

Linux (Kernel)

Linux (deutsch [ˈliːnʊks]) ist ein Betriebssystem-Kernel, der im Jahr 1991 von Linus Torvalds ursprünglich für die 32-Bit-x86-Architektur IA-32 entwickelt und ab Version 0.12 unter der freien GNU General Public License (GPL) veröffentlicht wird.^[4] Er ist heute Teil einer Vielzahl von Betriebssystemen.

Der Name *Linux* setzt sich zusammen aus dem Namen Linus und einem X für das als Vorbild dienende Unix. Er bezeichnet im weiteren Sinne mittlerweile nicht mehr nur den Kernel selbst, sondern übertragen davon ganze Linux-basierte Systeme und Distributionen. Dies führte zum GNU/Linux-Namensstreit.

Inhaltsverzeichnis

Grundlegende Technologie

- Aufgaben des Kernels
- Programmiersprache
- Funktionsweise
- Schnittstellen
- Architektur
- Portierbarkeit
 - Binärschnittstellen der Arm-Architektur
 - User Mode Linux
 - µClinux

Entwicklungsprozess

- Änderungen der Herkunftskontrolle
- Das Versionskontrollsystem Git
- Kernel-Versionen
- Versionsnummern-Schema
- Entwicklerversion
- Pflege der Kernel-Versionen
- Versionen mit Langzeitunterstützung

Versionsgeschichte

- Zeittafeln
- Versionsgeschichte bis Version 2.6
- Versionsgeschichte ab Version 2.6
- Neuerungen im Kernel 2.6
 - Neue Prozess-Scheduler
 - Präemptiver Kernel
 - Zugriffskontrolllisten
 - Inotify
 - Weitere wichtige Änderungen

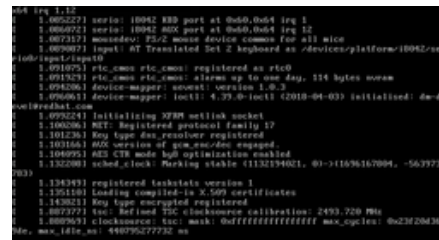
Lizenzbesonderheiten

- Proprietärer Code und Freiheitsbegriff
- Der Kernel unter der GPL 2

Literatur

Weblinks

Linux



Startender Systemkern (Version 3.0.0) in Debian

Entwickler	Linus Torvalds u. v. m.
Lizenz(en)	GPLv2 ^(only) , ^{[1][2]} enthält verschiedene Closed Source Binärblobs ^[3]
Erstveröff.	17. September 1991
Akt. Version	5.18 (22. Mai 2022)
Architektur(en)	IA-32 (inkl. x64), Alpha AXP, SPARC, Motorola 68k, PowerPC, POWER, Arm, Hitachi SuperH, z Systems, MIPS, PA-RISC, IA-64, AVR32, Renesas H8/300 und weitere
Sprache(n)	Englisch
	www.kernel.org (https://www.kernel.org/)

Die Binärschnittstelle des Kernels ist unerheblich, auf das komplette Betriebssystem kommt es an. Die Linux Standard Base (LSB) soll es ermöglichen, kommerzielle Programme unverändert zwischen Linux Betriebssystemen zu portieren. Die interne

Binärschnittstelle ist nicht stabil, und es gibt keinerlei Bestrebungen dies zu ändern; dies hat zur Folge, dass ein internes Modul, welches z. B. für Linux 3.0 kompiliert worden ist, höchstwahrscheinlich nicht mit Linux-Kernel 3.1 zusammenarbeiten wird. Dies ist eine ganz bewusste Entscheidung.^[5]

Architektur

Anwenderprogramme (z.B. Textverarbeitung, Tabellenkalkulation oder Browser)	User Mode
Complex Libraries (GLib, GTK+, Qt, SDL, EFL)	
Simple Libraries <i>sin</i> , <i>opendbm</i>	
C-Standard-Bibliothek: <i>glibc open, exec, sbrk, socket, fopen, calloc</i>	
Systemaufrufe <i>TRAP, CALL, BRK, INT</i> (je nach Hardware)	Kernel Mode
Kernel (Gerätetreiber, Prozesse, Netzwerk, Dateisystem)	
Hardware (Prozessor(en), Speicher, Geräte)	

Linux ist ein monolithischer Kernel. Die Treiber im Kernel und die Kernel-Module laufen im privilegierten Modus (x86: Ring 0), haben also unbeschränkten Zugriff auf die Hardware. Einige wenige Module des Kernels laufen im eingeschränkten Benutzermodus (x86: Ring 3). Die Level 1 und 2 der x86-Architektur werden von Linux nicht genutzt, da sie auf vielen anderen Architekturen nicht existieren und der Kernel auf allen unterstützten Architekturen im Wesentlichen gleich funktionieren soll.

Nahezu jeder Treiber kann auch als Modul zur Verfügung stehen und vom System dann dynamisch nachgeladen werden. Ausgenommen davon sind Treiber, die für das Starten des Systems verantwortlich sind, bevor auf das Dateisystem zugegriffen werden kann. Man kann allerdings den Kernel so konfigurieren, dass ein CramFS- oder Initramps-Dateisystem vor dem tatsächlichen Root-Dateisystem geladen wird, welches die weiteren für den Startprozess notwendigen Module enthält. Dadurch kann die Kernelgröße verringert und die Flexibilität drastisch erhöht werden.

Im System laufende Programme bekommen wiederum vom Kernel Prozessorzeit zugewiesen. Jeder dieser Prozesse erhält einen eigenen, geschützten Speicherbereich und kann nur über Systemaufrufe auf die Gerätetreiber und das Betriebssystem zugreifen. Die Prozesse laufen dabei im Benutzermodus (*user mode*), während der Kernel im Kernel-Modus (*kernel mode*) arbeitet. Die Privilegien im Benutzermodus sind sehr eingeschränkt. Abstraktion und Speicherschutz sind nahezu vollkommen, ein direkter Zugriff wird nur sehr selten und unter genau kontrollierten Bedingungen gestattet. Dies hat den Vorteil, dass kein Programm z. B. durch einen Fehler das System zum Absturz bringen kann.

Linux stellt wie sein Vorbild Unix eine vollständige Abstraktion und Virtualisierung für nahezu alle Betriebsmittel bereit (z. B. virtueller Speicher, Illusion eines eigenen Prozessors usw.).

Die Tatsache, dass Linux nicht auf einem Microkernel basiert, war Thema eines berühmten Flame Wars zwischen Linus Torvalds und Andrew S. Tanenbaum. Anfang der 1990er Jahre, als Linux entwickelt wurde, galten monolithische Kernels als obsolet (Linux war zu diesem Zeitpunkt noch rein monolithisch). Die Diskussion und Zusammenfassungen sind im Artikel Geschichte von Linux näher beschrieben.

Durch Erweiterungen wie FUSE und durch die zunehmende Verwendung von Kernel-Prozessen fließen mittlerweile auch zahlreiche Microkernel-Konzepte in Linux ein.

Portierbarkeit

Obwohl Linus Torvalds eigentlich nicht beabsichtigt hatte, einen portierbaren Kernel zu schreiben, hat sich Linux dank des GNU Compilers GCC weitreichend in diese Richtung entwickelt. Es ist inzwischen eines der am häufigsten portierten Systeme (nur noch NetBSD läuft auf etwa gleich vielen Architekturen). Das Repertoire reicht dabei von eher selten anzutreffenden Betriebsumgebungen wie dem iPAQ-Handheld-Computer, Digitalkameras oder Großrechnern wie IBMs System z bis hin zu normalen PCs.

Obwohl die Portierung auf die S/390 ursprünglich ein vom IBM-Management nicht genehmigtes Unterfangen war (*siehe auch: Skunk works*), plant IBM auch die nächste IBM-Supercomputergeneration Blue Gene mit einem eigenen Linux-Port auszustatten.

Ursprünglich hatte Torvalds eine ganz andere Art von Portierbarkeit für sein System angestrebt, nämlich die Möglichkeit, freie GPL- und andere quelloffene Software leicht unter Linux kompilieren zu können. Dieses Ziel wurde bereits sehr früh erreicht und macht sicherlich einen guten Teil des Erfolges von Linux aus, da es jedem eine einfache Möglichkeit bietet, auf einem

freien System freie Software laufen zu lassen.

Die ersten Architekturen, auf denen Linux lief, waren die von Linus Torvalds verwendeten Computer:^[6]

- IA-32 (x86 ab dem i386) – Linus hatte ab 1991 einen PC mit Intel-386DX-33-MHz-Prozessor, 4 MB RAM und 40 MB Festplatte.
- Alpha – Linus arbeitete von 1994 bis 1995 an der Portierung auf die 64-Bit-Alpha-Architektur (auf einem DEC-Alpha-Rechner, den er als Leihgabe erhalten hatte).

Damit war Linux sehr früh 64-Bit-fähig (Linux 1.2 erschien 1995) und durch die Portierung auf Alpha war der Weg für weitere Portierungen frei. Zeitgleich arbeitete der Student Dave Miller ab 1993 an der Portierung auf SPARC von Sun Microsystems, einer damals weit verbreiteten Architektur. Doch lief Linux 2.0 von Mitte 1996 offiziell auf IA-32 und Alpha, konnte aber bereits SMP.

Mit Linux 2.2 vom Januar 1999 kamen folgende Ports hinzu:^[6]

- SPARC/UltraSPARC von Oracle (ursprünglich von Sun)
- 68000 (m68k) von Motorola
- PowerPC (ppc) von Apple, IBM und Motorola (AIM-Allianz)

Mit Linux 2.4 vom Januar 2001 schließlich kamen folgende Architekturen hinzu:^[6]

- Itanium (IA-64) von Intel und HP
- S/390 von IBM
- SuperH von Hitachi
- MIPS

Trotz der unterstützten Befehlssatzarchitekturen (englisch *Instruction Set Architecture*, kurz ISA) ist für die Lauffähigkeit mehr nötig, sodass Linux gegenwärtig auf u. a. folgenden Plattformen und Architekturen läuft:

- Acorn Archimedes, A5000 und Risc-PC-Serie (Arm, StrongARM, Intel XScale usw.)
- Alpha von Compaq
- Atmel AVR32
- Axis Communications' CRIS
- Blackfin
- Hitachi H8/300
- Hewlett-Packard PA-RISC
- Itanium bzw. IA-64: Computer der nach 2017 eingestellten Itanium-Prozessorarchitektur von Intel und HP; von Intel „IA-64“ bezeichnet
- IBM S/390 und z Systems
- MIPS: Maschinen von Silicon Graphics ...
- Motorola 68020 und neuer: spätere Amigas, einige Atari und viele Apple Computer ab 1987 (Macintosh II) bis 1995 (siehe Linux68k)
- NEC v850e
- PowerPC: die meisten Apple Computer zwischen 1994 und 2006 (alle PCI-basierten Power Macintosh, der Nintendo GameCube; begrenzte Unterstützung für NuBus Power Macs bis Kernel 2.4.x, die weitere Entwicklung wurde in das Projekt *PPC/Linux for NuBus Power Macs* ausgelagert^[7]), Clones der Power Macs von Power Computing, UMAX und Motorola, mit einer „Power-UP“-Karte verbesserte Amigas (z. B. Blizzard oder CyberStorm) bzw. dessen Nachfolger AmigaOne, sowohl IBM Power als auch PowerPC-basierte IBM RS/6000-Systeme, verschiedene eingebettete PowerPC-Plattformen
- Sun/Oracle SPARC und UltraSparc: Workstations von Sun- bzw. Oracle
- SuperH von Hitachi: Sega Dreamcast
- OpenRISC
- x86-Architektur:
 - x86-16-Bit: im Rahmen des ELKS-Projektes werden IBM-PC-kompatible Computer mit den 16-Bit-Prozessoren 8086, 8088, 80186, 80188, 80286 und dazu kompatiblen unterstützt. Die Abkürzung „ELKS“ steht für *Embeddable Linux Kernel Subset* und stellt eine Untermenge des Linux-2.0/2.1-Kernels dar. Das angestrebte Ziel ist Kompatibilität zu Unix-V7 auf 16-Bit-x86-Systemen.

- IA-32, 32-Bit: IBM-PCs und kompatibel mit Intel 80386 (bis Kernel 3.7, Anfang 2013)^[8] bzw. Intel 80486 und dazu kompatiblen x86-Prozessor-Architekturen nachfolgender Generationen
- IA-32, 64-Bit: Unter der Bezeichnung „amd64“ oder x64 werden Prozessoren der IA-32-Architektur verstanden, die einen 64-Bit-Modus bieten (implementiert als AMD64 oder Intel 64). Dabei handelt es sich um die Nachfolger der ursprünglichen IBM-PC-kompatiblen Computer, die einen x86-Prozessor mit AMD Opteron, und dazu kompatible nachfolgende Generationen, nutzen, beispielsweise ab AMDs Athlon 64, Turion, Phenom, Phenom II, Bulldozer sowie Intels Core 2, Core i, Xeon

Binärschnittstellen der Arm-Architektur

Linux unterstützt zwei verschiedene Binärschnittstellen für Arm-Prozessoren. Die ältere Binärschnittstelle wird mit dem Akronym *OABI* (*old application binary interface*) bezeichnet und unterstützt die Prozessorarchitekturen bis einschließlich ARMv4, während die neuere Binärschnittstelle, die mit *EABI* (*embedded application binary interface*) bezeichnet wird, die Prozessorarchitekturen ab einschließlich ARMv4 unterstützt. Der bedeutendste Unterschied der Binärschnittstellen in Bezug auf Systemleistung ist die sehr viel bessere Unterstützung von Software-emulierten Gleitkommarechnungen durch EABI.^[9]

User Mode Linux

Ein besonderer Port ist das *User Mode Linux*. Prinzipiell handelt es sich dabei um einen Port von Linux auf sein eigenes Systemcall-Interface. Dies ermöglicht es, einen Linux-Kernel als normalen Prozess auf einem laufenden Linux-System zu starten. Der User-Mode-Kernel greift dann nicht selbst auf die Hardware zu, sondern reicht entsprechende Anforderungen an den echten Kernel durch. Durch diese Konstellation werden „Sandkästen“ ähnlich den virtuellen Maschinen von Java oder den jails von FreeBSD möglich, in denen ein normaler Benutzer Root-Rechte haben kann, ohne dem tatsächlichen System schaden zu können.

µClinux

µClinux ist eine Linux-Variante für Computer ohne Memory Management Unit (MMU) und kommt vorwiegend auf Mikrocontrollern und eingebetteten Systemen zum Einsatz. Seit Linux-Version 2.6 ist µClinux Teil des Linux-Projektes.

