

O'REILLY®

Stefan Pietraszak

Das Buch zu Raspberry Pi mit Linux



Stefan Pietraszak, Das Buch zu Raspberry Pi mit Linux, O'Reilly, ISBN 978-3-95561-698-4

Inhaltsverzeichnis

Über den Autor	3	2. Vorbereitungen	39
Vorwort	11	Weitere Anschaffungen	41
Raspberry Pi und Linux: ein starkes Team	13	Übertragung des Image auf SD-Karte unter Windows	43
Mission Statement	15	Installation des Image auf SD-Karte unter Linux	45
Danksagung	17	Übertragung des Image auf SD-Karte unter Mac OS X	47
1. Grundlagen	19	Exkurs: Anschluss mittels HDMI/VGA-Adapter	49
Das Board des Raspberry Pi	21	Der erste Start	51
Erforderliche Komponenten	23	Erste Einstellungen	53
Die Anschlüsse des Raspberry Pi	25	Der zweite Start	55
Linux – was ist das?	27	3. RaspBMC – RasPi als Medien-PC	57
Die unterschiedlichen Linux-Distributionen (1)	29	Vorbereitungen für den RasPi als Medien-PC	59
Die unterschiedlichen Linux-Distributionen (2)	31	WLAN statt Kabel	61
Die unterschiedlichen Linux-Distributionen (3)	33	Installation und Anschluss des RasPi als Medien-PC	63
Einsatzgebiete (1): Medien- und Desktop-PC	35	Die Benutzeroberfläche von RaspBMC	65
Einsatzgebiete (2): Medienserver und der Rest	37	Fine-Tuning der Darstellung	67
		Steuerung von RaspBMC	69

Exkurs: Fernsteuerung mittels FLIRC und Universalfernbedienung	71
Einbinden von Medien	73
Wiedergabe von Medien	75
Erweiterung von RaspBMC mittels Add-Ons	77
Troubleshooting beim Medien-PC	79

4. Raspbian	81
Vorbereitungen für Raspbian	83
Benutzeroberfläche	85
Einstellungsfragen: WLAN & Co.	87
Verbindung zur Kommandozeile: das Terminal	89
Raspbian-Browser Midori	91
Raspbian-Standardprogramme	93
Installation weiterer Programme	95
Softwareauswahl: Office Suite und Textverarbeitung	97
Softwareauswahl: Multimedia auf Raspbian	99
Softwareauswahl: Browser und Mail-Client	101
Softwareauswahl: Diverses	103
Raspbian herunterfahren	105
Troubleshooting bei Raspbian	107

5. Linux-Grundlagen	109
Der Beginn der Geschichte	111
Login und Fernzugriff mittels SSH	113
Das Kommandozeilen-Interface	115
Prompt zur Stelle	117
Ein Blick in die Umgebung	119
Hey man!	121
Hin und wieder zurück	123
Rechte ohne Pflichten	125
Super, User!	127
Befehlszeile heißt Textmodus	129
Das Linux-Dateisystem	131
Benutzerverwaltung: Anlegen weiterer Benutzer	133
Benutzerverwaltung: Benutzergruppen	135
Erstellen, Kopieren und Löschen	137
Ändern von Dateiberechtigungen	139
Erwerben, Aktualisieren und Entfernen von Software	141
Zu Befehl: eine Übersicht	143
Skripte: »Programmieren« auf der Kommandozeile	145
Ein Job für cron	147
Ein Skript im Einsatz: Ausführen mittels Cronjob	149

Adieu: Ausschalten des RasPi mittels Befehl	151	7. Kleinprojekte	187
Weitere Befehle	153	Ein RasPi für Musik oder Radio	189
6. Raspberry Pi als Medienserver	155	MusikPi: Einbinden der Musik	191
Speicherplatz: Vorbereiten der externen Festplatte	157	MusikPi: MOC – Musik auf der Konsole	193
Der erste Start	159	MusikPi: MOC – Erstellen und Wiedergeben der Playlist	195
Voreinstellungen	161	RadioPi: Streams aus dem Internet	197
Kopflloser RasPi	163	RadioPi: Füllen der Senderliste	199
Einbinden der externen Festplatte	165	Anschluss und Nutzung der Pi-Kamera	201
Permanentes Einbinden der externen Festplatte	167	Zeitrafferaufnahmen mit der Pi-Kamera	203
Tanze Samba mit mir	169	Umwandeln von Zeitrafferbildern in einen Film	205
Die Konfiguration von Samba: ein Überblick	171	Backup-Server	207
Samba-Konfiguration: Bereitstellen von Mediendaten	173	RasPi-Backup-Server: Einhängen einer externen Festplatte	209
Die letzten Kleinigkeiten	175	RasPi Backup-Server: Einrichten von Samba	211
Einbinden unter Windows	177	RasPi Backup-Server: Einrichten des Backups	213
Nutzung der Medien auf dem RasPi-Medien-PC	179	Übertakten des RasPi	215
Android tanzt mit	181	Sicher ist sicher: Abhärten des RasPi (1)	217
Bereitstellen der Medien	183	Sicher ist sicher: Abhärten des RasPi (2)	219
Troubleshooting beim Medien-Server	185	Sicher ist sicher: Abhärten des RasPi (3)	221
		Dieses und jenes: weitere Kleinprojekte	223

8. Hardware-Projekte – Basteln mit dem RasPi	225
Breadboards	227
Lochrasterplatine (1)	229
Lochrasterplatine (2)	231
Lochrasterplatine (3)	233
Geätzte oder gefräste Platinen	235
Bauelement: LED	237
Bauelement: Widerstand	239
Bauelement: Kondensator	241
Weitere Bauelemente	243
Schlangenbändigung mit Python	245
Die Python auf der Lauer	247
Lass es blinken: Kalte Theorie	249
Lass es blinken: Breadboard-Aufbau	251
Lass es blinken: Hilfe durch Python	253
Daumen drücken, und zwar auf den Taster	255
Tasterabfrage in Python	257

9. Das Universum von Raspberry-Pi	259
Weiterführende Lektüre	261
Bestelladressen	263
Hüllen in Hülle und Fülle	265
Hardware-Zubehör	267
RasPi-Communities	269
RasPi-Projekte (1)	271
RasPi-Projekte (2)	273
Jenseits des Pi	275
Das Ende	277
Index	279



