

LINUX, DAS ALLES KANN

Ubuntu als Haupt-Betriebssystem

Verfasst von Alexander Sofianos - +30 697 438 8332

alexander.sofianos@gmail.com

alexander-sofianos.com

Abstract

Distro-Auswahl

Neun Distros wurden teilweise als Multiboot-System auf zwei ThinkPads, W530 und T520, installiert.

Ubuntu hat sich als die goldene Mitte zwischen Linux-from-Scratch und Bloat-Distros durchgesetzt, und rangiert sehr gut in Bezug auf: Stabilität, Font-Wiedergabe, Vielseitigkeit und größte Hilfsgemeinde.

Betriebssystemklonen-Konzept

Ein eigenes Konzept für das Klonen des Betriebssystems (BS) durch den Einsatz von: Rsync, [Persistent-Live-USB mittels Mkusb](#), FSTAB-Konfiguration, [rEFInd-Boot-Manager](#), Grub-Reinstallation und -Updates.

Testen durch mehrfache und langfristige Einsetzung aktualisierter Klone als Vorgabe-BS.

Klon-Portabilität

BS-Klone sind wenig portabel: Eine „[Live CD Customization From Scratch](#)“ für eine eigene Distro-Erstellung wie auch die Virtualisierung einer physikalischen Installation anhand von [Virt-p2v](#) sind geplant.

KVMs

Mehrfache Ubuntu-KVMs laufen gleichzeitig, brauchen wenig CPU und können pausiert parat stehen. Sie werden vernetzt, geklont, synchronisiert und bekommen eine neue Identität.

KVM-CPU-Zyklen steigen bei Video-Anwendung trotz [Virgil3d-Treiber-Einsetzung](#). Neuere Hardware wird benötigt um GPU-Beschleunigung mittels [Intel-GVT-g](#) zu testen.

Windows-KVMs (WVMs) sind umständlich zu erstellen und durch Pool-Änderungen und VM-Umbenennungen, booten sie nicht mehr. Ihr Einsatz ist nicht zwingend nötig, da Linux viele versteckte Fähigkeiten hat.

Vernetzung

Alle Systeme werden problemlos durch SSH vernetzt und anhand der Fahne -X wird der X11-Fenstermanager weitergeleitet. NoMachine-Remote-Desktop wird ergänzend eingesetzt wie auch CIFS und Samba für Windows VMs (WVMs).

Gocryptfs - Rclone - Electrum-Bitcoin-Wallet

[Gocryptfs](#), ein gestapeltes Dateisystem, wird für Verzeichnisse-Verschlüsselung verwendet und [Rclone](#) für Verschlüsselung und Synchronisierung von Cloud-Backends. [Electrum-Bitcoin-Hot-Wallet](#): Sicherheitsmaßnahmen und Vorgehensweisen werden erklärt.

DE-Erweiterungen - Paketverwaltung - PDF-Bearbeitung

BS- und Desktop-Umgebung-Anpassung anhand Erweiterungen und vielseitiger Paketverwaltung. Ergänzende Programme und Tools für PDF-Bearbeitung und für jede Aufgabe.

Vorwort

Der Reiz zur Themenwahl kam durch die angeborene Übereinstimmung mit dem Linux-Spruch „frei wie in Freiheit, nicht wie in freies Bier“.

Am Anfang des Projekts im Jahr 2020 wurde ein Zertifikat in „Introduction to Linux“ von [edX](#) und [The Linux Foundation](#) erlangt, und danach wurde dieses Projekt verwirklicht, von Ideen die jahrelang im Kopf waren.

Danksagung

Lob an alle die teilweise freiwillige Arbeit leisten, und die Zeit nehmen in Foren, Blogs und Distro-Seiten, Wissen über Linux und Open-Source-Apps zu verbreiten.

Inhaltsverzeichnisse

Zusammenfassend

Abstract.....	1
Distro-Auswahl.....	1
Betriebssystemklonen-Konzept.....	1
Klon-Portabilität.....	1
KVMs.....	1
Vernetzung.....	1
Gocryptfs - Rclone - Electrum-Bitcoin-Wallet.....	1
DE-Erweiterungen - Paketverwaltung - PDF-Bearbeitung.....	1
Vorwort.....	2
Danksagung.....	2
Inhaltsverzeichnisse.....	2
Abkürzungen und Glossar.....	6
Einleitung.....	7
Testen und Experimente.....	7
Kapitel-Rangordnung.....	7
Ziele.....	7
Fortsetzungsprojekt.....	7
Literatur-Ressourcen klassifiziert.....	8
Manpages.....	8
Online-Ressourcen.....	8
Methodik und Analyse.....	10
Hardware.....	10
Distrofamilien.....	10
Paketverwaltungssysteme.....	10
Auswahl von Ubuntu als Haupt-BS.....	10
GPT-Partitionierung.....	12
Ubuntu-Paketverwaltung.....	16
Gnome.....	18
Verschiedene Anwendungen.....	20
PDF-Editors.....	24
LibreOffice.....	28
Bereinigung unnötiger Dateien.....	29
Netzwerken.....	31
BS-Backup und -Klon-Erstellung als Konzept des Verfassers.....	34
Virtuelle Maschinen - VMs.....	40
Rsync-Zusammenfassung.....	54
Electrum-Wallet für Bitcoins.....	55
Sicherheitsmaßnahmen und Datenverschlüsselung.....	59
Tar mit GPG.....	70

7z.....	70
Stapel-Umbenennung von Dateien.....	71
Schlussfolgerungen.....	72
Beweggrund.....	72
Befehlszeile-Lernen.....	72
Closed-Source-BSs vermeiden.....	72
Ubuntu-Zuverlässigkeit hat sich bewährt.....	72
Paketverwaltung.....	72
PDF-Bearbeitung.....	72
LibreOffice.....	72
Vernetzung.....	72
BS klonen und aktualisieren.....	73
BS-Portabilität.....	73
KVMs.....	73
Viren - Crypto-Wallets.....	73
Verschlüsselung.....	74
Literaturquellen.....	75

Vollständig

Abstract.....	1
Distro-Auswahl.....	1
Betriebssystemklonen-Konzept.....	1
Klon-Portabilität.....	1
KVMs.....	1
Vernetzung.....	1
Gocryptfs - Rclone - Electrum-Bitcoin-Wallet.....	1
DE-Erweiterungen - Paketverwaltung - PDF-Bearbeitung.....	1
Vorwort.....	2
Danksagung.....	2
Inhaltsverzeichnisse.....	2
Abkürzungen und Glossar.....	6
Einleitung.....	7
Testen und Experimente.....	7
Kapitel-Rangordnung.....	7
Ziele.....	7
Fortsetzungsprojekt.....	7
Literatur-Ressourcen klassifiziert.....	8
Manpages.....	8
Online-Ressourcen.....	8
Die größeren Foren und Gemeinde.....	8
Wiki.archlinux.org.....	8
Office.....	8
Kubuntu.....	8
VMs.....	8
Verschlüsselung und Crypto-Wallet.....	9
Tutorial-Blogs.....	9
Methodik und Analyse.....	10
Hardware.....	10
Distrofamilien.....	10
Paketverwaltungssysteme.....	10
Auswahl von Ubuntu als Haupt-BS.....	10
Font-Wiedergabe.....	10
Schwierigkeitsgrad der BS-Verwendung.....	11
Zuverlässigkeit.....	12
Kubuntu.....	12
Mint.....	12

Ubuntu.....	12
Canonical.....	12
GPT-Partitionierung.....	12
Warum GPT und nicht LVM?.....	12
Partitionierungs-Konzept.....	12
T520-Thinkpad Festplatte und externe Festplatte mit dazugehörigen BSs.....	13
W530-Thinkpad Festplatte.....	14
Fstab angepasste Einträge.....	15
Ubuntu-Paketverwaltung.....	16
Repos.....	16
Synaptic.....	16
Snap.....	17
PPA - Personal Package Archive - Persönliches Paket-Repo.....	17
Deb-Dateien Paketmanagement.....	17
Ubuntu-Software.....	18
Gnome.....	18
Erweiterungen.....	19
Verschiedene Anwendungen.....	20
DE bestimmte Programme.....	20
Dateimanager.....	20
Texteditor.....	22
PDF-Editors.....	24
Okular.....	24
Master-PDF-Editor und Qoppa-PDF-Studio.....	24
Linux-Acrobat abgeschafft.....	25
Foxit.....	25
LibreOffice-Draw als PDF-Editor.....	26
Anfängliche Probleme.....	26
Konfiguration für PDF-Dokumente.....	26
Minuspunkte.....	27
Pdf2odt.....	27
Arranger.....	28
Pdf-Fazit.....	28
LibreOffice.....	28
Vorlagen.....	28
Writer - der Mangel der geteilten Ansicht.....	29
Bereinigung unnötiger Dateien.....	29
Übriggebliebene Konfigurationsdateien entfernen.....	29
Unnötige Pakete mit autoremove löschen.....	30
Daten im Cache löschen.....	30
Daten im Snap-Cache löschen.....	30
Festplattenbelegungsanalyse.....	30
Netzwerken.....	31
SSH.....	31
Rsync über SSH.....	32
Remote-Desktop.....	32
Remote-Desktop gegen ssh -X.....	33
BS-Backup und -Klon-Erstellung als Konzept des Verfassers.....	34
Rsync.....	34
Woher soll der Befehl ausgeführt werden.....	35
Persistent-Live-USB.....	35
Wo sollen die Klone erstellt und aktualisiert.....	36
UUIDs in FSTAB ändern.....	37
Booten mit dem rEFInd-Bootmanager von Roderick Smith.....	37
Grub-Bootloader für unsere Arbeitspartition installieren.....	39
Portabilitätsbeschränkungen von Ubuntu-Installationen.....	39
Installationen externer Festplatten.....	40
Authentifizierungsfehler an der Bildschirmsperre.....	40