Entwicklungsprozess

Die Entwicklung von Linux liegt durch die GNU General Public License und durch ein sehr offenes Entwicklungsmodell nicht in der Hand von Einzelpersonen, Konzernen oder Ländern, sondern in der Hand einer weltweiten Gemeinschaft vieler Programmierer, die sich hauptsächlich über das Internet austauschen. Bei der Entwicklung kommunizieren die Entwickler fast ausschließlich über E-Mail, da Linus Torvalds behauptet, dass so die Meinungen nicht direkt aufeinander prallen. In vielen Mailinglisten, aber auch in Foren und im Usenet besteht für jedermann die Möglichkeit, die Diskussionen über den Kernel zu verfolgen, sich daran zu beteiligen und auch aktive Beiträge zur Entwicklung zu leisten. Durch diese unkomplizierte Vorgehensweise ist eine schnelle und stetige Entwicklung gewährleistet, die auch die Möglichkeit mit sich bringt, dass jeder dem Kernel Fähigkeiten zukommen lassen kann, die er benötigt.

Eingegrenzt wird dies nur durch die Kontrolle von Linus Torvalds und einigen besonders verdienten Programmierern, die das letzte Wort über die Aufnahme von Verbesserungen und Patches in die offizielle Version haben. Manche Linux-Distributoren bauen auch eigene Funktionen in den Kernel ein, die im offiziellen Kernel (noch) nicht vorhanden sind.



Linus Torvalds (2014)

Änderungen der Herkunftskontrolle

Der Entwicklungsprozess des Kernels ist wie der Kernel selbst ebenfalls immer weiterentwickelt worden. So führte der Rechtsprozess der SCO Group um angeblich illegal übertragenen Code in Linux zur Einführung eines „Linux Developer’s Certificate of Origin“, das von Linus Torvalds und Andrew Morton bekanntgegeben wurde.^[10] Diese Änderung griff das Problem auf, dass nach dem bis dahin gültigen Modell des Linux-Entwicklungsprozesses die Herkunft einer Erweiterung oder Verbesserung des Kernels nicht nachvollzogen werden konnte.

“These days, most of the patches in the kernel don't actually get sent directly to me. That not just wouldn't scale, but the fact is, there's a lot of subsystems I have no clue about, and thus no way of judging how good the patch is. So I end up seeing mostly the maintainers of the subsystem, and when a bug happens, what I want to see is the maintainer name, not a random developer who I don't even know if he is active any more. So at least for me, the `_chain_` is actually mostly more important than the actual originator.

There is also another issue, namely the fact than when I (or anybody else, for that matter) get an emailed patch, the only thing I can see directly is the sender information, and that's the part I trust. When Andrew sends me a patch, I trust it because it comes from him – even if the original author may be somebody I don't know. So the `_path_` the patch came in through actually documents that chain of trust – we all tend to know the „next hop“, but we do `_not_` necessarily have direct knowledge of the full chain.

So what I'm suggesting is that we start „signing off“ on patches, to show the path it has come through, and to document that chain of trust. It also allows middle parties to edit the patch without somehow „losing“ their names – quite often the patch that reaches the final kernel is not exactly the same as the original one, as it has gone through a few layers of people.”

„Zurzeit werden die meisten Patches für den Kernel nicht direkt an mich gesandt. Das wäre einfach nicht machbar. Tatsache ist, dass es eine Menge Untersysteme gibt, mit denen ich überhaupt nicht vertraut bin und ich somit keine Möglichkeit habe zu entscheiden, wie gut der Patch ist. Deshalb läuft es meist darauf hinaus, die Pfleger (Maintainer) des Untersystemes zu treffen. Falls ein Fehler auftritt, will ich den Namen eines Pflegers und nicht irgendeines Entwicklers sehen, von dem ich nicht einmal weiß, ob er noch aktiv ist. Daher ist für mich auf jeden Fall die `_Kette_` wichtiger als der tatsächliche Urheber. Auch gibt es ein anderes Problem, nämlich dass ich, falls man mir (oder irgendjemand anderem) einen Patch über E-Mail schickt, einzig die Senderinformation direkt sehen kann, und das ist der Teil, dem ich traue. Wenn Andrew mir einen Patch schickt, vertraue ich dem Patch, weil er von Andrew kommt – auch wenn der eigentliche Urheber jemand ist, den ich nicht kenne. Also belegt tatsächlich der `_Weg_`, den der Patch zu mir nahm, diese Kette des Vertrauens – wir alle neigen dazu, das jeweils nächste „Glied“ zu kennen, aber `_nicht_` unbedingt unmittelbares Wissen über die gesamte Kette zu haben. Was ich also vorschlage ist, dass wir anfangen, Patches „abzuzeichnen“, um den Weg, den sie genommen haben, aufzuzeigen und diese Kette des Vertrauens zu dokumentieren. Das erlaubt es darüber hinaus vermittelnden Gruppen, den Patch zu verändern, ohne dass dabei der Name von jemanden „auf der Strecke bleibt“ – ziemlich oft ist die Patchversion, die letztendlich in den Kernel aufgenommen wird, nicht genau die ursprüngliche, ist sie doch durch einige Entwicklerschichten gegangen.“

– LINUS TORVALDS: Linux-Kernel Archive, 23. Mai 2004^[11]

Das Versionskontrollsystem Git

Die Versionskontrolle des Kernels unterliegt dem Programm Git. Dies wurde speziell für den Kernel entwickelt und auf dessen Bedürfnisse hin optimiert. Es wurde im April 2005 eingeführt, nachdem sich abgezeichnet hatte, dass das alte Versionskontrollsystem BitKeeper nicht mehr lange für die Kernelentwicklung genutzt werden konnte.

Kernel-Versionen

Auf der Website *kernel.org* werden alle alten und neuen Kernel-Versionen archiviert. Die dort befindlichen Referenzkernel werden auch als Vanilla-Kernel bezeichnet (von umgangssprachlich engl. *vanilla* für *Standard* bzw. *ohne Extras* im Vergleich zu Distributionskernels). Auf diesem bauen die Distributionskernel auf, die von den einzelnen Linux-Distributionen um weitere Funktionen ergänzt werden. Die Kernel-Version des geladenen Betriebssystems kann mit dem Syscall `uname` abgefragt werden.

Versionsnummern-Schema

Die frühen Kernelversionen (0.01 bis 0.99) hatten noch kein klares Nummerierungsschema. Version 1.0 sollte die erste „stabile“ Linux-Version werden. Beginnend mit Version 1.0 folgen die Versionsnummern von Linux einem bestimmten Schema:

Die *erste Ziffer* wird nur bei grundlegenden Änderungen in der Systemarchitektur angehoben. Während der Entwicklung des 2.5er Kernels kam wegen der relativ grundlegenden Änderungen, verglichen mit dem 2.4er Kernel, die Diskussion unter den Kernel-Programmierern auf, den nächsten Produktionskernel als 3.0 zu deklarieren. Torvalds war aber aus verschiedenen Gründen dagegen, sodass der resultierende Kernel als 2.6 bezeichnet wurde.

Die *zweite Ziffer* gibt das jeweilige „Majorrelease“ an. Bisher wurden stabile Versionen (Produktivkernel) von den Entwicklern stets durch gerade Ziffern wie 2.2, 2.4 und 2.6 gekennzeichnet, während die Testversionen (Entwicklerkernel) immer ungerade Ziffern trugen, wie zum Beispiel 2.3 und 2.5; diese Trennung ist aber seit Juli 2004 ausgesetzt, es gab keinen Entwicklerkernel mit der Nummer 2.7, stattdessen wurden die Änderungen laufend in die 2.6er-Serie eingearbeitet.

Zusätzlich bezeichnet eine *dritte Zahl* das „Minorrelease“, das die eigentliche Version kennzeichnet. Werden neue Funktionen hinzugefügt, steigt die dritte Zahl an. Der Kernel wird damit zum Beispiel mit einer Versionsnummer wie 2.6.7 bestimmt.

Um die Korrektur eines schwerwiegenden NFS-Fehlers schneller verbreiten zu können, wurde mit der Version 2.6.8.1 erstmals eine *vierte Ziffer* eingeführt. Seit März 2005 (Kernel 2.6.11) wird diese Nummerierung offiziell verwendet.^[12] So ist es möglich, die Stabilität des Kernels trotz teilweise sehr kurzer Veröffentlichungszyklen zu gewährleisten und Korrekturen von kritischen Fehlern innerhalb weniger Stunden in den offiziellen Kernel zu übernehmen – wobei sich die vierte Ziffer erhöht (z. B. von 2.6.11.1 auf 2.6.11.2). Die Minorreleasenummer, also die dritte Ziffer, wird hingegen nur bei Einführung neuer Funktionen hochgezählt.

Im Mai 2011 erklärte Linus Torvalds, die nach der Version 2.6.39 kommende Version nicht 2.6.40, sondern 3.0 zu benennen.^[13] Als Grund dafür führte er an, dass die Versionsnummern seiner Meinung nach zu hoch wurden. Die Versionsnummer '3' stehe gleichzeitig für das dritte Jahrzehnt, welches für den Linux-Kernel mit seinem 20. Geburtstag anfangen. Bei neuen Versionen wird seitdem die zweite Ziffer erhöht und die dritte steht – anstelle der vierten – für Bugfixreleases.

Im Februar 2015 erhöhte Torvalds auf Version 4.0 statt Version 3.20^[14] nachdem er auf Google+ Meinungen hierzu eingeholt hatte.^[15] Seit März 2019 ist Linux 5.0 freigegeben. Dabei hat der Sprung von der letzten Versionsnummer 4.20 auf 5.0 keine tiefergehende Bedeutung. Die aktuelle Version soll eine modernere Speicherfunktion und mehr Geschwindigkeit liefern.^[16]

Entwicklerversion

Neue Funktionen finden sich im *-mm* Kernel des Kernelentwicklers Andrew Morton und werden anschließend in den Hauptzweig von Torvalds übernommen. Somit werden große Unterschiede zwischen Entwicklungs- und Produktionskernel und damit verbundene Portierungsprobleme zwischen den beiden Serien vermieden. Durch dieses Verfahren gibt es auch weniger Differenzen zwischen dem offiziellen Kernel und den Distributionskernel (früher wurden Features des Entwicklungszweiges von den Distributoren häufig in ihre eigenen Kernels rückintegriert). Allerdings litt 2004/2005 die Stabilität des 2.6er Kernels unter den häufig zu schnell übernommenen Änderungen. Ende Juli 2005 wurde deshalb ein neues Entwicklungsmodell beschlossen, das nach dem Erscheinen der Version 2.6.13 erstmals zur Anwendung kam: Neuerungen werden nur noch in den ersten zwei Wochen der Kernelentwicklung angenommen, wobei anschließend eine Qualitätssicherung bis zum endgültigen Erscheinen der neuen Version erfolgt.

Pflege der Kernel-Versionen

Während Torvalds die neuesten Entwicklungsversionen veröffentlicht, wurde die Pflege der älteren stabilen Versionen an andere Programmierer abgegeben. Gegenwärtig ist dafür Greg Kroah-Hartman verantwortlich – mit Ausnahme des von Ben Hutchings betreuten 3.16-Zweigs. Zusätzlich zu diesen offiziellen und über Kernel.org oder einen seiner Mirrors zu beziehenden Kernel-Quellcodes kann man auch alternative „Kernel-Trees“ aus anderen Quellen benutzen. Distributoren von Linux-basierten Betriebssystemen pflegen meistens ihre eigenen Versionen des Kernels und beschäftigen zu diesem Zwecke fest angestellte Kernel-Hacker, die ihre Änderungen meist auch in die offiziellen Kernels einfließen lassen.

Distributions-Kernel sind häufig intensiv gepatcht, um auch Treiber zu enthalten, die noch nicht im offiziellen Kernel enthalten sind, von denen der Distributor aber glaubt, dass seine Kundschaft sie benötigen könnte und die notwendige Stabilität respektive Fehlerfreiheit dennoch gewährleistet ist.

Versionen mit Langzeitunterstützung

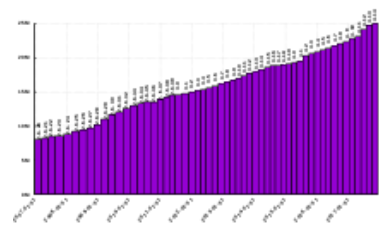
Folgende Versionen werden besonders lange mit Support (*Long Term Support*) versorgt:^[17]

Version	Veröffentlichung	Support-Ende (EOL)
4.4	10. Jan. 2016	Februar 2022
4.9	11. Dez. 2016	Januar 2023
4.14	12. Nov. 2017	Januar 2024
4.19	22. Okt. 2018	Dezember 2024
5.4	24. Nov. 2019	Dezember 2025
5.10	13. Dez. 2020	Dezember 2026
5.15	31. Okt. 2021	Oktober 2023

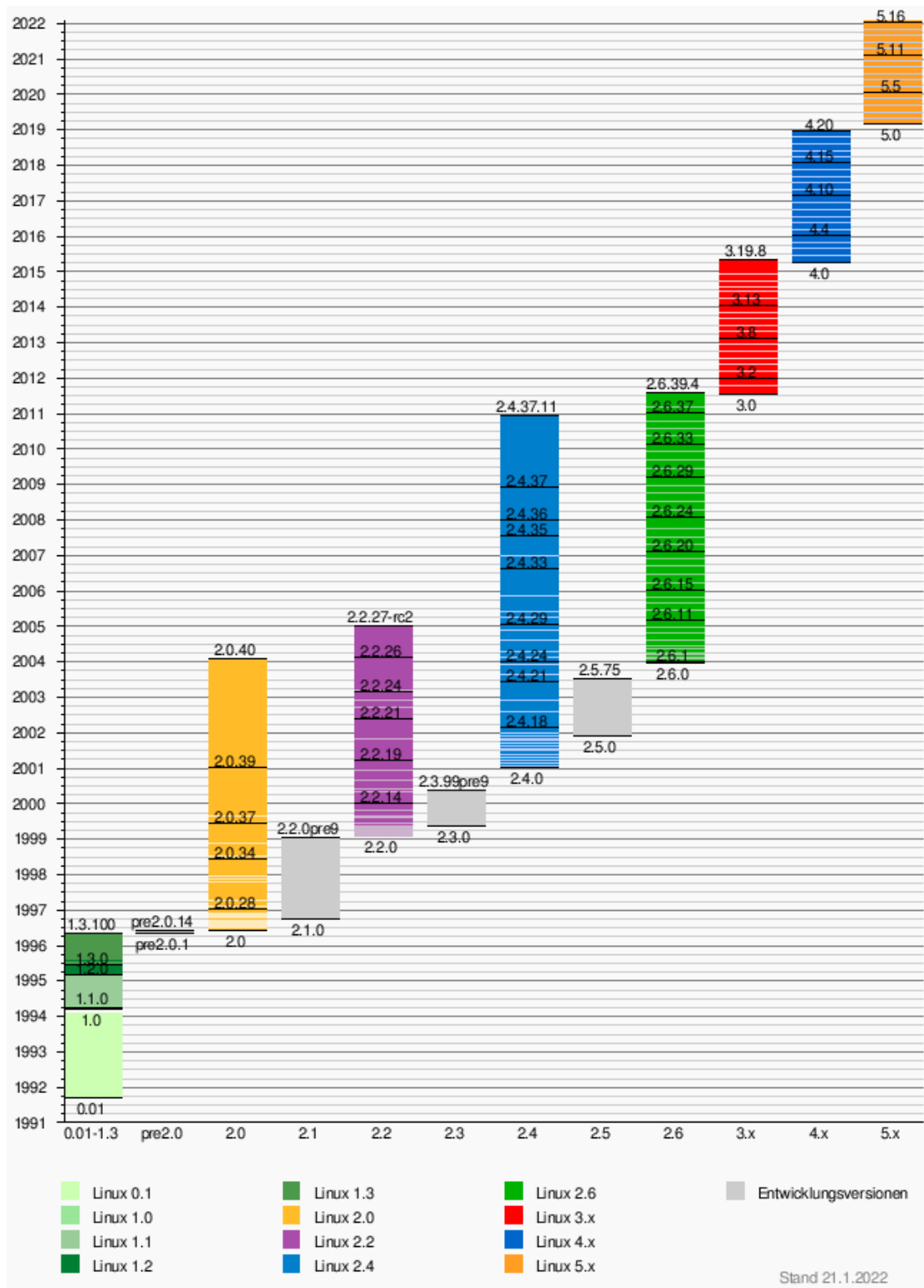
Versionsgeschichte

Zeittafeln

Das folgende Schaubild stellt einzelne Versionen des Linux-Kernels anhand der Erscheinungsdaten auf einer Zeittafel angeordnet dar und soll dem Überblick dienen.