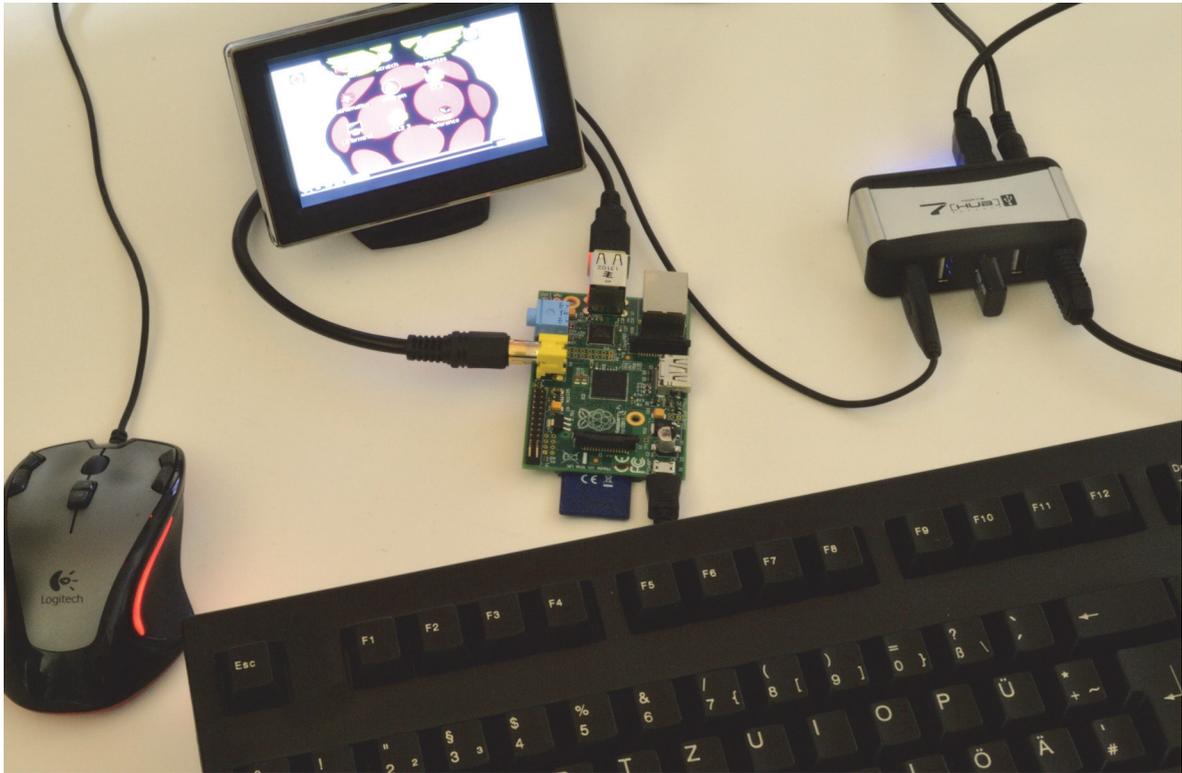
KAPITEL 4 | Raspbian

Neben dem Einsatz als **Medien-PC** ist die Nutzung des RasPi als **Desktop-PC**, mit dem wir durchs Web surfen, Mails abrufen, Texte schreiben und überhaupt alle grundlegenden Tätigkeiten erledigen können, für die ein »normaler« Computer üblicherweise so genutzt wird, ebenfalls ohne große Vorbereitungen und Grundkenntnisse möglich. Einfach die passende Linux-Distribution auf die SD-Karte übertragen, dann alles anschließen, RasPi starten, und schon kann es losgehen. Wer hierbei einen vollwertigen Ersatz für den heimischen PC erwartet, wird jedoch schnell enttäuscht. Der RasPi verfügt nicht über eine gewaltige Rechenleistung, der Arbeitsspeicher ist begrenzt und der USB 2.0-Port bremst angeschlossene Peripherie gegebenenfalls ordentlich aus. Dennoch kann er als »Zweit-PC« in einem anderen Zimmer, mit dem wir »nur mal eben« Mails lesen oder etwas im Internet recherchieren wollen, oder auch als »Notfall-PC«, falls der Hauptrechner streikt, gute Dienste leisten.

Wie schon in Kapitel 1 beschrieben, stehen uns zahlreiche Linux-Distributionen an Auswahl zur Verfügung. Die Entscheidung für Raspbian hat jedoch gute Gründe. Zunächst einmal handelt es sich um die Standarddistribution für den RasPi, was unter anderem bedeutet, dass durch die Community schnellere Hilfe zu erwarten ist, als bei etwas ausgefalleneren Distributionen. Des Weiteren basiert Raspbian auf Debian, dem soliden Arbeitspferd unter den Distros – , stets eine gute Wahl, wenn wir ein sicheres, stabiles und anpassbares System wünschen.

Proprietär? Open Source!

Software wird als »proprietär« bezeichnet, wenn sie nicht quelloffen ist, wir sie also verwenden können, der Quellcode aber nur von den Personen eingesehen oder geändert werden kann, die die Software programmiert haben. Bei Open Source Software hingegen ist der Quellcode für jeden zugänglich und kann entsprechend modifiziert werden.



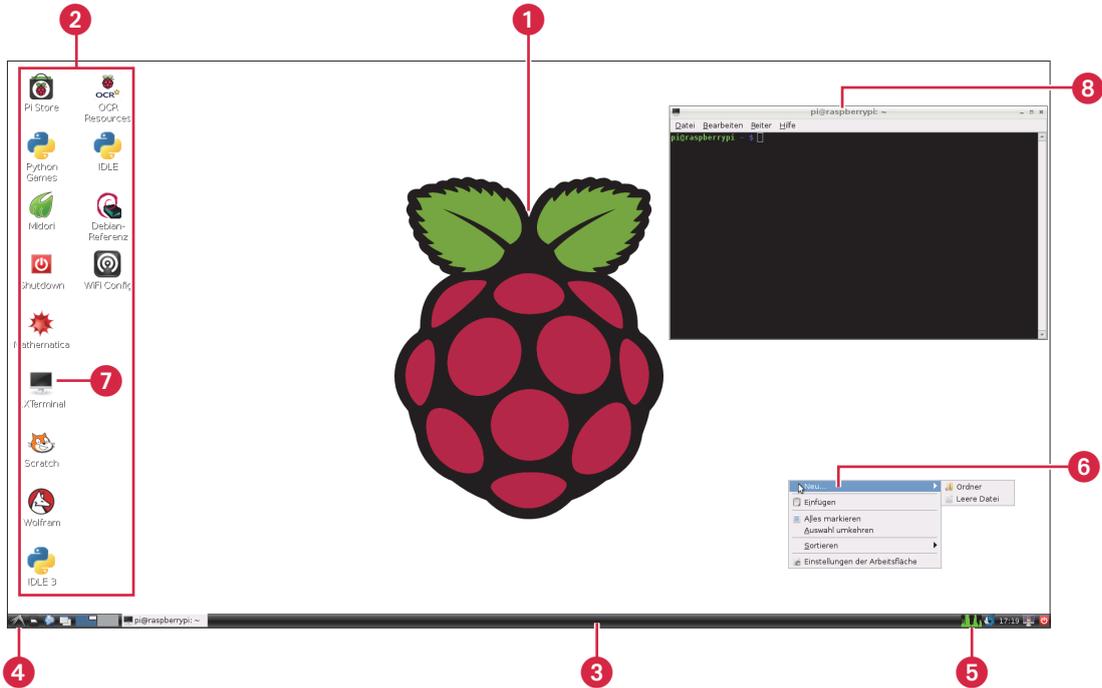
Vorbereitungen für Raspbian

Neben **RasPi**, **SD-Karte** und **USB-Tastatur** benötigen wir zur Nutzung der grafischen Oberfläche auch eine **USB-Maus**. Möglicherweise möchten wir später weitere USB-Geräte wie beispielsweise externe Festplatten oder Drucker anschließen. Da der RasPi aber nur mit zwei USB-Ports daherkommt, sollte auch ein **USB-Hub** mit am Start sein. Hierbei ist darauf zu achten, dass dieser über eine eigene Stromversorgung verfügt, denn die magere Versorgung durch den RasPi stößt schnell an ihre Grenzen.

Wenn wir mit dem RasPi im Internet surfen oder Mails checken möchten, ist eine Verbindung mit dem lokalen Netz erforderlich. Diese können wir per **Ethernetkabel** oder mittels eines **WLAN-Sticks** herstellen. Außerdem benötigen wir natürlich unbedingt einen **Monitor** oder **Fernseher** – idealerweise besitzt das Ausgabegerät einen HDMI-Anschluss. Wenn dies nicht der Fall ist, reicht auch ein Cinch-Kabel oder notfalls ein HDMI-auf-VGA-Adapter wie in Kapitel 2 beschrieben.

