Linux Kernel Security Process

or

"Why are there so many kernel CVEs now?"

Greg Kroah-Hartman
gregkh@linuxfoundation.org
git.sr.ht/~gregkh/presentation-security

THE LINUX FOUNDATION

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.

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

But hopefully I can convince them to agree with me.

Linux size – overall

85,000 files

38,640,000 lines

Kernel release 6.10.0

Linux size – what you use

5%-10%

~9 changes per hour

Kernel release 6.10.0

New* release model

- Release every 2-3 months
- All releases are stable

* As of January, 2004

“Cambridge promise”

- We will not break userspace

– July 2007

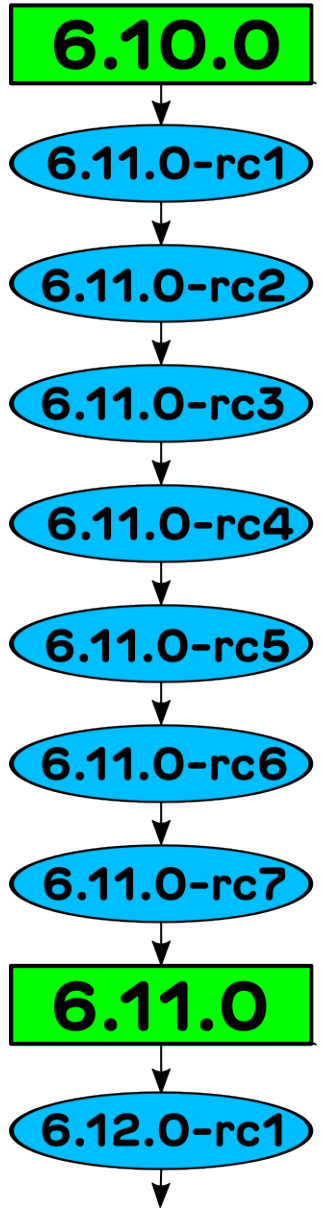
“Cambridge promise”

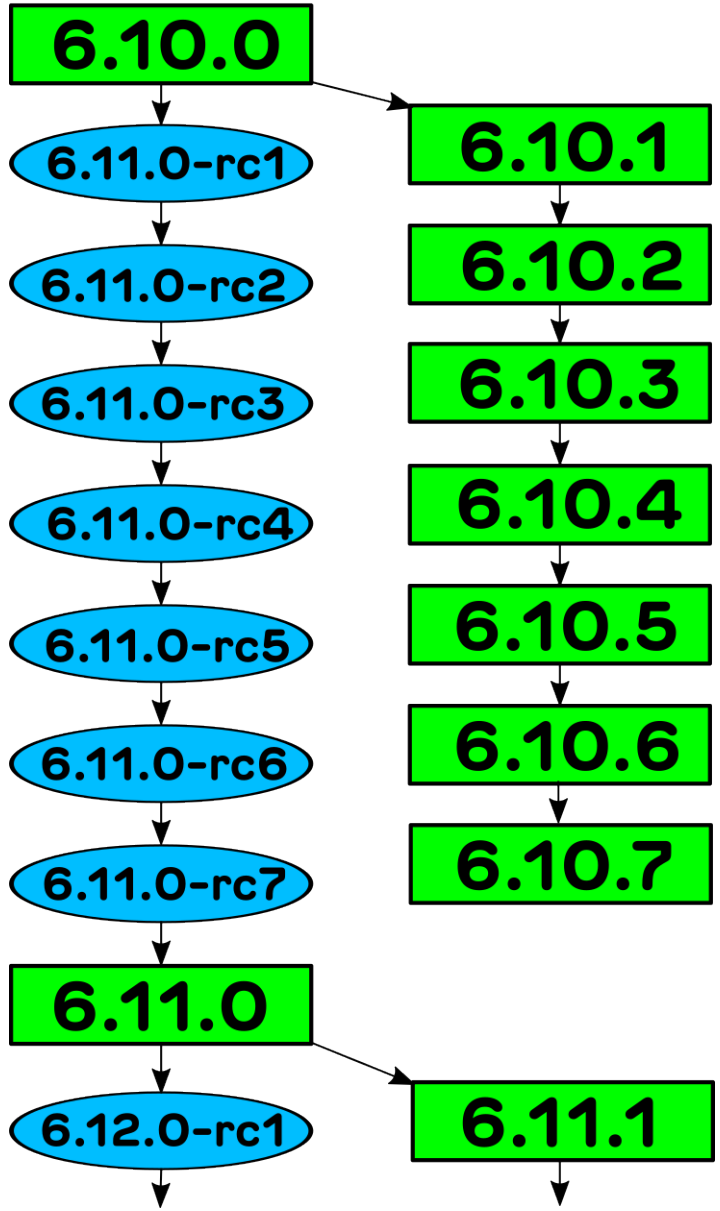
- We will not break userspace on purpose