Entwicklung der Anzahl
Quelltextzeilen



Versionsgeschichte bis Version 2.6

Zweig	Version	Veröffentlichung ^[18]	Dateien(a)	Quellcode- zeilen ^(b)	Größe in kB ^(c)	Bemerkungen
	0.01	17. September 1991	88	8.413	230	erste Veröffentlichung; startete auf Systemen mit Floppy-Diskettenlaufwerk und 386er-Prozessor, lud dabei die Treiber für das Minix-Dateisystem und eine finnische Tastatur, sowie als einziges Anwendungsprogramm den Kommandozeileninterpreter <i>bash</i> ^[19]
1.0	1.0.0	13. März 1994	563	170.581	1.259	erste "kommerziell verwendbare" Version. Tatsächlich sind jedoch die Buchstaben der letzten Version 0.99z ausgegangen, so führte man verfrüht die Version 1.0 ein.
1.1	1.1.0	6. April 1994	561	170.320	1.256	Entwicklungsversion
	1.1.95	2. März 1995			2.301	
1.2	1.2.0	7. März 1995	909	294.623	2.301	erste Portierungen auf weitere Prozessorarchitekturen, mit <i>Alpha</i> , <i>MIPS</i> und <i>SPARC</i> ^[20]
	1.2.13	2. August 1995			2.355	
1.3	1.3.0	12. Juni 1995	992	323.581	2.558	Entwicklungsversion mit erster Unterstützung für Mehrprozessorsysteme ^[20]
	1.3.100	10. Mai 1996			5.615	
2.0	2.0.0	9. Juni 1996	2.015	716.119	5.844	erste Unterstützung für symmetrische Mehrprozessorsysteme ^[20] , Einführung von Kernel-Modulen.
	2.0.40	8. Februar 2004			7.551	
2.1	2.1.0	30. September 1996	1.727	735.736	6.030	Entwicklungsversion
	2.2.0-pre9	21. Januar 1999			13.077	
2.2	2.2.0	26. Januar 1999	4.599	1.676.182	13.080	erste Unterstützung für das Netzwerkprotokoll IPv6 sowie Portierung auf die Plattformen <i>UltraSPARC</i> und <i>PA-RISC</i> ^[20]
	2.2.26	24. Februar 2004			19.530	
2.3	2.3.0	11. Mai 1999	4.721	1.763.358	13.804	Entwicklungsversion
	2.3.99-pre9	23. Mai 2000			20.882	
2.4	2.4.0	4. Januar 2001	8.187	3.158.560	24.379	erste Unterstützung für den Energieverwaltungsstandard <i>ACPI</i> und für den Datenbus <i>USB</i> , ^[20] <i>Large File Support</i> ^[21] , Einführung des <i>Netfilter</i> und der <i>iptables</i> ; ^[22] letzter gepflegter Zweig vor 2.6 ^[23]
	2.4.37	2. Dezember 2008			38.735	Letzte Version war 2.4.37.11 am 18. Dezember 2010
2.5	2.5.0	23. November 2001	9.893	3.833.603	29.405	Entwicklungsversion
	2.5.75	10. Juli 2003			40.969	
2.6	2.6.0	18. Dezember 2003	21.279	8.102.486	41.614	neues Versionsschema, dabei wurden die bisherigen Entwicklerzweige durch einen stetigen Entwicklungsprozess ersetzt ^[20]
→ für alle nachfolgenden Versionen siehe Abschnitt „Versionsgeschichte ab Version 2.6“						
Legende: Alte Version						

Versionsgeschichte ab Version 2.6

Bei Betrachtung der zuletzt erschienenen Versionen (siehe Tabelle) erfolgt die Entwicklung einer neuen Kernel-Version in durchschnittlich 82 Tagen. Der Kernel wird hierbei im Durchschnitt um 768 Dateien und 325.892 Quelltextzeilen (englisch *Lines of Code*) erweitert. Das mit dem Datenkompressionsprogramm *gzip* komprimierte *tar-Archiv* (.tar.gz) wächst im Mittel um rund 2 Megabyte mit jeder veröffentlichten Hauptversion.

Version	Veröffentlichung ^[18]	Anzahl der Dateien ^(a)	Quelltextzeilen ^(b)	Größe in kB ^(c)	Zeitraum in Tagen ^(d)	Bemerkungen
2.6.13	28. Aug. 2005					u. a. wurde <i>Inotify</i> aufgenommen ^[24]
2.6.20	4. Feb. 2007	21.280	8.102.486	54.548	66	erste Unterstützung für die Virtualisierungstechnik <i>KVM</i> ^[25]
2.6.21	26. Apr. 2007	21.614	8.246.470	55.329	80	
2.6.22	8. Juli 2007	22.411	8.499.363	56.914	74	
2.6.23	9. Okt. 2007	22.530	8.566.554	57.404	93	
2.6.24	24. Jan. 2008	23.062	8.859.629	59.079	107	
2.6.25	17. Apr. 2008	23.810	9.232.484	61.518	83	
2.6.26	13. Juli 2008	24.270	9.411.724	62.550	88	
2.6.27	9. Okt. 2008	24.354	9.709.868	63.721	88	wurde nachträglich mit Unterstützung für neue Hardware (wie SAS) erweitert; ^[26] Dieser Zweig hatte <i>Long Term Support</i> , die letzte Version war 2.6.27.62 am 17. März 2012
2.6.28	24. Dez. 2008	25.255	10.195.507	66.766	76	
2.6.29	23. März 2009	26.668	11.010.647	71.977	89	Aufnahme des <i>Btrfs</i> ^{[27][28]}
2.6.30	10. Juni 2009	27.879	11.637.173	75.768	78	<i>USB-3.0</i> -Unterstützung
2.6.31	9. Sep. 2009	29.111	12.046.317	78.279	92	Unterstützung für Festplatten mit nativen 4K-Sektoren (auch bekannt als <i>Advanced Format</i>)
2.6.32	3. Dez. 2009	30.485	12.610.030	81.901	84	Dieser LTS-Zweig wurde von Willy Tarreau betreut, die letzte Version war 2.6.32.71 am 12. März 2016
2.6.33	24. Feb. 2010	31.565	12.990.041	84.533	83	bildet die Grundlage für einen <i>Echtzeit</i> -Zweig; ^[29] Letzte Version war 2.6.33.20 am 7. November 2011 ^[30]
2.6.34	16. Mai 2010	32.297	13.320.934	86.520	82	Grafiktreiber für neuere <i>AMD-Radeon</i> -GPUs und die Grafikkern von einigen erst Anfang nächsten Jahres erwarteten <i>Intel</i> -Prozessoren, neue Dateisysteme: <i>LogFS</i> und <i>Ceph</i> ; ^[31] viele Änderungen an den Dateisystemen <i>Btrfs</i> , <i>ext4</i> , <i>NILFS2</i> , <i>SquashFS</i> und <i>XFS</i> ; ^[32] dem <i>SCSI</i> -Subsystem und dem Architektur-Code für <i>Arm</i> -, <i>Blackfin</i> - und <i>MicroBlaze</i> -CPUs; ^[33] bessere Unterstützung für neue und ältere <i>AMD</i> - und <i>Intel</i> -Chips; größere Umbaumaßnahmen am <i>Nouveau</i> -Treiber für <i>Nvidia</i> -Grafik; ^[34] Dieser LTS-Zweig wurde von Paul Gortmaker betreut, die letzte Version war 2.6.34.15 am 10. Februar 2014
2.6.35	1. Aug. 2010	33.316	13.545.604	88.301	77	neu sind unter anderem ein verbesserter Netzwerkdurchsatz, defragmentierbarer Arbeitsspeicher und die Unterstützung für die <i>Turbo-Core</i> -Funktion moderner <i>AMD</i> -Prozessoren; ^[35] die Unterstützung für die Stromsparfunktionen von <i>Radeon</i> -Grafikchips wurde verbessert, sowie die neuen <i>H.264</i> -Dekodierfunktionen für den Grafikkern in <i>Intels Core-i5</i> -Prozessoren (auch <i>Ironlake</i> genannt); ^[36] Letzte Version war 2.6.35.14 am 13. März 2012 ^[37]
2.6.36	20. Okt. 2010	34.301	13.499.457	88.707	80	neu sind – im auch „Flesh-Eating Bats with Fangs“ (englisch für „Fleischfressende Fledermäuse mit Reißzähnen“) genannten 36er Zweig ^[38] – unter anderem das Sicherheits-Framework <i>AppArmor</i> , die Schnittstelle „ <i>LIRC</i> “ (für das in Version 2.6.35 eingeführte System zur Nutzung von Infrarot-

						Fernbedienungen) und eine verbesserte Energieverwaltung für <u>Grafikprozessoren</u> ; ^[39] Unterstützung für Echtzeit-Virens Scanner; ^[40] Letzte Version war 2.6.36.4 am 17. Februar 2011
2.6.37	5. Jan. 2011	35.186	13.916.632	92.474	76	neu sind, neben den üblichen zusätzlichen Treibern (u. a. für USB 3.0), eine verbesserte Skalierung und Virtualisierung für <u>Mehrkernprozessoren</u> ; ^[41] Letzte Version war 2.6.37.6 am 27. März 2011
2.6.38	15. März 2011	35.864	14.208.866	94.144	69	Unterstützung für die <u>AMD-Radeon-HD-6000-Serie</u> und für <u>AMDs APUs</u> ; ^[42] sowie Verbesserungen für Audio- und Video-Verarbeitung als auch für <u>berührungsempfindliche Bildschirme</u> ; ^[43] Letzte Version war 2.6.38.8 am 3. Juni 2011
2.6.39	19. Mai 2011	36.705	14.533.582	95.994	65	Unterstützung für <u>Firewall-IP sets</u> ; ^{[44][45]} der <u>Big Kernel Lock</u> wird entfernt; ^[46] Letzte Version war 2.6.39.4 am 3. August 2011
3.0	22. Juli 2011	36.781	14.646.952	96.676	64	neues Versionsnummernschema, ^[47] zudem neue Treiber u. a. für (virtuelle) Netzwerkgeräte und den Fernseh-Standard <u>DVB-T2</u> , sowie allgemein verbesserte Virtualisierung (u. a. durch abschließende Arbeiten an dem Hypervisor <u>Xen</u>) und viele kleine Verbesserungen (wie z. B. Schreib- und Löschoptimierungen für Dateisysteme); ^{[48][49]} Dieser LTS-Zweig wurde von Greg Kroah-Hartman betreut, die letzte Version war <u>3.0.101</u> am 22. Oktober 2013
3.1	24. Okt. 2011	37.084	14.770.469	97.334	94	neu ist u. a. die Unterstützung der Prozessor-Plattform <u>OpenRISC</u> ; ^[50] Letzte Version war <u>3.1.10</u> am 18. Januar 2012
3.2	4. Jan. 2012	37.617	14.998.651	≈ 62.600	73	neu ist u. a. die Unterstützung für die Prozessor-Architektur <u>Hexagon</u> des Unternehmens <u>Qualcomm</u> ; zudem wurden u. a. einige Netzwerktreiber in das zugehörige Subsystem ausgelagert ^[30] Dieser LTS-Zweig wurde von Ben Hutchings betreut, die letzte Version war <u>3.2.102</u> am 31. Mai 2018
3.3	18. März 2012	38.082	15.166.074	≈ 75.300	74	neu ist u. a. die Unterstützung für die <u>Android-Plattform</u> ; ^[51] Letzte Version war <u>3.3.8</u> am 1. Juni 2012
3.4	20. Mai 2012	38.566	15.383.860	≈ 64.100	63	neunte <u>LTS-Freigabe</u> ; neu ist u. a. die Unterstützung des <u>Ruhezustand-Modus RC6</u> ; ^{[52][53][54]} Dieser LTS-Zweig wurde zuletzt von Li Zefan betreut, die letzte Version war <u>3.4.113</u> am 26. Oktober 2016
3.5	21. Juli 2012	39.096	15.596.378	≈ 77.200	62	die Unterstützung für die veralteten Netzwerk-Standards <u>Token Ring</u> und <u>Econet</u> wurde entfernt; ^[55] Letzte Version war <u>3.5.7</u> am 12. Oktober 2012
3.6	30. Sep. 2012	39.733	15.868.036	≈ 78.500	71	u. a. wurde die <u>Energiespar- oder Schlaffunktion</u> , für den <u>Bereitschaftsbetrieb</u> und <u>Ruhezustand</u> des gesamten Systems, weiterentwickelt; ^[56] Letzte Version war <u>3.6.11</u> am 17. Dezember 2012
3.7	11. Dez. 2012	40.905	16.191.690	≈ 79.800	71	u. a. <u>NAT für IPv6</u> , Unterstützung für den <u>Arm-64-Bit-Befehlssatz</u> und <u>Ext4-Größenänderungen</u> für Laufwerke die größer sind als 16 Terabyte, größere Veränderungen an den <u>Grafiktreibern</u> für <u>Nvidia</u> -, <u>Intel</u> - und <u>AMD-Grafik</u> und verbesserte <u>Hardware-Unterstützung</u> für <u>Helligkeitsregelung</u> und

