



by Georges Tarbouriech
<georges.t(at)linuxfocus.org>



FreeBSD, eine echte Alternative

Abstract:

About the author:

Georges ist ein langjähriger Unix-Benutzer. Er ist vernarrt in die freien Versionen dieses großartigen Systems.

Ja, Sie lesen LinuxFocus! Es existieren jedoch andere freie Unix-Versionen und diese sind auch von großem Interesse und voll großartiger Fähigkeiten. Dieser Artikel ist ein Überblick über ein anderes freies Unix: FreeBSD.

Warum FreeBSD ?

Die meisten Leute "denken Linux", wenn über freie Unix-Varianten gesprochen wird. Sicher ist Linux sehr verbreitet und jeder (naja, fast jeder) hat darüber schon gehört. Die Anzahl neuer Anwender wächst weiter und die Medien (entweder "spezialisiert" oder nicht) machen viel "Lärm" darum. Einer der Hauptgründe für diesen Erfolg rührt daher, dass Linux vor allem auf Intel-Maschinen läuft, welches die am weitesten verbreitete Computer-Architektur ist. Natürlich kann es auf vielen anderen Prozessoren laufen, da eine der Ideen hinter Unix war "wir kümmern uns nicht um Prozessoren". Nichtsdestoweniger wäre Linux sicher nicht so ein Erfolg geworden, wenn es für einen mehr "geheimen" Prozessor entwickelt worden wäre, oder es hätte zumindest viel länger gedauert.

Aber freies Unix ist nicht auf Linux beschränkt: es gibt auch die BSD-"Familie". Es gibt drei Hauptrichtungen: FreeBSD, NetBSD und OpenBSD. BSD steht für Berkeley Software Distribution. Wenn Sie mehr Informationen über die Unix-Geschichte wissen möchten, können Sie sich [Artikel 176](#) ansehen. (Lassen Sie sich gesagt sein, das dieser Artikel nicht über die Unterschiede zwischen BSD und System V spricht: dafür wäre ein Buch nötig.)

Jede Richtung hat etwas Besonderes:

1. NetBSD funktioniert auf jeder Plattform auf Erden (fast jeder), einschließlich sehr alter Computer: es dürften so 30 bis 40 verschiedene Plattformen sein. NetBSD ist der Beweis dafür, was schon angesprochen wurde: Unix kümmert sich nicht um Prozessoren. Es ist die richtige Wahl, wenn Sie ein nettes altes "Stück Müll" zu Hause oder in der Firma haben (aber natürlich nicht nur). Ich habe es persönlich noch vor einigen Jahren auf Amiga- oder Intelmaschinen benutzt und war sehr zufrieden damit.
2. OpenBSD ist als eine der sichersten Unix-Varianten bekannt. Es ist ideal für Server geeignet und es läuft natürlich auch auf verschiedenen Prozessoren. Es ist wahrscheinlich eines der am häufigsten genutzten System für Internet-Server. Man findet es sicherlich nicht auf Privatrechnern, aber jede/r kann tun, was er/sie möchte ...
3. FreeBSD ist eine "Mischung" der beiden vorher genannten. Es ist speziell auf Intel- und Alpha-Prozessoren ausgerichtet und hat einen Teil der Sicherheits-Philosophie von OpenBSD

"geerbt." Wie bei Linux macht es die Tatsache, dass es auf Intel-Prozessoren läuft, zu einem der weiter verbreiteten BSDs. Es ist eine gute Wahl für die Benutzung zu Hause. Das heißt aber nicht, dass man es nicht in Firmen einsetzen kann!

Es gibt noch etwas, was FreeBSD angeht: Diese Personen "teilen" ihre Arbeit mit Apple für die freie Software, die Teil von Mac OS X ist. Und da ich sehr vernarrt bin in OS X ... Bei dieser Gelegenheit, lassen Sie sich nicht verwirren: Obwohl OS X auf FreeBSD basiert, hat der Kernel nichts mit dem FreeBSD-Kernel zu tun. Es ist ein Mach-Kernel, der bereits bei NeXTstep, dem "Vater" von OS X, genutzt wurde: da NeXTstep bereits am Ende der 80er Jahre erschien, ist es offensichtlich, dass der Kernel sich etwas weiter entwickelt hat.

Lassen Sie uns nun etwas tiefer einsteigen.