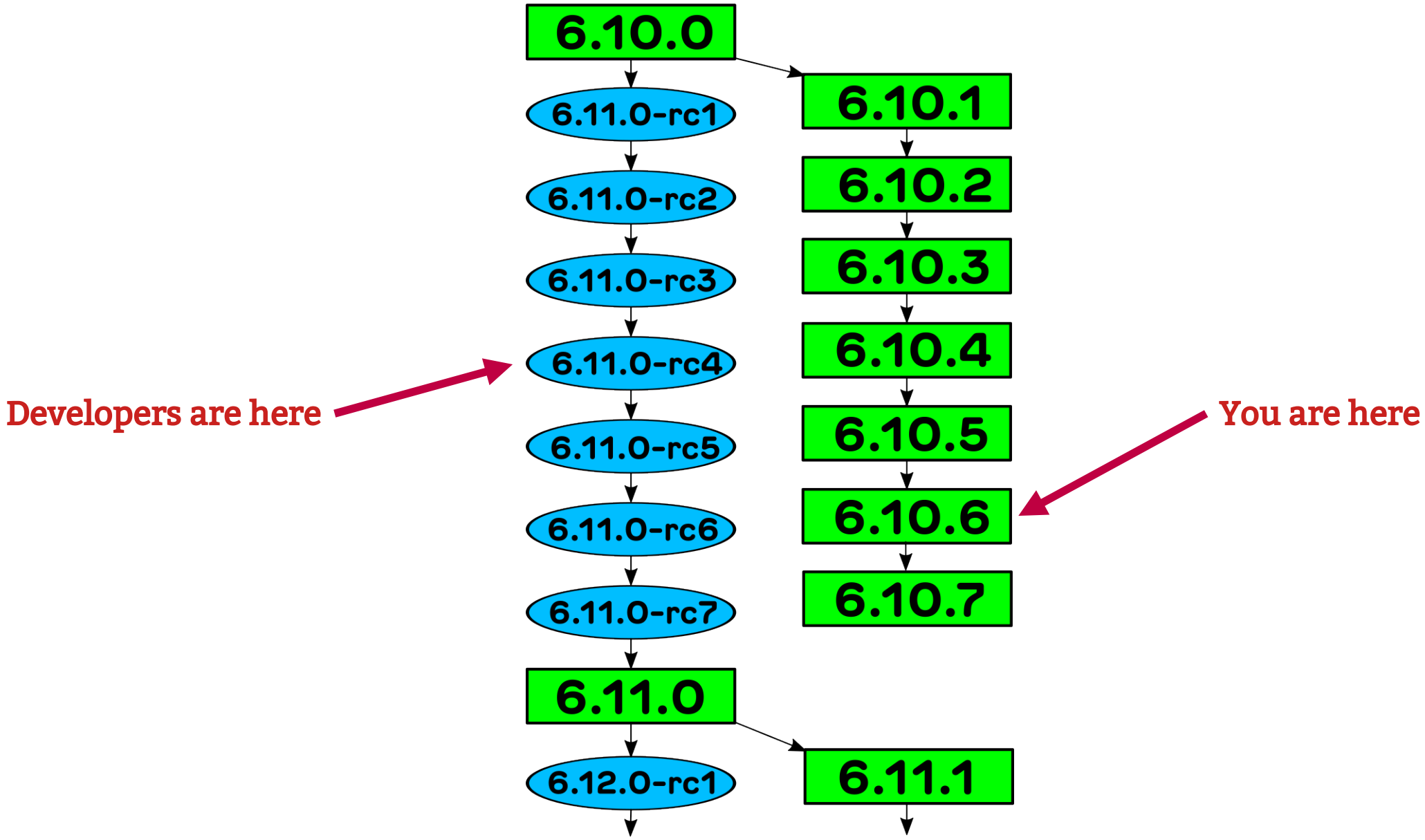
– July 2007

Version numbers mean nothing

2.6.x → 3.x	2011
3.x → 4.x	2015
4.x → 5.x	2019
5.x → 6.x	2022


6.10.0
6.11.0-rc1
6.11.0-rc2
6.11.0-rc3
6.11.0-rc4
6.11.0-rc5
6.11.0-rc6
6.11.0-rc7
6.11.0
6.12.0-rc1