						Stromsparfunktionen von Soundkarten; ^[57] Letzte Version war 3.7.10 am 27. Februar 2013
3.8	18. Feb. 2013	41.520	16.416.874	84.623	70	u. a. wurde die Unterstützung für das Dateisystem <i>F2FS</i> eingefügt ^[58] und die Unterstützung für <i>Intel 80386</i> -Prozessoren entfernt; ^[59] soll als Basis für die nächste <i>Android</i> -Version dienen; ^[60] Letzte Version war 3.8.13 am 11. Mai 2013
3.9	29. Apr. 2013	42.423	16.686.879	≈ 82.000	69	u. a. wurde die Unterstützung für 32-Bit-Prozessoren der Baureihe <i>HTP</i> (von <i>Meta</i>) ^[61] sowie <i>ARC 700</i> (von <i>Synopsys</i>) ^[62] hinzugefügt und die Treiber für das Dateisystem <i>Btrfs</i> um eine erste Unterstützung für <i>Raid 5</i> und <i>6</i> erweitert; ^{[60][63]} Letzte Version war 3.9.11 am 21. Juli 2013
3.10	30. Juni 2013	43.016	16.955.489	≈ 69.900	63	unter anderem bessere Unterstützung für den Unified Video Decoder (UVD) von <i>AMD-Radeon</i> -GPUs und bessere Unterstützung von <i>Nvidia Tegra</i> ; Zudem wurden die Echtzeitfähigkeiten verbessert sowie die Nutzung von <i>SSDs</i> als Cache für Festplatten ermöglicht. ^[64] Neu ist auch der Treiber für die Hochgeschwindigkeitsübertragungstechnologie <i>InfiniBand</i> ; ^[65] Dieser LTS-Zweig wurde zuletzt von Willy Tarreau betreut, die letzte Version war 3.10.108 am 4. November 2017
3.11	2. Sep. 2013	44.002	17.403.279	≈ 71.600	63	neben Optimierungen an den Grafik-Treibern, ist die Aufnahme von verschiedenen neuen WLAN- und LAN-Treibern sowie die Verbesserung der KVM- und Xen-Unterstützung auf <i>ARM64</i> vorgesehen; ^[66] wird der Arbeitsspeicher knapp kann dieser mit einem <i>Zswap</i> (siehe auch <i>Zip</i> und <i>Swap</i>) komprimiert werden; ^[67] Letzte Version war 3.11.10 am 29. November 2013 – Wegen ihrer Versionsnummer auch <i>Linux for Workgroups</i> genannt, angelehnt an <i>Microsoft's Windows 3.11 for Workgroups</i> . ^[68]
3.12	3. Nov. 2013	44.586	17.726.872	≈ 73.000	62	neben Optimierungen an den <i>Optimus</i> -Treibern, ist die Unterstützung eines <i>SYN-Proxy</i> hinzugekommen, ^[69] der <i>SYN-Flooding</i> -Angriffe verhindern soll; ^[70] <i>Multithreading</i> bei mit <i>Mdadm</i> angelegten <i>Raid-5</i> -Arrays und <i>Btrfs</i> beherrscht <i>Deduplikation</i> ; ^[71] Dieser LTS-Zweig wurde von Jiri Slaby betreut, die letzte Version war 3.12.74 am 9. Mai 2017
3.13	20. Jan. 2014	44.970	17.930.916	≈ 73.600	77	enthält die neue Firewall-Infrastruktur <i>Nftables</i> (welche die mit dem 2.4er eingeführten <i>Iptables</i> ablösen soll), bessere 3D-Leistung und Aktivierung des <i>Dynamic Power Management</i> (DPM) bei <i>AMD-Radeon</i> -Grafikkarten. ^{[72][73]} Moderneres Multiqueue-Storage-Interface; ^[74] Letzte Version war 3.13.11 am 22. April 2014
3.14	31. März 2014	45.935	18.271.989	≈ 74.900	70	ein <i>Scheduler</i> , der für Echtzeitsysteme geeignet ist, ^[75] hinzugefügt und u. a. ist nun <i>Xen 4.4</i> enthalten; ^[76] Unterstützung von neuen Grafikkerneln und korrekte Funktion des Unified Video Decoder (UVD) für <i>AMD</i> Grafikchips ab <i>HD 7000</i> ; ^[75] Dieser LTS-Zweig wurde von Greg Kroah-Hartman betreut, die letzte Version war 3.14.79 am 11. September 2016
3.15	8. Juni 2014	46.780	18.632.574	≈ 76.000	53	das Aufwachen aus dem <i>Suspend-to-RAM</i> -Modus wurde beschleunigt und <i>Open File Description Locks</i> eingeführt. Z. B.

						<p>Videosoftware soll nun einfacher Teile einer Datei auslesen können und atomares Austauschen von Dateien wird möglich. Die Unterstützung von FUSE, XFS und Flash-Speichern mit einem Dateisystem darauf wurde ausgebaut;^[77]</p> <p>Letzte Version war 3.15.10 am 14. August 2014</p>
3.16	3. Aug. 2014	47.425	18.879.129	≈ 76.900	56	<p>eingeflossen sind u. a. Robustheitsmaßnahmen beim Dateisystem <i>Btrfs</i>, die Grafiktreiber <i>Radeon</i> und <i>Nouveau</i> wurden optimiert;^[78]</p> <p>Dieser LTS-Zweig wurde von Ben Hutchings betreut, die letzte Version war 3.16.85 am 11. Juni 2020</p>
3.17	5. Okt. 2014	47.490	18.864.388	≈ 76.600	63	<p>Die Funktion <code>getrandom()</code> und Vorgaben zur Mindest-Entropie sorgen für sicherere Zufallszahlen. Neu sind Grundlagen für <i>Kdbus</i>, <i>Fences</i> in <i>Dma-Buf</i>, <i>MST (Multi Stream Transport)</i>, Teil von DisplayPort 1.2) für 4K-Monitore, sowie die Unterstützung für den <i>Xbox-One-Controller</i> und <i>ForcePad-Touchpads</i>. Optimierungen für <i>Thunderbolt</i> bei Apple-Geräten;^[79]</p> <p>Letzte Version war 3.17.8 am 8. Januar 2015</p>
3.18	7. Dez. 2014	47.971	18.994.096	≈ 77.300	64	<p>Aufnahme des <i>OverlayFS</i>; Verbesserungen bei <i>Btrfs</i> und <i>F2FS</i>; Tunnelung beliebiger Protokolle ist über UDP ermöglicht; Audio-Ausgabe mit dem <i>Nouveau</i>-Treiber über DisplayPort; Just-in-time-Kompilierung des <i>Extended Berkeley Packet Filter</i> (eBPF) für ARM64; der BPF ist nun durch den Syscall <code>bpf()</code> allgemein verfügbar; Modulparameter können als unsicher (englisch <i>unsafe</i>) markiert werden;^[80]</p> <p>Dieser LTS-Zweig wurde von Sasha Levin betreut, die letzte Version war 3.18.140 am 16. Mai 2019</p>
3.19	9. Feb. 2015	48.424	19.130.604	≈ 77.900	63	<p>u. a. Unterstützung für AMDs <i>Heterogeneous System Architecture</i> (HSA);^[81] zudem wurde das <i>Interprozesskommunikationsframework Binder</i> aufgenommen;^[82] welches ursprünglich für Android entwickelt wurde und auf <i>OpenBinder</i> aufbaut;^[83]</p> <p>Letzte Version war 3.19.8 am 11. Mai 2015</p>
4.0	12. Apr. 2015	48.945	19.312.370	≈ 78.500	63	<p>u. a. Unterstützung für <i>Updates</i> im laufenden Betrieb (<i>Kernel Live Patching</i>);^{[84][85]}</p> <p>Letzte Version war 4.0.9 am 21. Juli 2015</p>
4.1	22. Juni 2015	49.457	19.512.485	≈ 79.300	70	<p>u. a. Verschlüsselung für Ext4, Treiber für NV-DIMMs und Schaffung der Grundlagen zur 3D-Beschleunigung in <i>virtuellen Maschinen</i>;^[86]</p> <p>Dieser LTS-Zweig wurde von Sasha Levin betreut, die letzte Version war 4.1.52 am 28. Mai 2018</p>
4.2	30. Aug. 2015	50.795	20.311.717	≈ 82.000	70	<p>u. a. Update von UEFI nun aus Linux möglich, Unterstützung der AMD-Grafikkarten der Volcanic-Islands-Generation;^[87]</p> <p>Letzte Version war 4.2.8 am 15. Dezember 2015</p>
4.3	2. Nov. 2015	51.570	20.621.444	≈ 83.000	63	<p>Unterstützung für Skylake und Fiji-GPUs;^[88]</p> <p>IPv6 wird Voreinstellung;^[89]</p> <p>Letzte Version war 4.3.6 am 19. Februar 2016</p>
4.4	10. Jan. 2016	52.221	20.862.115	≈ 83.300	70	<p>Kernel mit Langzeitunterstützung;^[90]</p> <p>Grafiktreiber für den <i>Raspberry Pi</i> und weitere 3D-Grafikbeschleunigungen in <i>KVM</i> für virtuelle Maschinen, Verbesserungen bei <i>RAID</i> und <i>SSDs</i>, <i>BPF</i>-Aufrufe sind nun auch im <i>Userspace</i> möglich;^{[91][92]}</p>

						Letzte Version war 4.4.302 am 3. Februar 2022
4.5	14. März 2016	52.916	21.154.545	≈ 84.300	63	weitere Verbesserungen in den 3D-Treibern, unter anderem durch die Unterstützung für <i>PowerPlay</i> , und reibungslose Wechsel der Netzwerkverbindungen; ^{[93][94]} Verbesserungen beim Zugriff auf <i>NFS</i> -Server; ^[95] Letzte Version war 4.5.7 am 8. Juni 2016
4.6	15. Mai 2016	53.660	21.422.694	≈ 85.300	63	u. a. verbesserte Energieverwaltung durch die Möglichkeit der Taktraten-Einstellung bei <i>Tegra-X1</i> -GPUs und durch die Freischaltung der <i>Frame Buffer Compression</i> (FBC) für Prozessoren der <i>Haswell</i> -Architektur und <i>Broadwell</i> , des Weiteren wurde das Dateisystem <i>OrangeFS</i> freigeschaltet sowie zusätzliche Werkzeuge zur besseren Fehlererkennung (englisch <i>bug hunting</i>); ^{[96][97]} Letzte Version war 4.6.7 am 16. August 2016
4.7	24. Juli 2016	54.400	21.712.846	≈ 86.200	70	u. a. erste Unterstützung für neue <i>Radeon</i> -Grafikkarten (unter dem Decknamen <i>Polaris</i>), vier weitere ARM-Treiber und Unterstützung für Spiele-Steuergeräte; ^{[98][99]} zudem bessere SMR-Unterstützung und Beschleunigungen beim Tunneln; ^[100] im Vergleich zum Vorgänger (4.6) werden nun rund 500 weitere Hardware-Komponenten unterstützt; ^[101] Letzte Version war 4.7.10 am 22. Oktober 2016
4.8	2. Okt. 2016	55.503	22.071.048	≈ 87.700	70	neben der Unterstützung neuer Treiber für Haupt- und Grafik-Prozessoren von AMD, ARM (<i>Mali</i>), Intel und Nvidia wurde u. a. das Übertakten (oder <i>Overclocking</i>) für AMD-Grafiktreiber und eine neue GPU-Virtualisierungstechnik für Intel-Grafiktreiber eingeführt, des Weiteren wurde das <i>XFS</i> überarbeitet, so dass es auch Datenintegritäten prüfen sowie Datenduplizierung und das <i>Copy-On-Write</i> -Verfahren beherrscht; ^{[102][103]} Letzte Version war 4.8.17 am 9. Januar 2017
4.9	11. Dez. 2016	56.223	22.348.356	≈ 88.900	70	neben Verbesserungen in der Sicherheit – durch besseren Schutz vor Stapelüberläufen – erhalten u. a. die Treiber für das <i>XFS</i> eine <i>Shared Data Extents</i> genannte Erweiterung, welche auf die (bereits im 4.8er-Kern eingeführte) <i>Reverse-Mapping-Infrastruktur</i> aufsetzt und es künftig ermöglichen soll, dass sich mehrere Dateien einen Daten- oder Wert(e)bereich teilen können und dieser auch mehrere Besitzer haben kann; zudem wurden erste Unterstützungen für den sogenannten <i>Greybus</i> eingearbeitet, welcher ursprünglich für das ehemals von Google entwickelte modulare Smartphone Ara gedacht war und u. a. von Motorola in einem ihrer Geräte genutzt wird; ^{[104][105]} Aktuelle Version: 4.9.317 am 6. Juni 2022
4.10	19. Feb. 2017	57.172	22.839.541	≈ 89.900	71	Verbesserungen beim Schreiben auf Datenträger und Einführung einer schnelleren Fehlererkennung in RAID-Systemen, zudem wurden u. a. <i>EFI</i> -Zugriffe verbessert und der <i>LED</i> -Treiber <i>uleds</i> eingearbeitet; ^[106] sowie ein Verfahren zur Grafikbeschleunigung virtueller Maschinen eingeführt; ^[107] bei Intel-Prozessoren kann der Cache zwischen Prozessen aufgeteilt werden, die Funktionen für die ARM64-Architektur wurden ausgebaut, Überarbeitungen beim Routing, das Dateisystem UBIFS wurde um eine optionale Verschlüsselung ergänzt, während logs entfernt wurde; ^[108] Letzte Version war 4.10.17 am 20. Mai 2017