FreeBSD finden und installieren

Wie andere freie Software kann FreeBSD von der FreeBSD-Webseite abgerufen werden. Dies erfordert allerdings eine schnelle Verbindung. Keine Sorge: Sie können überall auf der Welt CDs erstellen. Die aktuelle Version (zum Zeitpunkt dieses Artikels) ist 4.6. Der Artikel befaßt sich mit der i386-Version. Die FreeBSD-Installation von CD-ROM (oder anderen Medien) ist einfach ... auch wenn es keine grafische Oberfläche dafür gibt! Stattdessen benutzen Sie ncurses. Folgen Sie einfach den Anweisungen und FreeBSD landet auf Ihrer Festplatte. Wie üblich müssen Sie Ihre Festplatte partitionieren und einige wenige Sachen wie Netzwerk oder sogar (optional) Ihren Kernel konfigurieren. Als nächstes treffen Sie auf `sysinstall`, ein sehr gutes Werkzeug, das – wie der Name sagt – viele der Installationsarbeiten übernimmt. Ob Sie den ganzen Haufen an Paketen installieren oder nur einige davon, ist Ihrer Wahl überlassen. Sie können sie jederzeit hinzufügen.

Die Paketverwaltung ist auch offensichtlich. Sie können wie unter Solaris die `pkg`-Befehle nutzen, oder Sie bevorzugen die `ports`. Kurz gesagt, entsprechen die `pkg`-Befehle den RPMs oder DEBs für Linux, während es sich bei den `ports` um Archive handelt, die mit `make` und `make install` kompiliert werden. In anderen Worten: nichts komplexes oder schwieriges.

Da wir mit Linux vergleichen, sollten wir hinzufügen, dass es ein Linux-Kompatibilitäts-Paket gibt, das Sie installieren sollten: dies gestattet es Ihnen, Linux-Anwendungen zu kompilieren und zu nutzen.

Erneut, alles ist sehr offensichtlich und – nur für den Fall – es wird ein sehr gutes Handbuch mitgeliefert, das jeden Schritt erläutert. Dieses existiert in verschiedenen Sprachen.

Entsprechend brauchen wir nicht allzu viel Zeit mit diesem Teil verbringen.

FreeBSD konfigurieren

Erneut werden wir nicht alles erwähnen können. Das Wichtigste ist zu wissen, wo sich die wesentlichen Dateien befinden. Wie üblich finden Sie diese im `/etc`-Verzeichnis. Sie werden `rc.irdendetwas` genannt und können viele verschiedene Teile verwalten: allgemeine Konfiguration, Netzwerk, `sysctl` usw.

Die Datei `rc.conf` ist sehr wichtig, da – wie es der Name sagt – hier die Konfiguration angepasst werden kann. In dieser Datei können Sie dem System mitteilen, ob einige Dämonen (`sshd`, `sendmail`, etc) gestartet werden sollen oder nicht, die Art Ihrer Firewall, ob Sie Kernel-Sicherheitsebenen aktivieren wollen oder nicht, ob Sie IPv6 nutzen wollen oder nicht, usw. Es gibt eine Menge an Optionen und Sie können alle in der `Default`-Datei sehen, die überraschenderweise `/etc/defaults/rc.conf` heißt. Lassen Sie sich nicht verwirren: diese Datei enthält die Standard-Einstellungen, keine Beispiele. D. h., die `/etc/rc.conf`-Datei überschreibt diese Standard-Einstellungen. In anderen Worten, kopieren Sie nicht die `/etc/defaults/rc.conf` nach `/etc/rc.conf` und versuchen Sie nicht, sie zu ändern. Die Änderungen müssen in `/etc/rc.conf` vorgenommen werden.

Zu Ihrer Information, die von `sysinstall` vorgenommenen Änderungen werden automatisch in `/etc/rc.conf` eingefügt (Netzwerkkarten-Konfiguration, Hostname, Sicherheitsebene usw.).

Ein sehr wichtiger Hinweis: FreeBSD kommt mit keinen aktivierten Diensten in `/etc/inetd.conf`. Das heißt, dass beim ersten Start die von `inetd` verwalteten Dämonen standardmäßig abgeschaltet sind. Meiner bescheidenen Meinung nach sollte das für jedes Unix-System verpflichtend sein. Auf jeden Fall ein guter Punkt!