SSH - Gebrochene Pakete.....	40
Installationen auf USB-Sticks.....	40
Den persönlichen Ordner in eine andere Installation kopieren.....	40
Virtuelle Maschinen - VMs.....	40
Leistungsvergleich - KVM gegen VirtualBox.....	40
KVM gegen Xen.....	41
Performance Evaluation of Xen, KVM, and Proxmox Hypervisors.....	41
KVM-Vorinstallationshandlungsweisen.....	41
KVM-Installation.....	41
Standard-Pool-Speicherstelle ändern.....	42
VMs-erstellen und -bearbeiten.....	43
Gasterweiterungen.....	44
Spicetreiber.....	44
VirtIO-Treiber.....	44
Gasterweiterungen für Windows-Gäste.....	44
Hohe CPU.....	44
Systemweit CPU-Zyklen.....	44
Videobezügliche CPU-Zyklen.....	45
VM-Netzwerken.....	46
Linux.....	46
Windows-VMs.....	48
Virtuelle Maschinen kopieren - updaten - umbenennen.....	49
VM klonen.....	49
VM löschen.....	50
VM aktualisieren.....	50
VM umbenennen.....	50
Benutzername ändern.....	50
Hostname ändern.....	51
VM-Name ändern.....	51
VM-Sicherungskopien.....	51
VM-Verwaltung und -Ressourcen.....	51
Probleme mit Windows-VMs.....	52
Speicher.....	52
VMs als Haupt-BS gegen Distroerstellung.....	53
Distroerstellung.....	53
Virtualisierungslösung.....	54
Virtualisierung einer physischen Installation.....	54
Netzwerk-Synchronisierung der VMs.....	54
Fazit: Distroerstellung gegen Virtualisierung.....	54
Rsync-Zusammenfassung.....	54
BS synchronisieren.....	54
EXT-Daten synchronisieren.....	54
EXT-DATEN über SSH synchronisieren.....	54
QCOW2-VD-Datei synchronisieren.....	54
NTFS-Daten synchronisieren.....	55
Ein Beispiel mit EXT-Daten.....	55
Electrum-Wallet für Bitcoins.....	55
Überprüfen der Schlüsselsignaturen.....	56
Optionale Abhängigkeiten fehlen.....	57
Cryptodomex.....	57
Zbar-tools.....	57
Startwert oder Seed.....	57
Privater Hauptschlüssel oder Master Private Key (MPK).....	57
Native-Segwit gegen Legacy.....	58
Transaktionsgebühren dynamisch gestalten.....	58
Wallets verschiedener Software erstellen.....	59
BTC Börsenplätze als Nebenbemerkung.....	59
Sicherheitsmaßnahmen und Datenverschlüsselung.....	59

Rclone.....	59
Client-ID erstellen.....	60
Remotes.....	61
Crypt.....	63
Duplikate.....	65
Gocryptfs.....	65
Daten-Korruption.....	69
Tar mit GPG.....	70
7z.....	70
Stapel-Umbenennung von Dateien.....	71
Schlussfolgerungen.....	72
Beweggrund.....	72
Befehlszeile-Lernen.....	72
Closed-Source-BSs vermeiden.....	72
Ubuntu-Zuverlässigkeit hat sich bewährt.....	72
Paketverwaltung.....	72
PDF-Bearbeitung.....	72
LibreOffice.....	72
Vernetzung.....	72
BS klonen und aktualisieren.....	73
BS-Portabilität.....	73
KVMs.....	73
Windows-VMs.....	73
KVM-GPU-Beschleunigung.....	73
Viren - Crypto-Wallets.....	73
Verschlüsselung.....	74
Rclone.....	74
Gocryptfs - GPG.....	74
Literaturquellen.....	75

Abkürzungen und Glossar

BS = Betriebssystem

Confs = Konfigurationen

DE = Desktop Environment - Desktop-Umgebung

Distro = Linux-Variante - Distribution

FOSS = Frei- und Open-Source-Software

Konf- oder Conf-Datei = Konfigurationsdatei

KVMs = Kernelbasierte Virtuelle Maschinen

Manpages = Manual-Pages - Handbuchseiten

PPA - Personal Package Archive - Persönliches Paket-Repo

Repos = Repositorys

SSH = Secure Shell

UVMs = Ubuntu Virtuelle Maschinen

VMs = Virtuelle Maschinen

WVMs = Windows Virtuelle Maschinen

Einleitung

Testen und Experimente

Das vorhandene Projekt ist ein Prozess um Linux so zu installieren, anzupassen und bereichern, dass es ein ganz individuelles Konzept darstellt, um alles zu bewältigen.

Diese Forschung bezieht sich auf eigenes Testen durch Experimente, nach kritischer Betrachtung der Bibliographie.

Kapitel-Rangordnung

Der Kern des Projekts liegt in den letzten Kapitel, und obwohl manche Leser wollen nur diese betrachten, die Kapitel-Rangordnung wurde so festgelegt, um einen natürlicheren Informationsaufbau durch die Projektstufen zu ermöglichen und eventuelle Zusammenhänge leichter zu erkennen.

Nachstehend werden aber die Ziele der Kapitel reihenfolgend vom Kern zum Rand aufgelistet um die schwerwiegendere Abschnitte zu unterstreichen.

Ziele

- BS klonen und updaten als eigenes Rsync-Konzept
- Portabilitätsbeschränkungen der BS-Klone erforschen
- KVMs einsetzen, klonen, updaten, umbenennen und Windows-KVMs als Plan-B-Test
- Partitionierungs-Konzept um alles getrennt zu speichern
- Distro-Auswahl durch Multiboot
- Rclone: Cloud-Speicher-Backends verschlüsseln und synchronisieren.
- Gestapeltes, kryptographisches Gocryptfs-Dateisystem und Gpg mit Tar lokal einsetzen
- Einfach und schnell alles mit SSH und RD vernetzen
- Electrum-Bitcoin-Wallet: Installation, Transaktionen, Sicherheitsvorkehrungen
- Paketverwaltung mit einer Vielfalt von Optionen
- Font-Wiedergabe verbessern
- Gnome-Anpassung durch Erweiterungen
- Lösungen für die PDF-Problematik finden
- LibreOffice-Mängeln überwinden
- Eine Anwendungen für jede Aufgabe parat haben
- Das System von unnötigen Dateien befreien

Fortsetzungsprojekt

Hervorgetretene Anliegen die vom Ausmaß der erforderlichen Arbeit und Hardware nicht mehr im Rahmen des vorliegenden Projekts bewältigt werden könnten, wurde betreffend nur die Bibliographie überblickt mit dem Vermerk dass sie als Fortsetzungsprojekt geplant sind.

Betreffend die wichtigsten Punkte sind: Distro-Erstellung, KVM-GPU-Beschleunigung und Virtualisierung einer physischen Installation.

Literatur-Ressourcen klassifiziert

Linux ist „im Bau befindlich“. Es muss ständig gegoogelt werden um Puzzlestücke zu sammeln die oft gar nicht zusammenpassen.

Es gibt keine Anleitung in der alles über ein BS oder Programm steht.

Manpages

In den Manpages sind die meisten Anwendungen gut dokumentiert. Eine Ausnahme ist ffmpeg, da vieles nicht dokumentiert wird, man kann viele Optionen auf ffmpeg.org auch schlecht finden.

Hierzu sind Blogs oder auch Youtube-Videos oft die einzigen Quellen. Ghostscript ist auch ein Fall wo Infos in den Manpages nicht komplett sind.

Online-Ressourcen

Die folgende hinweisende Liste versucht die Landschaft der unzähligen Quellen die beraten wurden zu schildern.

Die meisten Infos befinden sich in Frage-Foren, Foren, Blogs und Distro-Seiten.

Die größeren Foren und Gemeinde

Die größten Frage-Foren sind askubuntu.com und seine Verwandtforen: serverfault.com, superuser.com, stackoverflow.com und stackexchange.com.

