

OpenBSD Workshop



User ralli

17. Februar 2018

Inhaltsverzeichnis

Zusammenfassung	4
1 Einleitung	5
1.1 Vorwort	5
1.2 Über dieses Tutorial	5
2 UNIX Geschichte des Urgesteins	7
2.1 Was ist Unix	7
3 Was ist OpenBSD	10
3.1 Warum OpenBSD	11
3.1.1 OpenBSD ist frei	12
3.1.2 Die Dokumentation	12
3.1.3 Ports	13
3.1.4 Fazit	13
4 Vor der Installation	14
4.1 Mindestanforderungen	14
4.1.1 Plattformen	14
4.2 Beschaffung	15
5 Grundinstallation	16
5.1 Der Installer	16
5.2 Tastaturlayout	16
5.3 Hostname und Netzwerkschnittstelle	17
5.4 Root Passwort setzen	17
5.5 OpenSSH Daemon	17
5.6 X Window System	18
5.7 Benutzer anlegen	18

5.8	Zeitzone	18
5.9	Partitionierung	18
5.10	Basiskomponenten installieren	19
5.11	Abschluß der Installation und Neustart	20
5.12	Nach dem Neustart	21
5.13	Zusammenfassung	21
6	Nach der Grundinstallation	22
6.1	Aktualisierung und Updates	22
6.2	Benutzerverwaltung	22
6.2.1	Neuen Benutzer anlegen	22
6.3	Gruppenzugehörigkeit konfigurieren	23
6.4	Soft Updates	24
6.5	Sudo Ersatz doas einrichten	24
6.6	Paketverwaltung konfigurieren	25
6.7	Desktop XFCE installieren	25
6.8	Minimalsystem mit Windowmanager JWM	26
7	Multimedia	28
8	Internet	29
9	Wissenswertes	30
9.1	 Paketverwaltung	30
9.2	 Hilfe	30
9.3	Aktualität	31
9.4	Fazit	31
10	Spezielles	32
10.1	Der Editor Vi	32
10.1.1	Allgemeines	33
10.1.2	Bearbeitungsmodi	33
10.1.3	Dateiverwaltung	34
10.1.4	Editor	34
10.1.5	Im Text navigieren	35
10.1.6	Suchen und Ersetzen	35
10.1.7	Sonstiges	35

1 Einleitung

1.1 Vorwort

Seit 1974 konnte ich meinen Erfahrungshorizont im IT Bereich ausbauen.

- 1974 - Arbeit als Operator mit Systemfamilie Nixdorf 820 (Lochkarten basiert)
- 1980 - erster eigener Rechner Video Genie EG 3003 (TSR 80 kompatibel) mit Doppelfloppy und Newdos 80 als OS.
- 1983/85 - EDV Kurse BASIC im eigenen Computerraum mit eigenen Rechnern
- 1985/1995 - Programmentwicklung Audits für die Qualitätssicherung einer größeren Automobilfabrik
- 1985/1995 - Freiberuflicher EDV Dozent in der Automobilindustrie (Abendkurse für Mitarbeiter und Auszubildende)
- 1996 - erster Kontakt mit SuSE Linux und Abschied von Windows
- 2005 - Umstieg auf Debian
- 2012 - Umstieg auf FreeBSD
- 2017 - Umstieg auf OpenBSD

In all den Jahren war meine Schwerpunkttätigkeit die Programmierung von Datenbanken sowie deren Verwaltung und Administration.

1.2 Über dieses Tutorial

Dieses Tutorial ist kein Handbuchersatz. Für detailliertere Informationen sind stets die Manpages zu Rate zu ziehen. Es dient als schnelles Nachschlagewerk und ist somit eine Ergänzung für Benutzer, die ohne großen Aufwand ein funktionierendes aktuelles

OpenBSD mit dem XFCE Desktop oder ein Minimalsystem mit dem Windowmanager JWM aufsetzen und konfigurieren wollen.

Dieses Tutorial wäre niemals ohne die Kompetenz vieler Communitymitglieder unseres Forums www.bsdforen.de entstanden. Daher gilt mein Dank all Jenen, die mich mit Sachverstand dazu inspirierten und mich bei meinem Bemühen unterstützten, in OpenBSD selbst sattelfest zu werden.

2 UNIX Geschichte des Urgesteins

2.1 Was ist Unix

Das Betriebssystem Unix blickt auf eine lange Vergangenheit zurück. Ursprünglich wurde es für PDP-Rechner der Firma DEC (Digital Equipment Corporation) entwickelt und sollte hauptsächlich die Bedürfnisse von professionellen Softwareentwicklern befriedigen.

Die digitale Revolution wäre ohne Unix nie denkbar gewesen. Unix war die Ursuppe des Computeruniversums, der Urknall der digitalen Revolution. Mit Unix wurden moderne innovative und fundamentale Technologien wie weiter unten beschrieben eingeführt. Das bewährte sich so gut, das es später von allen anderen und späteren Betriebssystemen übernommen wurde. So inspirierte und beeinflusste Unix alle modernen Betriebssysteme nachhaltig bis zum heutigen Tage und etablierte sich als Meilenstein der Computergeschichte. Hier sehen wir



die Informatiker Legenden Ken Thompson und Dennis Ritchie bei der Arbeit an einer PDP 11.

