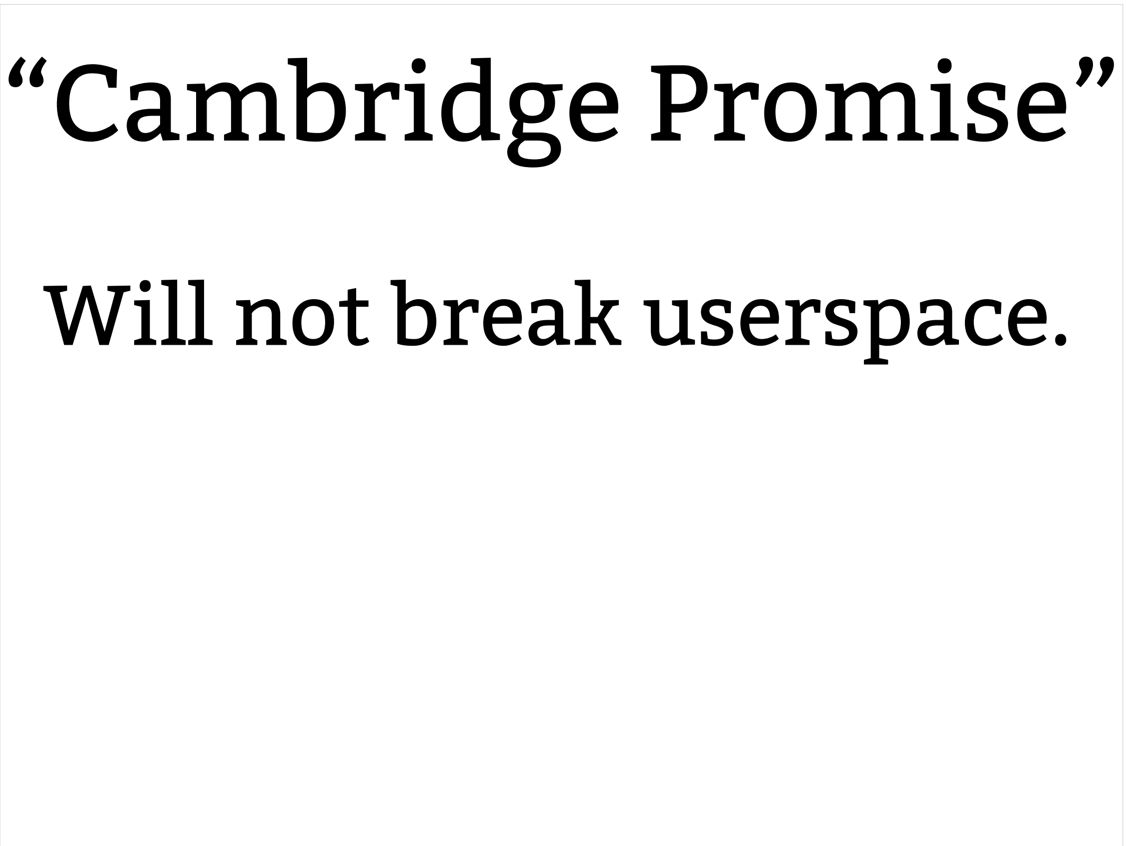
And the first largest on number of lines of code we have added, due to some very large drivers being added to the tree.

Old release model

2.2 – January 1999
2.4 – January 2001
2.6 – December 2003

“New” release model

Release every 2-3 months
All releases are stable

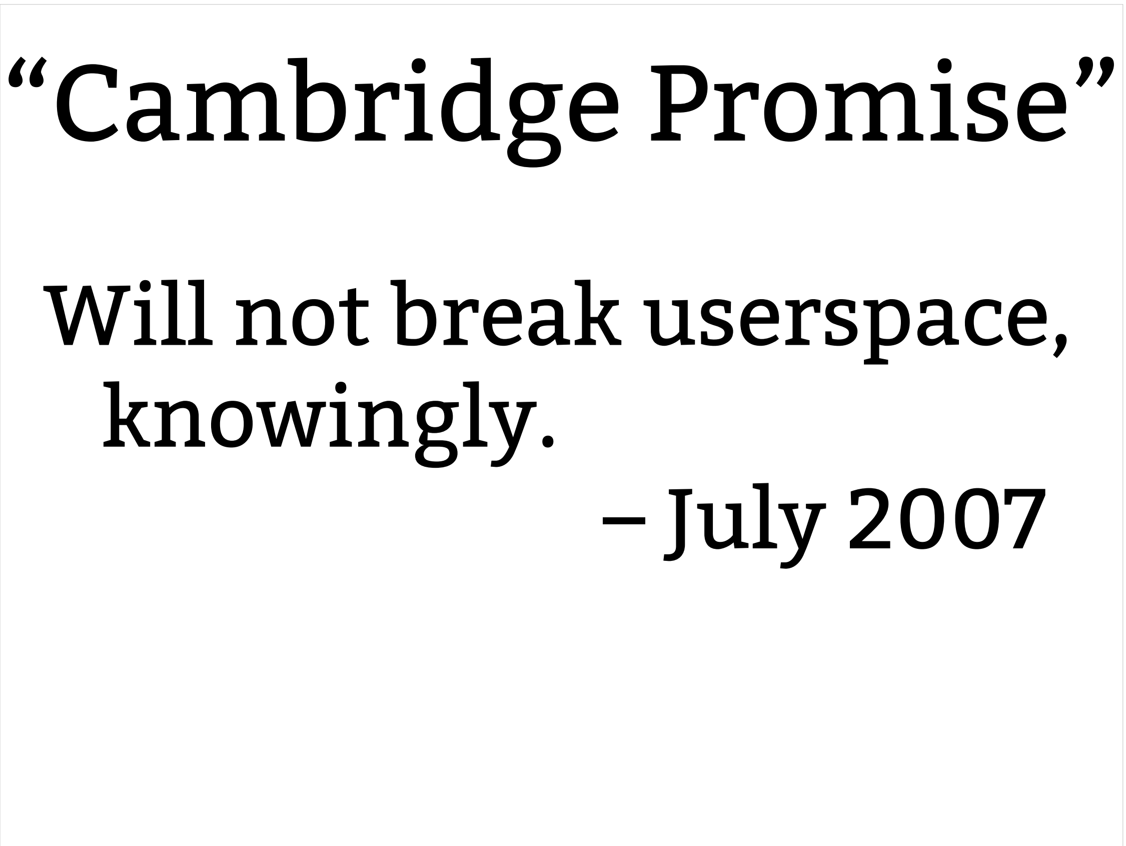


“Cambridge Promise”

Will not break userspace.

Kernel summit 2007 in Cambridge England

All kernel developers agreed that this is what we will do, in order to give users a reason to feel comfortable upgrading their kernels.



“Cambridge Promise”

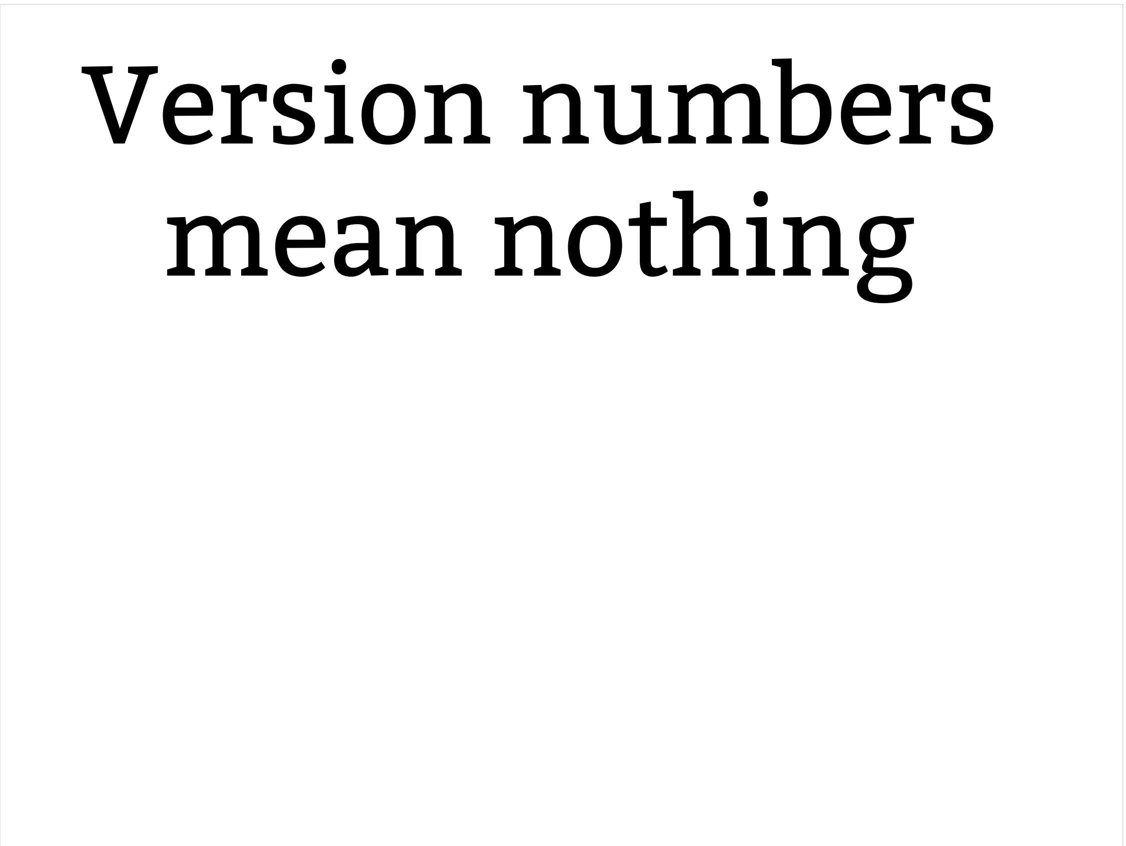
Will not break userspace, knowingly.

