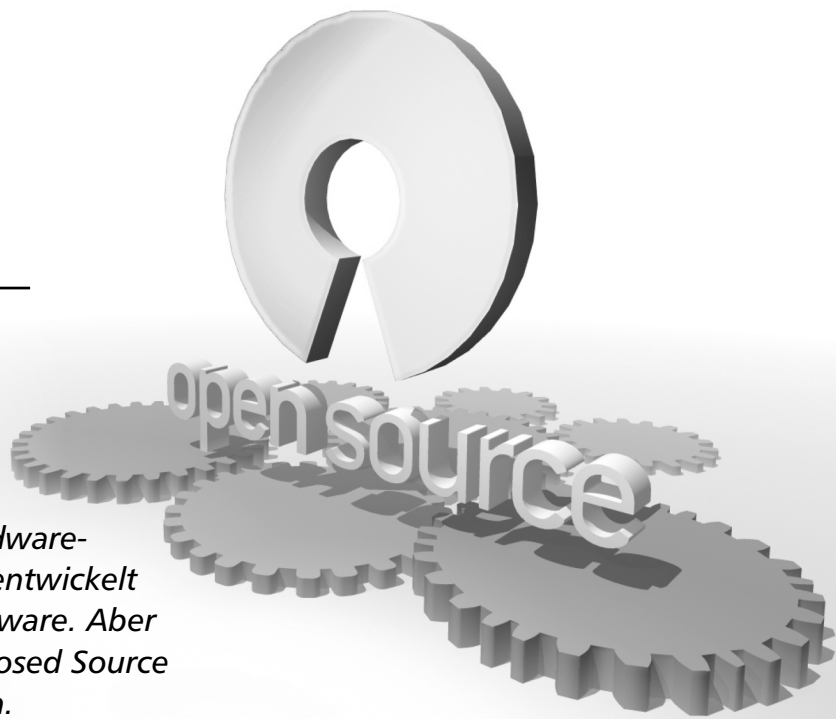


# No Blob!

JÜRGEN DANKOWEIT

*Open-Source-Betriebssysteme sind nicht gerade reich mit Gerätetreibern ausgestattet. Der eine oder andere Hardware-Hersteller erbarmt sich dann doch und entwickelt die für ein Betriebssystem wichtige Software. Aber leider liegt sie anschließend meist als Closed Source vor. Die Konsequenzen sind unerfreulich.*



Um es gleich vorweg zu nehmen: Es geht hier nicht um Sahnspinat oder Applikationen, die als binary-only angeboten werden. Dieses ernste Thema betrifft hauptsächlich Open-Source-Betriebssysteme: Gerätetreiber! Um das Verständnis für die Problematik zu schärfen, werden zuvor einige Begriffe erläutert. Den meisten Systemverwaltern sind Gerätetreiber ein Begriff und bedürfen daher keiner Erläuterung, ganz im Gegensatz zu den vielen normalen Benutzern und IT-Entscheidern, die für dieses Thematik auch sensibilisiert werden sollen.

Ein Gerätetreiber (Device Driver) ist ein Programm, das ein Gerät (Device) für den Betriebssystemkern – auch Kernel genannt – verfügbar macht. Der Kernel kommuniziert mit Hilfe dieses Treibers mit der Hardware (Bild 1).

## Closed-Source versus Open-Source versus NDA

Der Treiber wandelt Steuerbefehle in für das Gerät verständliche Befehle um. In der anderen Richtung werden Statusmeldungen und Fehlermeldungen für den Betriebssystemkern aufbereitet. Daneben transpor-

tiert der Treiber die Daten. Die Programmierer von Gerätetreibern tragen somit eine sehr große Verantwortung bei der Entwicklung dieser Art von Software. Ein kleiner Fehler hat fatale Auswirkungen wie beispielsweise Datenverlust.

Im Zusammenhang mit Software ist auch häufig von einem »Blob« die Rede. Der Begriff stammt aus dem

Datenbankbereich und bedeutet »Binary Large Object« (deutsch: großes binäres Objekt). Es handelt sich also um ein Stück Software, das in (ausschließlich) binärer Form vorliegt oder veröffentlicht wird. Im Umfeld der Systemsicherheit taucht oft der Begriff »Backdoor« (deutsch: Hintertür) auf. Eine Backdoor ist ein bewußt oder unbewußt (durch Feh-