Heute ist Unix auf Workstations mit 64-Bit-Prozessoren zugeschnitten. Zu den Neuerungen gehören die Einführung von Threads, mit denen ein Programm mehrere Funktionen gleichzeitig ausführen kann. Aber schon seit den Anfängen ist Unix mit folgenden Eigenschaften ausgestattet:

- hierarchisches, universelles Dateisystem
- Mehrprozessbetrieb (Multitasking)
- Mehrbenutzersystem (Multiuser-System)
- Netzwerkfähigkeit
- Plattformunabhängigkeit
- Das Unix-Werkzeugkasten-Prinzip und die Shell

Unix gehört historisch gesehen zu den älteren Betriebssystemen. Dennoch ist es gleichzeitig ein modernes Betriebssystem, das von Anfang an mit Merkmalen ausgestattet war, die von anderen Systemen erst viel später in einer vergleichbaren Form angeboten wurden. Unter Unix gab es von Anfang an echtes Multitasking, eine Trennung der Prozesse

voneinander (d. h. hohe Stabilität), klar definierte Zugriffsrechte für Dateien (d. h. hohe Sicherheit im Multiuser-Betrieb), ausgereifte Netzwerkfunktionen etc. Allerdings bot Unix vor ein bis zwei Jahrzehnten nur eine kommandozeilenorientierte Benutzeroberfläche und stellte hohe Hardware-Anforderungen. Deshalb wurde es fast ausschließlich auf teuren Workstations im wissenschaftlichen und industriellen Bereich eingesetzt.

Aber auch das ist Geschichte und die Einsatzzwecke haben sich grundlegend geändert. Heutzutage sind Unix Nachkommen auf jedem normalen PC lauffähig. Eins der sichersten und stabilsten Derivate ist OpenBSD, das wir uns nun detaillierter anschauen und behandeln.

3 Was ist OpenBSD

»Free, functional, and secure« – so lautet das Motto des Betriebssystems OpenBSD (nicht zu verwechseln mit dem Projekt OpenSSL). BSD ist eine Abkürzung und steht für »Berkeley Software Distribution«, welches ursprünglich eine Variante des UNIX-Systems bezeichnete. Heute bezeichnet man mit BSD eine Familie von Betriebssystemen, welche sich vom ursprünglichen BSD ableiten. Dies sind zum Beispiel FreeBSD, NetBSD, DragonFlyBSD und eben OpenBSD, um das es in diesem Workshop geht.

Das »Open« im Projektnamen deutet auf den kompromisslosen Einsatz der BSD-Lizenz (vereinfachte Form ISC-Lizenz) sowie die Haltung für Freie Software hin, welche das Entwicklerteam vertritt. So wurden schon viele, teils einfache, teils umfangreiche und komplexe Programme nachprogrammiert, um diese in der BSD-Lizenz nutzen zu können. Die BSD-Lizenz unterscheidet sich von der GPL hauptsächlich darin, dass sie kein Copyleft enthält und somit Änderungen am Quelltext unter einer anderen Lizenz fortgeführt werden können. Dadurch lässt sich unter BSD-Lizenz stehender Quelltext sehr gut auch in kommerziellen Produkten verwenden.

Als Abspaltung vom NetBSD-Projekt begann das Team um Theo de Raadt



1995 mit der Programmierung eines BSD-basierten Systems, das heute zu den sichersten Betriebssystemen überhaupt gezählt wird. Als Maskottchen wurde »Puffy« erkoren; ein Kugelfisch, der den Sicherheitsgedanken symbolisieren soll.

Alle sechs Monate erscheint ein neues Stable-Release, seit der aktuellen Version 6.1 gibt es kein CD Set mehr zu kaufen, OpenBSD wird nur noch zum Download angeboten.

3.1 Warum OpenBSD

Einige Gründe, warum wir denken, dass OpenBSD ein nützliches Betriebssystem ist:

- OpenBSD läuft auf vielen verschiedenen Hardware-Plattformen.
- OpenBSD wird als das sicherste UNIX-ähnliche Betriebssystem von vielen Sicherheitsexperten angesehen, als Ergebnis des nie endenden umfassenden Quellcode-Audits.

- OpenBSD ist ein voll funktionsfähiges UNIX-ähnliches Betriebssystem, das in Quell- und Binärform kostenlos zur Verfügung steht.
- OpenBSD integriert modernste Sicherheitstechnologie, die für den Aufbau von Firewalls und privaten Netzwerkdiensten in einer verteilten Umgebung geeignet ist.
- OpenBSD profitiert von einer starken Weiterentwicklung in vielen Bereichen und bietet Möglichkeiten, mit neuen Technologien und einer internationalen Community von Entwicklern und Endbenutzern zusammenzuarbeiten.
- OpenBSD versucht, die Notwendigkeit von Anpassung und Optimierung zu minimieren. Für die überwiegende Mehrheit der Nutzer arbeitet OpenBSD nur auf ihrer Hardware für ihre Anwendung.

3.1.1 OpenBSD ist frei

OpenBSD ist kostenlos. Die Binärdateien sind frei. Die Quelle ist frei. Alle Teile von OpenBSD haben angemessene Copyright-Bedingungen, die eine kostenlose Umverteilung ermöglichen. Mehr über OpenBSDs Urheberrechtspolitik finden Sie hier .

Die Betreuer von OpenBSD unterstützen das Projekt weitgehend aus ihren eigenen Taschen. Dazu gehören die Zeitaufwand für das Projekt, die Ausrüstung, die zur Unterstützung der vielen Ports verwendet wird, Netzwerkressourcen, die verwendet werden, um OpenBSD an Sie zu verteilen, und die Zeit, die beantwortet wurde, um Fragen zu beantworten und die Fehlerberichte der Benutzer zu untersuchen. Die OpenBSD-Entwickler sind nicht unabhängig reichen, und sogar kleine Beiträge von Zeit, Ausrüstung und Ressourcen machen einen großen Unterschied.