– July 2007

Well, we do not knowingly break userspace, we accidentally do it all the time, we are just human.

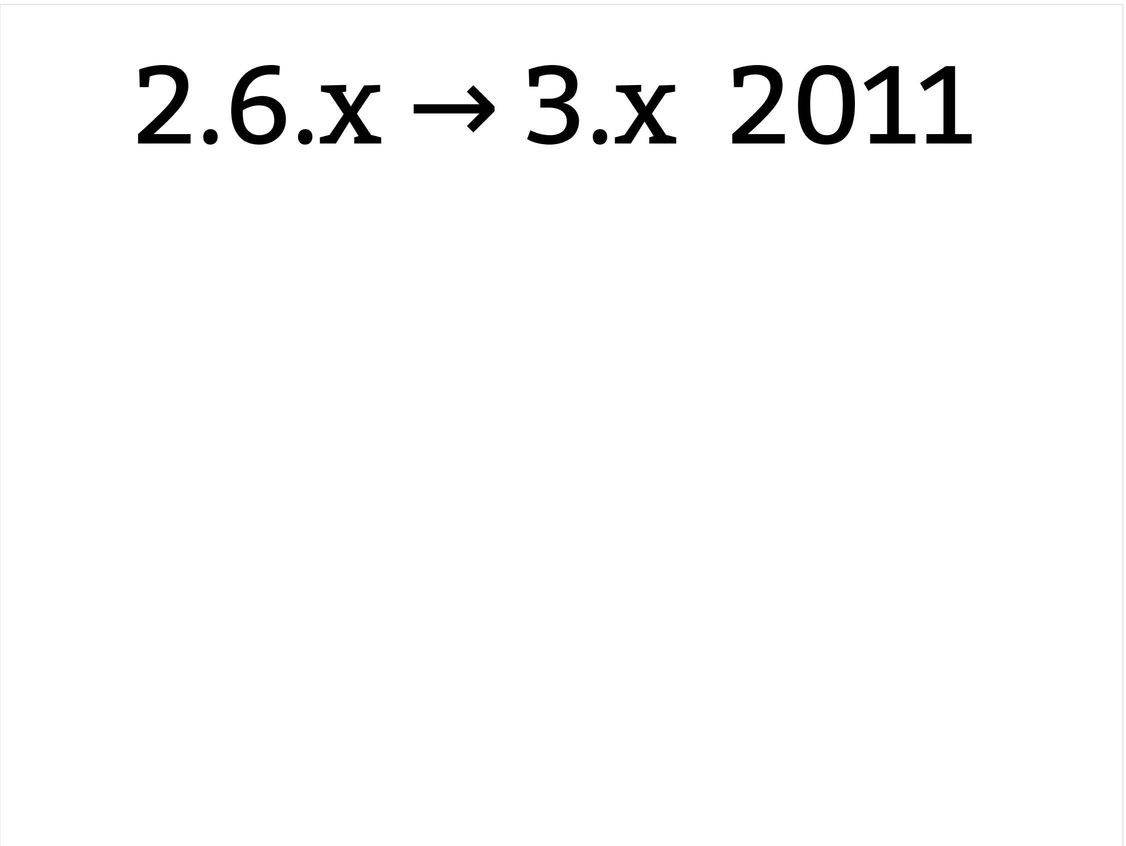
But we will work very hard to fix the issue.

Note, if no one notices userspace is broken, it isn’t.



Version numbers mean nothing

They only mean that one is newer than another.



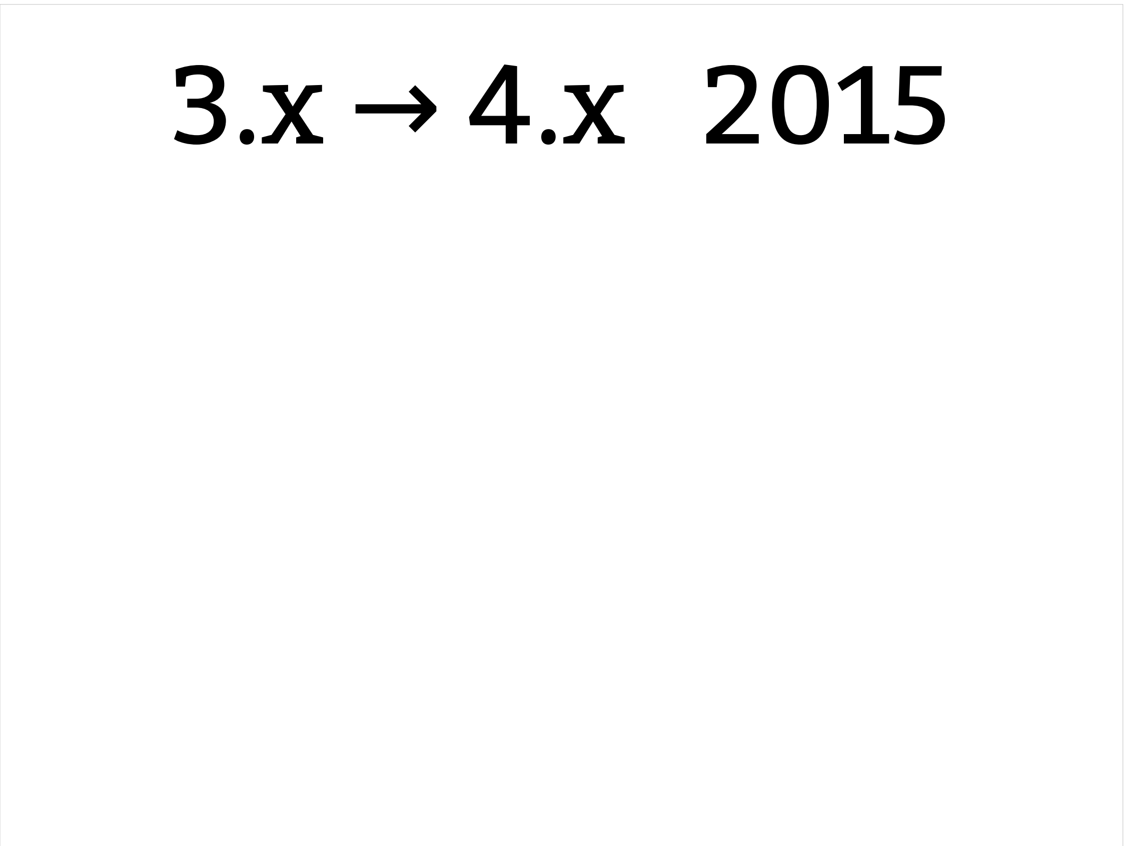
2.6.x → 3.x 2011

Big numbers seem to increment “smaller” over time than small numbers (brains are wierd)