Bezüglich Ubuntu und Gnome sind auch folgende Quellen vorhanden: help.ubuntu.com, ubuntu.com, ubuntuforums.org und wiki.ubuntu.com und extensions.gnome.org, bugs.launchpad.net

Reddit.com ist eine große Gemeinschaft mit zahlreichen Linux-Infos und github.com ist das Entwickler-Heim von Open-Source-Anwendungen.

Wiki.archlinux.org

Das ist ein Blog mit oft sehr präzise Infos und Anleitungen über komplizierte und exotische Themen.

Office

Für LibreOffice, ask.libreoffice.org ist sehr hilfreich wie auch documentation.libreoffice.org, extensions.libreoffice.org, help.libreoffice.org, libreofficehelp.com und wiki.documentfoundation.org

Docs.gimp.org und imagemagick.org bieten Hilfe für Bild-Dateien.

Kubuntu

Um Kubuntu zu testen, Infos finden sich unter kubuntu.org, community.kde.org, kde.org und userbase.kde.org.

VMs

Die größeren Foren sind sehr gut solange es um beliebte Themen geht. KVM- oder Xen-Virtualisierung sind etwa exotische Fächer die nicht viel in Foren diskutiert werden.

Die Distroseiten redhat.com und docs.fedoraproject.org, sowie das wiki.archlinux.org sind die Vorgabe-Quellen in dieser Hinsicht und anschließend spezifische Seiten wie libguestfs.org, libvirt.org, linux-kvm.org, spice-space.org und virgil3d.github.io.

Fedora und Redhat stellen außerdem zu Verfügung nützliche Treiber und Guest-Addons für Windows-Gäste.

Für Xen und Proxmox: xenproject.org, wiki.xenproject.org, citrix.com, proxmox.com

Hierzu wurde auch eine wissenschaftliche Arbeit „[Performance Evaluation of Xen, KVM, and Proxmox Hypervisors](#)“ betrachtet.

Hardware-Eigenschaften sollen auch recherchiert werden beispielsweise unter ark.intel.com für CPUs und unter crucial.com für Speicher.

Verschlüsselung und Crypto-Wallet

Rclone.org bietet detaillierte Anleitungen und ein sehr hilfreiches Forum unter forum.rclone.org.

Gocryptfs unter nuetzlich.net/gocryptfs. Der Entwickler ist sehr hilfsbereit und antwortet auf Fragen unter github.com/rfjakob/gocryptfs/issues.

Für den Crypto-Wallet: electrum.org, electrum.readthedocs.io, bitcointalk.org, bitcoin.stackexchange.com, en.bitcoin.it, bitcoinops.org, kraken.com

Tutorial-Blogs

Sehr gute Blogs mit Tutorials sind: fosslinux.com, geeksforgeeks.org, howtoforge.com, howtogeek.com, itsfoss.com, kifarunix.com, linuxconfig.org, linuxhint.com, linuxinsider.com, linuxize.com, linuxuprising.com, ostechnix.com, techrepublic.com, vitux.com.

Methodik und Analyse

Hardware

Zwei ThinkPads: W530 und T520. Zwei externe Toshiba-Festplatten und verschiedene USB-Sticks.

Distrofamilien

Linux-Distros können in drei Hauptfamilien geordnet werden: Debian, OpenSuse und RedHat. Weitere bemerkenswerte Familien sind Archlinux und Gentoo.

Ubuntu ist debianbasiert und viele Distros sind wiederum ubuntu-basiert.

Kommandozeile-Befehle sind im allgemeinen unterhalb der verschiedenen Familien kompatibel, mit der Ausnahme des Paketverwaltungssystems.

Paketverwaltungssysteme

Distro	Paketverwaltung
Debianbasiert	Apt
RedHat	Yum
Fedora	Dnf
OpenSuse	Zypper
Archlinux	Packman
Gentoo	Portage

Weitere Erläuterungen unter

https://en.wikipedia.org/wiki/List_of_software_package_management_systems

Auswahl von Ubuntu als Haupt-BS

Aufgrund des Linux-Seminars wurden anfänglich, außer Ubuntu, auch OpenSuse und CentOS installiert.

Nach dem Seminar, auch folgende Distros wurden installiert: Deepin, Manjaro, MX, Debian, Fedora und Kubuntu.

Mint wurde als bereits installiertes System getestet.

Font-Wiedergabe

Ohne viel Konfiguration, die Schriftwiedergabe lässt bei den meisten Distros viel zu wünschen übrig. Am Projektanfang, wurde einige Zeit mit Fedora um Verbesserung versucht, ohne zufriedenstellenden Erfolg.

Jedenfalls, in Ubuntu sehen wir von Anfang an, eine bessere Schriftwiedergabe, im Vergleich mit allen anderen Distros die installiert wurden, und in Übereinstimmung mit den allgemeinen Meinungen von Foren und Blogs.

Schrift-Rendering sieht am schlimmsten in Browsern aus, weil viele Webseiten, MS-Schriftarten verwenden, die nicht gut genug durch entsprechende Linux-Schriftarten ersetzt werden. Der erste Schritt hierzu ist das Paket [ttf-mscorefonts](#) zu installieren mit

```
$ sudo apt install ttf-mscorefonts
```

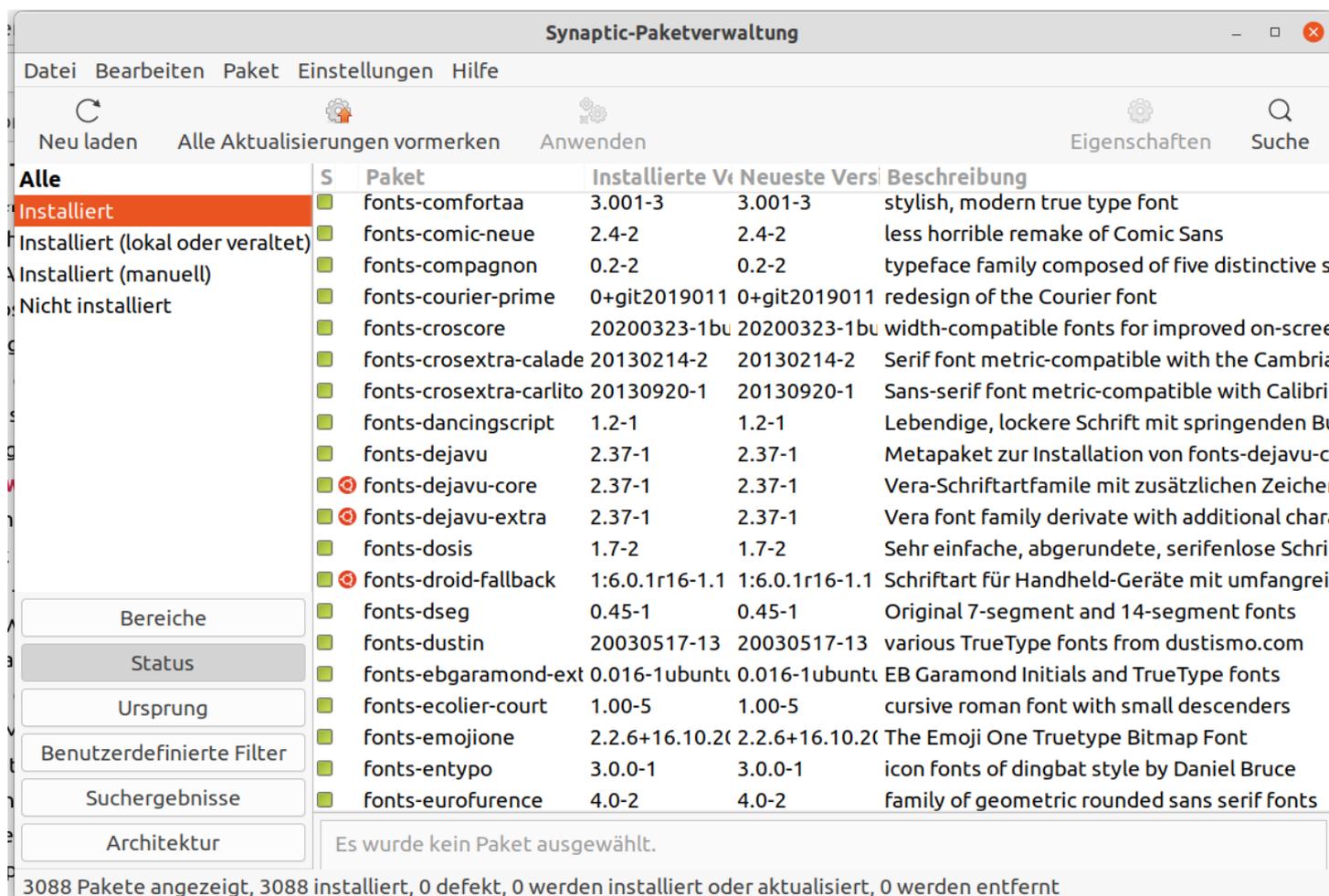
und in Firefox „Seiten das Verwenden von eigenen statt der oben gewählten Schriftarten erlauben“ nur fallweise aktivieren.