4.11	1. Mai 2017	57.964	23.137.284	≈ 91.000	71	Verringerung des Stromverbrauches von NVMe-SSDs durch die Stromspartechnik APST; Unterstützung für selbstverschlüsselnde SSDs; Verbesserungen bei der Abfrage von Metadaten für Verzeichnisse und Dateien; Überarbeitung der in 4.10 eingeführten Funktion Intel Turbo Boost Max 3.0; Verbesserungen bei der Grafikbeschleunigung virtueller Maschinen für AMD-Grafikeinheiten; ^[109] Anpassungen für ext4, um die Nutzung als Wegwerfdateisystem zu verbessern; ^[110] Letzte Version war 4.11.12 am 21. Juli 2017
4.12	2. Juli 2017	59.808	24.173.535	≈ 99.000	62	Einführung des Scheduler "Budget Fair Queueing" (BFQ) für bessere Performance bei Datenträgerzugriffen, Unterstützung von AMDs Grafikprozessor Radeon Vega; ^[111] Letzte Version war 4.12.14 am 20. September 2017
4.13	3. Sep. 2017	60.543	24.767.008	≈ 100.000	63	Performancegewinn im Protokoll HTTPS, Verzeichnisse im Ext4-Dateisystemen können nun bis zu 2 Milliarden Einträge enthalten; ^[112] Letzte Version war 4.13.16 am 24. November 2017
4.14	12. Nov. 2017	61.258	25.041.165	≈ 97.000	70	LTS-Version; ^[113] Unterstützung bis zu 4096 Terabyte Arbeitsspeicher; ^[114] Aktuelle Version: 4.14.282 am 6. Juni 2022
4.15	28. Jan. 2018	62.271	25.364.680	≈ 100.000	78	Einbau diverser Schutzmechanismen vor Meltdown und Spectre, der Treiber Amdgpu unterstützt nun AMDs Vega-Grafikkarten besser; ^[115] Letzte Version war 4.15.18 am 19. April 2018
4.16	1. Apr. 2018	62.883	25.558.670	100.606	63	Letzte Version war 4.16.18 am 26. Juni 2018
4.17	3. Juni 2018	61.332	25.379.428	99.772	63	HDCP-Unterstützung bei Intel-CPUs mit integriertem Grafikprozessor, erste Gegenmaßnahmen gegen Spectre v4 (Speculative Store Bypass), Entfernung Architektursupport Blackfin, AXIS CRIS und 6 weiterer; ^[116] Letzte Version war 4.17.19 am 24. August 2018
4.18	12. Aug. 2018	60.973	25.280.736	101.782	70	Vorarbeiten für leistungsgesteigerte Firewall Bpfilter; Unterstützung für GPU von Kaby Lake-G und der angekündigten Vega20; Erster Support für Qualcomms Snapdragon-845-Prozessor; ^[117] Letzte Version war 4.18.20 am 21. November 2018
4.19	22. Okt. 2018	61.700	25.588.319	103.117	71	LTS-Version; neuen „Code of Conduct“ für Entwickler fest in die Kernel-Dokumentation aufgenommen; ^[118] Unterstützung des neuen Wlan Standards 802.11ax, Performanceverbesserung der SATA-Treiber; ^[119] Aktuelle Version: 4.19.246 am 6. Juni 2022
4.20	23. Dez. 2018	62.446	25.955.384	104.258	62	Amdgpu-Treiber unterstützt neue GPUs von AMD, Behebung der Sicherheitslücke für die zweite Variante von Spectre; ^[120] Letzte Version war 4.20.17 am 19. März 2019
5.0	3. März 2019	63.135	26.211.072	102.776	70	u. a. neu hinzugekommen ist der Support für Freesync von AMD; ^[121] Letzte Version war 5.0.21 am 4. Juni 2019
5.1	6. Mai 2019	63.873	26.459.776	≈ 101.000	63	Datenträger können nun über asynchronem I/O (AIO) angesprochen werden; ^[122] Letzte Version war 5.1.21 am 28. Juli 2019
5.2	7. Juli 2019	64.587	26.552.127	≈ 102.000	63	u. a. Verzeichnisse in Ext4 können nun auch case-insensitive genutzt werden; die

						Performance von Meltdown- und Spectre v2-Patches wurde optimiert. ^[123] Letzte Version war 5.2.21 am 11. Oktober 2019
5.3	15. Sep. 2019	65.261	27.141.312	≈ 108.500	70	u. a. Unterstützung für neue AMD Navi-10-GPUs und Zhaoxin x86-CPUs, einige Neuerungen für Effizienzverbesserungen (Support für <i>umwait</i> -Instruktionen, Intel Speed Select, <i>Utilization Clamping</i> im Scheduler), 16 Millionen weitere IPv4-Adressen aus dem 0.0.0.0/8-Bereich werden verfügbar gemacht; ^[124] Letzte Version war 5.3.18 am 18. Dezember 2019
5.4	24. Nov. 2019	65.701	27.538.212	≈ 109.400	70	LTS-Version; Unterstützung für Microsofts exFAT-Dateisystem, Integration des Kernel Lockdown-Features, Support für weitere AMD GPU/APU-Produkte (Navi 12/14, Arcturus, Renoir), verbesserte Performance für Host-Dateisystemzugriffe aus einer virtuellen Maschine mit <i>virtio-fs</i> ; ^[125] Aktuelle Version: 5.4.197 am 6. Juni 2022
5.5	27. Jan. 2020	66.493	27.854.754	≈ 108.100	63	u. a. Grundlagen für den zukünftigen Support der VPN-Technik WireGuard gelegt, anfängliche Unterstützung des Raspberry Pi 4, modernisierter Code für die Lastverteilung des Schedulers, mit KUnit wurde ein Framework für Modultests integriert, Btrfs erhält neue RAID-1 und Hash-Algorithmen, verbesserte Performance und/oder Stabilität dank Multichannel-Support für CIFS/SMB, die Temperatur von NVMe-Laufwerken kann vom Kernel ausgelesen und bereitgestellt werden; ^[126] Letzte Version war 5.5.19 am 21. April 2020
5.6	29. März 2020	67.337	28.169.797	≈ 109.200	63	u. a. WireGuard wird vollständig unterstützt, weiter ausgebaut wurde der Support des Raspberry Pi 4 (PCIe-Controller), erste Bausteine für die USB4-Unterstützung aufgenommen, der k10temp-Treiber übermittelt nun Temperatur-, Stromstärke- und Spannungswerte für Zen-CPUs, <i>/dev/random</i> liefert Zufallszahlen auch bei leeren Entropiequellen, wenn der Cryptographic Random Number Generator (CRNG) einmalig initialisiert wurde (Programme, die Zufallszahlen auf diese Weise anfordern, werden so nicht mehr blockiert), Linux als Gastsystem einer VirtualBox-VM erlaubt dank des eingepflegten VirtualBox-Shared-Folder-Treibers (vboxsf) eine effiziente Einbindung von Verzeichnissen des Hostsystems, erstmals sind alle Voraussetzungen zur Bewältigung des Jahr-2038-Problems für 32-Bit-Betriebssysteme erfüllt; ^{[127][128]} Letzte Version war 5.6.19 am 17. Juni 2020
5.7	31. Mai 2020	67.939	28.442.333			Letzte Version war 5.7.19 am 27. August 2020
5.8	2. Aug. 2020	69.327	28.994.351			Letzte Version war 5.8.18 am 1. November 2020
5.9	11. Okt. 2020	69.972	29.461.217			Letzte Version war 5.9.16 am 21. Dezember 2020
5.10	13. Dez. 2020					Aktuelle Version: 5.10.121 am 9. Juni 2022
5.11	14. Feb. 2021					Letzte Version war 5.11.22 am 19. Mai 2021

5.12	25. Apr. 2021					Letzte Version war 5.12.19 am 20. Juli 2021
5.13	28. Juni 2021					Letzte Version war 5.13.19 am 18. September 2021
5.14	29. Aug. 2021					Support für Raspberry Pi 400. Verbesserte Unterstützung von USB 4. Grundlagen für Intel Alder Lake Prozessoren. Letzte Version war 5.14.21 am 21. November 2021
5.15	31. Okt. 2021					Aktuelle Version: 5.15.46 am 9. Juni 2022
5.16	9. Jan. 2022					Letzte Version war 5.16.20 am 13. April 2022
5.17	20. März 2022					Aktuelle Version: 5.17.14 am 9. Juni 2022
5.18	22. Mai 2022					Mit dieser Version wechselt der Kernel erstmals in der Geschichte die C-Sprachfassung: statt bisher C89 basiert Linux ab Version 5.18 auf dem C-Standard C11 (von 2011). ^[129] Aktuelle Version: 5.18.3 am 9. Juni 2022
5.19	2022					Aktuelle Version: 5.19 RC1 am 6. Juni 2022
Legende: ■ Ältere Version; nicht mehr unterstützt ■ Ältere Version; noch unterstützt ■ Aktuelle Version ■ Zukünftige Version						

Anmerkungen^{[130][131][132][133]}

- (a) Dateien gezählt mit: `find . -type f -not -regex '\./\..git/.*' | wc -l`
- (b) Quelltextzeilen gezählt mit: `find . -type f -not -regex '\./\..git.*' | xargs cat | wc -l`
- (c) Größe in kB bezogen auf ein mit gzip komprimiertes tar-Archiv (.tar.gz); ab Version 3.2 im Format tar.xz^[18]
- (d) Der genannte Entwicklungszeitraum bezieht sich lediglich auf die Zusammenführung bereits entwickelter Programmteile, welche selbst teilweise mehrere Jahre zuvor bis zur Zusammenführung entwickelt wurden.

Neuerungen im Kernel 2.6

Die Kernel-Reihe 2.6 wurde ab Dezember 2001 auf Basis der damaligen 2.4er-Reihe entwickelt und wies umfangreiche Neuerungen auf. Für die Entwicklung war der neue Quelltext übersichtlicher und leichter zu pflegen, während Anwender durch die Überarbeitung des Prozess-Schedulers sowie des I/O-Bereiches und von geringeren Latenzzeiten profitierten.^[134] Dies wurde durch eine Reihe von Maßnahmen erreicht, die im Folgenden aufgezeigt werden:

Neue Prozess-Scheduler

In einem Multitasking-fähigen Betriebssystem muss es eine Instanz geben, die den Prozessen, die laufen wollen, Rechenzeit zuteilt. Diese Instanz bildet der Prozess-Scheduler. Seit dem Erscheinen von Linux 2.6 wurde mehrfach grundlegend am Scheduler gearbeitet.

Für die ersten Kernel 2.6 war von Ingo Molnár ein gegenüber Linux 2.4 ganz neuer Scheduler konzipiert und implementiert worden, der O(1)-Scheduler. Dieser erhielt seinen Namen, weil die relevanten Algorithmen, auf denen der Scheduler basierte, die Zeitkomplexität **O(1)** haben. Dies bedeutet, dass die vom Scheduler für eigene Aufgaben benötigte Prozessorzeit unabhängig von der Anzahl der verwalteten Prozesse bzw. Threads ist. Insbesondere wurde etwa auf das Durchsuchen aller Prozesse nach dem momentan wichtigsten Prozess verzichtet.

Der O(1)-Scheduler arbeitete auch bei sehr vielen Prozessen überaus effizient und benötigte selbst sehr wenig Rechenzeit. Er verwendete prinzipiell zwei verkettete Listen, in denen die Prozesse eingetragen waren, die noch laufen wollten, und diejenigen, die bereits gelaufen sind. Wenn alle Prozesse in der zweiten Liste standen, wurden die Datenfelder getauscht, und das Spiel begann von neuem. Der Scheduler war darüber hinaus so ausgelegt, dass Prozesse, die große Mengen Rechenzeit in Anspruch nehmen wollen, gegenüber interaktiven Prozessen benachteiligt werden, wenn beide zur gleichen Zeit laufen wollen.

Interaktive Prozesse benötigen in der Regel nur sehr wenig Rechenzeit, sind dafür aber sehr zeitkritisch (so will der Benutzer beispielsweise nicht lange auf eine Reaktion der grafischen Oberfläche warten). Der O(1)-Scheduler besaß Heuristiken, um festzustellen, ob ein Prozess interaktiv ist oder die CPU eher lange belegt.

Der interne „Takt“ des Kernels wurde ab dem Kernel 2.6 von 100 auf 1000 Hertz erhöht, das heißt, die kürzestmögliche Länge einer Zeitscheibe beträgt nun eine Millisekunde. Auch hiervon profitieren besonders die interaktiven Prozesse, da sie früher „wieder an der Reihe sind“. Da dies aber zu einer erhöhten CPU-Last und somit zu einem größeren Stromverbrauch führt, entschied man, den Takt ab dem Kernel 2.6.13 auf 250 Hertz voreinzustellen. Bei der Konfiguration des Kernels sind jedoch auch noch die Werte 100, 300 und 1000 Hertz wählbar.

Mit der Kernelversion 2.6.23 wurde im Oktober 2007 der O(1)-Scheduler durch einen *Completely Fair Scheduler* (kurz CFS) ersetzt, der ebenfalls von Ingo Molnár entwickelt wurde. Der CFS als gegenwärtig einziger im Hauptentwicklungszweig verfügbarer Scheduler ist unter den Kernel-Entwicklern teilweise umstritten, da er seinen Schwerpunkt auf Skalierbarkeit auch bei Servern mit vielen Prozessorkernen legt. Entwickler wie Con Kolivas sind der Meinung, dass unter dieser Schwerpunktsetzung sowie einigen Designentscheidungen im CFS die Leistung auf typischen Desktop-Systemen leide.^[135]

Präemptiver Kernel

Der Kernel ist ab Version 2.6 in den meisten Funktionen präemptiv, d. h. selbst wenn das System gerade im Kernel-Modus Aufgaben ausführt, kann dieser Vorgang durch einen Prozess aus dem User-Modus unterbrochen werden. Der Kernel macht dann weiter, wenn der Usermodus-Prozess seine Zeitscheibe aufgebraucht hat oder selbst eine neue Zeitplanung (englisch *Re-Schedule*) anfordert, also dem Zeitplaner (englisch *Scheduler*) mitteilt, dass er einen anderen Task ausführen kann. Dies funktioniert, bis auf einige Kernel-Funktionen, die atomar (nicht unterbrechbar) ablaufen müssen, sehr gut und kommt ebenfalls der Interaktivität zugute.

Zugriffskontrolllisten

Mit dem Kernel 2.6 werden für Linux erstmals Zugriffskontrolllisten (englisch *access control lists*) nativ eingeführt. Diese sehr feinkörnige Rechteverwaltung ermöglicht es vor allem Systemadministratoren, die Rechte auf einem Dateisystem unabhängig vom Gruppen- und Nutzermodell zu gestalten und dabei faktisch beliebig viele spezielle Rechte pro Datei zu setzen. Die mangelnde Unterstützung von Zugriffskontrolllisten von Linux wurde vorher als massive Schwäche des Systems im Rahmen der Rechteverwaltung und der Möglichkeiten zur sicheren Konfiguration gesehen.

Die Unterstützung von Zugriffskontrolllisten funktioniert dabei mit den Dateisystemen ext2, ext3, jfs und XFS nativ.