Das Aufspielen von Raspbian wurde in Kapitel 2 bereits beschrieben. Sobald die SD-Karte entsprechend vorbereitet ist, schließen wir alle Geräte an den RasPi an, stecken die SD-Karte ein und versorgen den RasPi mit Strom. Beim ersten Starten folgt die altbekannte Konfigurationsroutine. Für den Einsatz als Desktop-PC muss dabei in der Konfiguration unbedingt bestätigt werden, dass der RasPi immer sofort die grafische Benutzeroberfläche starten soll. Auch dies kann in Kapitel 2 nachgelesen werden. Zur Erinnerung: Diese Option befindet sich unter Menüpunkt 3 der Raspi-Konfiguration.

Anschließend folgt der obligatorische Neustart und wir können loslegen. **Hallo, Raspbian!**



Benutzeroberfläche

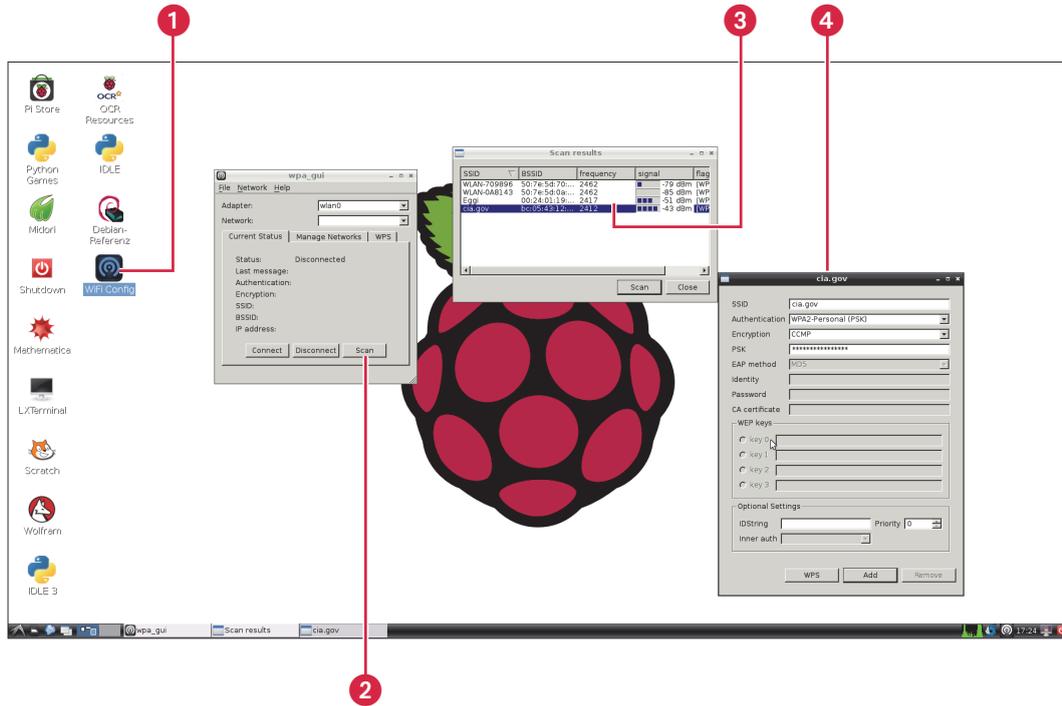
Sobald der RasPi fertig gebootet hat, werden wir vom Desktop begrüßt, der vom Raspberry-Pi-Logo ❶ dominiert wird. Ansonsten findet sich hier unaufgeregte Funktionalität: mehrere Icons am linken Rand der Arbeitsfläche ❷ sowie eine Taskbar am unteren Rand ❸, die unter anderem das Programmmenü ❹ und eine Anzeige für die Prozessorlast ❺ beherbergt, sowie Schaltflächen für alle geöffneten Programme – sofern wir Programme gestartet haben, versteht sich.

Im Prinzip funktioniert hier alles so, wie wir es von anderen Betriebssystemen her kennen: mittels eines Klicks erfolgt eine Auswahl, über einen Doppelklick wird ein Programm via Icon auf dem Desktop gestartet oder eine Datei geöffnet, bei einem Rechtsklick wird ein Kontextmenü ❻ angezeigt. Wenn ein Programm nicht als Icon auf dem Desktop abgelegt ist, finden wir es im thematisch gegliederten Programmmenü unten links ❷.

Beginnen wir mit einem Doppelklick auf das Icon **LXTerminal** ❸. Damit starten wir das Terminal, das unsere Verbindung von der grafischen Benutzeroberfläche zu der auf den ersten Blick furchterregenden Kommandozeile darstellt. Doch keine Angst, mit den Linux-spezifischen Kommandos und dem Umgang mit der Kommandozeile werden wir uns in Kapitel 5 ausgiebig beschäftigen – für den Moment nutzen wir diese nur, um unser System mit Updates zu versorgen. Wenn das Terminal-Fenster geöffnet und ausgewählt ist, geben wir folgendes Kommando ein, das wir ja bereits in Kapitel 2 kennengelernt haben: **sudo apt-get update && sudo apt-get upgrade**.

Anschließend folgt die Bestätigung mittels Eingabetaste. Nun bestätigen wir die Frage, ob wir fortfahren möchten, mit **J** für »ja«, lehnen uns entspannt zurück und beobachten das Update – oder nutzen die Zeit und besorgen uns ein Heißgetränk aus der Küche. Das Update ist fertig, sobald wieder die Eingabezeile (der sogenannte Prompt, siehe Kapitel 5) angezeigt wird.

Widmen wir uns nun dem Fine-Tuning.

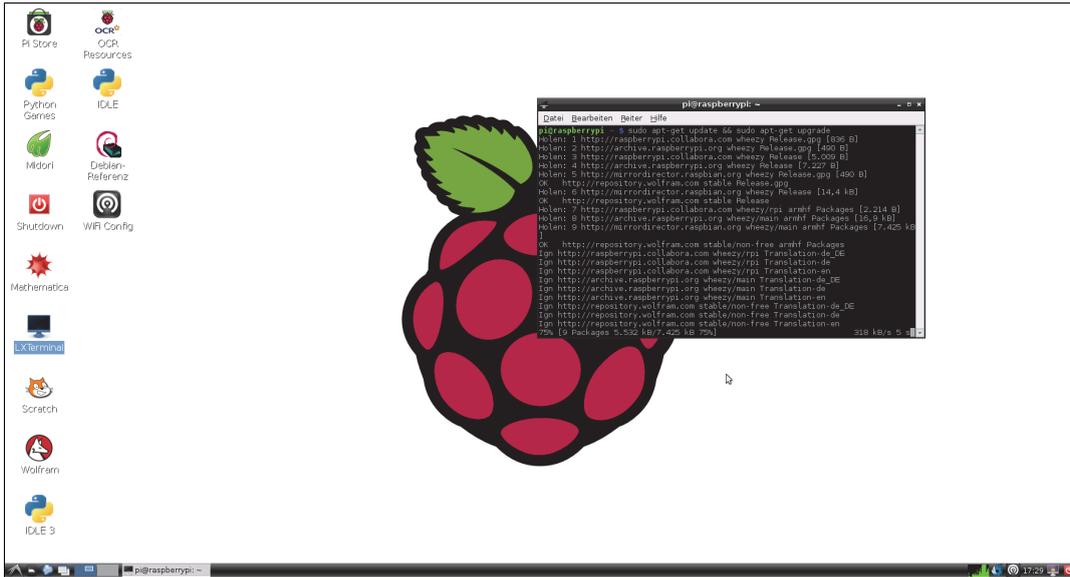


Einstellungsfragen: WLAN & Co.

Im Großen und Ganzen haben wir Raspbian nun bereits ordentlich eingerichtet. Was jedoch auf jeden Fall noch unsere Aufmerksamkeit erfordert, ist das WLAN, falls wir nicht mittels Kabel ins Netz gehen. Denn so findig Raspbian auch sein mag, es kann nicht den Schlüssel zu unserem WLAN erraten. Unser heimisches WLAN ist schließlich kein offenes Netz.