Da müssen auch viele zusätzlich Schriftarten installiert. [Texgyre](#) und [Roboto](#) sind ein gutes Beispiel für Fonts die wir in Browsern brauchen, die nicht vorinstalliert sind. Je mehr Schriftarten

installiert werden, desto erhöhen sich die Chancen, dass die Schriftart die wir lesen wollen, oder deren bestmöglichen Ersatz, verfügbar ist.

Zu diesem Zweck ist das [Fontcofig-Config-Paket](#) automatisch installiert: „Konfiguration Fontconfig ist eine Bibliothek zur Konfiguration und Anpassung von Schriften, die nicht vom X Window System abhängt. Sie wurde entwickelt, um Schriften innerhalb des Systems aufzufinden und gemäß der Anforderungen auszuwählen, die von Anwendungen gestellt werden.“ (Von der Paketbeschreibung.)

Für eine mehrfache Installation von kleinen Paketen wie Schriftarten, ist [Synaptic](#), die zuerst installiert werden muss, die bessere Lösung, und das war auch der einzige Fall im Projekt dass das Apt-Frontend zur Befehlszeile bevorzugt wurde.



Eine angepasste [Fonts.conf-Datei](#) kann auch eingesetzt werden. Das wurde in Fedora unternommen, aber in Ubuntu und Kubuntu war es nicht nötig. [Canonical](#), die Firma hinter Ubuntu, hat sogar die Entwicklung eigener Schriftarten finanziert, die sehr gelungen sind.

Schwierigkeitsgrad der BS-Verwendung

Für BSs wie CentOS und Debian muss man viel mehr Zeit widmen.

Bei CentOS die einzige Installations-ISO die funktionierte, war 11GB groß. Vom GUI fehlen auch viele „Selbstverständlichkeiten“.

Bei Debian, die übliche Installations-ISO, enthält nicht mal WLAN-Routertreiber. Man muss dann eine andere nehmen, die nicht nur „Super-Open-Source-Treiber“ enthält. Es fällt gleich auf, dass mit dem GUI und der Schriftartwiedergabe viel unternommen werden muss.

Zuverlässigkeit

Als Debian installiert wurde, auf einer nächstliegenden Partition war Fedora installiert. Nach der Debian-Installation, verschwand von Fedora jeder Hinweis von WLAN, also nichts unter den Einstellungen. Gleichzeitig waren auf der gleichen Festplatte auch Ubuntu, OpenSuse und Centos installiert und war ihnen nichts passiert. Darüber, dass in Fedora plötzlich das WLAN verschwindet, gibt es viele Ergebnisse beim Googeln.

Andererseits, Ubuntu erkennt oft WLAN-Router die andere Distros nicht erkennen.

Kubuntu

KDE ist eine Desktop-Umgebung, die für Anwender die von Windows kommen, angenehmer als Gnome sein kann. KDE ist aber nicht so ausgereift wie Gnome. Das ist der Grund dafür dass viele Distros, Gnome als Vorgabe-DE verwenden.

Für manche, KDE wirkt ein wenig wie Bloatware, das war aber hier nicht das Problem. Manche Programme und BS-Funktionen wie auch die Zuverlässigkeit des gesamten Systems, stehen nicht auf einer Stufe mit Ubuntu.

Das Schlimmste war das LibreOffice. Die Icons sind zu groß und mit falscher Skalierung. Man kann sie zwar deaktivieren aber nicht von überall.

Mint

Diese Distro hat wirklich sehr viel Bloat, und bringt sogar ein paar Konzepte mit, die wir nur von Windows kennen, wie z.B. dass der Benutzer keine BS-Updates ausschließen kann.

Ubuntu

Nach vielen Versuchen, die endgültige Wahl des Projekts fiel auf Ubuntu mit der Vorgabe-Gnome-Desktop-Umgebung.

Canonical

Bei manchen Distros existiert ein Zusammenhang mit einer Firma wie im Fall von RHEL, CentOS und Fedora, mit RedHat, so beim Ubuntu, mit der Firma Canonical.

Das ist nicht bedenklich, da Ubuntu Open-Source ist, und Canonical hauptsächlich Support verkauft.

GPT-Partitionierung

Warum GPT und nicht LVM?

Im Projekt wurde GPT mit Ext4-Partitionen verwendet.

LVM hat im Vergleich einige Vorteile, ist aber viel komplexer, weist eine höhere Gefahr von Datenkorruption und -verlust auf und hat zusätzliche Streitpunkte: Es wird deshalb in einem künftigen Projekt getestet werden.

Partitionierungs-Konzept

Wenn wir frei experimentieren wollen, ohne dass wir das BS ein zweites Mal von Grund auf installieren und konfigurieren müssen, sind viele bootfähige Klone eine sehr gute Lösung.

Um diesen Ansatz nicht unnötig zu komplizieren wurden keine einzelnen Partitionen für ein Home-Verzeichnis verwendet, sondern separate Partitionen für persönliche Daten.

Am W530, eine WD10JPLX Festplatte mit 1TB Speicher, wurde in 18 Partitionen zerteilt. Die ersten vier waren anfangs Windows-10 spezifisch, wurden aber nie dafür verwendet, weil im Laufe des Projekts das Interesse für Windows nachließ, so dass sie nur in VMs verwendet wurden.

Windows-10-Gäste waren als Plan-B gedacht, falls man unbedingt ein Programm braucht, das nur unter Windows vorhanden ist.

Dieser Ansatz wurde schließlich auch abgeschafft, für Gründe die weiter unten allmählich erläutert werden.

Zwei externe 1TB groß Toshiba-Festplatten sind für Backups vorhanden, aber auch für externe Ubuntu-Installationen eingesetzt.

T520-Thinkpad Festplatte und externe Festplatte mit dazugehörigen BSs

Der T520 wurde ähnlich wie der W530 partitioniert, aber mit einer 320GB-Festplatte.

Die Sda-Festplatte unten, stammt vom T520-ThinkPad, und die Sdb ist eine Toshiba externe Festplatte. Beide Festplatten wurden ähnlich partitioniert.

Die zwei ersten Partitionen sind ESP, also **EFI-System-Partitionen**, und es sind deswegen zwei, falls eine Distro eine eigene Partition für den Bootloader benötigt. **Fstype** ist **Vfat**, wenn eine Windows-Installation auch anwesend wäre, sonst könnten die **ESPs** auch mit **Ext4** formatiert werden.

Kubuntu, kubuntu-clone und kubuntu-test, sind bootfähige Klone, so angenommen ein BS unbehebbarer Fehler aufweist, kann die nächste Partition gebootet werden.

Auf den zwei letzten Partitionen sind OpenSuse und CentOS installiert.

```
ki@dusk: /mnt
ki@dusk:mnt$ lsblk -fm
NAME      FSTYPE LABEL          UUID                                 FSAVAIL FSUSE% MOUNTPOINT  SIZE OWNER  GROUP MODE
sda
--sda1   vfat   ESP1           53EB-46AC
--sda2   vfat   ESP2           5047-A658
--sda3   swap                                c021b07a-dfe5-499e-9593-77506edd4d47
--sda4   ext4   data           65b1286a-8239-43ca-9535-1add4caeb3bf
--sda5   ext4   ubuntu_lts2    c105d7c4-f0bd-4352-8655-7be80ad79ed4
--sda6   ext4   kubuntu       52ae2f0d-3964-4e3b-b9e4-193db2f966e6
--sda7   ext4   kubuntu-clone cf6212f3-0b32-46c9-9436-351559c48e23
--sda8   ext4   kubuntu-test  9a55f5fa-9721-4522-901a-47dccdcc541d
--sda9   ext4   opensuse      2c5010cb-6206-4a79-a1ea-59f67a54beab
--sda10  ext4   centos        4276f8c7-d406-40f5-887f-5e52873c1a23
sdb
--sdb1   vfat   ESP1           6716-EB8B                1014,2M  1% /boot/efi
--sdb2   vfat   ESP2           676C-0581
--sdb3   swap                                9ddb6672-7eb3-4922-8a26-a8e16d293962 [SWAP]
--sdb4   ntfs   datant        71712E6D5E305E59
--sdb5   ext4   dataei       dc1967fb-9f13-43ca-8ba0-907d2f185de2
--sdb6   ext4   vmisc3       e822d5b5-472f-43a2-abaf-95d7cde38cea
--sdb7   ext4   7ubui1       331505cb-c857-43f2-beda-569da3563b8a  44,6G   16% /
--sdb8   ext4   8ubui2       a08f3868-d6f2-4245-a3c6-0eea4d78e530
--sdb9   ext4   9ubui3       b6e84abf-c934-4804-9648-df19b9c64c6b
--sdb10  ext4   10ubuc2      48c8f16c-1ff2-4e5a-9ffc-b0d9f543f885
--sdb11  ext4   11ubuc5      96c4af41-5924-4726-a3ae-91868ce025ac
--sdb12  ext4   12ubuc6      738b36a9-49c2-402a-8fd2-cae31086d5eb
--sdb13  ext4   13kubui1     2cacc45-8dda-49a1-a691-5f5fc7bf0cd1
--sdb14  ext4   14kubui2     80e11bc1-e368-4dd7-87fb-09fa541be52b
--sdb15  ntfs   winbkup      405CAA5F37B13F5B
sr0
ki@dusk:mnt$
```

Sdb ist eine externe Festplatte, und von ihrer Sdb7-Partition ist Ubuntu 20.04 LTS gebootet, Sdb1 ist an **/boot/efi** angehängt und eine aktive Swap-Partition ist auch vorhanden.