LinuxCon Japan 2011

I bribed Linus with whisky

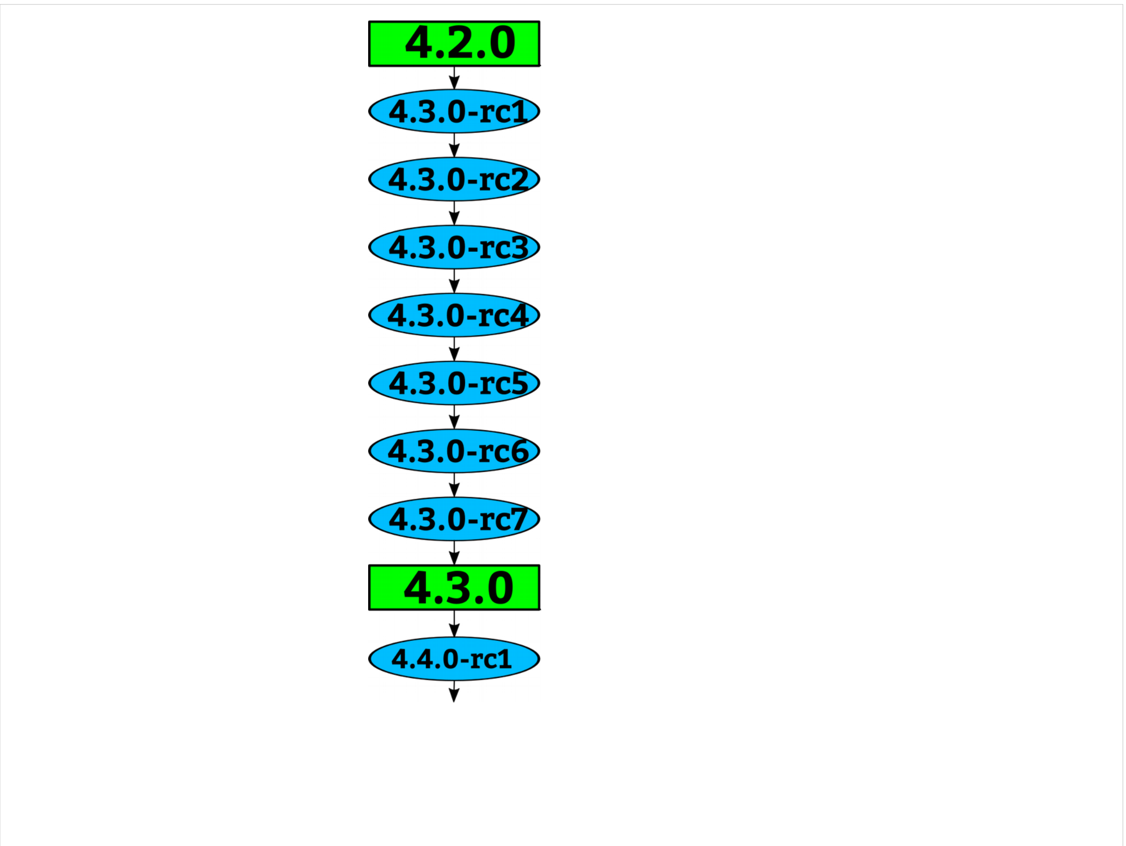
Kernel developers drank the bottle within minutes at the after-party.



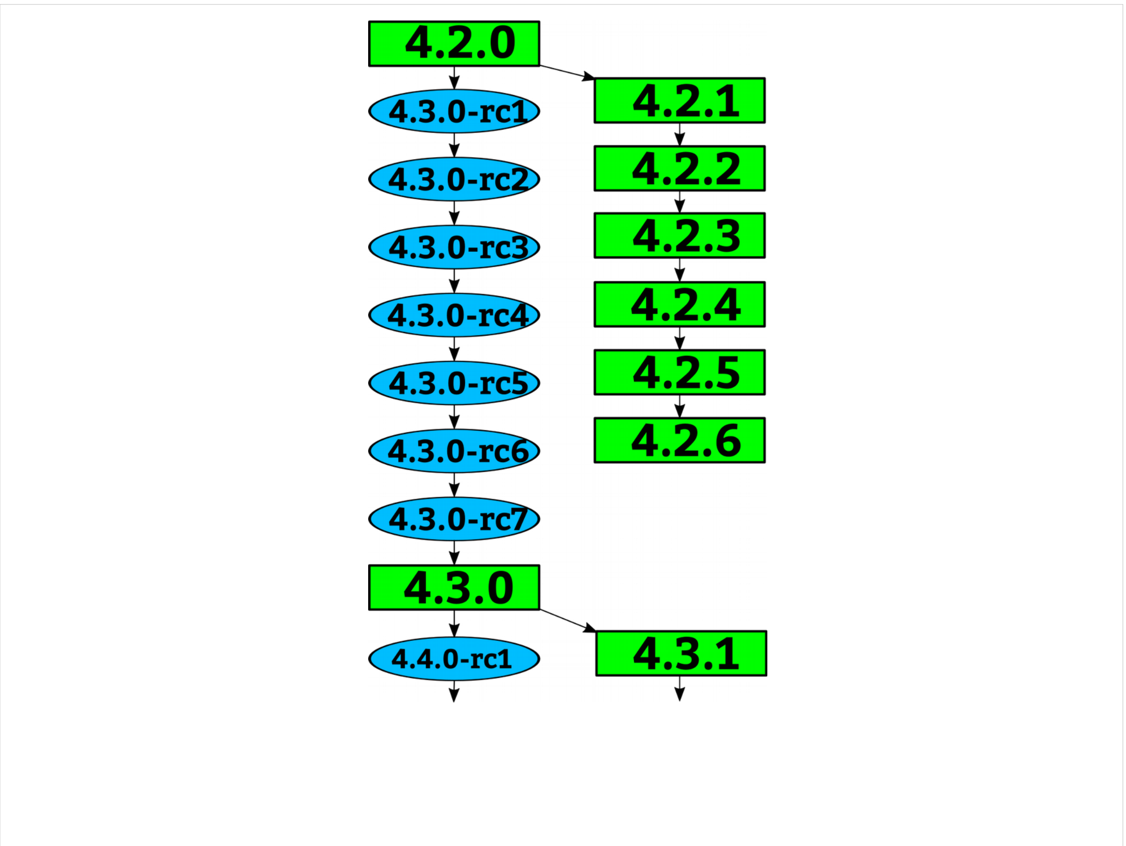
3.x → 4.x 2015

Big numbers seem to increment “smaller” over time than small numbers (brains are wierd)

2015



How a kernel is developed.
Linus releases a stable kernel
- 2 week merge window from subsystem maintainers
- rc1 is released
- bugfixes only now
- 2 weeks later, rc2
- bugfixes and regressions
- 2 weeks later,rc3
And so on until all major bugfixes and regressions are resolved and then the cycle starts over again.

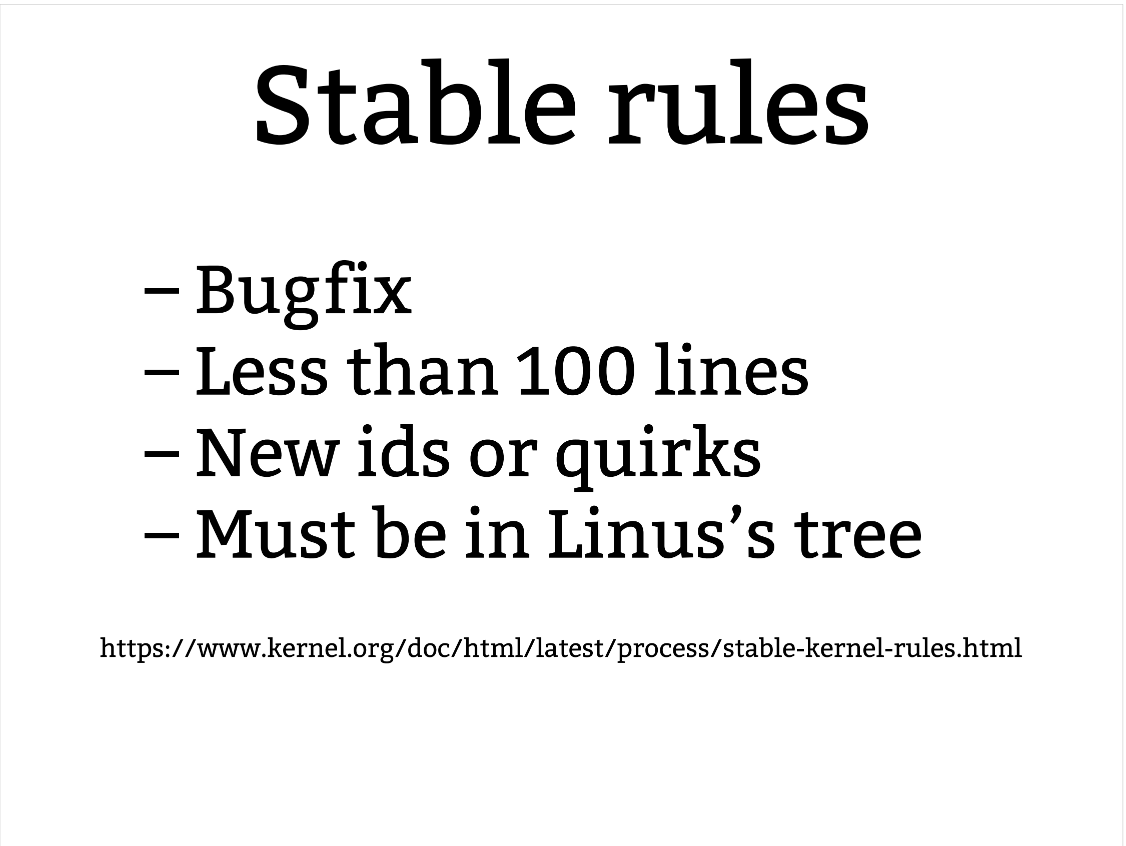


Greg takes the stable releases from Linus, and does stable releases with them, applying only fixes that are already in Linus's tree.