Raspbian hat zum Glück ein kleines Utensil am Start, mit dem sich das Einrichten der WLAN-Verbindung enorm erleichtern lässt. Über das Icon »Wifi Config« auf dem Desktop ❶ können wir alle nötigen Einstellungen vornehmen. Mit einem Doppelklick auf dieses Icon starten wir den Einrichtungsprozess. Im sich öffnenden Fenster namens »wpa-gui« klicken wir auf »Scan« ❷, woraufhin uns in einem weiteren Fenster namens »Scan results« eine Übersicht aller verfügbaren WLANs angezeigt wird. Dort wählen wir unser WLAN mittels Doppelklick an ❸, und es öffnet sich ein weiteres Fenster mit dem Namen unseres WLANs ❹, in dem wir alle nötigen Einstellungen vornehmen können – vermutlich reicht es, unter PSK den Schlüssel einzutragen, alle anderen Einstellungen sollten bereits korrekt sein. Und das war auch schon alles, Raspbian ist nun drahtlos mit unserem WLAN verbunden.

Viel mehr müssen wir gar nicht ändern. Nur wer ein mit NTFS formatiertes Laufwerk anschließen möchte, sollte folgenden Befehl im Terminal ausführen, um sicherzustellen, dass NTFS reibungslos funktioniert: **sudo apt-get install ntfs-3g**. Wer sich nicht sicher ist, ob ein Laufwerk mit NTFS formatiert ist oder nicht, kann diesen Befehl gefahrlos ausführen, auch wenn er nicht erforderlich ist.



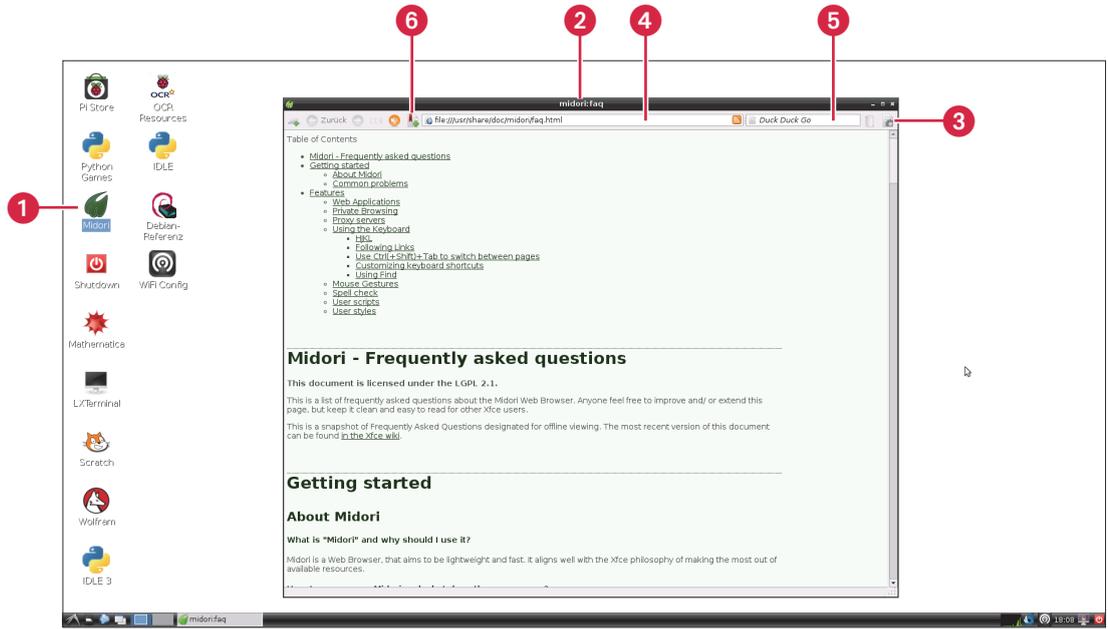
Verbindung zur Kommandozeile: das Terminal

Mit der Kommandozeile werden wir uns in Kapitel 5 noch ausführlich beschäftigen. Doch auch bei der Nutzung der grafischen Oberfläche führt manchmal kein Weg an der Kommandozeile vorbei, wie wir bereits festgestellt haben. Mit der Kommandozeile kommen wir ohnehin in der Regel schneller ans Ziel als über den Umweg einer grafischen Oberfläche. Dies gilt für Linux allgemein, für den Einsatz auf dem mageren RasPi aber doppelt und dreifach. So bieten Textbefehle mehrere handfeste Vorteile:

- Ein Terminal lässt sich deutlich schneller starten als fast alle anderen Programme.
- Befehle sind ohne großes Suchen in den Menüs sofort nutzbar.
- Bei einer Befehlseingabe gibt es keine Missverständnisse.

Im Folgenden werden wir immer wieder Befehle nutzen. Hierbei starten wir das **LXTerminal** durch Doppelklick auf das Desktop-Icon, tippen den angegebenen Befehl ein und bestätigen mit der Eingabetaste, woraufhin Raspbian den eingegebenen Befehl ausführt.

Wer es gar nicht erwarten kann, mehr über die Kommandozeile zu erfahren, kann an dieser Stelle gerne das Kapitel 5 vorziehen. Wirklich notwendig ist es aber an dieser Stelle noch nicht.



Raspbian-Browser Midori

Mit das Wichtigste, was wir heutzutage auf unseren Rechnern starten, ist der Browser. Bei Raspbian heißt der leichtgewichtige Standardbrowser **Midori**. Mittels Doppelklick auf das Midori-Icon auf dem Desktop ❶ wird er gestartet.

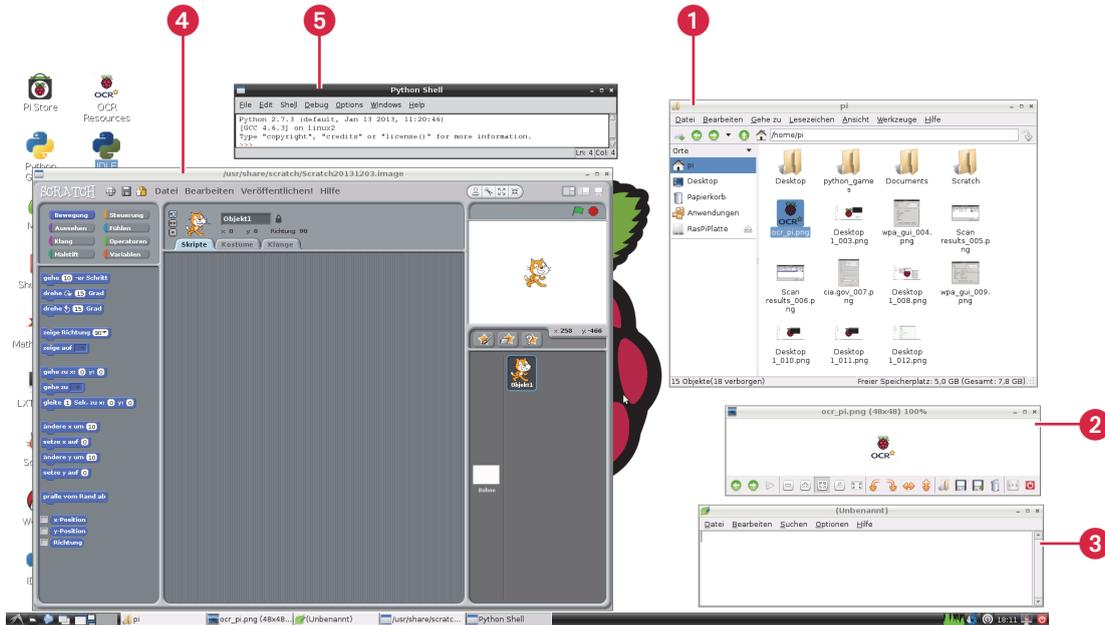
Begrüßt werden wir dann von dem »midori-faq« ❷. Hierbei handelt es sich um eine Startseite, die wir uns vermutlich nicht mehr als einmal ansehen möchten. Über »Einstellungen«, erreichbar über das Zahnradsymbol rechts oben ❸, können wir auch eine andere Seite festlegen – beispielsweise die Suchmaschine Duck Duck Go, die bei Midori ohnehin als Standardsuchmaschine integriert ist und eine nette Alternative zu Google darstellt.