6.10.0
6.11.0-rc1
6.11.0-rc2
6.11.0-rc3
6.11.0-rc4
6.11.0-rc5
6.11.0-rc6
6.11.0-rc7
6.11.0
6.12.0-rc1
6.10.1
6.10.2
6.10.3
6.10.4
6.10.5
6.10.6
6.10.7
6.11.1


6.10.0
6.11.0-rc1
6.11.0-rc2
6.11.0-rc3
6.11.0-rc4
6.11.0-rc5
6.11.0-rc6
6.11.0-rc7
6.11.0
6.12.0-rc1
6.10.1
6.10.2
6.10.3
6.10.4
6.10.5
6.10.6
6.10.7
6.11.1
Developers are here
You are here

Stable kernel 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

Longterm kernels

- One picked per year
- Maintained for at least 2 years*

4.19 5.4 5.10 5.15 6.1 6.6

* sometimes longer

Longterm kernels

4.19	14 changes / day
5.4	16 changes / day
5.10	21 changes / day
5.15	24 changes / day
6.1	29 changes / day
6.6	33 changes / day

Kernel releases

- Every release is stable
- 17+ year old guarantee to not break things
- No fear to ever upgrade

More release information in greater detail:

http://www.kroah.com/log/blog/2018/02/05/linux-kernel-release-model/

The world has changed

- 80%+ of the world's servers runs non-commercial distribution kernels*
- inter-company interactions achieve nothing
- The "community" does not sign NDAs

* Embedded it is like 99%, look at what is in your pocket or in your home

Kernel security team

Kernel security team

Reactive security, not proactive

Kernel security team

Other groups do proactive security

Kernel security team

- security@kernel.org
- Small group of kernel developers
- Do not represent any companies

Kernel security team

- Triage reports
- Drag in responsible developers
- Work to create a fix as soon as possible
- Get it merged into Linus's and stable trees

Kernel security team

- If you are brought in enough times, you are added to the alias to reduce the round-trip.

Kernel security team

- Fix the issue as soon as possible
- No embargoes longer than 7 days
- Do not do any kind of announcements

Linus in 2008

> On Wed, 16 Jul 2008, pageexec@freemail.hu wrote:
> >
> > you should check out the last few -stable releases then and see how
> > the announcement doesn't ever mention the word 'security' while fixing
> > security bugs

Umm. What part of "they are just normal bugs" did you have issues with?
I expressly told you that security bugs should not be marked as such,
because bugs are bugs.

> > in other words, it's all the more reason to have the commit say it's
> > fixing a security issue.

No.

> > I'm just saying that why mark things, when the marking have no meaning?
> > People who believe in them are just _wrong_.
>
> what is wrong in particular?

You have two cases:

- people think the marking is somehow trustworthy.

  People are WRONG, and are misled by the partial markings, thinking that
  unmarked bugfixes are "less important". They aren't.

- People don't think it matters

  People are right, and the marking is pointless.
  In either case it's just stupid to mark them. I don't want to do it,
  because I don't want to perpetuate the myth of "security fixes" as a
  separate thing from "plain regular bug fixes".

  They're all fixes. They're all important. As are new features, for that
  matter.

> when you know that you're about to commit a patch that fixes a security
> bug, why is it wrong to say so in the commit?

It's pointless and wrong because it makes people think that other bugs
aren't potential security fixes.

What was unclear about that?

Linus

Above email:

https://lore.kernel.org/lkml/alpine.LFD.1.10.0807151620450.2867@woody.linux-foundation.org/

Whole thread:

https://lore.kernel.org/lkml/20080703035807.GA8190@kroah.com/

Reporting Security bugs:

https://www.kernel.org/doc/html/latest/admin-guide/security-bugs.html

Kernel security policy

- Almost all bugs can be a “security” issue

Kernel security policy

- A fix for a known bug is better than the potential of a fix causing a future problem as future problems, when found, will be fixed then.