Requiring fixes to be in Linus's tree first ensures that there is no divergence in the development model.

After Linus releases a new stable release, the old stable series is dropped.

With the exception of “longterm” stable releases, those are special, the stick around for much longer…

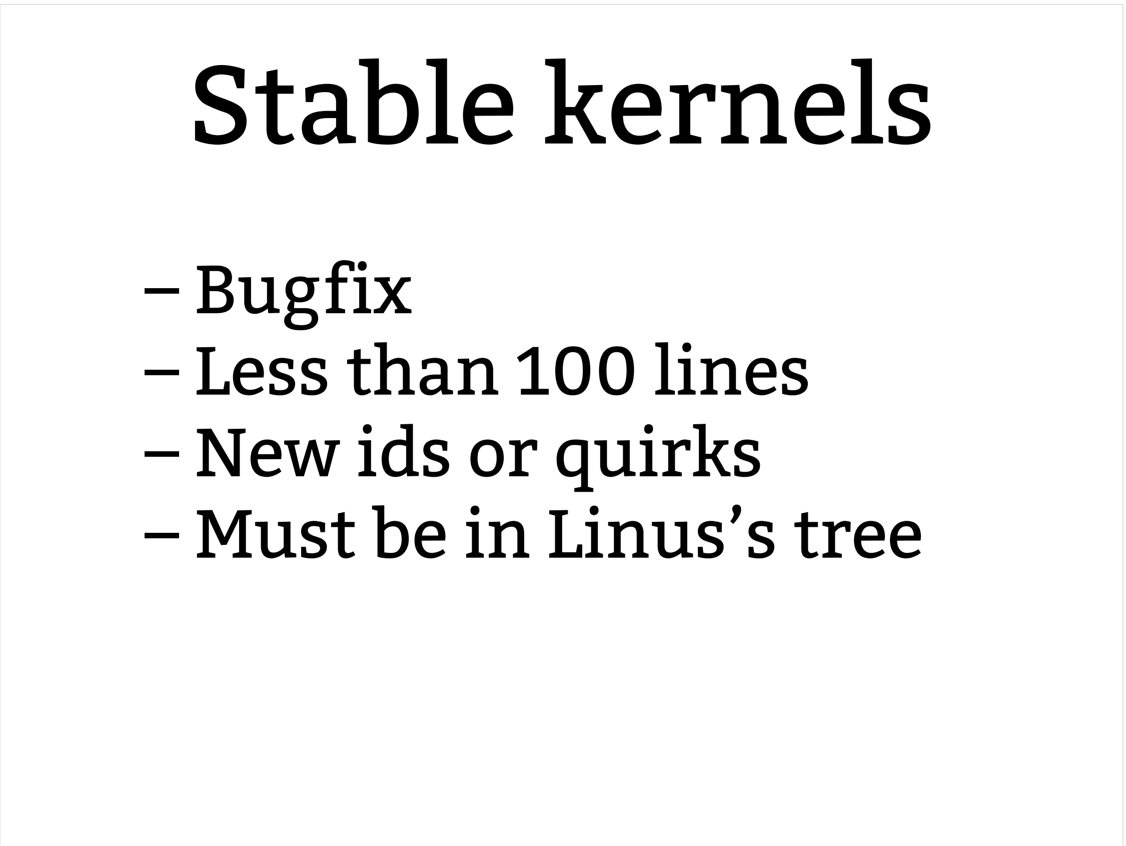


Stable rules

- Bugfix
- Less than 100 lines
- New ids or quirks
- Must be in Linus’s tree

https://www.kernel.org/doc/html/latest/process/stable-kernel-rules.html

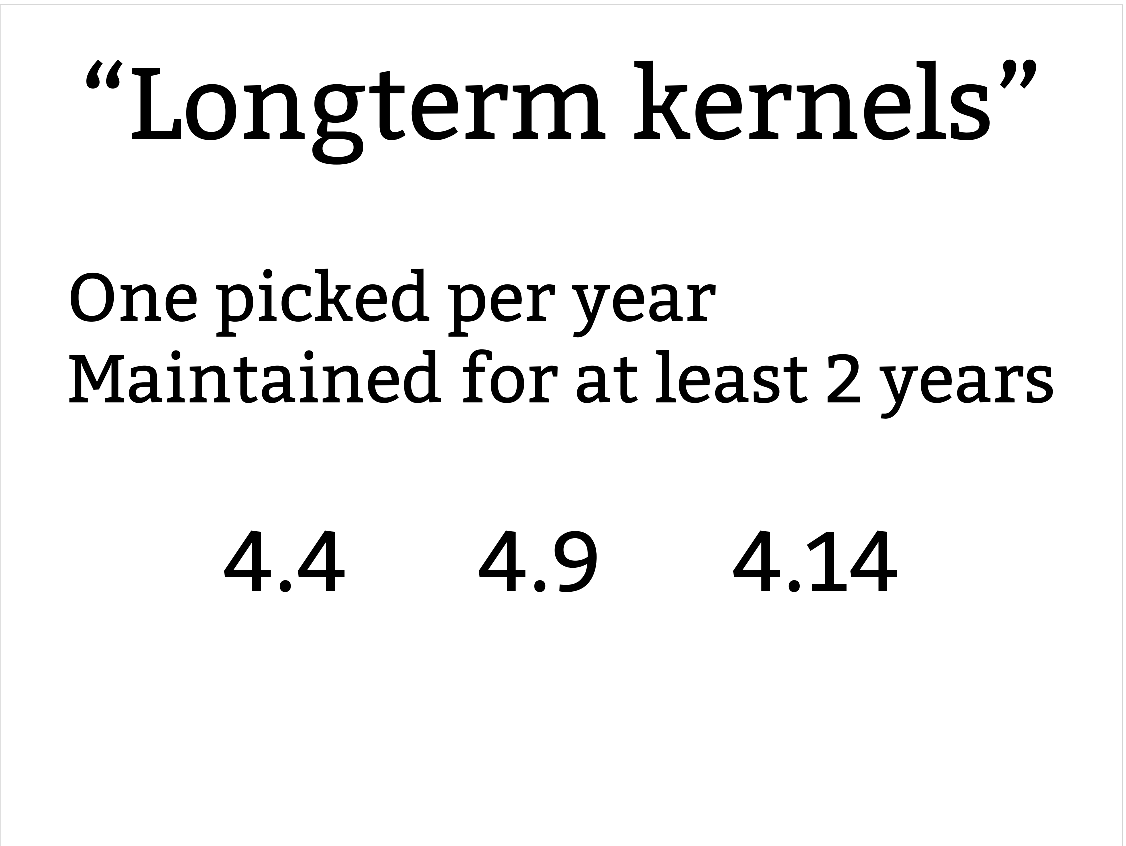
They only mean that one is newer than another.



Stable kernels

- – Bugfix
- – Less than 100 lines
- – New ids or quirks
- – Must be in Linus’s tree

They only mean that one is newer than another.