7ubui1, 8ubui2 und 9ubui3 sind ebenfalls Klone, im Sinne der Kubuntu-Klone an der Sda.

10ubuc2, 11ubuc5 und 12ubuc6, sind Klone des Ubuntu-BS vom W530-Thinkpad, und die 13kubui1 und 14kubui2, sind Klone des Kubuntu-Haupt-BS auf dem T520-Thinkpad. Auf der Partition Vmisc3 befinden sich Kopien virtueller Maschinen.

Die Sda4-data, ist eine größere Partition wo alle Personaldaten gespeichert werden, so dass unabhängig davon welche Partition gebootet wird, der Zugriff auf die privaten Daten nur das Einhängen der Data-Partition erfordert.

Die Dataei-Partition hat Sicherungskopien der persönlichen Notebookdaten, und Datani ist eine kleine NTFS-Datenpartition.

W530-Thinkpad Festplatte

Die gleichen Grundgedanken wie bei dem T520 wurden hier auch befolgt.

Zwei ESPs, eine Swap-Partition, Datae ist die persönliche Datenpartition, Vms, Vms2, Vms3 und Vms4 sind Partitionen für virtuelle Maschinen, Datan ist für Ntfs-Daten, 14ubuk2, 15ubuk3 und 16ubuk4, sind drei Klone des BSs.

```

bi@green: /mnt
sda
-sda1 vfat ESP1 70C9-ECC1 1014,2M 1% /boot/efi 931,5G |
-sda2 vfat ESP2 7A75-E0CE 1G |
-sda3 swap 63c9cfc-dbb4-415a-8259-734fdb023b3b [SWAP] 48G |
-sda4 ntfs datan 432857D23F67627D 20G |
-sda9 ext4 datae 012e8f8b-d0a0-429b-b520-823149a0ed7e 139,6G 43% /mnt/datae 271,5G |
-sda10 ext4 vms 25e9e0e5-dbc8-4d26-9690-90d461cf5c45 32,2G 65% /mnt/vms 110G |
-sda11 ext4 vms2 8d87329c-ce2f-4033-863f-aa6e02a6a83c 33,1G 37% /mnt/vms2 58G |
-sda12 ext4 vms3 666fc3fe-17af-439e-a9e1-6741e58c23e2 16,5G 66% /mnt/vms3 58G |
-sda13 ext4 13cryfs f8563746-c8dc-49ac-b033-20f3a3d9d48f 58G |
-sda14 ext4 14ubuk2 bc3bc37a-9b66-4039-913c-87df10a043ae 25,2G 51% / 58G |
-sda15 ext4 15ubuk3 22b0b57b-4c35-430c-89e0-aef14b1ed721 58G |
-sda16 ext4 16ubuk4 935552e2-f275-4aa7-a6f1-7f23166eb522 58G |
-sda17 ext4 4e2596d2-8656-418b-b9e7-5950ce7d4c91 58G |
-sda18 ext4 vms4 8c2180a2-2ee2-49a4-b611-1236010452c8 15,5G 68% /mnt/vms4 58G |
sdb 29,8G |
sr0 1024M |
bi@green:mnt$

```

Fstab angepasste Einträge

Damit schnell und leicht mit dem Ein- und Aushängen von Partitionen geht, können wir in Fstab, angepasste Einhängpunkte eintragen.

```
GNU nano 4.8 /etc/fstab
1 # /etc/fstab: static file system information.
2 #
3 # Use 'blkid' to print the universally unique identifier for a
4 # device; this may be used with UUID= as a more robust way to name devices
5 # that works even if disks are added and removed. See fstab(5).
6 #
7 # <file system> <mount point> <type> <options> <dump> <pass>
8 # / was on /dev/sda13 during installation
9 UUID=bc3bc37a-9b66-4039-913c-87df10a043ae / ext4 errors=remount-ro 0 1
10 # /boot/efi was on /dev/sda5 during installation
11 UUID=70C9-ECC1 /boot/efi vfat umask=0077 0 1
12 # swap was on /dev/sda7 during installation
13 UUID=63c9cfcd-dbb4-415a-8259-734fdb023b3b none swap sw 0 0
14 # custom mounts
15 # LABEL=15ubuk3 /mnt/15ubuk3 ext4 user,noauto,exec,rw,suid,dev 0 2
16 # LABEL=16ubuk4 /mnt/16ubuk4 ext4 user,noauto,exec,rw,suid,dev 0 2
17 LABEL=7ubui1 /mnt/7ubui1 ext4 user,noauto,exec,rw,suid,dev 0 2
18 LABEL=8ubui2 /mnt/8ubui2 ext4 user,noauto,exec,rw,suid,dev 0 2
19 LABEL=9ubui3 /mnt/9ubui3 ext4 user,noauto,exec,rw,suid,dev 0 2
20 LABEL=umore2 /mnt/umore2 ext4 user,noauto,exec,rw,suid,dev 0 2
21 LABEL=umore1 /mnt/umore1 ext4 user,noauto,exec,rw,suid,dev 0 2
22 LABEL=umore3 /mnt/umore3 ext4 user,noauto,exec,rw,suid,dev 0 2
23 LABEL=vms /mnt/vms ext4 user,noauto,exec,rw,suid,dev 0 2
24 LABEL=vms2 /mnt/vms2 ext4 user,noauto,exec,rw,suid,dev 0 2
25 LABEL=vms3 /mnt/vms3 ext4 user,noauto,exec,rw,suid,dev 0 2
26 LABEL=vms4 /mnt/vms4 ext4 user,noauto,exec,rw,suid,dev 0 2
27 LABEL=vmsc /mnt/vmsc ext4 user,noauto,exec,rw,suid,dev 0 2
28 LABEL=vmsc2 /mnt/vmsc2 ext4 user,noauto,exec,rw,suid,dev 0 2
29 LABEL=vmsc3 /mnt/vmsc3 ext4 user,noauto,exec,rw,suid,dev 0 2
30 LABEL=datae /mnt/datae ext4 user,noauto,exec,rw,suid,dev 0 2
31 LABEL=dataei /mnt/dataei ext4 user,noauto,exec,rw,suid,dev 0 2
32 LABEL=dataei2 /mnt/dataei2 ext4 user,noauto,exec,rw,suid,dev 0 2
33 LABEL=bkp1-st /mnt/bkp1-st ext4 user,noauto,exec,rw,suid,dev 0 2
34 LABEL=datan /mnt/datan ntfs permissions,uid=1000,gid=1000,dmask=002,fmask=113,noauto,
exec,rw,suid,dev,utf8 0 2
35 LABEL=usbdata /mnt/usbdata ntfs permissions,uid=1000,gid=1000,dmask=002,fmask=113,noa
uto,exec,rw,suid,dev,utf8 0 2
^G Hilfe ^O Speichern ^W Wo ist ^K Ausschneide ^J Ausrichten ^C Textmarke
^X Beenden ^R Datei öffne ^\ Ersetzen ^U Text einfüge ^T Rechtschr. ^_ Zu Zeile gehen
```

Diese sind dann mit **Label** und nicht mit **UUID** gekennzeichnet, damit sie leicht erkennbar sind.

Ein Label kann während der Erstellung einer Partition mit verschiedenen Programmen zugewiesen werden, wie mit dem **Mkfs-Program**.

Nachträglich kann ein Label mit

```
$ sudo e2label /dev/sdxx <label>
```

eingetragen werden.

Die zwei letzten Einträge sind von Ntfs-Partitionen. Es muss hier bemerkt werden dass sie ganz andere Einhängoptionen als die Ext4-Partitionen brauchen.

Ubuntu-Paketverwaltung

Repos

Unsere Pakete dürfen mittels zwei Befehlen aktualisiert werden.