3.1.2 Die Dokumentation

OpenBSD kommt mit einer umfangreicher Dokumentation in Form von Man-Pages. Sie sind die maßgebliche Informationsquelle für OpenBSD, und es werden erhebliche Anstrengungen unternommen, um sicherzustellen, dass sie aktuell und genau sind. Von Entwicklern, die eine Änderung des Systems vornehmen, wird erwartet, dass sie die Man-Seiten zusammen mit ihrer Änderung des Systemcodes aktualisieren. Es wird erwartet, dass die Benutzer die Man-Pages überprüfen, bevor sie um Hilfe bitten.

3.1.3 Ports

OpenBSD empfiehlt die Benutzung von fertigen Binärpaketen.

3.1.4 Fazit

OpenBSD ist eine Unix-Distribution für jeden denkbaren Einsatzzweck, die genau an spezifische Aufgaben angepasst werden kann. Sie ist schnell, kompakt, flexibel und die meisten Teile unter der Haube sind ziemlich einfach zu verstehen und einzustellen. OpenBSD ist daher auch eine gute Distribution, um die Grundlagen zu lernen. Es werden keine Konfigurationsassistenten zur Verfügung gestellt, so dass Du schnell lernen wirst, das System von der Kommandozeile aus zu konfigurieren.

4 Vor der Installation

4.1 Mindestanforderungen

Die Mindestanforderungen variieren von System zu System und sind auch vom Einsatzzweck abhängig.

4.1.1 Plattformen

OpenBSD unterstützt offiziell folgende Plattformen:

- alpha - Digital Alpha-basierte Systeme
- amd64 - AMD64-basierte Systeme
- armv7 - ARM-basierte devices, suche als BeagleBone, BeagleBoard, PandaBoard ES, Cubox-i, SABRE Lite, Nitrogen6x und Wandboard
- hppa - Hewlett-Packard Precision Architecture (PA-RISC) Systeme
- i386 - Standard PC und Clones basierend auf der Intel i386 Architektur und compatible Prozessoren
- landisk - IO-DATA Landisk Systeme (suche als USL-5P) basierend auf der SH4 Cpu
- loongson - Loongson 2E- and 2F-basierende Systeme, suche als the Lemote Fuloong and Yeeloong, Gdium Liberty, etc.
- luna88k - Omron LUNA-88K und LUNA-88K2 workstations
- macppc - Apple Neue Welt PowerPC-basierte Maschinen, von dem iMac aufwärts
- octeon - Cavium Octeon-basierte MIPS64 Systeme
- sgi - SGI MIPS-basierte workstations

- sparc64 - Sun UltraSPARC und Fujitsu SPARC64 Systeme

Dieses Tutorial behandelt ausschließlich die Grundinstallation auf einem normalen handelsüblichen PC mit amd64 Architektur.

4.2 Beschaffung

Du kannst Dir OpenBSD hier besorgen:



<https://www.openbsd.org/>

Das ISO Image install62.iso brennen wir auf eine CD. Danach starten wir unseren Rechner neu und booten von dieser Installations-CD.

5 Grundinstallation

5.1 Der Installer

Nach dem Booten meldet sich der Installer im Textmodus. Die Grundinstallation beschränkt sich auf das Wesentliche und ist in 5 Minuten erledigt.

```
vga1 at pci0 dev 2 function 0 "InnoTek VirtualBox Graphics Adapter" rev 0x00
vga1: aperture needed
wsdisplay0 at vga1 mux 1: console (80x25, vt100 emulation)
em0 at pci0 dev 3 function 0 "Intel 82540EM" rev 0x02: apic 1 int 19, address 08
:00:27:1c:59:9c
"InnoTek VirtualBox Guest Service" rev 0x00 at pci0 dev 4 function 0 not configu
red
"Intel 82801AA AC97" rev 0x01 at pci0 dev 5 function 0 not configured
ohci0 at pci0 dev 6 function 0 "Apple Intrepid USB" rev 0x00: apic 1 int 22, ver
sion 1.0
"Intel 82371AB Power" rev 0x08 at pci0 dev 7 function 0 not configured
usb0 at ohci0: USB revision 1.0
uhub0 at usb0 "Apple OHCI root hub" rev 1.00/1.00 addr 1
isa0 at mainbus0
pckbc0 at isa0 port 0x60/5
pckbd0 at pckbc0 (kbd slot)
pckbc0: using irq 1 for kbd slot
wskbd0 at pckbd0: console keyboard, using wsdisplay0
softraid0 at root
scsibus1 at softraid0: 256 targets
root on rd0a swap on rd0b dump on rd0b
erase ^?, werase ^W, kill ^U, intr ^C, status ^T

Welcome to the OpenBSD/amd64 5.7 installation program.
(I)nstall, (U)pgrade, (A)utoinstall or (S)hell? w_
```

Hier geben wir (I) für die Grundinstallation ein.

5.2 Tastaturlayout

Es erscheint folgendes Fenster:

```
Choose your keyboard layout ('?' or 'L' for list) [default] L
Available layouts: be br cf de dk es fr hu is it jp la lt lv nl no pl pt ru sf s
g si sv tr ua uk us
Choose your keyboard layout ('?' or 'L' for list) [default] us_
```

Wir wählen jetzt das Tastaturlayout aus. Für Deutschland ist das de.

Für weitere Informationen können wir ein Fragezeichen eingeben, mit “L” bekommen wir eine Liste aller verfügbaren Tastaturlayouts angezeigt.

Nach der Eingabe von de drücken wir <ENTER>.