Adressleiste ❹ und Suchfeld ❺ funktionieren wie beim altbekannten Firefox. Den Lesezeichen wurde ein separat von der Adressleiste angefügter Button ❻ spendiert.

Da es sich bei Flash, das beispielsweise zum Abspielen einiger Videos benötigt wird, um proprietäre Software handelt, ist es nicht standardmäßig am Start. Wenn wir Flash in Midori nutzen möchten, greifen wir zur Flash-Alternative **Gnash**. Wir schließen Midori, falls es geöffnet ist, und geben folgendes Kommando im Terminal ein:

sudo apt-get install gnash && sudo apt-get install browser-plugin-gnash

Wenn wir das nächste Mal Midori starten, können wir uns nun auch Flash-Videos ansehen.



Raspbian-Standardprogramme

Das **LXTerminal** haben wir ja bereits kennengelernt. Ein weiteres zentrales Programm ist der **Dateimanager** ❶, mit dem wir unsere Daten beliebig organisieren können. Mithilfe der Schaltflächen auf der linken Seite springen wir zwischen den einzelnen Orten hin und her, und auch weitere angeschlossene Laufwerke werden hier angezeigt.

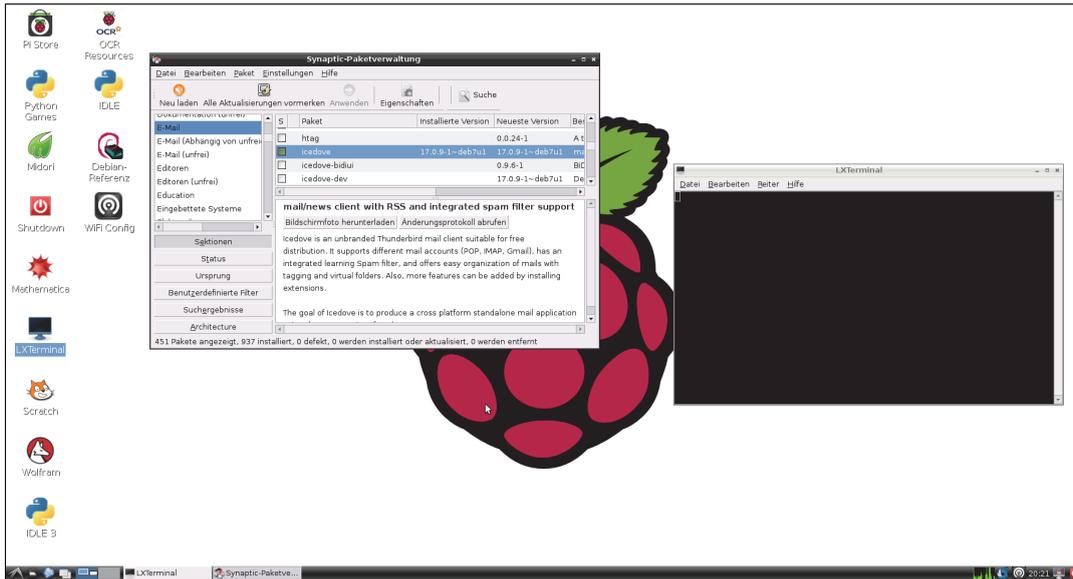
Mit der Dateistruktur unter Linux werden wir uns in Kapitel 5 noch eingehend beschäftigen. Alle unsere »normalen« Dateien befinden sich ohnehin im Benutzerverzeichnis – da der Standardbenutzer »pi« heißt, ist hier auch das standardmäßig zu nutzende Heimatverzeichnis abgelegt.

Der **Bildbetrachter** ❷ kommt zum Einsatz, wenn wir schnell ein paar ... nun ja, Bilder betrachten möchten. Wenn wir im Dateimanager ein Bild mittels Doppelklick auswählen, wird automatisch der Bildbetrachter gestartet. Mit ihm lassen sich auch einfache Änderungen an den Bildern vornehmen.

Für das einfache Verfassen von Texten steht das schmale **Leafpad** ❸ zur Verfügung. Dabei handelt es sich nicht so sehr um ein Textverarbeitungsprogramm im eigentlichen Sinne, sondern vielmehr um einen einfachen Texteditor.

Scratch ❹ und **IDLE** ❺ sind für das Programmieren auf dem RasPi gedacht. Scratch bietet hier eine grafische Oberfläche, bei der mittels Drag and Drop die Programmteile aneinandergereiht werden können. Es ist hervorragend für Programmierneinsteiger geeignet. Bei IDLE und IDLE 3 hingegen handelt es sich um Programmierumgebungen für die Sprache Python (und nein, es ist kein Zufall, es besteht tatsächlich ein Bezug zwischen IDLE und Eric Idle von Monty Python). Standardmäßig kommt auf Raspbian Python 2 zum Einsatz; wer auf das aktuellere Python 3 nicht verzichten möchte, nutzt **IDLE 3** zur Programmierung.

Apropos Python: Hinter den »Python Games« auf dem Desktop verbergen sich bereits vorinstallierte Spiele, die in dieser Programmiersprache geschrieben wurden. Viel Spaß!



Installation weiterer Programme

Die Standardprogramme bringen uns möglicherweise noch nicht ans Ziel. Für einige Aufgaben stehen leistungsfähigere Programme zur Verfügung (immer mit dem Hinweis, dass der RasPi schnell an seine Grenzen stößt), und zusätzliche Software für andere Einsatzgebiete gibt es natürlich auch.

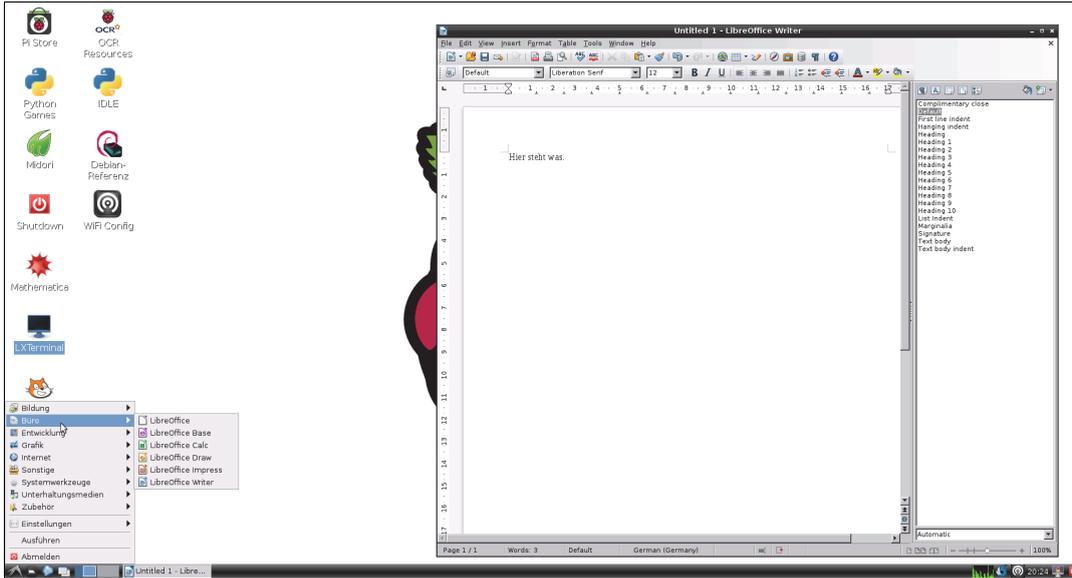
Wie wir diese Software installieren, bleibt uns überlassen. Ohne viel Federlesen lässt sich hier die Kommandozeile in Form des LXTerminals verwenden: einfach das Terminal starten und den Befehl **sudo apt-get install** gefolgt von dem Namen des zu installierenden Programms eingeben, mit der Eingabetaste bestätigen und die Frage nach dem Fortfahren mit für »Ja« beantworten. Bei allen im Folgenden angeführten Programmen ist der am Ende des Befehls zu ergänzende Name komplett in Kleinbuchstaben und bei mehrteiligen Namen zusammen zu schreiben. Der vollständige Befehl zum Installieren des Musikwiedergabeprogramms LXMusik beispielsweise heißt **sudo apt-get install lxmusic**, und das Textverarbeitungsprogramm LibreOffice wird mit **sudo apt-get install libreoffice** installiert.