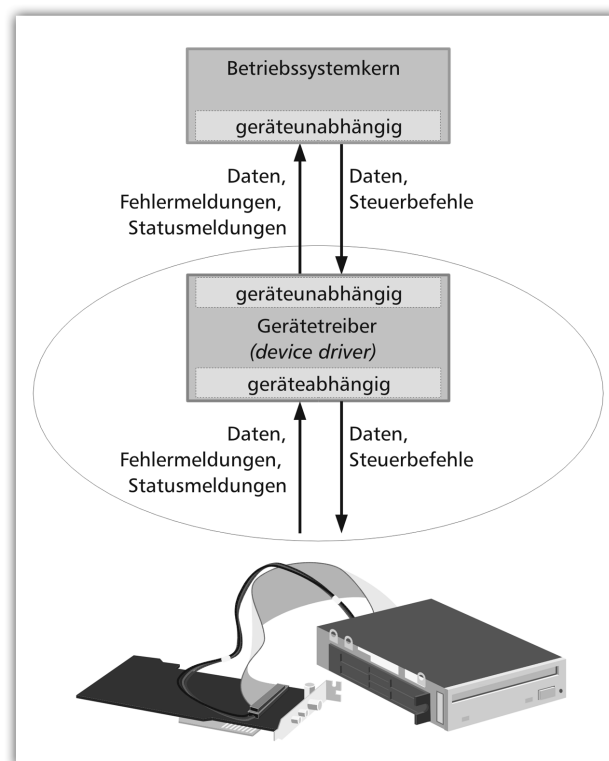


Bild 1:  
Kommunikation  
zwischen Kernel  
und Gerätetreiber

ler) programmierter Einsprungspunkt. Meist werden sie dazu benutzt, um die Kommunikation von Treiber und Daemon-Prozessen zu gewährleisten oder man verwendet sie für Debug-mechanismen.

Der Leser wird sich nun fragen, was der Unterschied zwischen Closed-Source- und Open-Source-Treibern ist. Closed-Source-Treiber erhält der User immer nur in binärer Form. Der Vorteil ist, daß der Treiber nicht erst aufwendig kompiliert werden muß. Es reicht das Einlegen der Diskette oder CD aus und schon beginnt der Installationsprozeß. Bei Open-Source-Treibern ist das meist anders. Hier ist ein Compilerlauf notwendig, manchmal muß auch der gesamte Kernel kompiliert werden oder schlimmstenfalls ist auch ein Neubau des gesamten Betriebssystems angesagt. Das schreckt viele Anwender ab.

Closed-Source-Treiber haben aber auch entscheidende Nachteile. Auf zwei dieser Nachteile möchte der Autor näher eingehen: Sicherheit und Backdoors.

### Sicher! Sicher?

In einer Zeit, in der Sicherheit vor Angriffen aus dem Internet eine immer wichtigere Rolle spielt, sollte sich jeder User oder Administrator sich seine Gedanken gemacht haben, wie »sicher« sein bevorzugtes Betriebssystem sein sollte oder könnte. Dabei spielen zwei Aspekte eine Rolle: die Sicherheit nach außen und die innere Sicherheit.

Nach außen hin werden die Betriebssysteme mit Paketfiltern oder ähnlichem vor unerwünschten Zugriffen hinreichend geschützt. Die innere Sicherheit ist meist durch gute Virens Scanner abgedeckt. Was ist aber mit Software, die durch einen Fehler den Datenbestand eines Servers gefährdet? Was ist, wenn ein Treiber für einen SCSI-Hostadapter oder SATA-Controller bei der Übertragung eines bestimmten Datenvolumens einen Fehler produziert? Manche Administratoren oder User werden jetzt denken, dann rufe ich meinen Händ-

### Hersteller-Unterstützung

Es gibt glücklicherweise Hardware-Hersteller, die von Open-Source überzeugt sind und die Open-Source-Gemeinde mit Dokumentation unterstützen. Die folgende Tabelle zeigt eine kleine Übersicht, die Ansporn für alle anderen Produzenten sein soll, ihre Verweigerungshaltung zu überdenken. Die Konkurrenz ist schon bei Open Source!

Hardware-Hersteller	freundlich	Open-Source-zurückhaltend	ablehnend
3COM		X	
AMD		X	
APC	X		
Adaptec			X
Apple			X
Atheros			X
Atmel	X		
Dell			X
Epson	X		
Exabyte	X		
HP	X		
Intel		X	
JMicron	X		
Linksys		X	
LSI	X		
Logitech			X
M-Audio		X	
Marvell			X
Matrox			X
Microsoft			X
Netcomm			X
Novatel			X
NVIDIA			X
Ralink	X		
Realtek	X		
Ricoh			X
Seagate		X	
SiS		X	
Soekris	X		
SUN		X	
TI			X

Diese Übersicht entstand durch Recherchen in Mailinglisten und erhebt damit keinerlei Anspruch auf Vollständigkeit.

ler an oder schaue auf der Internetseite des Hardware-Herstellers nach Neuerungen zur Software. Es gibt Hersteller, die nur in bestimmten Zyklen Fehlerkorrekturen zu ihrer Software liefern (prominentes Beispiel: Microsofts Patchday). Schwierigkeiten treten dann auf, wenn an so einem Patchday die Fehler im Treiber nicht behoben werden.