Die anderen `rc`-Dateien erlauben Ihnen entweder die Konfiguration der Firewall oder des `sysctl`-Teils und vieler anderer Bereiche. Dies bringt uns natürlich zur Feineinstellung von FreeBSD.

FreeBSD tunen

Eines der besten Programme für die Feineinstellung nennt sich `sysctl` (auch unter Linux!). Sie können es von der Befehlszeile nutzen, um einen speziellen Wert zu definieren oder eine `sysctl.conf`-Datei erstellen (empfohlen, es sei denn, Sie benutzen die Befehlszeile für Testzwecke).

Wenn Ihre Maschine z. B. als Gateway genutzt wird, können Sie mittels `sysctl` die IP-Weiterleitung erlauben: `sysctl -w net.inet.ip.forwarding=1`. Zur erhöhten Sicherheit können Sie überprüfen, ob die Pakete, die über eine Schnittstelle hereinkommen, auch die entsprechende IP-Adresse enthalten mittels des Befehls: `sysctl -w net.inet.ip.check_interface=1`. Sie können fast alle Aspekte Ihres Systems mit `sysctl` steuern; lesen Sie die Handbuchseite. Wenn Sie diese Befehle in eine `sysctl.conf`-Datei schreiben wollen, fügen Sie einfach die Argumente ohne den `sysctl`-Befehl selbst ein. Die Datei benutzt das *Variable = Wert*-Format. Die obigen Beispiele werden zu:

```
net.inet.ip.forwarding=1
net.inet.ip.check_interface=1
```

Natürlich wird Ihre eigene `sysctl.conf`-Datei mehr als zwei Zeilen enthalten, da sie alles auf Kernebene steuern kann, naja, fast alles!

Am Anfang dieses Artikels sprachen wir über Sicherheitsebenen. Es gibt vier verschiedene Ebenen, von `-1` bis `3`. `-1` repräsentiert die Unsicherheits-Ebene und `3` repräsentiert die höchste Sicherheitsebene. Es ist sehr wichtig zu verstehen, was jede Ebene bewirkt, bevor eine ausgewählt wird. Die Benutzung anderer Ebenen als `-1` oder `0` kann zu einem System führen, das nicht so arbeitet wie Sie es erwarten. Der erste Rückschlag kommt mit Ebene `1`, weil es verhindern kann, dass Sie den `X`-Server richtig nutzen können. Außerdem können Sie z. B. keine Kernel-Module laden oder entladen. Sie sind gewarnt!

Diese hohen Sicherheitsebenen können jedoch für spezielle Server erforderlich sein, die hohe Sicherheit benötigen. Als Privatanwender sollten Sie die Ebene `0` nutzen, aber es liegt an Ihnen ...

Der `chflags`-Befehl bezieht sich auf die Sicherheitsebenen. Untersuchen Sie es, lesen Sie die Handbuchseite. Es ist ein sehr mächtiger Befehl.

Hier weiterzumachen, würde diesen Artikel zu einem über "Sicherung von FreeBSD" machen, deswegen gehen wir zum nächsten Kapitel.

FreeBSD verbessern

Lassen Sie uns annehmen, dass Sie gerade einen CD-Satz mit FreeBSD 4.5 erworben haben. Einige Tage später wird FreeBSD 4.6 veröffentlicht. Welch ein Pech!

Machen Sie sich keine Sorgen: FreeBSD enthält ein Aktualisierungssystem per CVS. Sie können entweder anonymes CVS oder CVSup nutzen. Ersteres ist leichter zu nutzen, während das zweite effizienter ist. Mit diesen Werkzeugen besteht die Aufgabe nur darin, den neuen Quellcode-Baum abzurufen und zu synchronisieren.

Als nächstes müssen Sie das ganze neue Zeug mittels `make buildworld` kompilieren. Wie Sie dies machen, ist im zur Verfügung gestellten Handbuch sehr gut erläutert. Der wichtige Punkt ist, dass dieses Verfahren Ihnen erlaubt, ein neues System immer aktuell zu halten.