5.3 Hostname und Netzwerkschnittstelle

Jetzt wird der Host und die Netzwerkschnittstelle konfiguriert:

```
System hostname? (short form, e.g. 'foo') example
Available network interfaces are: em0 vlan0.
Which network interface do you wish to configure? (or 'done') [em0] _
```

Als Hostname gebe ich immer arbeit ein. Für das Netzwerkinterface übernehme ich die ermittelte Vorgabe und bestätige das mit <ENTER>.

Wer IPv6 nicht benötigt, überspringt das mit <ENTER>.

5.4 Root Passwort setzen

Nun setzen wir im nächsten Schritt das Root-Passwort für den Administrator.

```
Password for root account? (will not echo)
Password for root account? (again)
```

Es wird empfohlen, hier ein belastbares Passwort zu vergeben.

5.5 OpenSSH Daemon

Nach der Einrichtung des Root Passwortes fragt der Installer nach dem Standardstart von

- sshd - OpenSSH Daemon
- ntpd - Network Time Protocol Daemon (Zeitserver)

und dem X Window System.

Wenn Du nichts über diese Daemonen weiß, übernehme die Vorgabe und damit die Standardwerte.

5.6 X Window System

Wir übernehmen die Vorgaben einfach mit mehrmaligem <ENTER>.

Wir wollen keinen Login Manager installieren, weil das System mittels einer selbstangelegten `.xinitrc` manuell gestartet wird.

5.7 Benutzer anlegen

Einen neuen Benutzer legen wir separat nach der erfolgten Grundinstallation an.

5.8 Zeitzone

Jetzt richten wir eine Zeitzone ein:

```
Setup a user? (enter a lower-case loginname, or 'no') [no] example
Full name for user example? [example] mr Example
Password for user example? (will not echo)
Password for user example? (again)
What timezone are you in? ('?' for list) [Europe/Moscow]
```

Normal ermittelt das System automatisch die richtige Zeitzone. Wir können uns mit dem Fragezeichen eine Liste aller verfügbaren Zeitzonen anzeigen lassen.

Bei uns ist das [Europe/Berlin].

Wir bestätigen das mit <ENTER>

5.9 Partitionierung

```
Available disks are: wd0.
Which disk is the root disk? ('?' for details) [wd0]
Use DUIDs rather than device names in fstab? [yes]
'' is not a valid choice.
Use DUIDs rather than device names in fstab? [yes] yes
MBR has invalid signature; not showing it.
Use (W)hole disk or (E)dit the MBR? [whole] W
Setting OpenBSD MBR partition to whole wd0...done.
The auto-allocated layout for wd0 is:
#          size      offset  fstype  [fsize bsize  cpg]
a:         851.8M         64  4.2BSD   2048 16384    1 # /
b:         183.5M    1744448      swap
c:         3072.0M          0  unused
d:         1707.4M    2120320  4.2BSD   2048 16384    1 # /usr
e:          328.5M    5617152  4.2BSD   2048 16384    1 # /home
Use (A)uto layout, (E)dit auto layout, or create (C)ustom layout? [a]
```

Standardmäßig ermittelt das System eine sinnvolle Partitionierung für die gesamte Festplatte.

Deshalb benutzen wählen wir (A) für ein automatisches Layout.

Natürlich können wir an dieser Stelle auch ein individuelles Layout anlegen, dazu bedarf es aber einiger Vorüberlegungen und Erfahrung.

5.10 Basiskomponenten installieren

Hier eine kurze Übersicht der auszuwählenden Sets von CD:

- bsd - Kernel
- bsd.mp - Kernel für Multiprozessorsystem
- bsd.rd - Kernel mit RAM Laufwerk
- base61.tgz - Basis OpenBSD System
- comp61.tgz - Entwicklungswerkzeuge, Compiler, Bibliotheken und Header
- man61.tgz - Manual Seiten
- game61.tgz - Spiele
- xbase61.tgz - Basisinstallation von X11
- xfont61.tgz - Schriftart-Server und Schriftarten X11
- xserv61.tgz - X11 Server
- xshare61.tgz - Handbücher, Locales, und anderes für das X Window System

```
Let's install the sets!
Location of sets? (cd disk http or 'done') [cd] cd
Available CD-ROMs are: cd0.
Which CD-ROM contains the install media? (or 'done') [cd0] cd0
Pathname to the sets? (or 'done') [5.7/amd64]

Select sets by entering a set name, a file name pattern or 'all'. De-select
sets by prepending a '-' to the set name, file name pattern or 'all'. Selected
sets are labelled '[X]'.
[X] bsd          [X] base57.tgz    [X] game57.tgz    [X] xfont57.tgz
[X] bsd.rd      [X] comp57.tgz   [X] xbase57.tgz  [X] xserv57.tgz
[ ] bsd.mp      [X] man57.tgz   [X] xshare57.tgz
Set name(s)? (or 'abort' or 'done') [done]
```

Sie können einzelne Sets abwählen. Wir empfehlen, die Vorgabe aller Sets zu übernehmen.

Das bestätigen wir durch <ENTER>

Wir sehen nun den Installationsprozess und den Installationsfortschritt:

```
Set name(s)? (or 'abort' or 'done') [done]
[X] bsd          [X] base57.tgz    [X] game57.tgz    [X] xfont57.tgz
[X] bsd.rd       [X] comp57.tgz    [X] xbase57.tgz  [X] xserv57.tgz
[ ] bsd.mp       [X] man57.tgz     [X] xshare57.tgz
Set name(s)? (or 'abort' or 'done') [done] done
Directory does not contain SHA256.sig. Continue without verification? [no] yes
Installing bsd          100% |*****| 9794 KB 00:01
Installing bsd.rd       100% |*****| 7414 KB 00:00
Installing base57.tgz  100% |*****| 56035 KB 00:13
Extracting etc.tgz     100% |*****| 110 KB 00:00
Installing comp57.tgz  100% |*****| 50045 KB 00:12
Installing man57.tgz   100% |*****| 8773 KB 00:02
Installing game57.tgz  100% |*****| 2724 KB 00:00
Installing xbase57.tgz 100% |*****| 16660 KB 00:04
Extracting xetc.tgz   100% |*****| 9169 00:00
Installing xshare57.tgz 100% |*****| 4413 KB 00:02
Installing xfont57.tgz 35% |*****| 13952 KB 00:05 ETA
```

Das geht ziemlich flott und dauert 2 bis 3 Minuten je nach Leistungsstärke des Rechners.