Einige Programme, beispielsweise die Office Suite LibreOffice, finden sich im **Pi Shop**, der bereits installiert und auch via Icon auf dem Desktop erreichbar ist. Manche Programme kosten Geld, aber andere sind auch kostenfrei, wie beispielsweise LibreOffice.

Wer eine grafische Oberfläche zum Installieren von Programmen wünscht, aber im Pi Shop nicht die gesuchte Software findet, kann Synaptic verwenden, das mit folgendem Befehl im Terminal installiert wird:

sudo apt-get install synaptic

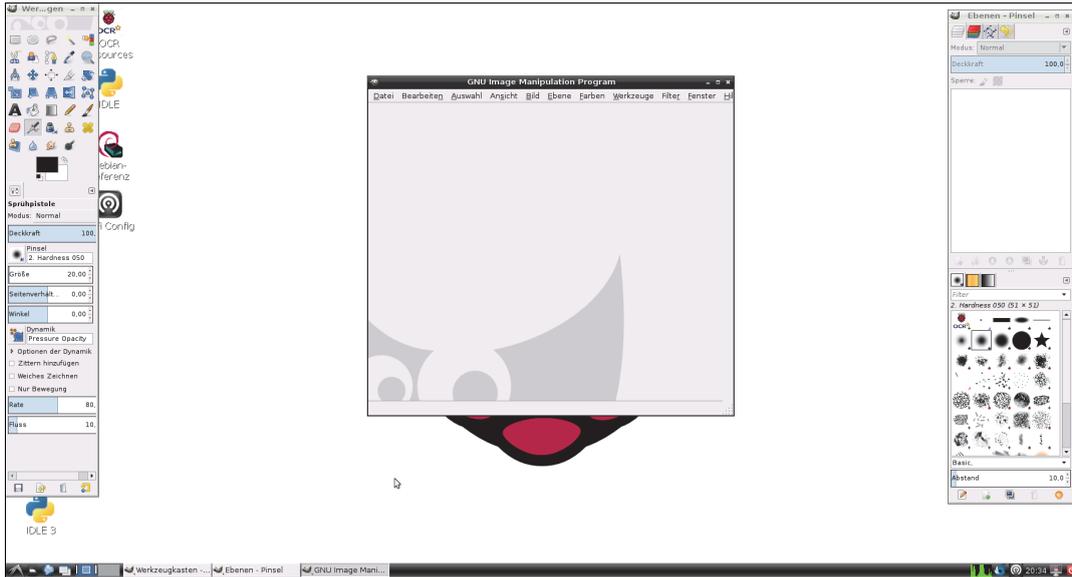
Der altbekannte Hinweis, dass der RasPi auf diese Weise schnell seine Leistungsgrenzen findet, darf leider nicht fehlen – schneller und flüssiger bleibt der Weg über die Kommandozeile.



Softwareauswahl: Office Suite und Textverarbeitung

Die erste Wahl für eine komplette Office Suite mit Textverarbeitung, Tabellenkalkulation, Präsentationsanwendung und mehr ist **LibreOffice**, ein Ableger des momentan noch etwas besser bekannten OpenOffice. Umfangreicher und leistungsfähiger geht es fast nicht – was auf dem RasPi natürlich mit Vorsicht zu genießen ist. Trotzdem kann es sich lohnen, LibreOffice einzusetzen, denn damit lässt sich nicht nur gut arbeiten, sondern es können auch so ziemlich alle bekannten Office-Formate geöffnet werden, allen voran das berüchtigte doc(x)-Format von Microsoft Word. Wenn wir LibreOffice erst einmal installiert haben, steht uns im Programmmenü eine ansehnliche Sammlung an Office-Software zur Verfügung: »Writer« ist die Textverarbeitung und für viele sicher das Herzstück von LibreOffice, bei »Calc« handelt es sich um die Tabellenkalkulation, mit »Impress« erstellen wir Präsentationen, »Draw« hilft uns beim Erstellen von Diagrammen und Skizzen, bei »Base« handelt es sich um eine Datenbankverwaltung und »Math« lässt uns mathematische Gleichungen darstellen.

Wer lieber eine etwas flüssigere Textverarbeitung möchte und auf die anderen Programme verzichten kann, greift stattdessen zu **Abiword**. Dieses Programm ist wesentlich spartanischer gehalten als die Textverarbeitung von Libre Office, was jedoch auf dem RasPi nicht von Nachteil ist.

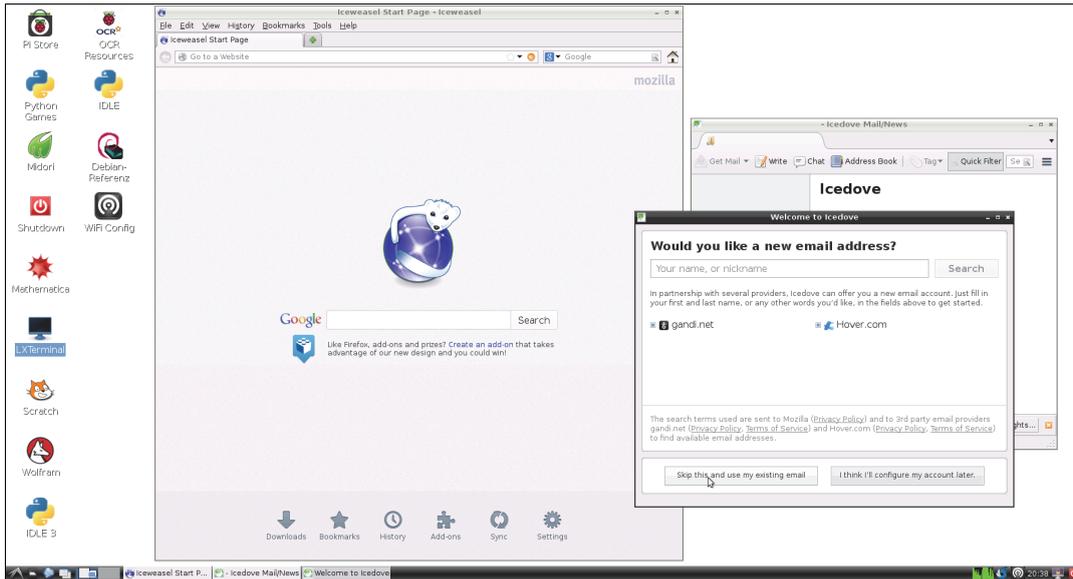


Softwareauswahl: Multimedia auf Raspbian

Wer Bilder mit dem RasPi bearbeiten möchte, findet mit **GIMP** eine umfassende Lösung. GIMP ist sozusagen die eierlegende Wollmilchsau der Grafikbearbeitung und braucht den Vergleich mit Photoshop kaum zu scheuen. Beim Einsatz auf dem RasPi müssen wir hier jedoch mal wieder ein ordentliches Maß an Geduld mitbringen. Falls sich unsere Bearbeitungswut in Grenzen hält, reicht vielleicht schon der bereits installierte Bildbetrachter.