Mangels Quelltext hat ein des Programmierens kundiger Administrator keinerlei Möglichkeit, Analysen anzustellen, wo sich der Fehler befindet und ihn eventuell zu beheben. So blieben ihm zwei Möglichkeiten: beten und warten, bis der Hersteller einen Bugfix liefert. Bei älteren Geräten besteht dann durchaus der Zwang, neue Hardware anzuschaffen, was auch mit nicht unerheblichen Primär- und Sekundärkosten verbunden ist. Unter Primärkosten fallen die Anschaffungskosten des neuen Geräts, die Sekundärkosten entstehen durch ausgiebige Testphasen, Systemausfälle während der Umbauphase und Personalkosten.

### Securitas per definitionem?

Systemverwalter und User müssen sich vor Augen halten, welches Gefahrenpotential in Treiber-Backdoors stecken kann. Angenommen, ein Daemon-Prozeß nutzt einen Treiber-Backdoor, um den Netzwerkverkehr zu überwachen. Durch einen kleinen Fehler im Treiber kommt es dazu, daß der Daemon-Prozeß die Daten manipuliert oder falsch interpretiert. Dadurch können nicht unerhebliche Schäden – auch finanzieller Art! – auftreten.

Leider denken viele Hardware-Hersteller genau in diese Richtung: Eine Software ist nur dann sicher, wenn der Quelltext im Verborgenen bleibt. Dabei verfährt man nach dem Motto »was ich nicht weiß, macht mich nicht heiß«, und das im doppelten Wortsinn.

Zum einen gehen die Hardware-Hersteller davon aus, daß durch Offenlegung der Quelltexte zu den



Gerätetreibern bestimmte interne Logikstrukturen der Geräte bekannt werden, die der Mitbewerber nicht wissen darf. Man kann also froh sein, daß CPU-Hersteller den Befehlssatz und die interne Logik ihrer CPUs nicht geheimhalten. Das trifft auch auf die Entwickler des PCI-Busses zu. Wären die Spezifikationen (Bus-Timings, Trigger-Phasen oder Arbitrierungslogik) Closed Source, dann gäbe es wahrscheinlich keine PCI-Steckkarten. Man denke dabei an IBMs Desaster mit dem Microchannel-Bus! Andererseits wird argumentiert, sobald die Spezifikation eines Geräts offengelegt ist, würden durch falsche Programmierung der Register eventuell bestimmte Richtlinien nicht mehr eingehalten.

Wireless-LAN-Netzwerkkarten sind hierfür ein gutes Beispiel. Hier sträuben sich die meisten Hardware-Hersteller, Dokumentationen freizugeben. Es wird hauptsächlich damit begründet, daß durch falsche Programmierung der PLL-Bausteine die Sendefrequenz in einen unzulässi-

gen Bereich verschoben wird. Das mag stimmen, aber wer garantiert, daß der Entwickler des Closed-Source-Treibers den Steuerbaustein korrekt programmiert hat?

Es drängen sich Fragen auf. Wieso bauen die Hardware-Hersteller nicht eine Programmierschnittstelle (API), an die ein Open-Source-Treiber sauber andocken kann? Wieso muß die CPU immer mehr Aufgaben übernehmen, die prinzipiell von der Peripherie erledigt werden sollten (GDI-Drucker, Win-Modems)? Wieso wird die Firmware eines Peripherie-Geräts im Arbeitsspeicher des PCs gehalten und nicht in einem Flash-ROM des Geräts (manche WLAN-Karten)? Im Open-Source-Bereich haben viele Programmierer die Möglichkeit, das korrekte Funktionieren der Software zu prüfen. Es finden sich fast immer Leute, die auftretende Fehler binnen kürzester Zeit beheben. Gleiches gilt für die Optimierung von Treibern. Das Mehraugenprinzip ist dabei nicht zu unterschätzen.

Die Beweggründe für die Verweigerung der Hardware-Hersteller sind vielfältig. Meist ist es die Angst, die Konkurrenz würde Einblick in bestimmte Algorithmen erhalten. Dieses Argument ist verständlich und auch nicht von der Hand zu weisen. Es läßt sich doch dadurch lösen, daß die Firmware als separate Datei vorliegt, die vom Treiber in das Peripherie-Gerät geladen werden muß. Damit der Treiber mit der Firmware kommunizieren kann, ist das API genauestens zu dokumentieren. Die Hersteller geben auf diese Weise keine für sie wichtige Algorithmen preis und die Entwicklergemeinschaft hat die Chance, Treiber

#### allBSD – eine neues Betriebssystem?

allBSD ist kein neues Betriebssystem oder Distribution von irgendwas. Es ist ein Zusammenschluß von Administratoren, IT-Consultants und Users, die es sich zur Aufgabe gemacht haben, über die BSD-Betriebssysteme nicht nur normale Benutzer, sondern und vor allem auch IT-Entscheidungsträger in kleinen und mittelständischen Firmen (KMU) zu informieren. Um ein möglichst breites Publikum zur erreichen, ist die Vereinigung auf Messen und anderen IT-Veranstaltungen vertreten. Neben persönlichen Gesprächen, Beratung und Installationshilfen werden auch Informationsschriften in Form von Flyern angeboten.