5.11 Abschluß der Installation und Neustart

```
CONGRATULATIONS! Your OpenBSD install has been successfully completed!
To boot the new system, enter 'reboot' at the command prompt.
When you login to your new system the first time, please read your mail
using the 'mail' command.
#
```

Die Grundinstallation ist nun abgeschlossen.

Herzlichen Glückwunsch, Ihr OpenBSD wurde erfolgreich und komplett installiert!

Es ist nun ein Neustart erforderlich den wir mit der Eingabe von reboot einleiten.

5.12 Nach dem Neustart

```
kern.securelevel: 0 -> 1
creating runtime link editor directory cache.
preserving editor files.
starting network daemons: smtpd sndiod.
Path to firmware: http://firmware.openbsd.org/firmware/5.7/
No devices found which need firmware files to be downloaded.
starting local daemons: cron.
Mon May  4 17:43:51 MSK 2015

OpenBSD/amd64 (example.my.domain) (ttyC0)

login: example
Password:
OpenBSD 5.7 (GENERIC) #825: Sun Mar  8 10:59:14 MDT 2015

Welcome to OpenBSD: The proactively secure Unix-like operating system.

Please use the sendbug(1) utility to report bugs in the system.
Before reporting a bug, please try to reproduce it with the latest
version of the code.  With bug reports, please try to ensure that
enough information to reproduce the problem is enclosed, and if a
known fix for it exists, include that as well.

You have new mail.
$
```

Wir sehen folgendes Bild und können uns nun als root einloggen, um die Konfiguration zu vervollständigen.

5.13 Zusammenfassung

OpenBSD ist ein sehr zuverlässiges Betriebssystem.

Und es ist auch eine gute Wahl für den Produktiveinsatz wie auch dem Desktop.

6 Nach der Grundinstallation

6.1 Aktualisierung und Updates

Nach der Grundinstallation ist eine Aktualisierung des Systems erforderlich, um es auf den aktuellen Stand zu bringen.

Neu ist, das Base System Updates über syspatch abgedeckt werden.

Als root:

```
# syspatch
```

holt alle Patches seit der Release Ausgabe und installiert sie.

Hier gibt es ein 3rd Party Script openup zum Download und zur Aktualisierung der installierten Binärpakete:

```
https://www.mtier.org/solutions/apps/openup/
```

openup ist ein Shell Script, das wir ausführbar machen müssen:

```
chmod +x openup
```

Jetzt starten wir die Aktualisierung der Binärpakete:

Als root:

```
# sh openup
```

Openup prüft für jedes installierte Binärpaket, ob es eine neuere Version gibt und holt und installiert sie automatisch. Das ist sehr praktisch und wichtig!

6.2 Benutzerverwaltung

6.2.1 Neuen Benutzer anlegen

mit adduser, weil:

- adduser scriptgesteuert und interaktiv ist
- es erledigt den Eintrag für das neue Passwort in `/etc/passwd` und `/etc/group` gleich mit

Wir legen nun mit `adduser` einen neuen Benutzer an:

```
# adduser
```

6.3 Gruppenzugehörigkeit konfigurieren

Wir editieren die Datei `/etc/group` und fügen den neuen Benutzer in die folgenden Gruppen ein

- wheel
- operator
- staff

Die Gruppe `staff` bedarf unserer besonderen Aufmerksamkeit, wir müssen nicht nur uns als Benutzer hinzufügen, sondern anpassen:

1.) Den user in die Gruppe `staff` aufnehmen

2.) in `/etc/login.conf` folgendes hinzufügen bzw. die Gruppe `staff` dort wie folgt anpassen:

Code:

```
staff:\
:datasize-cur=infinity:\
:datasize-max=infinity:\
:datasize=infinity:\
:openfiles-cur=1024:\
:stacksize-cur=16M:\
:maxproc-max=512:\
:maxproc-cur=512:\
:ignorenologin:\
:requirehome@:\
```

:tc=default:

3.) kern.shminfo.shmall erhöhen:

Code:

```
# echo kern.shminfo.shmall=268435456 >> /etc/sysctl.conf
```

```
# sysctl kern.shminfo.shmall=268435456
```

4.) Ausloggen und wieder einloggen.

6.4 Soft Updates

Ich benutze vi:

Im Terminal:

Code:

```
# vi /etc/fstab
```

Dann im vi:

Code:

```
:%s/rw/rw,softdep/g [Enter-Taste drücken]
```

```
:wq [Enter-Taste drücken]
```

Dann OpenBSD rebooten und das gesamte Filesystem sollte dann soft updates unterstützen.

6.5 Sudo Ersatz doas einrichten

Eine neue /etc/doas.conf anlegen und eintragen:

```
permit nopass user as root
```

Jetzt ist der hinzugefügte user mit root Rechten ausgestattet.