Inotify

Mit dem Kernel 2.6.13 hielt erstmals eine *Inotify* genannte Funktion Einzug in den Kernel. Diese ermöglicht eine andauernde Überwachung von Dateien und Verzeichnissen – wird eines der überwachten Objekte geändert oder ein neues Objekt im Überwachungsraum erschaffen, gibt *Inotify* eine Meldung aus, die wiederum andere Programme zu definierten Tätigkeiten veranlassen kann. Dies ist insbesondere für Such- und Indexierungsfunktionen der Datenbestände von entscheidender Bedeutung und ermöglicht erst den sinnvollen Einsatz von Desktop-Suchmaschinen wie Strigi oder Meta Tracker. Ohne eine solche Benachrichtigungsfunktion des Kernels müsste ein Prozess die zu überwachende Datei oder das zu überwachende Verzeichnis in bestimmten Zeitintervallen auf Änderungen überprüfen, was im Gegensatz zu *Inotify* zusätzliche Performance-Einbußen mit sich bringen würde.

Weitere wichtige Änderungen

Soweit es möglich ist, wurde in Linux 2.6 die Maximalzahl für bestimmte Ressourcen angehoben. Die Anzahl von möglichen Benutzern und Gruppen erhöhte sich von 65.000 auf über 4 Milliarden, ebenso wie die Anzahl der Prozess-IDs (von 32.000 auf 1 Milliarde) und die Anzahl der Geräte (Major/Minor-Nummern). Weitere leistungssteigernde Maßnahmen betrafen die I/O-Scheduler, das Threading mit der neuen Native POSIX Thread Library und den Netzwerk-Stack, der nun ebenfalls in den meisten Tests O(1) skaliert ist. Außerdem wurde für die Verwaltung der I/O-Geräte dateien das früher genutzte devfs durch das neuere udev ersetzt, was viele Unzulänglichkeiten, wie zum Beispiel ein zu großes */dev*-Verzeichnis, beseitigt. Außerdem kann so eine einheitliche und konsistente Gerätebenennung erfolgen, die beständig bleibt, was vorher nicht der Fall war.

Lizenzbesonderheiten

Proprietärer Code und Freiheitsbegriff

Die heute von Linus Torvalds herausgegebene Fassung des Kernels enthält proprietäre Objekte in Maschinsprache (BLOBs) und ist daher nicht mehr ausschließlich Freie Software. Richard Stallman bezweifelt sogar, dass sie legal kopiert werden darf, da diese BLOBs im Widerspruch zur GPL stünden und die Rechte aus der GPL daher erlöschen würden.^[136] Resultierend daraus rät die Free Software Foundation deshalb dazu, nur BLOB-freie Versionen von Linux einzusetzen, bei denen diese Bestandteile entfernt wurden. Linux-Distributionen mit dem Kernel Linux-libre erfüllen diesen Anspruch.

Der Kernel unter der GPL 2

Die bei GPL-Software übliche Klausel, dass statt der Version 2 der GPL auch eine neuere Version verwendet werden kann, fehlt beim Linux-Kernel.^[137] Die Entscheidung, ob die im Juni 2007 erschienene Version 3 der Lizenz für Linux verwendet wird, ist damit nur mit Zustimmung aller Entwickler möglich. In einer Umfrage haben sich Torvalds und die meisten anderen Entwickler für die Beibehaltung der Version 2 der Lizenz ausgesprochen.

Literatur

- Wolfgang Mauerer: *Linux-Kernelarchitektur. Konzepte, Strukturen und Algorithmen von Kernel 2.6*. Hanser Fachbuchverlag, München u. a. 2003, ISBN 3-446-22566-8.
- Robert Love: *Linux-Kernel-Handbuch. Leitfaden zu Design und Implementierung von Kernel 2.6*. Addison-Wesley, München u. a. 2005, ISBN 3-8273-2204-9.
- Jonathan Corbet, Alessandro Rubini, und Greg Kroah-Hartman: *Linux Device Drivers*. 3. Auflage, O'Reilly. 2005, ISBN 0-596-00590-3.

Weblinks

 **Commons: Linux (Kernel)** (https://commons.wikimedia.org/wiki/Linux_kernel?uselang=de) – Album mit Bildern, Videos und Audiodateien

Englisch:


- The Linux Kernel Archives (<https://www.kernel.org/>) – offizielle Website
- Elixir Cross Referencer (<https://elixir.bootlin.com/linux/latest/source>)
- Interactive Linux Kernel Map (http://www.makelinux.net/kernel_map/)
- Linux Kernel Newbies (<https://kernelnewbies.org/>) – Infos für angehende Kernel-Programmierer
- Oldlinux.org (<http://www.oldlinux.org/>) – eine Sammlung historischer Kernel
- Linux 0.01 News (http://draconux.free.fr/os_dev/linux0.01_news.html) – Seite zu Weiterentwicklungen des 0.01-Kernels für neuere GCC-Versionen
- Anatomy of the Linux kernel (<https://developer.ibm.com/articles/l-linux-kernel/>) – *IBM*, am 6. Juni 2007

Deutsch:

- Linux-Kernel (<https://www.golem.de/specials/linux-kernel/>) – *Golem*
- *Linux Kernel* (https://www.heise.de/thema/Linux_Kernel) – *Heise*
- *Linux Kernel Download* (<https://www.computerbase.de/downloads/system/linux-kernel/>) – *ComputerBase*

Einzelnachweise

1. *COPYING*. (<https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable.git/tree/COPYING>) Abgerufen am 18. August 2017 (englisch).
2. Reiko Kaps: *Linus Torvalds kritisiert den zweiten GPLv3-Entwurf*. (<https://heise.de/-146474>) In: *Heise online*. 28. Juli 2006. Abgerufen am 30. Oktober 2012.
3. *GNU Linux-libre*. (<https://directory.fsf.org/wiki/Linux-libre>) Abgerufen am 18. August 2017 (englisch).
4. Linus Torvalds: *RELEASE NOTES FOR LINUX v0.12*. (<https://www.kernel.org/pub/linux/kernel/Historic/old-versions/RELNOTES-0.12>) In: *kernel.org*. Abgerufen am 22. April 2017 (englisch).
5. index: kernel/git/torvalds/linux.git (<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/tree/Documentation/process/stable-api-nonsense.rst>)
6. *UNIX/Linux History*. (<https://digital-domain.net/lug/unix-linux-history.html>) Abgerufen am 11. März 2017 (englisch).

7. *PPC/Linux for NuBus Power Macs*. (<https://sourceforge.net/projects/nubus-pmac/>) sourceforge.net, abgerufen am 12. April 2014 (englisch).
8. Katherine Noyes: *Linux 3.8: Hello 2013, Goodbye 386 Chips*. (<https://www.linux.com/news/software/linux-kernel/701695-linux-38-hello-2013-goodbye-386-chips>) In: *Linux News*. Linux Foundation, 20. Februar 2013, abgerufen am 14. April 2013 (englisch).
9. Andres Calderon, Nelson Castillo: *Why ARM's EABI matters*. (<http://linuxdevices.org/why-arms-eabi-matters/>) LinuxDevices.com, 14. Mai 2007, abgerufen am 10. April 2019 (englisch).
10. Pressemitteilung OSDL: *Developer's Certificate of Origin*. (http://www.osdl.org/newsroom/press_releases/2004/2004_05_24_dco.html) 2004.
11. Linux-Kernel Archive: *[RFD] Explicitly documenting patch submission* (<https://lkml.org/lkml/2004/5/23/10>), 23. Mai 2004 (englisch)
12. Thorsten Leemhuis: *Neue Kernel-Serie mit Linux 2.6.11.1 gestartet*. (<https://heise.de/-140974>) In: *Heise online*. 7. März 2005. Abgerufen am 26. April 2015.
13. Linus Torvalds: *Linux 3.0-rc1*. (<https://web.archive.org/web/20161130121934/http://thread.gmane.org/gmane.linux.kernel/1147415>) (Nicht mehr online verfügbar.) thread.gmane.org, 30. Mai 2011, archiviert vom Original (<https://giftbot.toolforge.org/deref.fcgi?url=http%3A%2F%2Fthread.gmane.org%2Fgmane.linux.kernel%2F1147415>) am 30. November 2016; abgerufen am 16. November 2014 (englisch).  **Info:** Der Archivlink wurde automatisch eingesetzt und noch nicht geprüft. Bitte prüfe Original- und Archivlink gemäß [Anleitung](#) und entferne dann diesen Hinweis.
14. Linus Torvalds: *index: kernel/git/torvalds/linux.git*. (<https://git.kernel.org/cgiit/linux/kernel/git/torvalds/linux.git/commit/?id=c517d838eb7d07bbe9507871fab3931deccff539>) git.kernel.org, 23. Februar 2015, abgerufen am 23. Februar 2015 (englisch).
15. Linus Torvalds: *Linus Torvalds — Google+*. (<https://plus.google.com/+LinusTorvalds/posts/jmtzzLiiejc>) plus.google.com, 13. Februar 2015, abgerufen am 23. Februar 2015 (englisch).
16. *Linux 5.0 ist da: Geschwindigkeit zurückerobern und moderner speichern* (<https://www.heise.de/ct/artikel/Linux-5-0-ist-da-Geschwindigkeit-zurueckerobern-und-moderner-speichern-4307315.html>). heise.de/ct. 4. März 2019. Abgerufen am 4. März 2019.
17. *Active kernel releases*. (<https://www.kernel.org/category/releases.html>) kernel.org, abgerufen am 1. November 2021 (englisch).
18. *The Linux Kernel Archives*. (<https://www.kernel.org/>) kernel.org, abgerufen am 6. September 2010 (englisch, Versionsübersicht bei [kernel.org](https://www.kernel.org/)).
19. Oliver Diedrich: *Happy Birthday, Tux*. (<https://heise.de/-155501>) In: *Heise online*. 25. August 2006. Abgerufen am 24. Oktober 2014.
20. Dr. Oliver Diedrich: *Die Woche: Linux wird 3.0*. (<https://heise.de/-1254128>) In: *Heise online*. 1. Juni 2011. Abgerufen am 26. April 2015.
21. Andreas Jaeger: *Large File Support in Linux* (https://users.suse.com/~aj/linux_lfs.html). SUSE GmbH. 15. Februar 2015.
22. Oliver Diedrich: *Neuer Firewall-Code für den Linux-Kernel*. (<https://heise.de/-1982608>) In: *Heise online*. 21. Oktober 2013. Abgerufen am 28. Oktober 2013.
23. LKML: Willy Tarreau: *Linux 2.4.37.10 + 2.4 EOL plans* (<https://lkml.org/lkml/2010/9/6/15>) – Mitteilung bei der *Linux-Kernel Mailing List*, vom 6. September 2010, abgerufen am: 16. September 2012 (englisch)
24. *Linux-Kernel 2.6.13 veröffentlicht* (<https://www.heise.de/newsticker/meldung/Linux-Kernel-2-6-13-veroeffentlicht-126180.html>) – Thorsten Leemhuis, bei *Heise*, am 29. August 2005
25. Thorsten Leemhuis: *Die Woche: Xen hat KVM vorbeiziehen lassen*. (<https://heise.de/-1261765>) In: *Heise online*. 16. Juni 2011. Abgerufen am 24. Oktober 2014.
26. Jörg Thoma: *Linux-Kernel: Updates für Long-Term-Zweige*. (<https://www.golem.de/1105/83224.html>) Golem.de, 4. Mai 2011, abgerufen am 24. Oktober 2014.
27. Clever schachteln: *Das neue Linux-Dateisystem Btrfs im Detail* (<http://www.admin-magazin.de/Das-Heft/2009/04/Das-neue-Linux-Dateisystem-Btrfs-im-Detail>) – *Admin-Magazin*, Ausgabe 04/2009
28. Dr. Oliver Diedrich: *Das Dateisystem Btrfs*. (<https://heise.de/-221863>) In: *Heise online*. 7. Juli 2009. Abgerufen am 14. August 2016.
29. Thorsten Leemhuis: *Kernel-Log: Entwicklung von 2.6.39 angelaufen*. (<https://heise.de/-1212541>) In: *Heise online*. 22. März 2011. Abgerufen am 26. April 2015.
30. Thorsten Leemhuis: *Hauptentwicklungsphase von Linux 3.2 abgeschlossen*. (<https://heise.de/-1374691>) In: *Heise online*. 8. November 2011. Abgerufen am 26. April 2015.
31. Thorsten Leemhuis: *Die Neuerungen von Linux 2.6.34*. (<https://heise.de/-999161?seite=3>) In: *Heise online*. 17. Mai 2010. S. 3: *Linux 2.6.34: Dateisysteme, Netzwerk*. (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-2-6-34-999161.html?seite=3>) Abgerufen am 30. November 2015.