Kernel security policy

- We do NOT know your use case!

Kernel security policy

- We do NOT know your use case!
- We do NOT know what code you use!

Kernel security policy

- We do NOT know your use case!
- We do NOT know what code you use!
- We do NOT want to know any of this!

“It's hard to capture the fact that a bug can be super serious in one type of deployment, somewhat important in another, or no big deal at all -- and that the bug can be all of this at the same time. Vulnerability remediation is hard.”

– Ben Hawkes

https://blog.isosceles.com/what-is-a-good-linux-kernel-bug/

Kernel security policy

- Fix known bugs as soon as possible
- Get releases out to users quickly
- Does not work for hardware bugs*

* Hardware vendors think they are special,
They are not, they are just slow...

Hardware security issues

- Handled separately
- Encrypted restricted email list
- No NDAs
- Cross company / OS coordination
- Embargos are tolerated*

* for now...

Hardware security issues

- How this works:

  `https://www.kernel.org/doc/html/latest/process/embargoed-hardware-issues.html`

Kernel security team

- Does not do any kind of annoucements
- Can not assign CVEs*
- No early annoucement list

* This is on purpose

No pre-disclosure at all!

- All “early notice” lists are leaks and should be considered public.

No pre-disclosure at all!

- All “early notice” lists are leaks and should be considered public.
- Unless your project is not used by anyone.

No pre-disclosure at all!

- All “early notice” lists are leaks and should be considered public.
- Unless your project is not used by anyone.
- Otherwise, why would your government allow it to exist?

Security fixes

- Happen at least once a week
- Look like any other bugfix
- Many not known to be security related until years later
- No differentiation between bug types

CVEs used to mean nothing for Linux

https://kernel-recipes.org/en/2019/talks/cves-are-dead-long-live-the-cve/

CVEs now mean something!

https://www.cve.org/Media/News/item/news/2024/02/13/kernel-org-Added-as-CNA

http://www.kroah.com/log/blog/2024/02/13/linux-is-a-cna/

kernel.org is now a CNA!

- Responsible for all kernel CVEs
- Community is in control
- CVEs assigned for all “vulnerabilities” fixed

Linux CVE resources

- Contact us:

  cve@kernel.org

- Process documentation:

  https://docs.kernel.org/process/cve.html

- Public git repo:

  https://git.kernel.org/pub/scm/linux/security/vulns.git/

- List of all assigned CVEs:

  https://lore.kernel.org/linux-cve-announce/

What is a vulnerability?

“An instance of one or more weaknesses in a Product that can be exploited, causing a negative impact to confidentiality, integrity, or availability; a set of conditions or behaviors that allows the violation of an explicit or implicit security policy.”

What is a kernel vulnerability?

- Any user-triggerable crash
- Memory use-after-free / leak / overflow
- Incorrect boundary checks
- Denial of service
- Logic errors
- Lots of other things

Why is triggering `WARN_ON` a CVE?

- If `panic_on_warn` is enabled, reboot happens
- Billions of Linux systems have this enabled
- Android is slowly moving away from this.
- Some want any warning to reboot

What is a not a kernel vulnerability?

- Data corruption / loss
- Performance issues
- Bugfixes that are not externally triggered

How are they assigned

- CVE team reviews every stable commit
- Votes if is, or is not, a vulnerability
- Each team member uses different methods
- Commits that are agreed get assigned
- Commits that are disagreed are discussed
- Review happens in public

How are they assigned

- Community requests

CVEs for everyone!

- Averaging 60 CVEs per week
- Every CVE says what files are affected
- Every CVE says what versions are affected
- Very few CVEs are applicable for you!

Most CVEs are not for you!