Häufig sollen Bilder auch gar nicht bearbeitet werden, sondern man möchte sie einfach nur betrachten. Zwar ist dies ebenfalls schon mit dem mitgelieferten Bildbetrachter möglich, doch **gThumb** ist da um einiges komfortabler.

Um Musik wiederzugeben, können wir zu **LXMusic** greifen. Hierbei handelt es sich um das Standardmusikprogramm der Desktopumgebung LXDE, die bei Raspbian zum Einsatz kommt (ganz nebenbei ist so auch geklärt, warum das Terminal »LXTerminal« genannt wird). Mit diesem können wir Musik abspielen sowie Playlists anlegen, verwalten und einsetzen. Zugegeben, es ist nicht die schönste Musik-App, aber sie ist recht funktional und passt deshalb hervorragend zum RasPi.

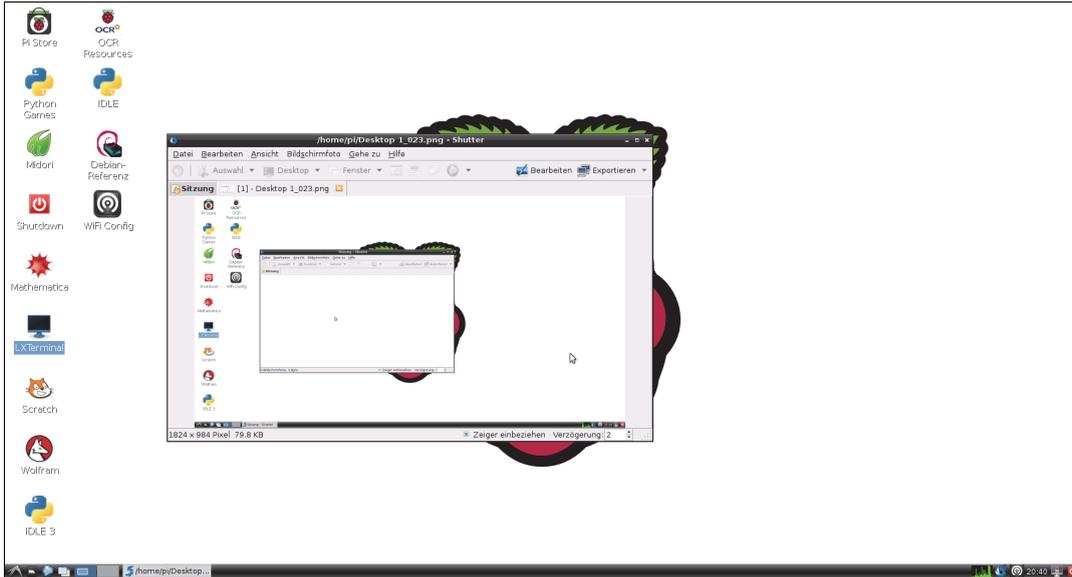


Softwareauswahl: Browser und Mail-Client

Vom heimischen Desktop-PC sind wir vermutlich den Umgang mit Firefox gewohnt. Wer diesen Browser anstelle von Midori verwenden möchte, kann sich **Iceweasel** installieren – hierbei handelt es sich um einen Klon von Firefox, der aus markenrechtlichen Gründen umbenannt wurde.

Zwar ist Midori die ressourcenschonendere Wahl, doch wer partout nicht ohne den Firefox kann (was ich gut verstehe), findet hier adäquaten Ersatz. Prinzipiell funktioniert Iceweasel wie der Firefox, wer also mit diesem klarkommt, hat beim Umstieg keine Probleme.

Analog zum Gespann Firefox/Iceweasel heißt der Ableger von Thunderbird **Icedove**. Mit ihm können wir vom RasPi aus bequem unsere Emails abrufen, sobald diese Möglichkeit entsprechend eingerichtet ist. Wer stattdessen Mails lieber mit dem Browser liest, kann das natürlich auch weiterhin tun, so ersparen wir uns die Installation eines eigenen Mail-Clients.



Softwareauswahl: Diverses

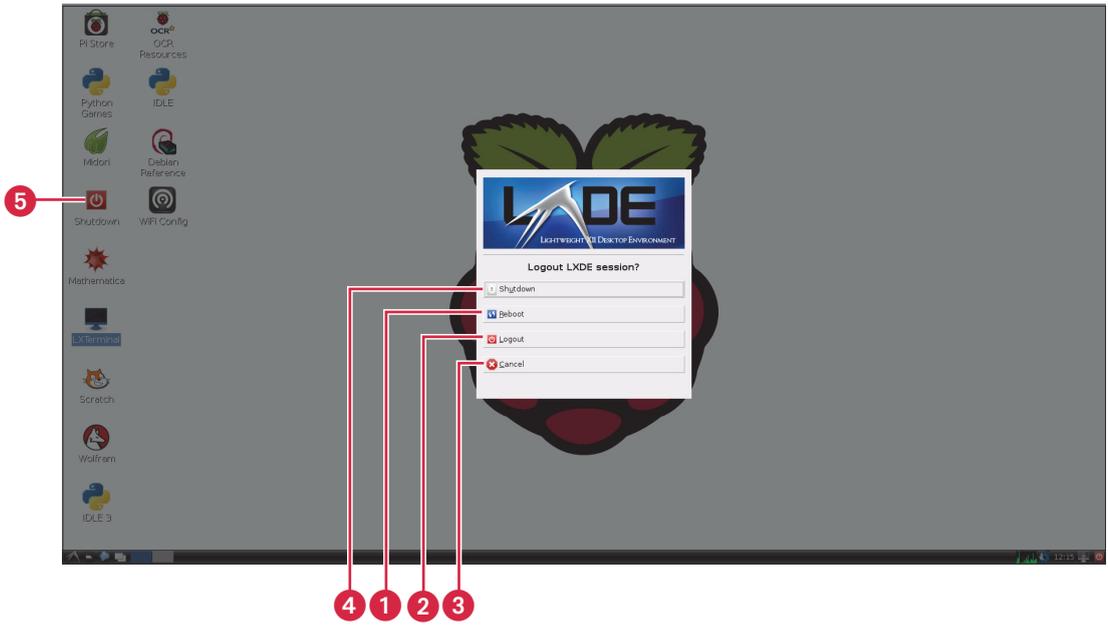
Wer Screenshots anfertigen möchte, ist mit **Shutter** eigentlich gut beraten. Allerdings ist es ein Leistungsfresser und zwingt den RasPi beim ersten Start in die Knie – einmal gestartet funktioniert es aber relativ flüssig. Eine leichtgewichtige Wahl wäre **Scrot**, doch hierbei handelt es sich um ein Kommandozeilentool – Scrot kann zwar anstandslos auch Bilder vom Desktop anfertigen, doch starten und konfigurieren müssen wir es von der Kommandozeile. Dabei stehen zahlreiche Optionen zur Verfügung – die Befehle **scrot -h** und **man scrot** sollen hier erste Anhaltspunkte geben.