“Longterm kernels”

One picked per year
Maintained for at least 2 years

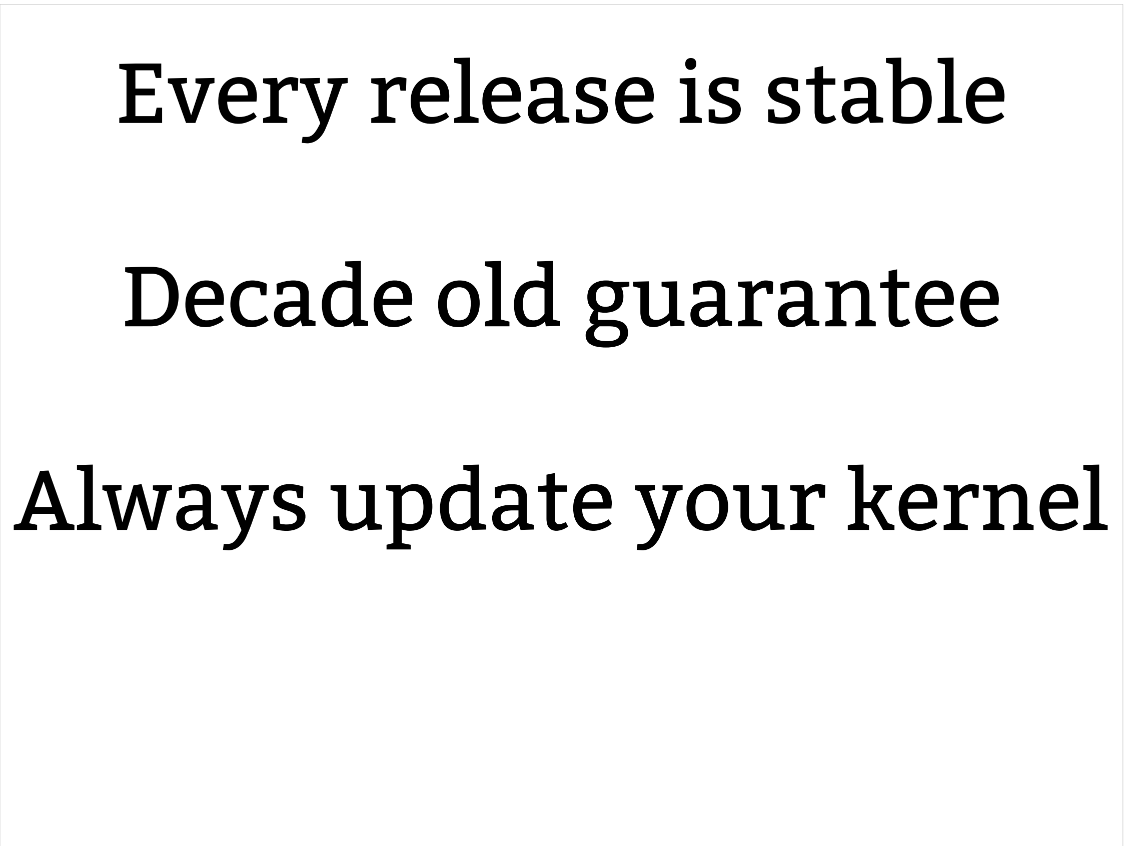
4.4 4.9 4.14

I pick one kernel release per year to maintain for longer than one release cycle. This kernel I will maintain for at least 2 years.

This means there are 2 longterm kernels being maintained at the same time.

4.4 and 4.9 are the longterm kernel releases I am currently maintaining

The LTSI project is based on the longterm kernels.

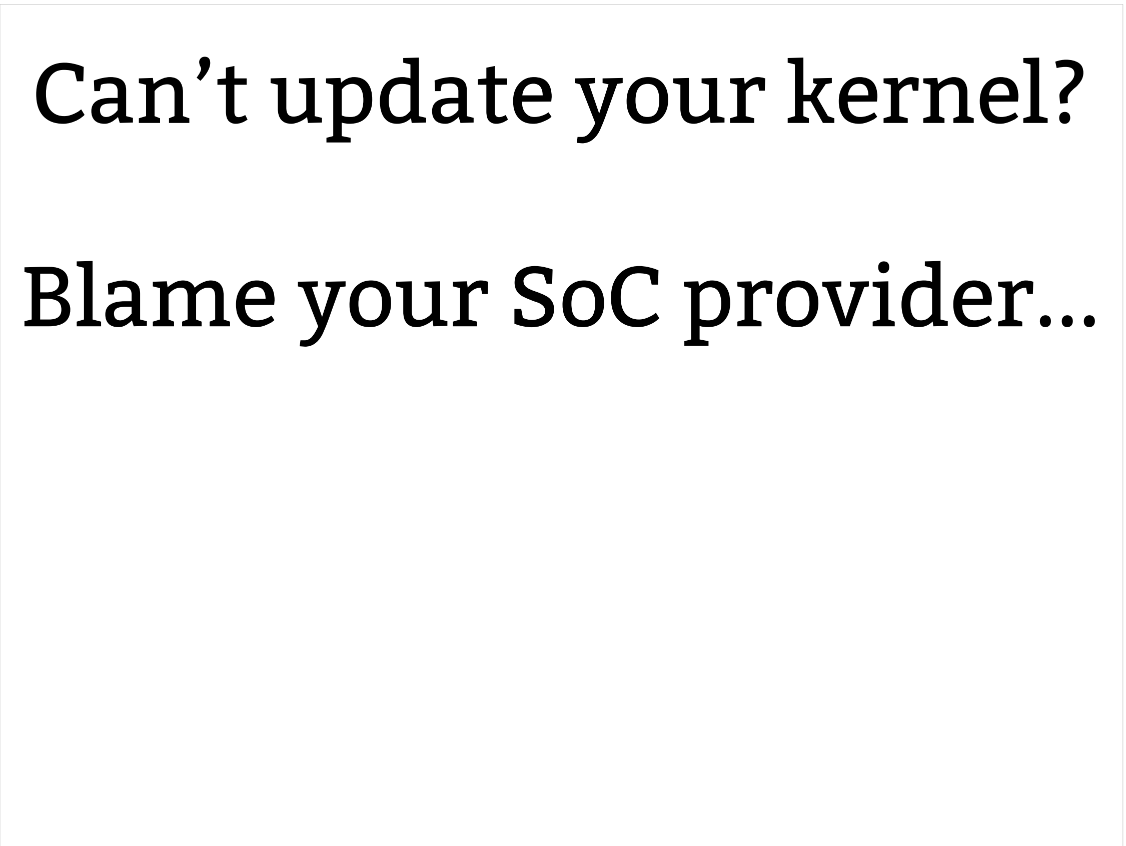


Every release is stable

Decade old guarantee

Always update your kernel

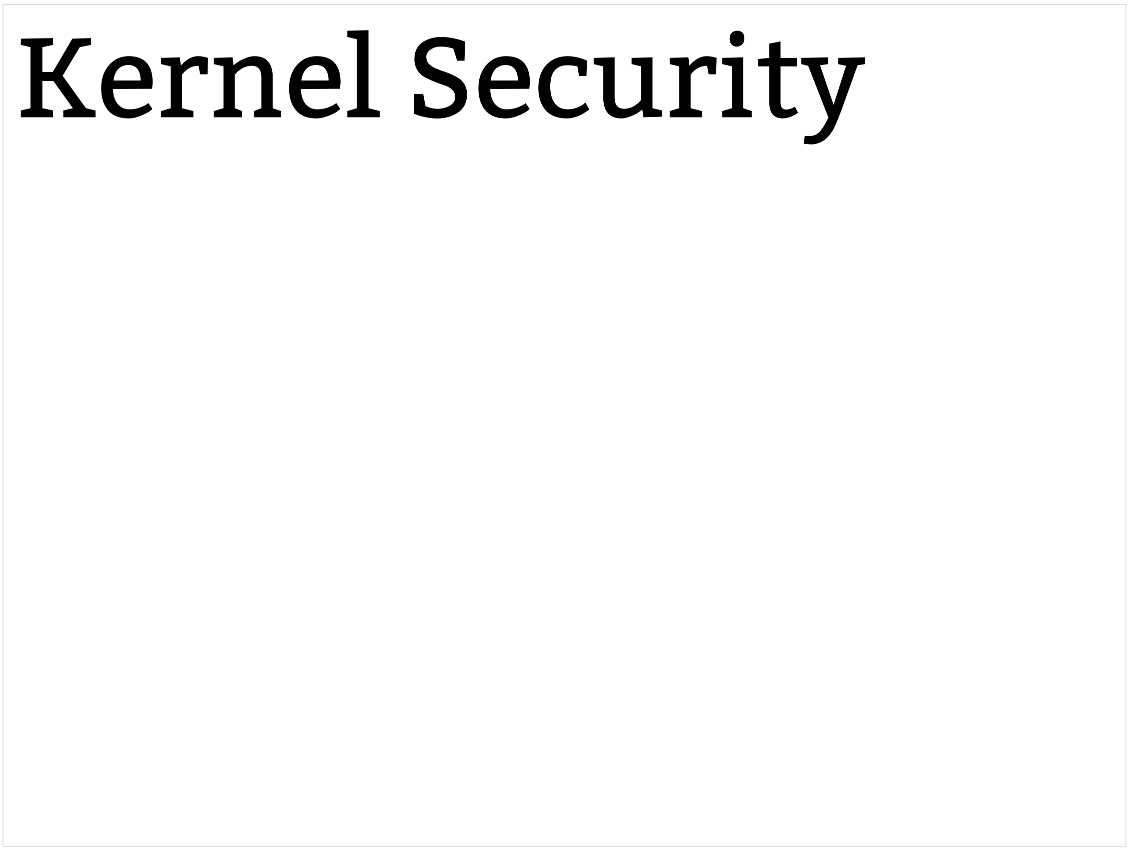
Wait what? Why update?