Dies ist jedoch noch nicht das Ende. Wie immer, wenn es um freie Software geht, werden Verwundbarkeiten gefunden und sehr schnell gelöst. Im Gegensatz zu vielen proprietären Systemen, wo eine Verwundbarkeit für immer (und 1 Tag) ungelöst bleiben kann, reagieren Leute, die sich mit freier Software beschäftigen, mit Lichtgeschwindigkeit und versorgen Sie mit Fehlerkorrekturen. Natürlich ist es Ihre Aufgabe zu prüfen, was so läuft. Die FreeBSD-Webseite verfügt über einen Sicherheitsabschnitt, der nach Release geordnete Hinweise enthält. Sie können dort .asc-Dateien (Textdateien) abrufen, die Ihnen mitteilen, was Sie wissen müssen: eine Beschreibung des Problems und dessen Lösung. Auch andere Webseiten bieten Ihnen diese Informationen. Lassen Sie uns z. B. [linuxsecurity](#), [CERT](#), usw. erwähnen. Das bedeutet natürlich, dass Sie Ihr System aktualisieren müssen.

FreeBSD aktualisieren

Sie möchten doch keine bekannten Verwundbarkeiten in Ihrem System behalten, oder? FreeBSD bietet Ihnen – wie die anderen Unix-Varianten – Patches. Besorgen Sie sich diese Patches und wenden Sie diese an. Für Linux-Benutzer besteht der Hauptunterschied daran, dass sie kein korrigiertes Paket erhalten, sondern stattdessen einen Quelltext-Patch. D. h., dass Sie den Kernel nach dem Patchen des Quellcodes neu kompilieren müssen, was natürlich voraussetzt, dass der Quellcode bereits auf Ihrem Rechner installiert ist. Dies gilt auch für Linux, aber die Personen rufen eher z. B. ein modifiziertes rpm-Paket zumindest für die korrigierte Anwendung ab, aber auch für korrigierte Kernel.

Einer der wichtigsten Unterschiede zwischen Linux und FreeBSD findet sich im Kernelbereich. FreeBSD basiert auf BSD 4.4 und hat keine verschiedenen Kernel-Versionen wie die Linuxwelt. Das bedeutet, dass der Kernel stabil ist, aber im Laufe der Zeit auch immer größer wird, da er eher monolithisch ist, obwohl er auch von Modulen profitieren kann. Der Nachteil ist, dass Sie Ihren Kernel öfter als unter Linux kompilieren müssen, wenn Sie Ihr System so sicher wie möglich halten wollen.

Ich bevorzuge jedenfalls diese Philosophie gegenüber der von Linux, aber das ist nur meine Meinung. Ich glaube nicht, dass z. B. 3 verschiedene Kernel "in Arbeit" zu besonderer Stabilität führen. Ich möchte auch nur die Anwendungen erwähnen, die von Version zu Version portiert werden müssen, besonders diejenigen, die ihren Schwerpunkt auf Sicherheit legen. Ein weiteres Beispiel könnte das Paketfilter-System sein: 3 Kernel-Versionen, 3 verschiedene Paketfilter-Softwareversionen! Und so weiter ...

Ich habe viel Respekt für die Leute, die diesen großartigen Job machen, aber ist das der richtige Weg. Verbesserung ist immer ein Bedürfnis, aber ist es kohärent, immer alles zur gleichen Zeit zu ändern? Macht nichts!

Natürlich können Sie auch einen neuen Kernel kompilieren, um ihn an Ihre Bedürfnisse anzupassen ... und dann ist es wie unter Linux. Der Weg dazu ist sehr detailliert im Handbuch beschrieben, damit können wir uns etwas anderem zuwenden.

Anwendungen

Wie bereits gesagt, ein großer Teil freier Software-Anwendungen (wenn nicht alle) arbeiten auch unter FreeBSD. Sie brauchen nicht die oben erwähnten ports oder packages. Sie können die meisten der verfügbaren Archive kompilieren, einerlei um welchen Softwaretyp es sich handelt.

Für die Oberflächen-Abhängigen sind Gnome und KDE verfügbar ... und in der Distribution enthalten. Das gleiche gilt für Window-Manager. Das großartige GNUstep-Rahmenwerk läuft auch unter FreeBSD sehr gut. Zu dessen Kompilierung benötigen Sie eine neuere gcc-Version als die in der Distribution enthaltene: kein Problem! Holen Sie sich das entsprechende Archiv und kompilieren Sie den neuen Compiler. Natürlich arbeiten die GNUstep-Anwendungen wie GNUMail.app, Gorm.app oder ProjectBuider.app auch gleichermaßen gut.

Wirklich, all die Software, die Sie unter Linux benutzen, ist auch für FreeBSD verfügbar ... und es gibt viele andere!