32. Thorsten Leemhuis: *Kernel-Log – Was 2.6.34 bringt (2): Dateisysteme*. (<https://heise.de/-984551>) In: *Heise online*. 23. April 2010. Abgerufen am 9. Dezember 2016.
33. Thorsten Leemhuis: *Kernel-Log: Cluster-Dateisystem Ceph in 2.6.34, Kernel- und KVM-Vorträge von den CLT2010*. (<https://heise.de/-959928>) In: *Heise online*. 22. März 2010. Abgerufen am 9. Dezember 2016.
34. Thorsten Leemhuis: *Kernel-Log – Was 2.6.34 bringt (3): Grafik*. (<https://heise.de/-993746>) In: *Heise online*. 5. Mai 2010. Abgerufen am 9. Dezember 2016.
35. Oliver Diedrich: *Entwicklung von Linux 2.6.35 läuft an*. (<https://heise.de/-1012414>) In: *Heise online*. 31. Mai 2010. Abgerufen am 9. Dezember 2016.
36. Thorsten Leemhuis: *Eine Flut neuer Stable-Kernel und ein Ausblick auf Linux 2.6.35*. (<https://heise.de/-1033772>) In: *Heise online*. 6. Juli 2010. Abgerufen am 9. Dezember 2016.
37. Andi Kleen: *The longterm Linux 2.6.35.14 kernel is released*. (<https://lkml.org/lkml/2011/8/1/324>) lkml.org, 1. August 2011, abgerufen am 7. Mai 2012 (englisch).
38. Thorsten Leemhuis: *Die Neuerungen von Linux 2.6.36*. (<https://heise.de/-1102066>) In: *Heise online*. 21. Oktober 2010. Abgerufen am 17. August 2015.
39. Thorsten Leemhuis: *Kernel-Log: 2.6.36, neue Stable-Kernel, frische Treiber*. (<https://heise.de/-1053519>) In: *Heise online*. 11. August 2010. Abgerufen am 17. August 2015.
40. Thorsten Leemhuis: *Hauptentwicklungsphase des Linux-Kernels 2.6.36 abgeschlossen*. (<https://heise.de/-1058700>) In: *Heise online*. 16. August 2010. Abgerufen am 17. August 2015.
41. Thorsten Leemhuis: *Hauptentwicklungsphase des Linux-Kernel 2.6.37 abgeschlossen*. (<https://heise.de/-1128535>) In: *Heise online*. 1. November 2010. Abgerufen am 15. Februar 2015.
42. Thorsten Leemhuis: *Kernel-Log: Wunderpatch integriert, verbesserte Grafiktreiber am Start*. (<https://heise.de/-1165324>) In: *Heise online*. 7. Januar 2011. Abgerufen am 10. März 2015.
43. Thorsten Leemhuis: *Kernel-Log – Was 2.6.38 bringt (6): Treiber*. (<https://heise.de/-1202105>) In: *Heise online*. 9. März 2011. Abgerufen am 15. Februar 2015.
44. Thorsten Leemhuis: *Kernel-Log: Erster Release Candidate von Linux 2.6.39*. (<https://heise.de/-1217118>) In: *Heise online*. 30. März 2011. Abgerufen am 8. Februar 2015.
45. Thorsten Leemhuis: *Die Neuerungen von Linux 2.6.39*. (<https://heise.de/-1242011>) In: *Heise online*. 19. Mai 2011. Abgerufen am 8. Februar 2015.
46. Arnd Bergmann: *BKL: That's all, folks*. (<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=4ba8216cd90560bc402f52076f64d8546e8aefcb>) git.kernel.org, 25. Januar 2011, abgerufen am 18. August 2017.
47. Thorsten Leemhuis: *Erste Vorabversion von Linux 3.0 erschienen*. (<https://heise.de/-1252084>) In: *Heise online*. 30. Mai 2011. Abgerufen am 24. Januar 2015.
48. Jörg Thoma: *Kernel: Linux 3.0 ist fertig*. (<https://www.golem.de/1107/85080.html>) golem.de, 22. Juli 2011, abgerufen am 6. Januar 2015.
49. Thorsten Leemhuis: *Linux-Kernel 3.0 freigegeben*. (<https://heise.de/-1279618>) In: *Heise online*. 22. Juli 2011. Abgerufen am 6. Januar 2015.
50. Thorsten Leemhuis: *Kernel-Log – Was 3.1 bringt (3): Architektur, Infrastruktur, Virtualisierung*. (<https://heise.de/-1344563>) In: *Heise online*. 17. September 2011. Abgerufen am 6. Januar 2015.
51. Mirko Dölle: *Android-Treiber sollen in Kernel 3.3 einfließen*. (<https://heise.de/-1400888>) In: *Heise online*. 23. Dezember 2011. Abgerufen am 16. November 2014.
52. Jörg Thoma: *Stromsparoption RC6 in Linux-Kernel funktionstüchtig*. (<https://www.golem.de/news/sandy-bridge-stromsparoption-rc6-in-linux-kernel-funktionstuechtig-1202-89809.html>) golem.de, 15. Februar 2012, abgerufen am 21. Mai 2012.
53. Thorsten Leemhuis: *Kernel-Log – Was 3.3 bringt (4): Treiber*. (<https://heise.de/-1464001>) In: *Heise online*. 8. März 2012. Abgerufen am 21. Mai 2012.
54. Thorsten Leemhuis: *Die Neuerungen von Linux 3.4*. (<https://heise.de/-1572878>) In: *Heise online*. 21. Mai 2012. Abgerufen am 21. Mai 2012.
55. Thorsten Leemhuis: *Kernel-Log – Was 3.5 bringt (1): Netzwerk*. (<https://heise.de/-1624233>) In: *Heise online*. 25. Juni 2012. Abgerufen am 27. Juni 2012.
56. Thorsten Leemhuis: *Kernel-Log: Entwicklung von Linux 3.6 läuft*. (<https://heise.de/-1656817>) In: *Heise online*. 2. August 2012. Abgerufen am 11. Mai 2014.
57. Thorsten Leemhuis: *Linux-Kernel 3.7 veröffentlicht*. (<https://heise.de/-1760312>) In: *Heise online*. 11. Dezember 2012. Abgerufen am 13. Dezember 2012.
58. Thorsten Leemhuis: *Kernel-Log – Was 3.8 bringt (1): Dateisysteme und Storage*. (<https://heise.de/-1788185>) In: *Heise online*. 21. Januar 2013. Abgerufen am 16. Februar 2013.
59. Thorsten Leemhuis: *Kernel-Log – Was 3.8 bringt (2): Infrastruktur*. (<https://heise.de/-1800418>) In: *Heise online*. 8. Februar 2013. Abgerufen am 16. Februar 2013.

60. Jörg Thoma: *Linux Kernel: Torvalds schließt Merge-Fenster für Linux 3.9.* (<https://www.golem.de/news/linux-kernel-torvalds-schliesst-merge-fenster-fuer-linux-3-9-1303-97966.html>) golem.de, 4. März 2013, abgerufen am 6. März 2013.
61. Meta SoC Processors (<https://web.archive.org/web/20130315125508/http://www.imgtec.com/meta/meta.asp>) (Memento vom 15. März 2013 im *Internet Archive*) – Seite bei *Imagination Technologies*, abgerufen am 7. März 2013 (englisch).
62. DesignWare ARC 700 Processor Core Family (<https://www.synopsys.com/designware-ip/processor-solution/s/arc-processors/arc-700-family.html>) – Seite bei *Synopsys*, abgerufen am 7. März 2013 (englisch).
63. Hans-Joachim Baader: *Linux-Kernel 3.9 tritt in die Testphase ein.* (<https://www.pro-linux.de/news/1/19520/linux-kernel-39-tritt-in-die-testphase-ein.html>) *Pro-Linux*, 4. März 2013, abgerufen am 7. März 2013.
64. Thorsten Leemhuis: *Kernel-Log – Was 3.10 bringt (4): Treiber.* (<https://heise.de/-1897241>) In: *Heise online*. 28. Juni 2013. Abgerufen am 10. Juli 2013.
65. Thorsten Leemhuis: *Die Neuerungen von Linux 3.9.* (<https://heise.de/-1844759?seite=3>) In: *Heise online*. 29. April 2013. S. 3: *Fazit, Trends, Statistik.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-3-9-1844759.html?seite=3>) Abgerufen am 10. Juli 2013.
66. Thorsten Leemhuis: *Die Neuerungen von Linux 3.10.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-3-10-1898006.html?artikelseite=3>) Heise, 1. Juli 2013, abgerufen am 10. Juli 2013.
67. Thorsten Leemhuis: *Kernel-Log – Was 3.11 bringt (2): Infrastruktur.* (<https://www.heise.de/ct/artikel/Kernel-Log-Was-3-11-bringt-2-Infrastruktur-1937862.html>) Heise, 19. August 2013, abgerufen am 20. August 2013.
68. Thorsten Leemhuis: *"Linux for Workgroups": Funktionsumfang von Linux 3.11 steht.* (<https://www.heise.de/newsticker/meldung/Linux-for-Workgroups-Funktionsumfang-von-Linux-3-11-steht-1917174.html>) Heise, 15. Juli 2013, abgerufen am 13. März 2018.
69. Thorsten Leemhuis: *Linux 3.12 bringt Optimus- und SYN-Proxy-Unterstützung.* (<https://www.heise.de/newsticker/meldung/Linux-3-12-bringt-Optimus-und-SYN-Proxy-Unterstuetzung-1958491.html>) Heise, 17. September 2013, abgerufen am 18. September 2013.
70. Thorsten Leemhuis: *Die Neuerungen von Linux 3.12: Netzwerk.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-3-12-1981573.html?artikelseite=2>) Heise, 4. November 2013, abgerufen am 6. November 2013.
71. Thorsten Leemhuis: *Kernel-Log – Was 3.12 bringt (1): Dateisysteme & Storage.* (<https://www.heise.de/ct/artikel/Kernel-Log-Was-3-12-bringt-1-Dateisysteme-Storage-1973671.html>) Heise, 8. Oktober 2013, abgerufen am 6. November 2013.
72. Thorsten Leemhuis: *Die Neuerungen von Linux 3.13.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-3-13-2087964.html>) Heise online, 20. Januar 2014, abgerufen am 20. Januar 2014.
73. Thorsten Leemhuis: *Kernel-Log – Was 3.13 bringt (5): Grafiktreiber.* (<https://www.heise.de/ct/artikel/Kernel-Log-Was-3-13-bringt-5-Grafiktreiber-2074557.html>) Heise online, 10. Januar 2014, abgerufen am 20. Januar 2014.
74. Thorsten Leemhuis: *Kernel-Log – Was 3.13 bringt (1): Dateisysteme und Storage.* (<https://www.heise.de/ct/artikel/Kernel-Log-Was-3-13-bringt-1-Dateisysteme-und-Storage-2062929.html>) Heise online, 10. Dezember 2013, abgerufen am 20. Januar 2014.
75. Jörg Thoma: *Prozessfristen: Linux 3.14 erhält Deadline Scheduler.* (<https://www.golem.de/news/prozessfristen-linux-3-14-erhaelt-deadline-scheduler-1403-105475.html>) golem.de, 31. März 2014, abgerufen am 31. März 2014.
76. Peter Siering: *Xen 4.4: virtuelle Maschinen für ARM.* (<https://www.heise.de/newsticker/meldung/Xen-4-4-virtuelle-Maschinen-fuer-ARM-2139439.html>) heise.de, 10. März 2014, abgerufen am 11. März 2014.
77. Thorsten Leemhuis: *Kernel-Log – Was 3.15 bringt (1): Dateisysteme und Storage.* (<https://www.heise.de/ct/artikel/Kernel-Log-Was-3-15-bringt-1-Dateisysteme-und-Storage-2183256.html>) heise.de, 6. Mai 2014, abgerufen am 6. Mai 2014.
78. Thorsten Leemhuis: *Die Neuerungen von Linux 3.16.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-3-16-2278810.html>) heise.de, 4. August 2014, abgerufen am 17. August 2014.
79. Thorsten Leemhuis: *Die Neuerungen von Linux 3.17.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-3-17-2404284.html>) heise.de, 6. Oktober 2014, abgerufen am 10. November 2014.
80. Hans-Joachim Baader: *Linux-Kernel 3.18 freigegeben.* (<https://www.pro-linux.de/news/1/21815/linux-kernel-3-18-freigegeben.html>) *Pro-Linux*, 8. Dezember 2014, abgerufen am 8. Dezember 2014.
81. Thorsten Leemhuis: *Linux 3.19 freigegeben: HSA-Support für AMD-Chips.* (<https://www.heise.de/newsticker/meldung/Linux-3-19-freigegeben-HSA-Support-fuer-AMD-Chips-2543019.html>) Heise online, 9. Februar 2015, abgerufen am 9. Februar 2015.
82. Jörg Thoma: *Linux 3.19: Mehr für die Grafik und für das Netzwerk.* (<https://www.golem.de/1502/112227.html>) golem.de, 9. Februar 2015, abgerufen am 9. Februar 2015.

83. Zeitgemäße Linux-IPC (<http://www.linux-magazin.de/Ausgaben/2015/02/Android-IPC>) – *Linux-Magazin*, Ausgabe 02/2015
84. Thorsten Leemhuis: *Linux-Kernel macht Versionssprung auf 4.0.* (<https://www.heise.de/newsticker/meldung/Linux-Kernel-macht-Versionssprung-auf-4-0-2557302.html>) heise.de, 23. Februar 2015, abgerufen am 23. Februar 2015.
85. Jörg Thoma: *Version 4.0 patcht sich selbst im laufenden Betrieb.* (<https://www.golem.de/1502/112528.html>) golem.de, 23. Februar 2015, abgerufen am 23. Februar 2015.
86. Thorsten Leemhuis: *Linux 4.1 freigegeben: Ext4 verschlüsselt jetzt selbst.* (<https://www.heise.de/newsticker/meldung/Linux-4-1-freigegeben-Ext4-verschluesselt-jetzt-selbst-2717732.html>) heise.de, 22. Juni 2015, abgerufen am 22. Juni 2015.
87. Thorsten Leemhuis: *Linux 4.2-rc1: Neuer Grafiktreiber von AMD integriert.* (<https://www.heise.de/newsticker/meldung/Linux-4-2-rc1-Neuer-Grafiktreiber-von-AMD-integriert-2735351.html>) heise.de, 6. Juli 2015, abgerufen am 19. Juli 2015.
88. Thorsten Leemhuis: *Linux-Kernel 4.3 wird neue AMD- und Intel-Grafikprozessoren unterstützen.* (<https://www.heise.de/newsticker/meldung/Linux-Kernel-4-3-wird-neue-AMD-und-Intel-Grafikprozessoren-unterstuetzen-2806331.html>) heise.de, 5. September 2015, abgerufen am 15. September 2015.
89. Thorsten Leemhuis: *Linux 4.3 wird IPv6-Unterstützung standardmäßig nutzen.* (<https://www.heise.de/newsticker/meldung/Linux-4-3-wird-IPv6-Unterstuetzung-standardmaessig-nutzen-2811927.html>) heise.de, 14. September 2015, abgerufen am 15. September 2015.
90. Greg Kroah-Hartman: *The Linux 4.4 kernel will be the next LTS (long term support) release...* (<https://plus.google.com/+gregkroahhartman/posts/WMEWz4noxFm>) Greg Kroah-Hartman, 28. Oktober 2015, abgerufen am 2. November 2015 (englisch).
91. Thorsten Leemhuis: *Linux 4.4 freigegeben: Neue Grafiktreiber und Optimierungen für SSDs.* (<https://www.heise.de/newsticker/meldung/Linux-4-4-freigegeben-Neue-Grafiktreiber-und-Optimierungen-fuer-SSDs-3055350.html>) heise.de, 11. Januar 2016, abgerufen am 13. August 2016.
92. Sebastian Grüner: *Kernel: Linux 4.4 erscheint mit Grafiktreiber für Raspberry Pi.* (<https://www.golem.de/news/kernel-linux-4-4-erscheint-mit-grafiktreiber-fuer-raspberry-pi-1601-118449.html>) golem.de, 11. Januar 2016, abgerufen am 13. August 2016.
93. Thorsten Leemhuis: *Linux 4.5: Verbesserte 3D-Treiber und reibungslose Verbindungswechsel.* (<https://www.heise.de/newsticker/meldung/Linux-4-5-Verbesserte-3D-Treiber-und-reibungslose-Verbindungswechsel-3082334.html>) heise.de, 25. Januar 2016, abgerufen am 13. August 2016.
94. Ferdinand Thommes: *Kernel: Linux 4.5-rc1 wertet AMDGPU mit Powerplay auf.* (<https://www.golem.de/news/kernel-linux-4-5-rc1-wertet-amdgpu-mit-powerplay-auf-1601-118712.html>) golem.de, 11. Januar 2016, abgerufen am 13. August 2016.
95. Thorsten Leemhuis: *Die Neuerungen von Linux 4.5.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-4-5-3132326.html?seite=2>) heise.de, 14. März 2016, abgerufen am 15. Mai 2016.
96. Ferdinand Thommes: *Kernel: Linux 4.6-rc1 mit neuem Dateisystem OrangeFS.* (<https://www.golem.de/news/kernel-linux-4-6-rc1-mit-neuem-dateisystem-orangefs-1603-119995.html>) golem.de, 28. März 2016, abgerufen am 13. August 2016.
97. Thorsten Leemhuis: *Linux 4.6 wird 3D-Beschleunigung aktueller GeForce-Chips unterstützen.* (<https://www.heise.de/newsticker/meldung/Linux-4-6-wird-3D-Beschleunigung-aktueller-GeForce-Chips-unterstuetzen-3152433.html>) heise.de, 29. März 2016, abgerufen am 15. Mai 2016.
98. Thorsten Leemhuis: *Linux 4.7 soll AMDs neue Radeon-Grafikkarten unterstützen.* (<https://www.heise.de/newsticker/meldung/Linux-4-7-soll-AMDs-neue-Radeon-Grafikkarten-unterstuetzen-3221749.html>) heise.de, 30. Mai 2016, abgerufen am 2. Juli 2016.
99. Ferdinand Thommes: *Linux 4.7-rc1 unterstützt AMDs Polaris.* (<https://www.golem.de/news/kernel-linux-4-7-rc1-unterstuetzt-amds-polaris-1605-121157.html>) golem.de, 30. Mai 2016, abgerufen am 12. Dezember 2016.
100. Thorsten Leemhuis: *Linux 4.7 vermeidet Netzwerk-Latenzen und unterstützt SMR-Festplatten besser.* (<https://www.heise.de/newsticker/meldung/Linux-4-7-vermeidet-Netzwerk-Latenzen-und-unterstuetzt-SMR-Festplatte-besser-3238450.html>) heise.de, 15. Juni 2016, abgerufen am 15. Juni 2016.
101. Thorsten Leemhuis: *Linux 4.7 freigegeben: RX-480-Treiber und verbesserte Sicherheit.* (<https://www.heise.de/newsticker/meldung/Linux-4-7-freigegeben-RX-480-Treiber-und-verbesserte-Sicherheit-3266704.html>) heise.de, 25. Juli 2016, abgerufen am 18. August 2016.
102. Thorsten Leemhuis: *Linux 4.8 bringt Treiber für neue ARM-, Intel- und Nvidia-Grafikprozessoren.* (<https://www.heise.de/newsticker/meldung/Linux-4-8-bringt-Treiber-fuer-neue-ARM-Intel-und-Nvidia-Grafikprozessoren-3289567.html>) heise.de, 8. August 2016, abgerufen am 16. August 2016.
103. Jörg Thoma: *Betriebssysteme: Linux 4.8 übertaktet AMD-Grafikkarten.* (<https://www.golem.de/news/betriebssysteme-linux-4-8-uebertaktet-amd-grafikkarten-1608-122574.html>) golem.de, 8. August 2016, abgerufen am 16. August 2016.