Can’t update your kernel?

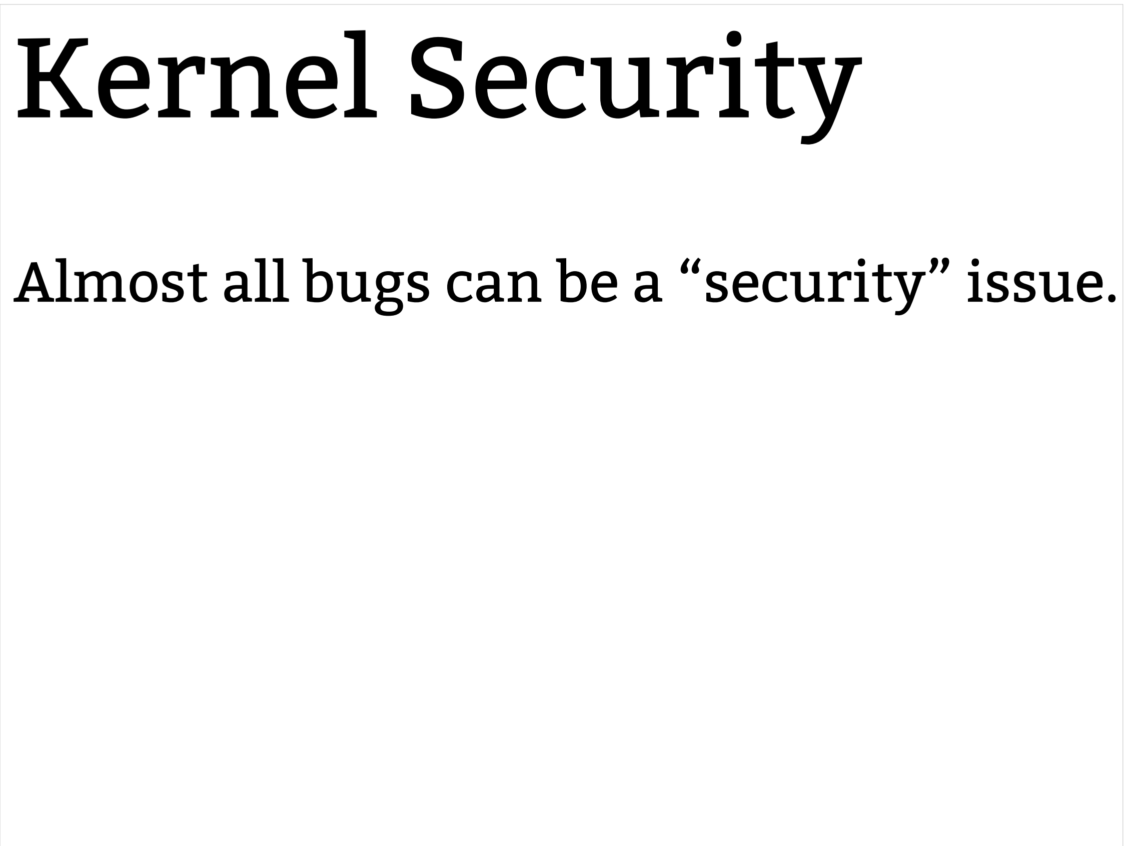
Blame your SoC provider…

SoC kernels suck ass.



Kernel Security

Let’s talk about kernel security.

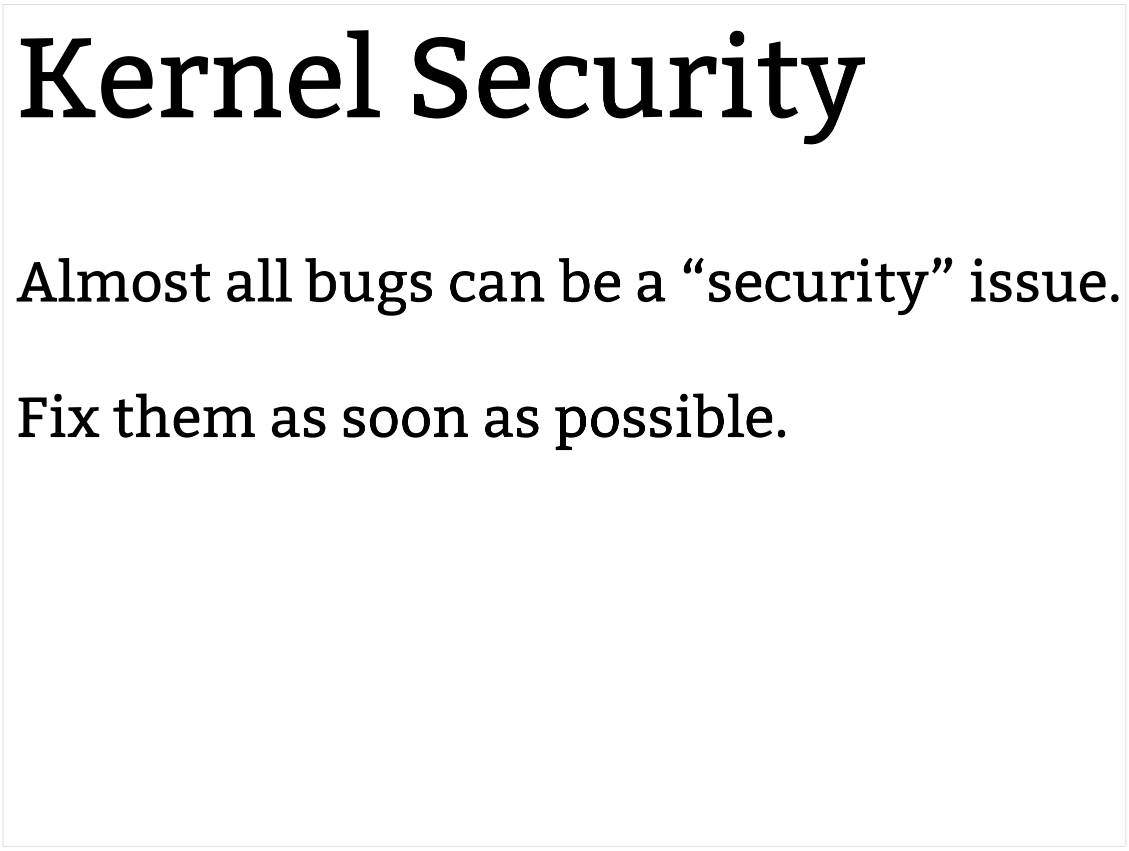


Kernel Security

Almost all bugs can be a “security” issue.

Anything that goes wrong in the kernel can usually be turned into a “security” problem.

Be it a DoS, or a reboot, or local root exploit, or worst case, a remote root exploit (very rare, thankfully.)