6.6 Paketverwaltung konfigurieren

Wir erstellen mit einem Editor unserer Wahl im Verzeichnis `/etc` eine Datei `installurl` und tragen einen Mirror ein:

```
https://ftp.hostserver.de/pub/OpenBSD/
```

Nun können wir Binärpakete mit `pkg_add <Paketname>` installieren

6.7 Desktop XFCE installieren

```
pkg_add consolekit2 xfce xfce-extras
```

Der XFCE Desktop bringt folgende Programme bereits mit:

Thunar (Dateimanager)

Mousepad (Texteditor)

Ristretto (Bildanzeigeprogramm)

Terminal

Taskmanager

Darüber hinaus gibt es noch kleine Helferlein wie Orage als Kalender, Xarchiver zum Packen und Entpacken von Archiven sowie ein Programm für Bildschirmfotos.

Nach der Installation des XFCE Desktops müssen wir noch im Homeverzeichnis des Benutzers eine `vi .xinitrc` mit folgendem Inhalt anlegen:

```
export LANG=de_DE.UTF-8
```

```
setxkbmap de
```

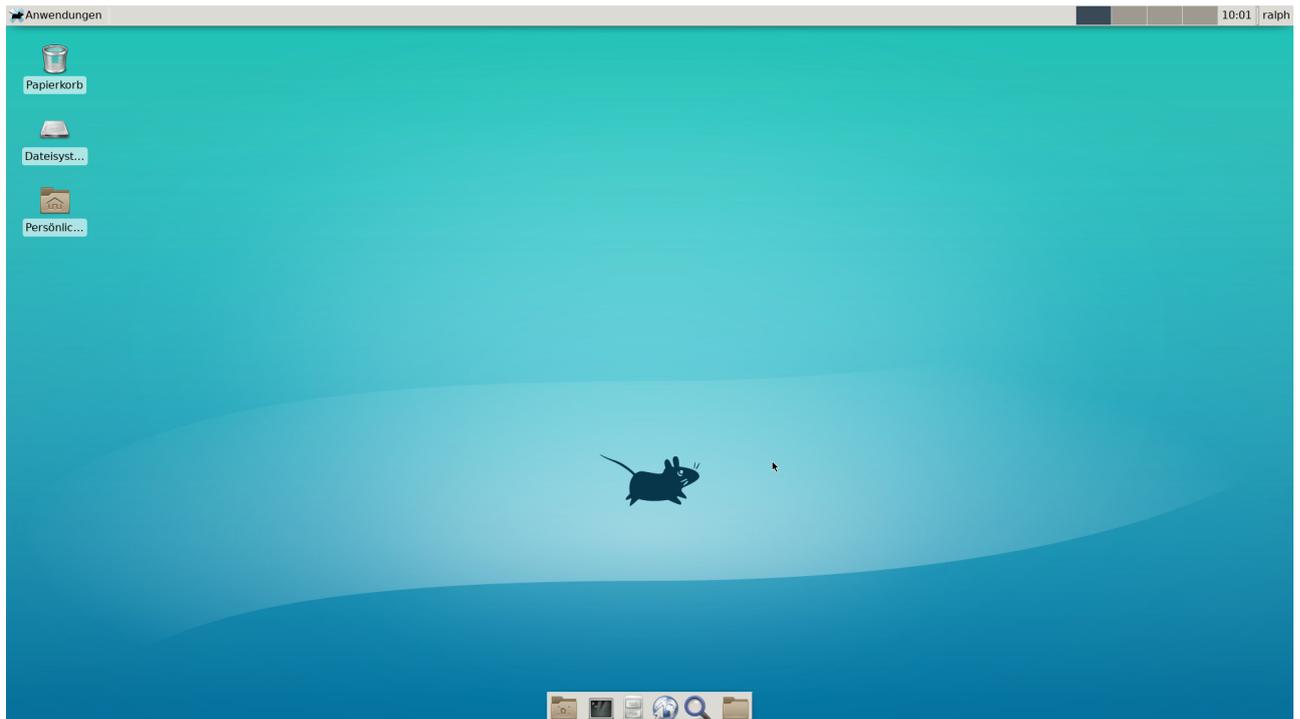
```
xset -b
```

```
numlockx on # das Paket numlockx muß installiert sein
```

```
xrandr -s 1920x1080 # gewünschte Bildschirmauflösung setzen
```

```
exec startxfce4
```

Nach einem reboot loggen wir uns ein und starten unser X Window System mit dem XFCE Desktop mit `startx`.



Wir sollten noch `pkg_add mupdf numlockx` installieren.

Mupdf ist ein ressourcensparender PDF Betrachter, numlockx schaltet in der `.xinitrc` den numerischen Block der Tastatur ein.

6.8 Minimalsystem mit Windowmanager JWM

Nach der Grundinstallation installieren wir den Windowmanager JWM:

```
pkg_add jwm
```

Einen Dateimanager unserer Wahl: rox-filer, pcmanfm, thunar

Einen Texteditor - leafpad oder mousepad

Einen Bildbetrachter - gpicview oder ristretto

Einen PDF Betrachter - mupdf

Nun kopieren wir die `/etc/system.jwmrc` (eine Beispielkonfigurationsdatei) in unser Homeverzeichnis und benennen sie um in:

```
.jwmrc
```

Wir laden die Datei `.jwmrc` in einen Texteditor unserer Wahl und passen sie an:

Ich habe eine fertig angepaßte .jwmrc erstellt, die hier heruntergeladen und sofort benutzt und eingesetzt werden kann:

<https://www.bsdforen.de/threads/mwm-verschwunden.34108/#post-299156>

Sie muß lediglich wieder umbenannt werden in .jwmrc



Have Fun!