104. Thorsten Leemhuis: *Linux 4.9 wird Sicherheit verbessern und neue Treiber bringen.* (<https://www.heise.de/newsticker/meldung/Linux-4-9-wird-Sicherheit-verbessern-und-neue-Treiber-bringen-3351310.html>) heise.de, 17. Oktober 2016, abgerufen am 18. Oktober 2016.
105. Ferdinand Thommes: *Kernel: Linux 4.9rc1 erscheint mit Protokoll für Project-Ara-Module.* (<https://www.golem.de/news/kernel-linux-4-9rc1-erscheint-mit-protokoll-fuer-project-ara-module-1610-123853.html>) golem.de, 17. Oktober 2016, abgerufen am 18. Oktober 2016.
106. Hans-Joachim Baader: *Erste Vorschau auf Linux-Kernel 4.10.* (<https://www.pro-linux.de/news/1/24318/erste-vorschau-auf-linux-kernel-410.html>) pro-linux.de, 26. Dezember 2016, abgerufen am 5. November 2017.
107. Thorsten Leemhuis: *Linux 4.10 Neue Wege zur Grafikbeschleunigung in virtuellen Maschinen.* (<https://www.heise.de/newsticker/meldung/Linux-4-10-Neue-Wege-zur-Grafikbeschleunigung-in-virtuellen-Maschinen-3597644.html>) heise.de, 17. Januar 2017, abgerufen am 5. November 2017.
108. Hans-Joachim Baader: *Linux-Kernel 4.10 freigegeben.* (<https://www.pro-linux.de/news/1/24471/linux-kernel-4-10-freigegeben.html>) In: *pro-linux.de*. 19. Februar 2017, abgerufen am 19. Februar 2017.
109. Linux-Kernel 4.11: Längere Akkulaufzeit durch NVMe-Stromspartechnik (<https://www.heise.de/newsticker/meldung/Linux-Kernel-4-11-Laengere-Akkulaufzeit-durch-NVMe-Stromspartechnik-3643585.html>) – Heise, am 6. März 2017
110. Thorsten Leemhuis: *Die Neuerungen von Linux 4.11.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-4-11-3641334.html>) heise.de, 16. April 2017, abgerufen am 5. November 2017.
111. Thorsten Leemhuis: *Die Neuerungen von Linux 4.12.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-4-12-3712705.html>) heise.de, 3. Juli 2017, abgerufen am 23. Juli 2017.
112. Thorsten Leemhuis: *Die Neuerungen von Linux 4.13.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-4-13-3771362.html>) heise.de, 4. September 2017, abgerufen am 24. September 2017.
113. Greg Kroah-Hartman: *As no one seemed to make 4.9 blow up too badly, let's try this again!* (<https://plus.google.com/+gregkroahhartman/posts/DmKQJZMVTXG>) plus.google.com, 17. Juni 2017, abgerufen am 17. Juni 2017 (englisch).
114. Thorsten Leemhuis: *Die Neuerungen von Linux 4.14.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-4-14-3831941.html>) heise.de, 13. November 2017, abgerufen am 18. März 2018.
115. Thorsten Leemhuis: *Die Neuerungen von Linux 4.15.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-4-15-3900646.html>) heise.de, 28. Januar 2018, abgerufen am 18. März 2018.
116. Thorsten Leemhuis: *Die Neuerungen von Linux 4.17.* (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-4-17-4061166.html>) heise.de, 4. Juni 2018, abgerufen am 4. Juni 2018.
117. Thorsten Leemhuis: *Linux 4.18 freigegeben.* (<https://www.heise.de/ct/artikel/Linux-4-18-freigegeben-Raspi-3-Support-und-Vorboten-neuer-Firewall-Technik-4078605.html>) heise.de, 13. August 2018, abgerufen am 13. August 2018.
118. Thorsten Leemhuis: *Kernel-Log: Linux 4.19: Flackerfrei starten und Strom sparen.* (<https://www.heise.de/ct/ausgabe/2018-22-Linux-4-19-Flackerfrei-starten-und-Strom-sparen-4184869.html>) heise.de, 12. Oktober 2018, abgerufen am 24. Oktober 2018.
119. Thorsten Leemhuis: *Linux 4.19: Schöner starten und bereit für das WLAN von Morgen.* (<https://www.heise.de/ct/artikel/Linux-4-19-Schoener-starten-und-bereit-fuer-das-WLAN-von-Morgen-4144037.html>) heise.de, 22. Oktober 2018, abgerufen am 24. Oktober 2018.
120. Thorsten Leemhuis: *Linux 4.20 freigegeben: Performance-Optimierungen und neue Treiber.* (<https://www.heise.de/ct/artikel/Linux-4-20-freigegeben-Performance-Optimierungen-und-neue-Treiber-4223066.html>) heise.de, 12. Dezember 2018, abgerufen am 24. Dezember 2018.
121. Thorsten Leemhuis: *Linux 5.0 ist da: Geschwindigkeit zurückerobern und moderner speichern.* (<https://www.heise.de/ct/artikel/Linux-5-0-ist-da-Geschwindigkeit-zurueckerobern-und-moderner-speichern-4307315.html>) heise.de, 4. März 2019, abgerufen am 24. Juni 2019.
122. Thorsten Leemhuis: *Linux 5.1: Performance-Verbesserungen und neue Speichertechnik.* (<https://www.heise.de/ct/artikel/Linux-5-1-Performance-Verbesserungen-und-neue-Speichertechnik-4411986.html>) heise.de, 6. Mai 2019, abgerufen am 24. Juni 2019.
123. Linux 5.2 freigegeben: Änderungsrekord und Geschwindigkeitsverbesserungen (<https://www.heise.de/ct/artikel/Linux-5-2-freigegeben-Aenderungsrekord-und-Geschwindigkeitsverbesserungen-4424484.html?seite=all>) – *c't Magazin*, am 9. Juli 2019
124. Thorsten Leemhuis: *Linux 5.3 freigegeben: Prioritäten deckeln und Trouble für Nvidia.* (<https://www.heise.de/ct/artikel/Linux-5-3-freigegeben-Prioritaeten-deckeln-und-Trouble-fuer-Nvidia-4470638.html>) Abgerufen am 26. November 2019.
125. Thorsten Leemhuis: *Linux 5.4 freigegeben: exFAT-Support und Einschränkungen für Root.* (<https://www.heise.de/ct/artikel/Linux-5-4-freigegeben-exFAT-Support-und-Einschraenkungen-fuer-Root-4541639.html>) Abgerufen am 26. November 2019.

126. Thorsten Leemhuis: *Linux 5.5 freigegeben: Wireguard-Fundament und Performance-Verbesserungen*. (<http://www.heise.de/ct/artikel/Linux-5-5-freigegeben-Wireguard-Fundament-und-Performance-Verbesserungen-4605827.html>) Abgerufen am 27. Januar 2020.
127. Thorsten Leemhuis: *Linux 5.6 freigegeben: Wireguard- und USB4-Support*. (<https://www.heise.de/ct/artikel/Linux-5-6-freigegeben-Wireguard-und-USB4-Support-4655269.html>) Abgerufen am 6. April 2020.
128. Thorsten Leemhuis: *Linux 5.6 unterstützt Wireguard und USB4*. (<https://www.heise.de/ct/artikel/Linux-5-6-unterstuetzt-Wireguard-und-USB4-4679177.html>) Abgerufen am 6. April 2020.
129. Oliver Müller: *Linux 5.18 kommt als „kleine Revolution“*. (<https://heise.de/-7121451?seite=3>) In: *Heise online*. 24. Mai 2022. S. 3: *Probleme mit altem C*. (<https://www.heise.de/news/Linux-5-18-kommt-als-kleine-Revolution-7121451.html?seite=3>) Abgerufen am 27. Mai 2022.; Zitat: „Der neue Linux-Kernel ... wechselt erstmalig zu einem neueren C-Standard. ... Bislang war das Programmieren nach dem C89-Standard in Kernel-Kreisen Pflicht. ... [Jakob] Koschel hatte eine Unsicherheit identifiziert, die auf ein in C89 nicht lösbares Problem zurückzuführen war. ... Erstmals in der Geschichte des Kernel wechselt der C-Standard weg von C89 zu C11. Damit ist fortan der Sprachstandard aus dem Jahr 2011 die Grundlage für die Programmierung des Kernels. Was nach einer Formalie aussieht, ist ein kontrolliertes Wagnis.“.
130. Thorsten Leemhuis: *Die Neuerungen von Linux 2.6.25*. (<https://www.heise.de/ct/artikel/Verbesserungen-aller-orten-Die-Neuerungen-von-Linux-2-6-25-221493.html>) Daten und Zahlen zu den jüngsten Versionen des Linux-Kernels. Heise online, 17. April 2008, S. 6, abgerufen am 28. September 2010.
131. Thorsten Leemhuis: *Die Neuerungen von Linux 2.6.31*. (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-2-6-31-763959.html>) Daten und Zahlen zu den jüngsten Versionen des Linux-Kernels. Heise online, 10. September 2009, S. 6, abgerufen am 28. September 2010.
132. Thorsten Leemhuis: *Die Neuerungen von Linux 2.6.38*. (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-2-6-38-1203073.html>) Daten und Zahlen zu den jüngsten Versionen des Linux-Kernels. Heise online, 15. März 2011, S. 8, abgerufen am 15. März 2011.
133. Thorsten Leemhuis: *Die Neuerungen von Linux 3.0*. (<https://www.heise.de/ct/artikel/Die-Neuerungen-von-Linux-3-0-1279228.html>) Daten und Zahlen zu den jüngsten Versionen des Linux-Kernels. Heise online, 22. Juli 2011, S. 4, abgerufen am 25. Juli 2011.
134. Dr. Oliver Diedrich: *The Next Generation – Linux 2.6: Fit für die Zukunft*. (<https://www.heise.de/ct/artikel/The-Next-Generation-289090.html>) in der c't 24/2003 (am 17. November 2003), Seite 194
135. Marcel Hilzinger: *Con Kolivas meldet sich mit neuem Scheduler zurück*. (<http://www.linux-magazin.de/NEWS/Con-Kolivas-meldet-sich-mit-neuem-Scheduler-zurueck>) *Linux-Magazin*, 2. September 2009, abgerufen am 27. April 2014.
136. Richard M. Stallman: *Linux, GNU, and freedom*. (<https://www.gnu.org/philosophy/linux-gnu-freedom.html>) gnu.org, 10. Juli 2014, abgerufen am 12. Februar 2015 (englisch).
137. Linus Torvalds: *GPL V3 and Linux – Dead Copyright Holders*. (<http://article.gmane.org/gmane.linux.kernel/371935>) 25. Januar 2006, abgerufen am 24. März 2009 (englisch, Nachricht auf der Linux-Kernel-Mailingliste).

Abgerufen von „[https://de.wikipedia.org/w/index.php?title=Linux_\(Kernel\)&oldid=223567921](https://de.wikipedia.org/w/index.php?title=Linux_(Kernel)&oldid=223567921)“

Diese Seite wurde zuletzt am 9. Juni 2022 um 15:22 Uhr bearbeitet.

Der Text ist unter der Lizenz „Creative Commons Attribution/Share Alike“ verfügbar; Informationen zu den Urhebern und zum Lizenzstatus eingebundener Mediendateien (etwa Bilder oder Videos) können im Regelfall durch Anklicken dieser abgerufen werden. Möglicherweise unterliegen die Inhalte jeweils zusätzlichen Bedingungen. Durch die Nutzung dieser Website erklären Sie sich mit den Nutzungsbedingungen und der Datenschutzrichtlinie einverstanden.
Wikipedia® ist eine eingetragene Marke der Wikimedia Foundation Inc.