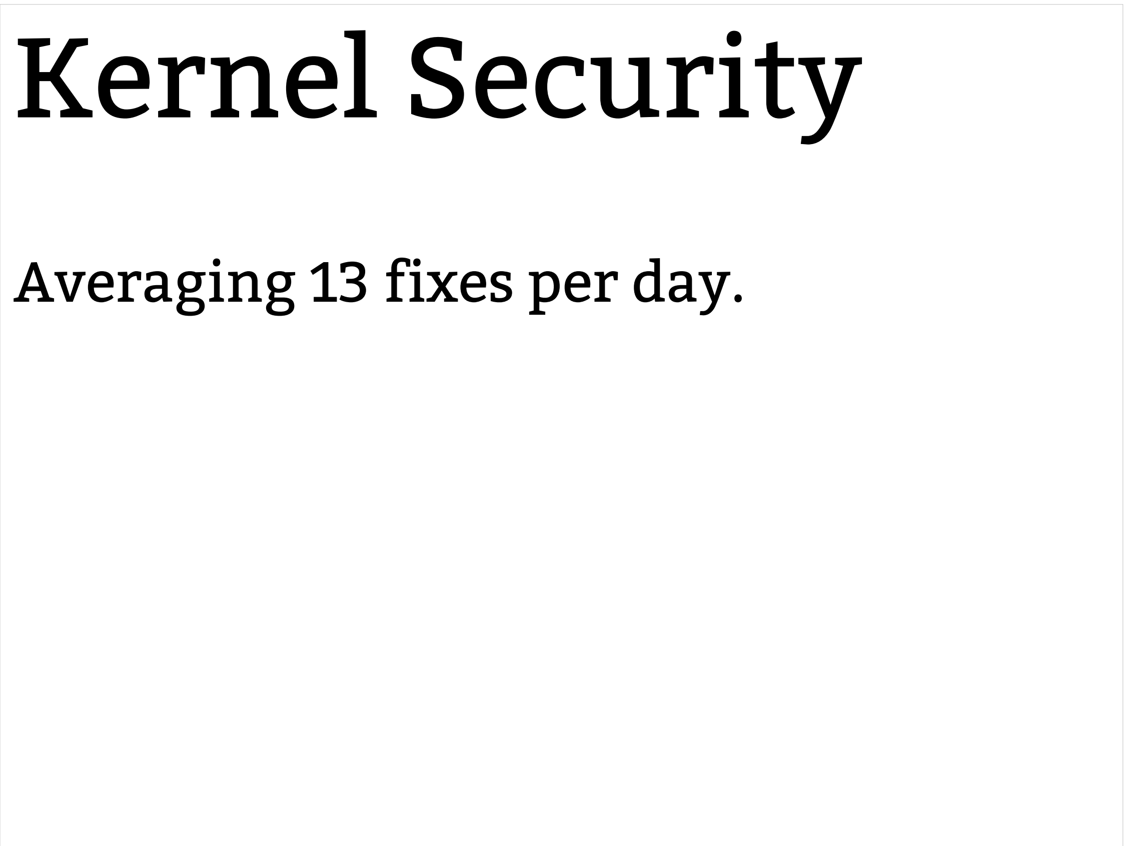
Kernel Security

Almost all bugs can be a “security” issue.

Fix them as soon as possible.

Because it’s really hard to determine if a bug is a “security” issue, our response is that we fix all bugs as soon as possible once we learn about them.

TTY bug in RH



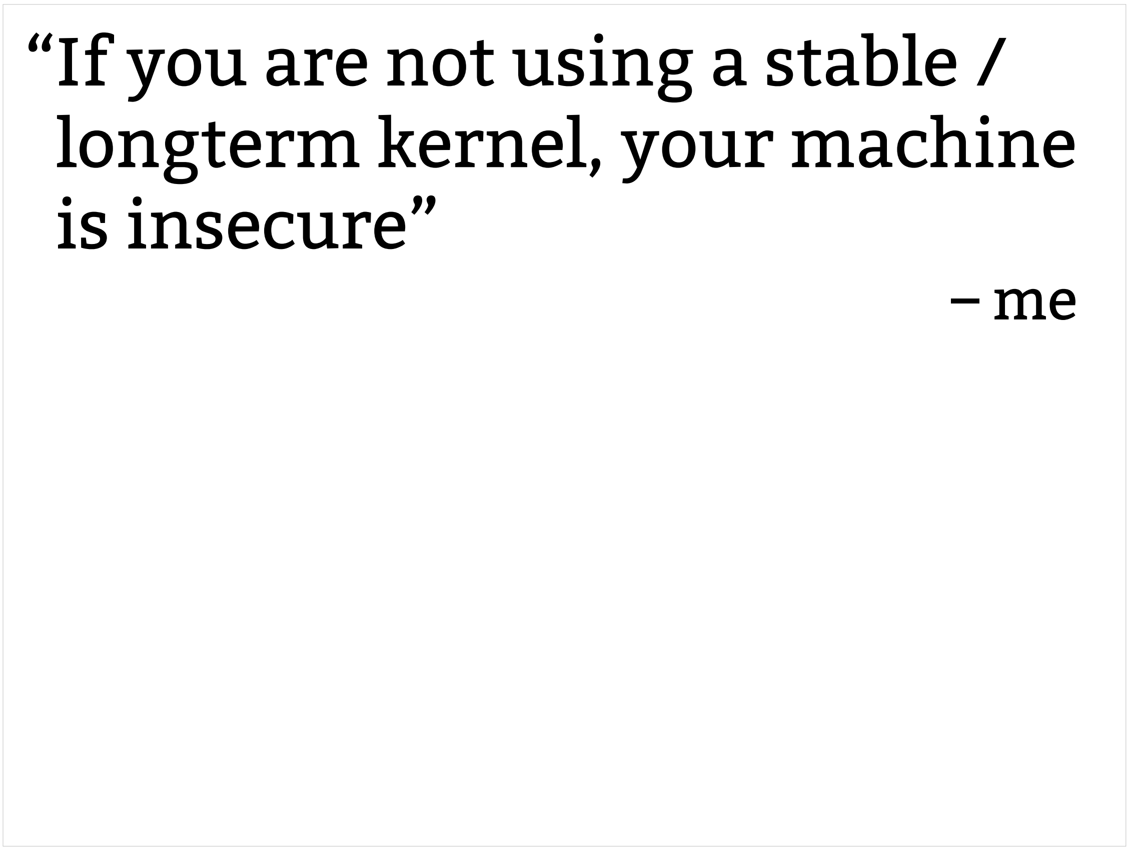
Kernel Security

Averaging 13 fixes per day.

If you look at the number of patches flowing into the stable tree, we are averaging 13 patches a day, every single day.

Now not all of them are “security” fixes. But some small percentage is.

This is for the latest kernel release, the 4.4 kernel is averaging 9 fixes a day, and 4.9 is still running at 13 fixes a day!



"If you are not using a stable / longterm kernel, your machine is insecure"

– me

Your infrastructure HAS to support updating the kernel. If you can't do that, you are insecure.

Even the "enterprise" kernels aren't keeping up with this rate of change, the exception being Debian.

If you use these kernels, you HAVE to keep up to date.

Android example.

“The kernel needs airbags”
– Konstantin Ryabitsev

slides.com/mricon/giant-bags-of-mostly-water#/

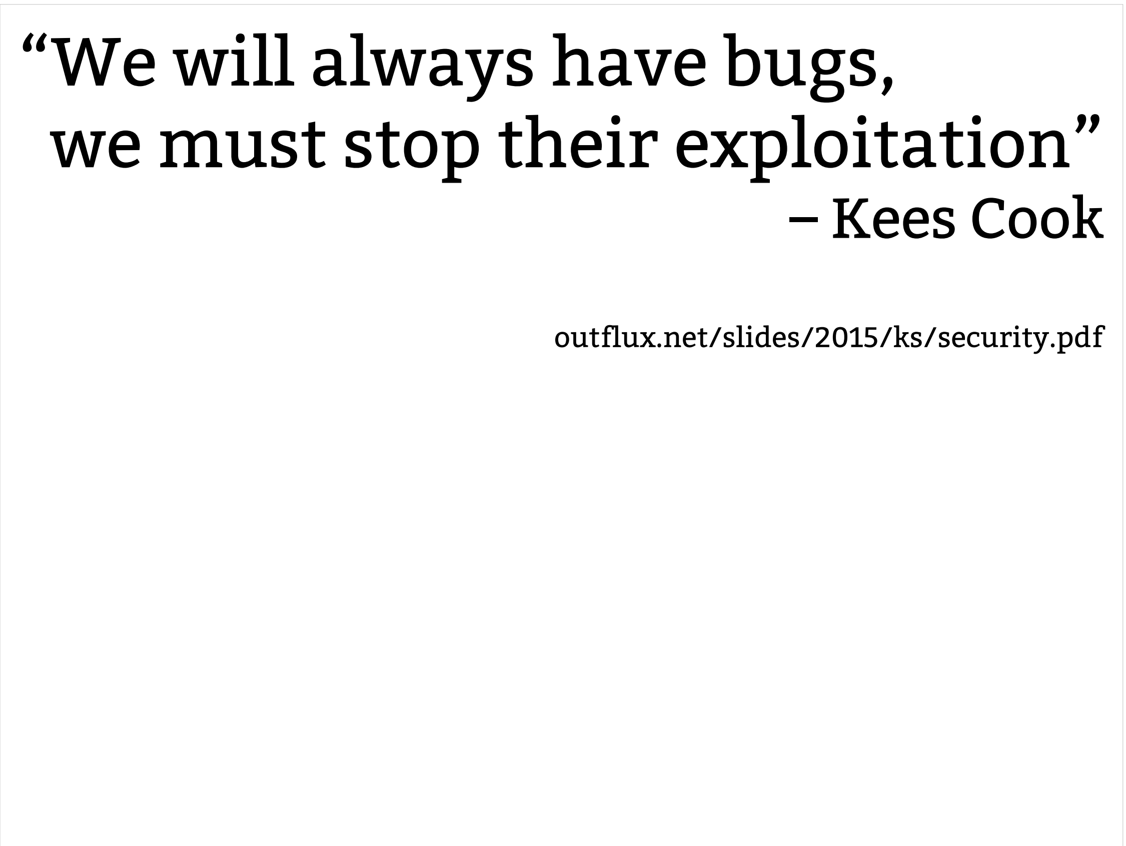
kernel.org sysadmin, in charge of the LF sysadmin team, Fedora infrastructure developer.