7 Multimedia

Ich benutze und empfehle

- moc - Konsolenplayer für verschiedene Audioformate wie mp3, ogg, wav und flac
- mpv - Videoplayer für alle gängigen Videoformate
- abcde - zum Rippen von Musik CD's

```
pkg_add moc mpv abcde
```

Benutzer die den Windowmanager jwm einsetzen und über keine Multimediatasttaur verfügen, sollten das Paket

pkg_add volumeicon installieren. Dies Paket beinhaltet einen Lautstärkeregler.

Sie sollten dann am Ende der .jwmrc einfügen:

```
<StartupCommand>volumeicon</StartupCommand>
```

8 Internet

Wir müssen noch einen Browser und einen Mail Clienten installieren. Ich benutze

- Firefox als Browser
- Claws-Mail als Mail Clienten

```
pkg_add firefox
```

```
pkg_add firefox-i18n-de (deutsche Lokalisation)
```

Natürlich stehen noch andere Pakete wie Seamonkey, Chrome oder Thunderbird zur Auswahl.

9 Wissenswertes

9.1 Paketverwaltung

OpenBSD benutzt für seine ganze Paketverwaltung 3 einfache Befehle:

- `pkg_add` - zum Installieren und Aktualisieren von Softwarepaketen
- `pkg_delete` - zum Löschen von zuvor installierten Softwarepaketen
- `pkg_info` - zum Anzeigen von Informationen über Softwarepakete

Mit `pkg_info > pakete` erzeugen wir eine Datei `pakete`, in der alle bereits installierten Softwarepakete angezeigt werden.

Mit `pkg_info -m` werden alle Pakete angezeigt, die manuell und nicht als Abhängigkeit installiert wurden.

Mit `pkg_delete -a` werden alle nicht mehr notwendigen Abhängigkeiten deinstalliert, allerdings nur, wenn diese nicht von irgendwelchen installierten Paketen abhängig sind.

9.2 Hilfe

Bei Problemen bietet sich an, das deutsche BSD Forum zu besuchen: <https://www.bsdforen.de>

Hier triffst Du viele hilfsbereite und kompetente Communitymitglieder.

Die Chancen stehen hoch, dass ein anderer OpenBSD Benutzer schon die gleiche Frage hatte wie Du, und dass sie bereits beantwortet wurde. Frag einfach mal herum!

Außerdem steht Dir mit dem Wiki des Forums eine große Hilfe zur Verfügung. Blättere einfach die Dokumentation durch oder nutze die Suche, um Hilfe zu dem Thema zu finden, zu dem Du Hilfe brauchst.

Hilfreich ist auch die FAQ der Homepage von OpenBSD: <https://www.openbsd.org/faq/index.html>.

Oder schau Dich einfach mal auf der Homepage des OpenBSD Projektes um : <https://www.openbsd.org>.

Hier gibt es immer Neues zu entdecken!

9.3 Aktualität

In halbjährlichen Abstand, also ca. alle 6 Monate erscheint eine neue aktualisierte Version von OpenBSD. Zur Zeit ist die Version 6.2 aktuell.

9.4 Fazit

Also, kurzum: OpenBSD ist eine mächtiges Betriebssystem, das für die Bedürfnisse kompetenter UNIX-Nutzer entworfen wurde. Sie soll einfach zu warten sein, so dass sie eine ideale Distribution für Server- und Einzelplatz-Rechner bildet. Mach damit (und mach daraus), was immer Du willst!

10 Spezielles

10.1 Der Editor Vi

Der vi ist unter Informatikern geliebt und gehaßt. Für die liebenden, die Puristen, gilt er als eines der mächtigsten Werkzeuge überhaupt, überall verfügbar, stabil wie sonst nichts, und wer ihn bedienen kann, gehört einfach dazu, jedenfalls in bestimmten Kreisen, in anderen wird man gemieden. Gehaßt wird er, weil seine Bedienung nicht erlernbar ist, ohne einen Pakt mit dem Teufel zu schließen. Geschrieben im Jahre 1973, also schon fast dreißig Jahre alt, fehlt dem vi jede graphische Benutzerführung. Menüs oder gar die Maus sind ihm unbekannt. Jede Eingabe erfolgt über Tastenkombinationen, die so kryptisch sind, daß selbst der Erwerb von Grundkenntnissen malaiischer Sprachen wie ein Kinderspiel wirkt. Genug der Polemik. Woher kommt die Motivation, sich dennoch mit dem vi zu beschäftigen?

- Wie gesagt: vi ist überall verfügbar. Er gehört zur Unix-Standardausstattung. Das bedeutet, wenn kein anderer Editor installiert ist, wird man auf den vi ausweichen müssen. Insbesondere Systemadministratoren kommen daher um den vi nicht herum.
- vi ist wirklich sehr mächtig. Wer nicht vor monatelangem Studieren der zahllosen Einführungen zurückschreckt und mit viel Versuch und Irrtum diverse Arbeit zu-nichte gemacht hat, wird irgendwann in der Lage sein, mit einem einzigen Befehl die komplexesten Änderungen in seinen Programmquellen durchzuführen. (Man darf aber in der Zeit des Wissenserwerb um den vi nichts anderes tun, insbesondere keine anderen Programme nutzen oder mit Menschen reden, sonst kommt es zu einem Phänomen, das ich mal als kognitive Interferenz bezeichnen möchte.)
- Wenn man statt vi vim (vi improved) benutzt, hat man etwas mehr Komfort.
- Hauptargument für die Auseinandersetzung mit vi ist, daß es einem leicht passieren kann, ihn auf einmal gestartet zu haben und nicht mehr herauszufinden. Das liegt

daran, daß manche Programme standardmäßig vi als Texteditor verwenden. Dazu gehören elm, more und less. Durch Setzen der Umgebungsvariablen EDITOR läßt sich dies manchmal ändern.