Z. B. kommt FreeBSD mit einer Menge an Programmen, die die Sicherheit betreffen.

Überwachungsprogramme, Verwaltungsprogramme usw. sind Teil der Distribution.

Firewalls, Proxies, Port-Scanner, IDS usw. sind auch verfügbar. So können Sie z. B. zwischen IPsec oder ipfw auswählen. Sie können nessus, nmap, portsentry usw. benutzen. Erneut, die meisten von ihnen sind Teil der Distribution und nichts hält Sie davon ab, die aktuellsten Versionen abzurufen und zu kompilieren.

Wir haben bereits über Arbeitsumgebungen und Window-Manager gesprochen, aber wir haben noch kein Wort darüber verloren, was zu deren Benutzung erforderlich ist: das X Window System. XFree 4.* ist nun der Standard. Nur zur Information: bis zu FreeBSD 4.5 war XFree 3.* der Standard. D. h., XFree 4.* war in der Distribution enthalten, aber der Installationsprozeß wählte XFree 3.* aus. Nun ist XFree 4.* der Standard, aber Sie können immer noch Version 3.* benutzen, wenn Sie dies vorziehen.

Hardware

Da wir über die i386-Version sprechen, sollten wir sagen, dass fast alle verfügbare Hardware unter FreeBSD arbeitet, selbst wenn sie etwas älter ist. Z. B. wird eine alte 3Com Combo-Netzwerkkarte ohne Probleme arbeiten, sobald Sie den Medientyp im ifconfig-Befehl definieren. Wenn Sie eine BNC-Verbindung benutzen, reicht es, *media 10base2/BNC* am Ende der ifconfig-Zeile in */etc/rc.conf* hinzuzufügen.

Das gleiche gilt für einige alte Grafikkarten: Die Benutzung von XFree 4.* wird die meisten der Probleme lösen. Dies ist besonders wahr für ziemlich alte ATI-Karten.

Jedenfalls sollten Sie die Hardware-Hinweise auf der FreeBSD-Webseite konsultieren, um sicher zu sein.

Sicherheit

Wir erwähnten bereits einige Programme wie Firewalls, Port-Scanner usw. FreeBSD kommt mit vielen weiteren Sicherheitsprogrammen. OpenSSH, OpenSSL, Kerberos, MD5 usw. sind Teil der Distribution. Wie üblich, kann man ohne diese Werkzeuge nicht leben und natürlich müssen Sie auf eventuelle Verwundbarkeiten achten.

Ein aktuelles Beispiel zeigt die Wichtigkeit, sehr vorsichtig zu sein. Nehmen wir an, Ihnen wurde mitgeteilt, dass z. B. in OpenSSH eine Verwundbarkeit entdeckt wurde. Das richtige Verhalten ist, sich auf die Webseite zu bewegen und die korrigierte Version abzurufen.

Ja, aber das ist nicht genug! Bitte überprüfen Sie die Prüfsumme des Paketes.

Das Beispiel von OpenSSH wurde nicht willkürlich gewählt. Vor einigen Wochen ersetzte jemand das Original-Archiv mit einer manipulierten Version. Wie üblich reagierten die Menschen bei OpenSSH recht schnell. Wenn jedoch jemand in dieser kurzen Zeitspanne das Paket abgerufen hat, ohne die Prüfsumme zu verifizieren, dann wird eine "schlechte" Version installiert. Daher der Bedarf, eine Prüfsumme zu nutzen ... und diese zu kontrollieren.

Selbst wenn es außerhalb dieses Artikels liegt, sollten wir auf dem Hinweis beharren, dass Sicherheit vor allem eine Verhaltensweise ist.

Kommunikation

Vernetzung unter FreeBSD ist recht gut entwickelt. Die IPv6-Implementation ist sehr gut verwaltet. Die aktuellen Mittel zur Kommunikation werden nicht außer acht gelassen. Wenn Sie ein ADSL- oder

Kabel-Modem nutzen, ist die Konfiguration sehr einfach.

Viele ISPs benutzen heutzutage PPPoE. FreeBSD verwaltet dies auf eine sehr einfache Art: einige wenige Zeilen in der ppp.conf-Datei reichen aus. Hinsichtlich PPPoA (ATM) wird derzeit nur das Alcatel-Modem unterstützt. Dies sollte sich in naher Zukunft ändern (vielleicht gibt es schon andere unterstützte Modems).