Great presentation on how you, as a sysadmin, can implement secure practices for your network. Full checklist and guide has been published.

But, even with those practices, we need low-level changes in order to save ourselves from the accidents that will happen.

We need “airbags” in the kernel, and elsewhere.

Things like SELinux, grsec, openwall we need them.



“We will always have bugs,
we must stop their exploitation”
– Kees Cook

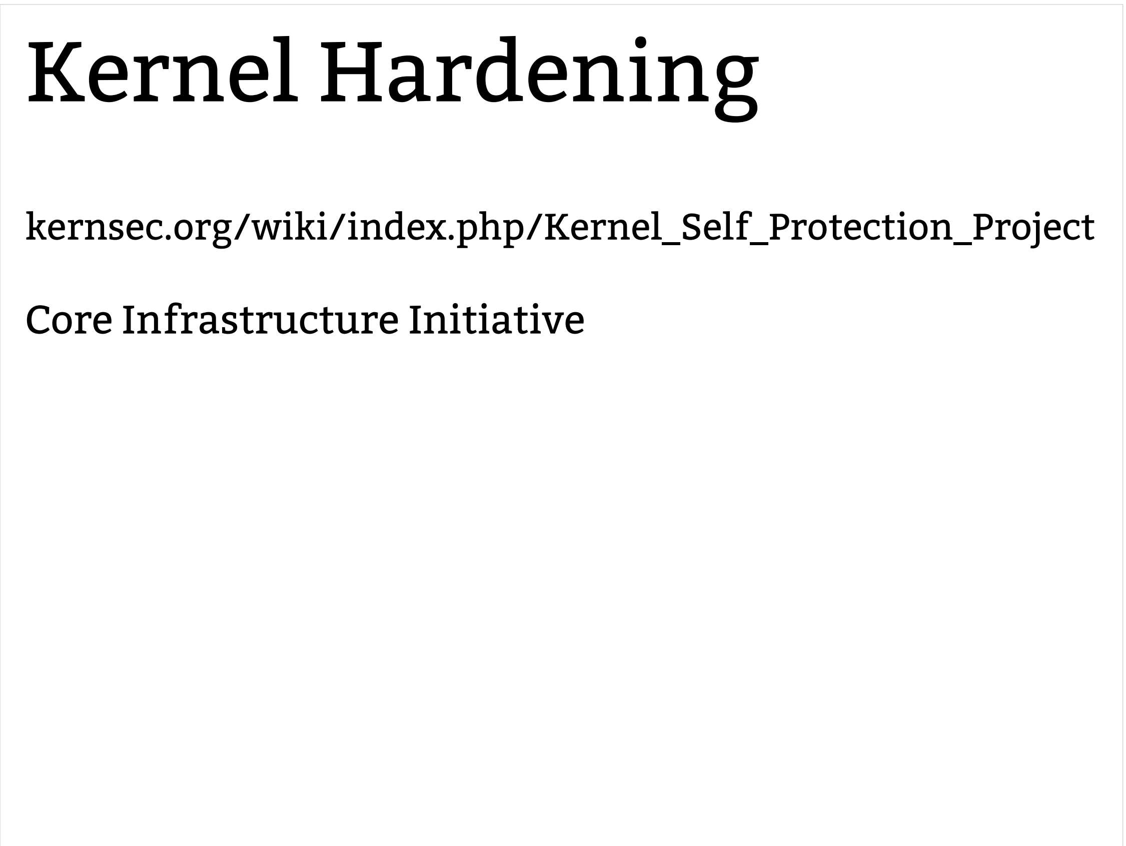
outflux.net/slides/2015/ks/security.pdf

Kees Cook, kernel security developer, presentation at kernel summit last year.

We need to start doing things to make the kernel more “robust” from a security standpoint.

Even if it makes things harder for the developers.

Everyone agreed.



Kernel Hardening

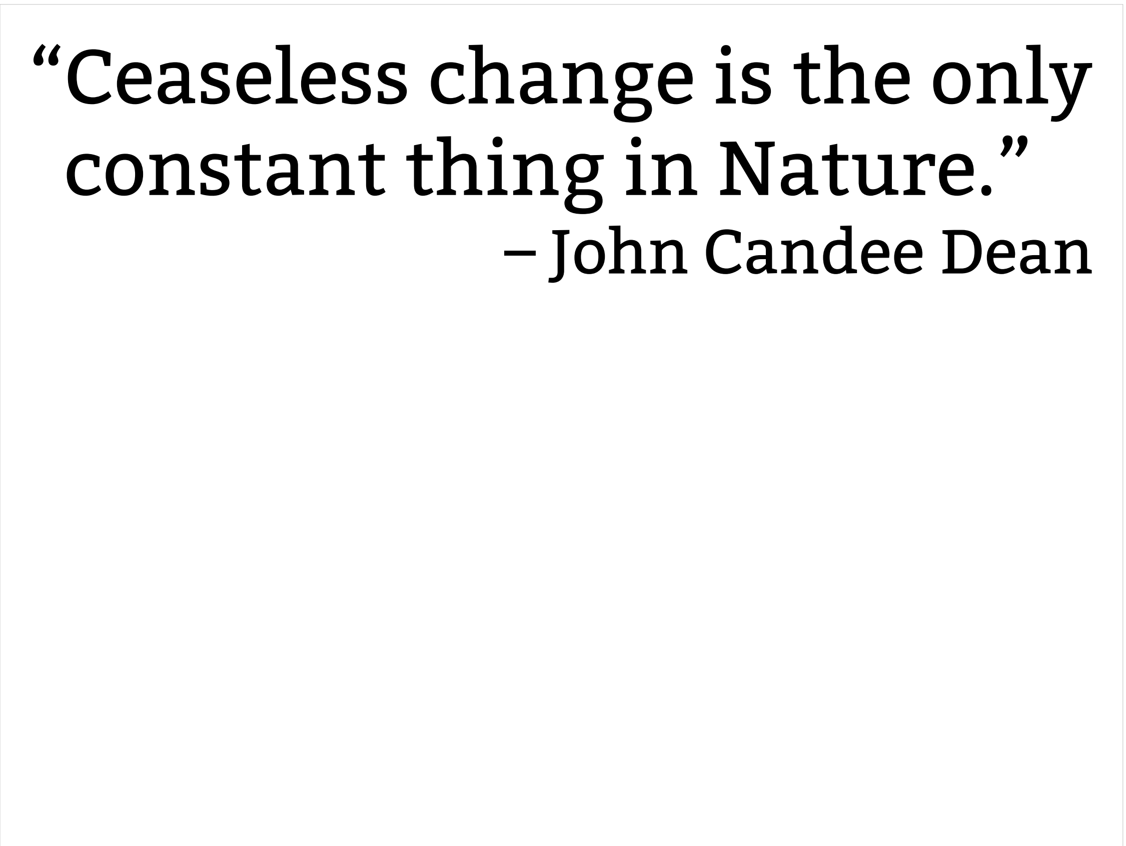
kernsec.org/wiki/index.php/Kernel_Self_Protection_Project

Core Infrastructure Initiative

Kernel hardening project.

New security features are being added in each release, but if you don’t upgrade, you don’t get those features, and protection.

CII is helping to fund this, if you want to work on it, we need developers, and we will pay for it.



“Ceaseless change is the only constant thing in Nature.”

– John Candee Dean

1911 astronomer.

If your operating system isn’t constantly changing, then it is dead. The world doesn’t stop changing, learn to embrace the change in order to survive.

“static systems” die.

github.com/gregkh/kernel-development

Obligatory Penguin Picture

[redacted]