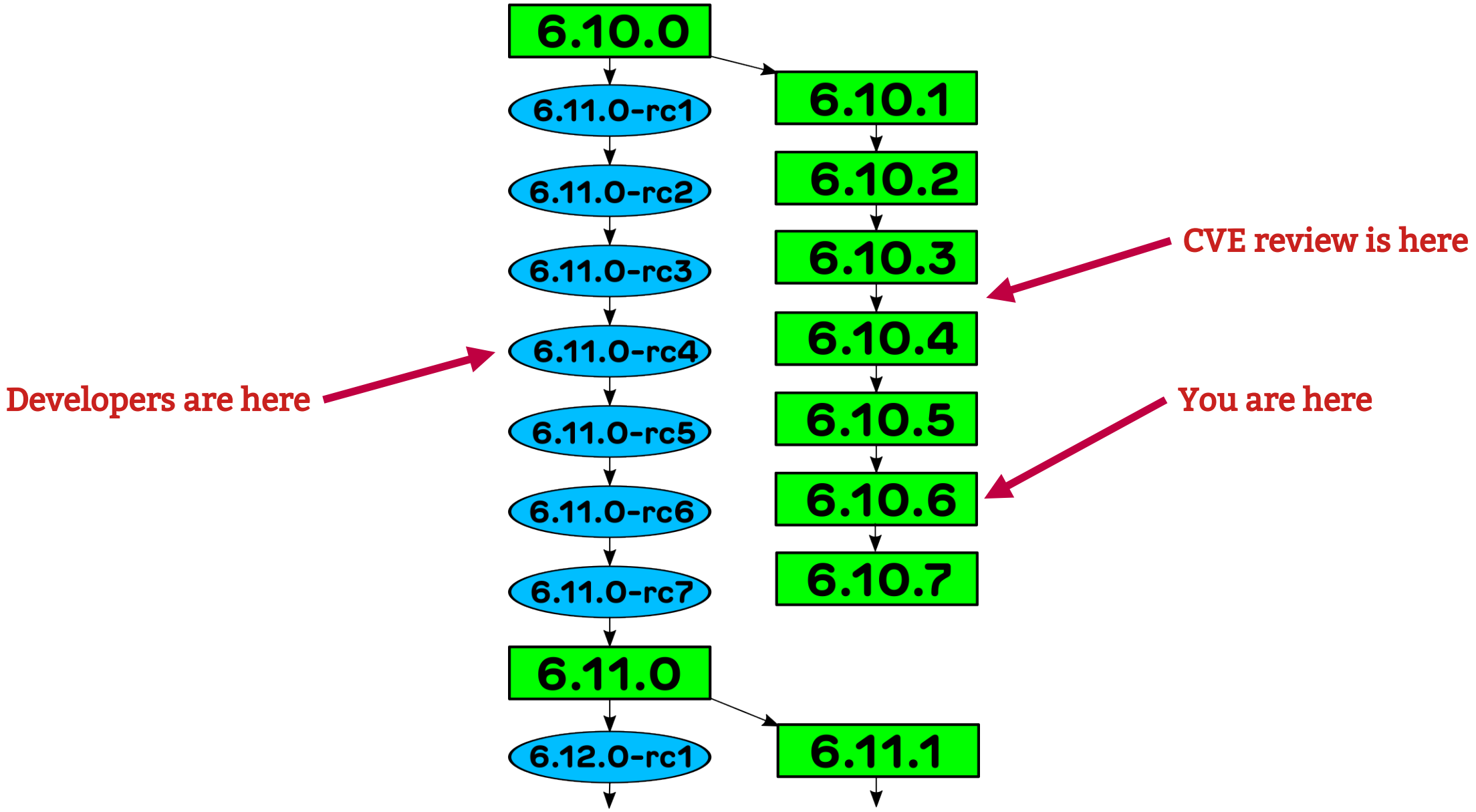
- Only you know your usecase!
- Only you know what files you use!

Some CVEs are for you!

- Real issues get fixed every week
- Ignoring them will affect your security

CVEs are assigned after-the-fact

- Usually 1-2 week delay
- Allows systems to be updated before public announcements
- CVEs only reference specific fix, not any previous changes needed.
- CVE fixes are NOT tested independently


6.10.0
6.11.0-rc1
6.11.0-rc2
6.11.0-rc3
6.11.0-rc4
6.11.0-rc5
6.11.0-rc6
6.11.0-rc7
6.11.0
6.12.0-rc1
6.10.1
6.10.2
6.10.3
6.10.4
6.10.5
6.10.6
6.10.7
6.11.1
Developers are here
CVE review is here
You are here

Staying secure

- Take all stable/longterm updates
- Community supported and tested as a whole
- Bonus is you get data corruption fixes and performance improvements for free!