\$ sudo apt update - Paketinfos holen

\$ sudo apt upgrade - Pakete aktualisieren

Wir suchen nach Paketen in den Repos mit Schlüsselwörtern wie folgt:

\$ unbuffer apt search -f <schlüsselwort> | less -R



```
bi@green: ~  
bi@green:~$ unbuffer apt search python | less -R
```

Die Ergebnisse vom „apt search“ sind teilweise farbig. Aber um die Bild-auf- und Bild-ab Taste zu verwenden, pfeifen wir less dazu, aber gehen dann die Farben verloren.

Unbuffer, das zuerst installiert werden muss, arbeitet mit der -R Fahne von less zusammen, um die farbige Ausgabe zu bewahren.

Die -f Option bedeutet full, deshalb kann sie abwechselnd eingesetzt werden, um die Ausgabe zusammenfassend oder mit vollen Beschreibungen zum Drucken.

```
python-crypto/focal 2.6.1-13ubuntu2 amd64  
  Kryptografische Algorithmen und Protokolle für Python  
  
python-crypto-dbg/focal 2.6.1-13ubuntu2 amd64  
  Kryptografische Algorithmen und Protokolle für Python (Debug-Erweiterung)  
  
python-cryptography/focal-updates,focal-security 2.8-3ubuntu0.1 amd64  
  Bibliothek zur Bereitstellung kryptographischer Rezepte und Primitive (Python 2)  
  
python-cryptography-doc/focal-updates,focal-updates,focal-security,focal-security 2  
.8-3ubuntu0.1 all  
  Bibliothek zur Bereitstellung kryptographischer Rezepte und Primitive (Dokumentat  
ion)  
  
python-cryptography-vectors/focal,focal 2.8-1 all  
  Testvektoren für python-cryptography (Python 2)
```

Um ein Paket zu installieren,

\$ sudo apt install <paketname>

Wir entfernen Pakete mit

\$ sudo apt remove <paketname>

Es bleiben aber die Konfigurationen gespeichert falls wir das Paket nochmals installieren wollen. Ein Paket und seine Konfigurationen entfernen wir mit

\$ sudo apt purge <paketname>

Synaptic

Die Paketverwaltung ist auch mittels Synaptic möglich, dass weiter oben erwähnt wurde, ist ein GUI-Frontend für Apt, und muss zuerst installiert werden.

Snap

Entsprechend der Paketverwaltung durch die Repos, müssen die Software-Entwickler ein einzelnes Installationspaket für jedes Paket-Verwaltungssystem: Apt, Yam, Dnf, Zypper usw., erstellen müssen.

Snap wurde 2014 von Canonical freigegeben und 2016 wurde in die Distro-Mehrheit portiert. Snaps können universal installiert werden und vereinfachen die Entwicklerarbeit.

Durch Snaps ist es auch möglich freie Versionen von Closed-Source-Anwendungen, wie Skype zu installieren, was durch die Repos unmöglich ist.

Folgend werden Snapbefehle-Beispiele angeführt.

```
$ snap find <paketname>
```

```
$ snap install <paketname>
```

```
$ snap remove <paketname> - Confs werden nicht entfernt
```

```
$ snap remove --purge <paketname> - Confs werden entfernt
```

```
$ snap list - Installierte Snaps anzeigen
```

PPA - Personal Package Archive - Persönliches Paket-Repo

PPAs sind private Repos, wodurch Pakete installiert werden können, die in den offiziellen Repos nicht vorhanden sind, aber weil sie auch Repos sind, deren Pakete werden zusammen mit den offiziellen aktualisiert.

Um ein PPA hinzufügen und ein dazugehöriges Paket zu installieren werden folgende Befehle gebraucht.

```
$ sudo add-apt-repository ppa:<ppa übergeordnetes verzeichnis>/<paketverzeichnis>
```

```
$ sudo apt update
```

```
$ sudo apt install <paketname>
```

Deb-Dateien Paketmanagement

Closed-Source-Programme wie Foxit-PDF-Reader und Master-PDF-Editor, werden oft als Dateien mit einer `.deb` Erweiterung angeboten.

Die werden installiert mit

```
$ sudo dpkg -i <paketpfad>
```

Es ist auch möglich

```
$ sudo apt install <paketpfad>
```

es kann aber sein dass `apt install`, keine Meldung über fehlende Abhängigkeiten gibt.

Wenn `apt search` einen Paketzustand als `[installed, local]` anzeigt, es handelt sich um ein Deb-Paket dass manuell installiert wurde.

Wenn das Paket aktualisiert werden soll, aber kein offensichtlicher GUI-Weg dafür verfügbar ist, suchen wir das Paket zuerst suchen mit

```
$ dpkg -l | grep <paketname>
```

Paket entfernen mit

```
$ dpkg -r <paketname>
```

Der Befehl

```
$ locate <paketname>
```

zeigt immer noch Dateien wie `postinst` und `porstrm`.

`apt search` zeigt deren Zustand als `[residual config]`

Man sollte diese Pakete nicht entfernen, so dass wenn das neue Paket installiert wird, können die alten Konfigurationen dadurch aufgenommen werden.

Ubuntu-Software

Ubuntu-Software ist die einfachere GUI-Variante ein Programm zu installieren, was im Projekt nie angewendet wurde.

Gnome

Beim ersten Kontakt, wirkt Gnome etwas fremd für Benutzer die von Windows kommen. Es sollten zuerst die Pakete: [Optimierungen](#) und [Dconf-Editor](#) installiert werden, und daraufhin verschiedene Erweiterungen. So kann das Look-and-Feel sehr angepasst werden.

Wo nicht angepasst werden kann, das sind die Desktop-Verknüpfungen. Es können Icons zum Desktop nur wenn in den Ordner [/home/<benutzer>/Desktop](#) eingefügt werden.

Wenn wir Linux lernen wollen, müssen wir unausweichlich einiges anders machen als bei Windows. Der Desktop sollte nur um ein schönes Bild zu zeigen verwendet werden. Da können auch Icons als Deko verwendet, aber lieber nicht als Verknüpfungen.

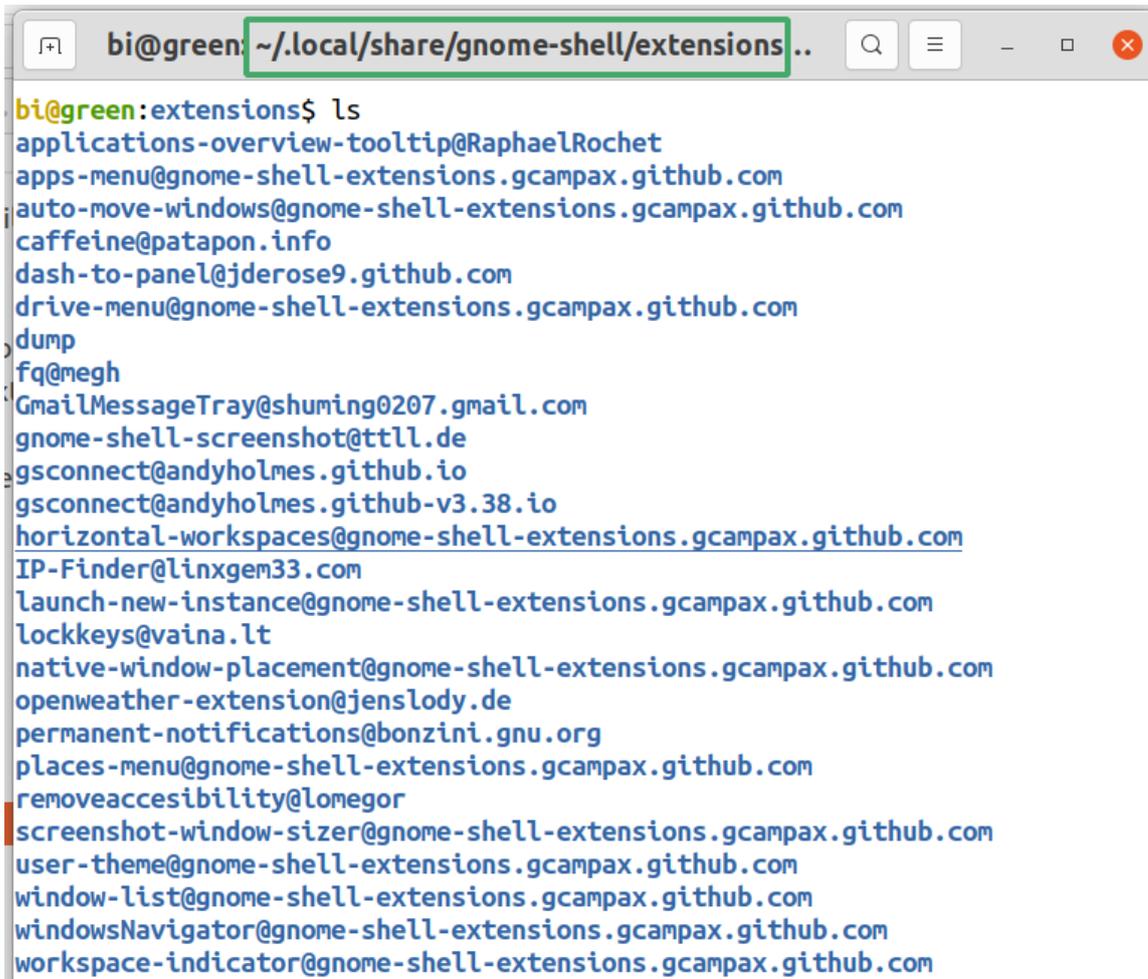
Erweiterungen

Einige Extensions dürfen durch die Paketverwaltung installiert werden.