Bild 2:  
Das Plakat  
zur Kampagne

zu entwickeln. Damit es zu keinen Mißverständnissen kommt: Die Firmware darf Closed Source sein, dagegen ist nichts einzuwenden!

### Kampagne von allBSD

In der letzten Zeit wurde seitens einiger weniger Treiberentwickler – namentlich Greg Kroah-Hartmann – aus dem Linux-Umfeld vorgeschlagen, daß Treiberentwickler ein sogenanntes *Non Disclosure Agreement* (NDA) mit den Hardware-Herstellern eingehen sollten. Das würde gewährleisten, daß immer die neueste Dokumentation zur Entwicklung von Treibern bereitstehen würde. Ist das die Lösung des Problems? Ganz sicher nicht! Die Open-Source-Gemeinde wäre damit wieder den Herstellern ausgeliefert. Jeder Hardware-Produzent kann in sein NDA hinschreiben, was er möchte. Die Wahrscheinlichkeit ist hoch, daß durch dieses Hintertürchen Closed-Source-Treiber Einzug in ein Open-Source-

Betriebssystem halten. Ist das der Geist von Open Source? Es bleibt weiterhin die Frage zu klären, ob ein NDA an eine Person oder an einem Entwicklerteam gebunden ist. Falls es nur an einer Person gebunden sein sollte, müssen sich die Befürworter von NDA-Abkommen für den Zeitpunkt Gedanken machen, wenn die Vertragsperson aussteigt! Außerdem darf dabei nicht vergessen werden, daß nur die Vertragsperson Einblick in die Dokumentation haben darf. Wie sollen nach dem Mehraugenprinzip andere Entwickler Fehler in der Software beheben dürfen? Wie werden unterschiedliche Lizenzmodelle (GPL versus BSD)

berücksichtigt? all-BSD hat sich entschlossen, mit einer Kampagne im europäischen Raum auf die Situation eindringlich hinzuweisen, daß viele Hardware-Hersteller nicht gewillt sind, Informationen zu ihren Geräten für Open-Source-Entwickler zugänglich zu machen.

Zu diesem Zweck wurde ein Plakat entworfen, das in markigen Worten alle Beteiligten zum Nachdenken anregen soll (Bild 2).

### Es geht um Ihre Sicherheit

Es zeigt das für alle BSD-Betriebssysteme gemeinsame Maskottchen Beastie mit einem großen Hammer, das symbolisch eine Diskette mit Closed-Source-Treibern einstampft. Damit möchte allBSD zeigen, daß Closed-Source-Treiber in Open-Source-Betriebssystemen nicht zu suchen haben. Neben diesem Plakat informiert zusätzlich ein Flyer über das Anliegen von allBSD und seiner Kampagne.

Um es deutlich zu machen: Diese Kampagne möchte nicht diffamieren oder marktschreierisch auftreten, sondern sie soll bei allen Beteiligten durch Information zu einem Umdenken animieren. Daher folgender Aufruf an die Hardware-Hersteller: Arbeiten Sie mit Open-Source-Entwicklern zusammen. Sie sollten immer den Gedanken im Hinterkopf behalten, daß die Konkurrenz schon Open-Source-Treiber anbietet und dadurch mehr Umsatz und Gewinn macht. Außerdem hat sie deswegen ihren Marktanteil gesteigert! Eine gesunde Symbiose zwischen Open-Source-Entwicklern und Hardware-Herstellern ist von gegenseitigem Nutzen. ◆

#### Links:

No-Blob!-Kampagne: [www.allbsd.de](http://www.allbsd.de), [www.bsdgroup.de](http://www.bsdgroup.de)

Hintergrundinformation: [www.openbsd.org/papers/opencon06-drivers/index.html](http://www.openbsd.org/papers/opencon06-drivers/index.html), [www.vendorwatch.org](http://www.vendorwatch.org)

NDA bei Linux: [www.heise.de/newsticker/meldung/85495](http://www.heise.de/newsticker/meldung/85495).

Allgemeine Informationen zu den BSD-Systemen: [www.freebsd.org](http://www.freebsd.org), [www.netbsd.org](http://www.netbsd.org), [www.dragonflybsd.org](http://www.dragonflybsd.org).