“If you are not using the latest a stable / longterm kernel, your system is insecure”

– me

Linux Kernel Security Process

or

"Why are there so many kernel CVEs now?"

Greg Kroah-Hartman
gregkh@linuxfoundation.org
git.sr.ht/~gregkh/presentation-security

THE LINUX FOUNDATION

Bonus Slides!

Why talk about this now?

- European Union Cyber Resiliance Act (CRA)

  https://www.linuxfoundation.org/blog/understanding-the-cyber-resilience-act

  https://linuxfoundation.eu/cyber-resilience-act

- Companies keep asking to join the “security team”
- Companies keep asking for “early security notices”
- Hardware embargoed issues are a pain

Reporting security issues

On Wed, 16 Jul 2008, pageexec@freemail.hu wrote:
>
> we went through this and you yourself said that security bugs are *not*
> treated as normal bugs because you do omit relevant information from such
> commits

Actually, we disagree on one fundamental thing. We disagree on
that single word: "relevant".

I do not think it's helpful _or_ relevant to explicitly point out how to
tigger a bug. It's very helpful and relevant when we're trying to chase
the bug down, but once it is fixed, it becomes irrelevant.

You think that explicitly pointing something out as a security issue is
really important, so you think it's always "relevant". And I take mostly
the opposite view. I think pointing it out is actually likely to be
counter-productive.

Reporting security issues

For example, the way I prefer to work is to have people send me and the kernel list a patch for a fix, and then in the very next email send (in private) an example exploit of the problem to the security mailing list (and that one goes to the private security list just because we don't want all the people at universities rushing in to test it). THAT is how things should work.

Should I document the exploit in the commit message? Hell no. It's private for a reason, even if it's real information. It was real information for the developers to explain why a patch is needed, but once explained, it shouldn't be spread around unnecessarily.

Linus

Above email:

https://lore.kernel.org/lkml/alpine.LFD.1.10.0807151716510.2867@woody.linux-foundation.org/

From: Steve Bergman <steve@rueb.com>
To: linux-kernel@vger.kernel.org
Subject: Proper procedure for reporting possible security vulnerabilities?
Date: Mon, 10 Jan 2005 10:46:57 -0600

There seems to be some confusion in certain quarters as to the proper
procedure for reporting possible kernel security issues.
REPORTING-BUGS says send bug reports to the maintainer of that area of
the kernel. However, what about areas for which a maintainer is not
listed? (e.g. VM) It seems that some take that to mean send it
directly to Linus and if you don't hear something back quickly, release
an exploit to the wild.

So what is the preferred procedure and is it documented somewhere?
Should it be made more prominent?

Thanks for any information,
Steve Bergman

https://lore.kernel.org/lkml/41E2B181.3060009@rueb.com/

```
From: Chris Wright <chrisw@osdl.org>
To: torvalds@osdl.org
Cc: akpm@osdl.org, alan@lxorguk.ukuu.org.uk,
    marcelo.tosatti@cyclades.com, linux-kernel@vger.kernel.org
Subject: [PATCH] Security contact info
Date: Wed, 9 Mar 2005 01:05:50 -0800

Add security contact info and relevant documentation.

Signed-off-by: Chris Wright <chrisw@osdl.org>

 MAINTAINERS                |    5 +++++
 REPORTING-BUGS             |    4 ++++
 Documentation/SecurityBugs |   38 ++++++++++++++++++++++++++++++++++++++
 3 files changed, 47 insertions(+)
```

https://lore.kernel.org/all/20050309090550.GW28536@shell0.pdx.osdl.net/

```
From: Chris Wright <chrisw@osdl.org>
To: torvalds@osdl.org
Cc: akpm@osdl.org, alan@lxorguk.ukuu.org.uk,
    marcelo.tosatti@cyclades.com, linux-kernel@vger.kernel.org
Subject: [PATCH] Security contact info
Date: Wed, 9 Mar 2005 01:05:50 -0800

Add security contact info and relevant documentation.

Signed-off-by: Chris Wright <chrisw@osdl.org>

...


+3) Non-disclosure agreements
+
+The Linux kernel security team is not a formal body and therefore unable
+to enter any non-disclosure agreements.
```

https://lore.kernel.org/all/20050309090550.GW28536@shell0.pdx.osdl.net/