Dokumentation

Das mitgelieferte Handbuch ist nicht die einzige verfügbare Dokumentation. Sie finden ferner eine FAQ, Installations-Handbuch usw. auf der FreeBSD-Webseite. Eine weitere wichtige Informationsquelle findet sich unter <http://www.freebsd-howto.com>.

Was kommt als nächstes?

Dies ist eine recht kurze Übersicht zu FreeBSD. Dieses System ist wie die anderen freien Unix-Varianten ein sehr interessantes. Es hat viele große Fähigkeiten und verdient wirklich einen Versuch. Unglücklicherweise kann dieser Artikel keine vollständige Besprechung sein und eine Menge wichtiger Eigenschaften wurde gar nicht erwähnt.

FreeBSD erfordert mehr Unix-Wissen als einige "Windows-artige"-Linux-Distributionen, aber es bleibt doch recht einfach zu nutzen.

Natürlich kann jemand ohne Unix-Wissen in Schwierigkeiten kommen, wenn er nach Aktualisierungen des Systems den Kernel neu kompilieren muss. Dies ist jedoch ein guter Weg des Lernens, selbst wenn es "etwas" Zeit in Anspruch nimmt. Die Menschen scheinen heutzutage zu vergessen, dass Unix von Entwicklern für Entwickler erschaffen wurde, nicht für den Heim- oder Normal-Benutzer.

Zum Glück für diesen Personenkreis haben freie und proprietäre Unix-Versionen diesen Ansatz geändert. Dank an die vielen Entwickler, die dies ermöglicht haben. In den 80er Jahren wären viele der heutigen Linux-Benutzer vor Unix weggelaufen, allein schon wegen des Installationsprozesses. Die Distributionen wurden auf Band ausgeliefert und die Installationsphase konnte einen ganzen Arbeitstag in Anspruch nehmen, der Konfigurationsprozeß mindestens einen weiteren. Dabei reden wir noch gar nicht vom Konfigurieren der Anwendungen! Und heute mit Systemen wie Mac OS X können Sie Unix benutzen, ohne es zu kennen. Dies war schon zu Beginn der 90er Jahre möglich mit dem "Vater" von OS X, NeXTstep.

Dies scheint nicht zum Thema zu gehören, aber nichtsdestoweniger wies NeXTstep den Weg für die freien und proprietären Unix-Varianten ... und einige andere! Linux dem Normalbenutzer nahe zu bringen, war eine Herausforderung, aber es ist geschehen. Glückwunsch an alle die Menschen, die an dieser Herausforderung beteiligt waren, einerlei, ob sie für Geld oder frei gearbeitet haben. Diejenigen, die ohne Entlohnung gearbeitet haben, verdienen eine besondere Erwähnung und viel mehr Dank ;-)

Die Tatsache, dass FreeBSD mit Apple "gemeinsame Sache" macht, sieht für mich sehr gut aus. Jeder profitiert vom anderen in der richtigen Art. Proprietäre Software hat schon immer freie Software genutzt ... ohne das allzusehr "herauszuposaunen". Apple macht das und so auch SGI. Einige andere scheinen das auch zu tun ... jedoch mit Hintergedanken. Einerlei, ich denke weiterhin, dass es "Raum" für alle gibt ... naja, mit Ausnahme des Einen, der bereits den ganzen "Raum" hält und dies hat nichts mit Unix zu tun!

Der "Lärm" um Linux hält die anderen Unix-Versionen im Dunklen: so weit, so gut. Dies kann auch eine gute Seite haben ...und dies ist der Hauptgrund, warum ich FreeBSD nutze. Natürlich behalte ich Linux (und viele andere): Ich liebe Vielfalt!

Wenn Ihnen nun danach ist, FreeBSD einmal zu testen, dann war dieser Artikel nicht umsonst. Testen Sie es selbst und Sie werden eine Menge an Sachen entdecken, die hier nicht erwähnt wurden.

Ich habe es bereits gesagt: Wir leben in einer großartigen Zeit!

<p><u>Webpages maintained by the LinuxFocus Editor team</u> © Georges Tarbouriech "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org</p>	<p>Translation information: en --> -- : Georges Tarbouriech <georges.t(at)linuxfocus.org> en --> de: Hermann-Josef Beckers <beckerst/at/lst-online.de></p>
---	--

2005-01-11, generated by lfparsr_pdf version 2.51