Wer nicht nur mit normalen (Pixel-)Bildern, sondern auch mit Vektorgrafiken arbeitet, findet mit **Inkscape** ein umfassendes Programm. Einen Ausflug ins Desktop Publishing ermöglicht **Scribus**. Ein gutes Programm zur Musik- und Tonverarbeitung nennt sich frech **Audacity**.

Wenn wir einen USB-Drucker an den RasPi anschließen möchten, leistet **CUPS** (das »Common Unix Printing System«) gute Dienste. Je nach Drucker erfolgt die Einrichtung anschließend automatisch.

SSH ist eine feine Sache, doch wenn wir aus der Ferne nicht nur auf die Kommandozeile, sondern auf den Desktop zugreifen möchten, benötigen wir auf dem RasPi noch ein Programm wie den **TightVNC Server**. Auf den anderen Rechnern, von denen aus auf den RasPi zugegriffen werden soll, müssen wir dann noch den **TightVNC-Client** installieren und einrichten. Eine Anleitung hierzu findet sich unter http://elinux.org/RPi_VNC_Server – besonderes Augenmerk ist hier auf die Sicherheitsaspekte zu richten, denn TightVNC ist in der Standardkonfiguration nicht abgesichert.

Und damit wäre Raspbian für fast alle Eventualitäten vorbereitet. Da wäre nur noch eine Kleinigkeit ...

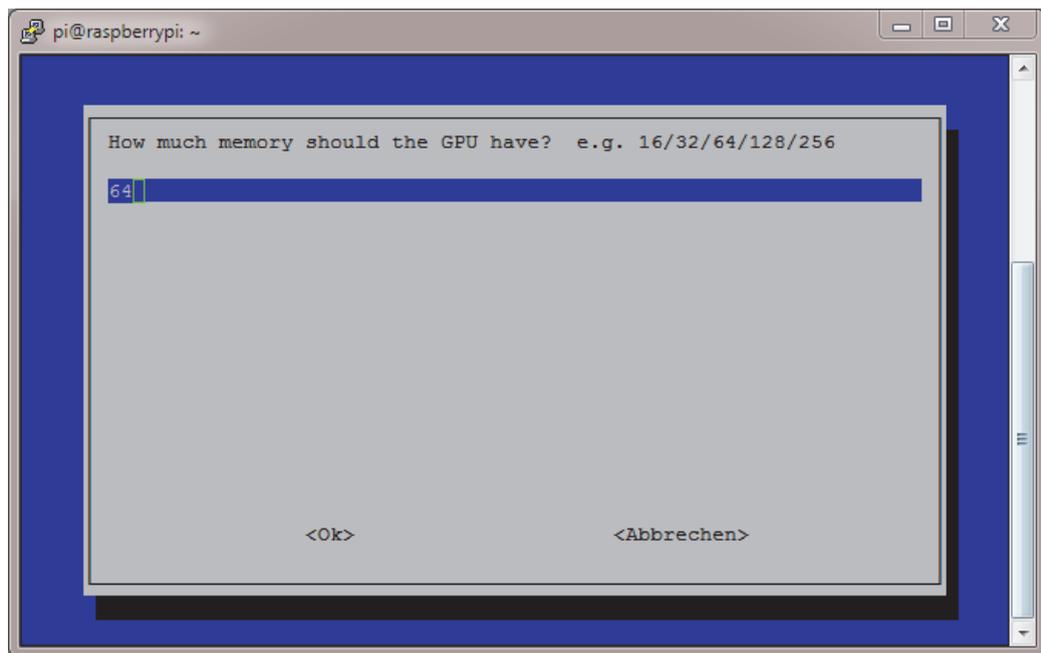


Raspbian herunterfahren

Zwar ist der RasPi sparsam im Verbrauch, doch ab und an wollen wir ihn sicher trotzdem ausschalten. Wer den RasPi einfach so vom Strom trennt, riskiert – beim RasPi wie bei anderen Computern auch – Schäden an den angeschlossenen Speichermedien, also beispielsweise an den internen oder externen Festplatten, oder eben an der SD-Karte im RasPi. Im schlimmsten Fall wird das Speichermedium ganz unbrauchbar. Wir müssen also – erneut: wie bei anderen Computern auch – dem RasPi mitteilen, dass er herunterfahren soll.

Hierzu verfügt die Taskbar rechts unten über ein Symbol zum Herunterfahren. Wenn wir es mit der linken Maustaste anklicken, öffnet sich ein Menü, bei dem mehrere Optionen zur Auswahl stehen. Hier können wir den RasPi neu starten ❶, uns abmelden ❷, durch Abbruch ❸ zurück zum Desktop gelangen oder eben den RasPi herunterfahren ❹. Alternativ kann auch einfach ein Doppelklick auf das entsprechende Desktop-Icon ❺ erfolgen.

Nach getaner Arbeit sollten wir diesen Weg wählen, der RasPi wird es uns danken.



Troubleshooting bei Raspbian

- **Englische Tastaturbelegung:** Falls wir bei der Erstkonfiguration entsprechend unachtsam waren, beglückt uns Raspbian unter Umständen mit der englischen Tastatur, bei der unter anderem »y« und »z« vertauscht und Sonderzeichen an anderen Stellen verortet sind. In diesem Fall starten wir das LXTerminal und geben folgenden Befehl ein: **sudo nano /etc/default/keyboard** (der Slash »/« befindet sich bei der englischen Tastaturbelegung auf dem Minuszeichen). Daraufhin wird der Editor nano mit der Konfigurationsdatei für die Tastaturbelegung gestartet. Neben »XKBMODEL« sollte bereits »pc105« angegeben sein – falls dies nicht der Fall ist, nehmen wir die entsprechende Änderung vor. Unter »XKBLAYOUT« ist vermutlich »gb« angeführt, was wir in »de« ändern. Mit Strg+C beenden wir nano und bestätigen dann mittels , dass wir die Änderungen übernehmen möchten. Zurück auf der Befehlszeile geben wir **sudo dpkg-reconfigure keyboard-configuration** ein (das Minuszeichen findet sich auf dem »ß«), wodurch die Tastaturkonfiguration gestartet wird. Wir wählen nun nacheinander »Generische PC-Tastatur mit 105 Tasten (Intl)«, »Deutsch«, »Der Standard für die Tastenbelegung«, »Keine Compose-Taste« und »Ja«. Manche Bluetooth-Tastaturen lassen sich jedoch auch durch dieses Vorgehen nicht zur Verwendung des deutschen Tastaturschemas bewegen. Hierbei handelt es sich um einen Bug. Der einfachste Weg, diesem Bug aus dem Weg zu gehen, ist der Einsatz einer anderen Tastatur.
- **Falls der RasPi zu langsam arbeitet:** Je nach gestellter Aufgabe macht es Sinn, den Hauptspeicher anders zu bemessen. Hierzu muss zunächst das Terminal neu gestartet, dann **sudo raspi-config** eingegeben und dies durch die Eingabetaste bestätigt werden. Nach Eingabe des Benutzerpassworts (voreingestellt: »raspberrypi«) startet die Konfigurationsroutine, und wir können den Arbeitsspeicher neu festlegen.
- **WLAN funktioniert nicht:** Ist der WLAN-Stick mit dem RasPi kompatibel? Steckt er richtig im USB-Slot? Funktioniert der Stick an anderen Computern? Wurden alle Einstellungen (insbesondere der Schlüssel) korrekt vorgenommen?
- **Schwarze Balken auf dem Bildschirm:** Wir müssen das sogenannte »Overscan« aktivieren, damit der gesamte Bildschirm von Raspbian genutzt wird. Hierzu starten wir im Terminal die Systemkonfiguration mittels **sudo raspi-config**. Unter »Advanced Options« aktivieren wir »Overscan«, schließen die Konfiguration ab und bestätigen den Neustart.