Zusätzliche Erweiterungen können entweder von extensions.gnome.org installiert werden oder auch manuell:

Zuerst unter `/home/<benutzer>/.local/share/gnome-shell/` einen Ordner namens `<extensions>` erstellen.

Dann von extensions.gnome.org eine Erweiterung in den `<extensions>` Ordner herunterladen.



```
bi@green: ~/.local/share/gnome-shell/extensions ..
bi@green:extensions$ ls
applications-overview-tooltip@RaphaelRochet
apps-menu@gnome-shell-extensions.gcampax.github.com
auto-move-windows@gnome-shell-extensions.gcampax.github.com
caffeine@patapon.info
dash-to-panel@jderose9.github.com
drive-menu@gnome-shell-extensions.gcampax.github.com
dump
fq@megh
GmailMessageTray@shuming0207.gmail.com
gnome-shell-screenshot@ttll.de
gsconnect@andyholmes.github.io
gsconnect@andyholmes.github-v3.38.io
horizontal-workspaces@gnome-shell-extensions.gcampax.github.com
IP-Finder@linxgem33.com
launch-new-instance@gnome-shell-extensions.gcampax.github.com
lockkeys@vaina.lt
native-window-placement@gnome-shell-extensions.gcampax.github.com
openweather-extension@jenslody.de
permanent-notifications@bonzini.gnu.org
places-menu@gnome-shell-extensions.gcampax.github.com
removeaccessibility@lomegor
screenshot-window-sizer@gnome-shell-extensions.gcampax.github.com
user-theme@gnome-shell-extensions.gcampax.github.com
window-list@gnome-shell-extensions.gcampax.github.com
windowsNavigator@gnome-shell-extensions.gcampax.github.com
workspace-indicator@gnome-shell-extensions.gcampax.github.com
```

Entpacken und die Datei `metadata.json` aufmachen.

Die [UUID-Beschreibung](#) kopieren, die innerhalb der Anführungszeichen steht, und damit den Erweiterung-Ordner umbenennen.

```
bi@green:extensions$ cd lockkeys@vaina.lt
bi@green:lockkeys@vaina.lt$ ls
extension.js  LICENSE  lockkeys.pot  prefs.js  stylesheet.css
icons        locale   metadata.json schemas    utils.js
bi@green:lockkeys@vaina.lt$ cat metadata.json
{
  "_generated": "Generated by SweetTooth, do not edit",
  "description": "Numlock & Capslock status on the panel",
  "name": "Lock Keys",
  "original-authors": "Kazimieras Vaina, Pierre Ossman, erguille, jonnius,
Philipp Wolfer, Mariusz Lisowski, Cristian Beroiza, warmsun0220, Rasmus Kaj
, Pablo Martin-Gomez Bouska, Rapha\u00ebl Rochet, Luiz Nickel, Jesse, Du\u00
161an Kazik",
  "shell-version": [
    "3.18",
    "3.20",
    "3.22",
    "3.24",
    "3.26",
    "3.28",
    "3.30",
    "3.34",
    "3.32",
    "3.36",
    "3.38"
  ],
  "url": "https://github.com/kazysmaster/gnome-shell-extension-lockkeys",
  "uuid": "lockkeys@vaina.lt",
  "version": 41
}
bi@green:lockkeys@vaina.lt$
```

`Alt+F2` und im Kommandofenster „r“ geben um die Shell neu zu starten.

[Dash-to-Panel](#) ist eine Extension die uns sehr viele Anpassungsmöglichkeiten gibt. Damit ist eine vielseitige Taskleiste gemeint, die die Ubuntu-Erfahrung um einiges verbessert, und womit Neankömmlinge etwas freuen können.

[Verschiedene Anwendungen](#)

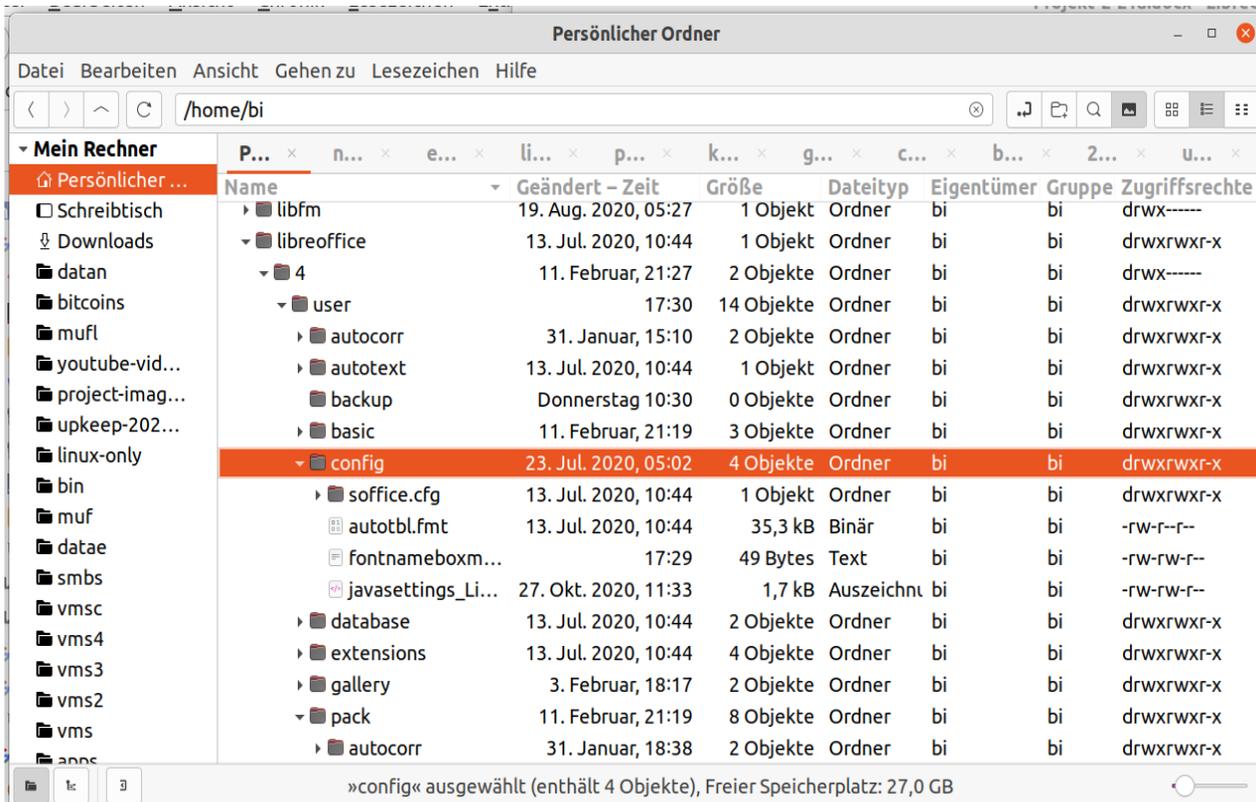
[DE bestimmte Programme](#)

Jede Desktop-Umgebung hat Vorgabe-Programme. Es ist aber möglich solche Programme auch in anderen DEs zu installieren.