Das wichtigste zum Umgang mit vi ist zu wissen, wie man ihn beendet, und zwar sowohl mit als auch ohne Speichern etwaiger Änderungen in einer Datei. Zunächst aber zum Aufruf: vi dateiname (s)

Mit vi + zeilennummer dateiname wird der Cursor gleich in der Zeile zeilennummer positioniert. Statt vi kann natürlich auch vim aufgerufen werden.

Der vi hat zwei verschiedenen Modi, den Kommandomodus und den Einfügemodus. Nach dem Start befindet man sich im Kommandomodus. Was hier eingegeben wird, wird als Kommando interpretiert, nicht als Text. Das ist sicher sehr ungewohnt. In den meisten Editoren kann man direkt mit der Eingabe von Text loslegen. Mit i (insert) wechselt man in den Einfügemodus. Jetzt kann man schreiben, aber z.B. nicht mit den Cursortasten im Text herumwandern. Zurück in den Kommandomodus kommt man mit ESC. Die wichtigsten Kommandos sind:

10.1.1 Allgemeines

Alle vim-Kommandos (beginnen mit ":") können in die Konfigurationsdatei eingetragen werden. Diese kann heissen: .gvimrc oder .vimrc.

10.1.2 Bearbeitungsmodi

ESC :	Kommandomodus
i	Einfügemodus (an der aktuellen Cursorposition)
A	Einfügemodus (am Ende der aktuellen Zeile)
:	Execute Modus (Eingabe eines Kommandos)

10.1.3 Dateiverwaltung

:e Dateiname	Datei laden
:e Verzeichnis	Dateimanager (Der Parameter ist ein Verzeichnis)
:w	Aktuelle Datei speichern
:w Dateiname	Aktuelle Datei speichern unter einem neuen Namen (Bearbeitet wird weiter das)
:sav Dateiname	Aktuelle Datei speichern unter einem neuen Namen (Neue Datei wird bearb)
:buffer Dateiname	Zwischen den Buffers (geöffnete Dateien) umschalten
:next	Zum nächsten Buffer wechseln
:prev	Zum vorherigen Buffer wechseln
:q	VIM beenden (nur wenn alle Buffers gespeichert, sonst Meldung)
:q!	VIM beenden (nicht gespeicherte Buffers gehen verloren)
:wq	Speichern und Schliessen (Alternativ: [ZZ])

10.1.4 Editor

u:	Aktion rückgängig machen (undo)
Ctrl+r:	Aktion wiederherstellen (redo)

yy	Zeile kopieren
dd	Zeile ausschneiden (löschen)
dw	Wort ausschneiden (ab der aktuellen Position nach rechts)
db	Wort ausschneiden (ab der aktuellen Position nach links)
d^	Bis Zeilenanfang löschen (ab der aktuellen Position)
d\$	Bis Zeilenende löschen (ab der aktuellen Position)
: 'a, 'bd	Von Marke "a" bis Marke "b" löschen
p	Einfügen
v	Anfang des zu markierenden Bereichs setzen (Mit den Cursortasten Bereich markieren)
y	Markierten Bereich kopieren
"xy	Markierten Bereich in das Puffer "x" kopieren (es können bis zu 26 Puffer belegt werden)
"xp	Text aus dem Puffer "x" einfügen
VG	Alles markieren (ab der aktuellen Position nach unten)

mx	Marke setzen ("x" ist ein bel. Kleinbuchstabe -> es sind 26 Marken möglich)
'x	Zur Marke "x" springen

J	CR/LF am Ende einer Zeile löschen (Zeilen zusammenfügen)
:set list	Nicht darstellbare Zeichen anzeigen
:set ic	Gross/Kleinschreibung ignorieren
Ctrl+n	Wort vervollständigen
:%!xxd	HEX-Editor-Modus (xxd muss installiert sein)

10.1.5 Im Text navigieren

gg	An die erste Position des Textes springen
GG	An die letzte Position des Textes springen
[m][Cursor]:	Um "m" Zeichen/Zeilen springen (mit den Cursortasten -> links, rechts, oben, unten)
w:	Wortweise nach rechts springen (Mit [b] nach links) -> auch hier kann eine Zahl vorangestellt werden

10.1.6 Suchen und Ersetzen

/Suchwort:	Nach "Suchwort" suchen ("Suchwort" ist ein reg. Ausdruck). Mit [n] weiter springen
/Suchwort/i:	Nach "Suchwort" suchen, Gross/Kleinschreibung ignorieren
:1,\$s/Unix/Linux/g:	"Unix" durch "Linux" ersetzen (Bereich: 1,\$ -> vom Anfang bis Ende, g: Alle Zeichen)
:1,.s/^/#/g:	Alle Zeilen bis zur aktuellen Cursorposition (.) auskommentieren

10.1.7 Sonstiges

:split:	Hauptfenster wird horizontal aufgespaltet. Wechseln mit [Ctrl+w+w].
:r !date:	Ausgabe eines UNIX-Kommandos in den Text einfügen (hier Datum)
:syntax on:	Syntaxeinfärbung einschalten
:set nu!:	Zeilennummern ein-/ausblenden.
:set ai:	Automatisches Einrücken (AutoIdent)
:set cin:	Automatisches Einrücken im Still von C (Einrücken in Funktionsrümpfen)
:set softtabstop=3:	Tabulator entspricht drei Leerzeichen
:set shiftwidth=3:	Autom. Einrücken entspricht drei Leerzeichen
:ab w1 wort:	Autom. Ersetzen von "w1" durch "wort" während der Eingabe