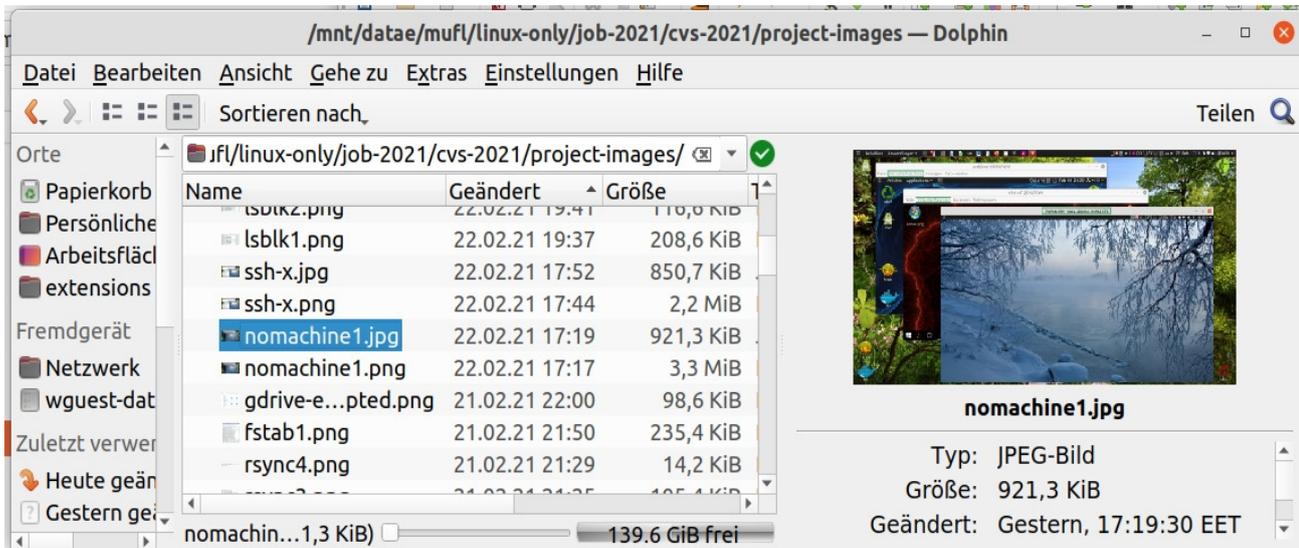
[Dateimanager](#)

Der Dateimanager von Gnome, Nautilus, ist nicht so anpassbar und atemberaubend, kann er aber leichter als andere Dateimanager, Netzwerken mit Samba-Shares und Android unter anderem.

Als täglicher Dateimanager wirkt aber Nemo von Cinnamon-DE, viel besser.



Wiederum, wenn uns Dateien-Vorschauen wichtig ist, da wäre Dolphin von KDE besser.



Alle drei Dateimanager können ergänzend verwendet werden.

Texteditor

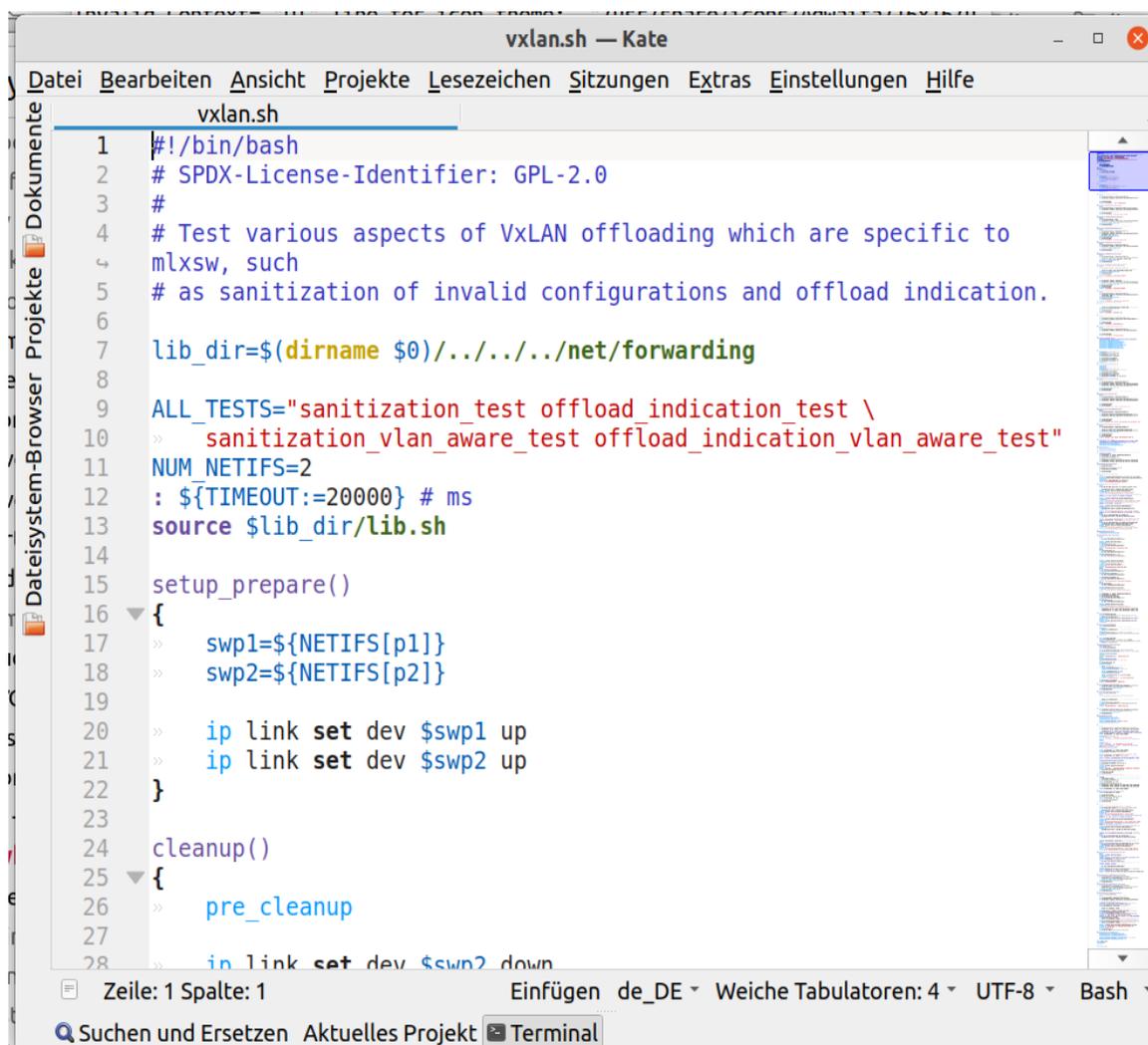
Im Terminal ein guter und leicht zum lernen Editor, ist Nano.

```
bi@green: /mnt/datae/no-upload/vmconfs
GNU nano 4.8 vxlan.sh
9 ALL_TESTS="sanitization_test offload_indication_test \
10 sanitization_vlan_aware_test offload_indication_vlan_aware_test"

11 NUM_NETIFS=2
12 : ${TIMEOUT:=20000} # ms
13 source $lib_dir/lib.sh
14
15 setup_prepare()
16 {
17     swp1=${NETIFS[p1]}
18     swp2=${NETIFS[p2]}
19
20     ip link set dev $swp1 up
21     ip link set dev $swp2 up
22 }
23
24 cleanup()
25 {
26     pre_cleanup
27
28     ip link set dev $swp2 down
29     ip link set dev $swp1 down
30 }
31
32 sanitization_single_dev_test_pass()
33 {
34     ip link set dev $swp1 master br0
35     check_err $?
36     ip link set dev vxlan0 master br0
37     check_err $?
38
39     ip link set dev $swp1 nomaster

^G Hilfe      ^O Speichern  ^W Wo ist    ^K Ausschneiden ^J Ausrichten
^X Beenden    ^R Datei öffnen \ Ersetzen    ^U Text einfügen ^T Rechtschr.
```

Betreffend GUI-Editoren, der Vorgabe-Gnome-Texteditor ist auch nichts besonders. Kate, wieder von KDE, hat nur Vorteile und ist nicht so schwer wie andere.



```
vxlan.sh — Kate
Datei Bearbeiten Ansicht Projekte Lesezeichen Sitzungen Extras Einstellungen Hilfe
vxlan.sh
1 #!/bin/bash
2 # SPDX-Lizenz-Identifizierer: GPL-2.0
3 #
4 # Test various aspects of VxLAN offloading which are specific to
5 # mlxsw, such
6 # as sanitization of invalid configurations and offload indication.
7
8 lib_dir=$(dirname $0)/../../../../net/forwarding
9
10 ALL_TESTS="sanitization_test offload_indication_test \
11 >> sanitization_vlan_aware_test offload_indication_vlan_aware_test"
12 NUM_NETIFS=2
13 : ${TIMEOUT:=20000} # ms
14 source $lib_dir/lib.sh
15
16 setup_prepare()
17 {
18     swp1=${NETIFS[p1]}
19     swp2=${NETIFS[p2]}
20     ip link set dev $swp1 up
21     ip link set dev $swp2 up
22 }
23
24 cleanup()
25 {
26     pre_cleanup
27
28     ip link set dev $swp2 down
```

Zeile: 1 Spalte: 1 Einfügen de_DE Weiche Tabulatoren: 4 UTF-8 Bash

Suchen und Ersetzen Aktuelles Projekt Terminal

PDF-Editors

Diese Programme sind ein Spezialfall wegen der allgemeinen Ansicht, dass wir unter Linux große Schwierigkeiten mit Pdf-Editors haben, was nur teilweise trifft.

Poppler, die PDF-Darstellungsbibliothek mit der die meisten Open-Source-Frontends arbeiten, muss allerdings noch weiterentwickelt werden.

Okular

Eine gute Auswahl ist der PDF-Editor Okular, schon wieder von KDE. Bei den meisten Dokumenten ist die Fontwiedergabequalität sehr gut.

Es können zwar mit Okular Dokumente editiert werden, aber zwischen anderem, Formen werden nicht unterstützt, und die Schreibfunktion ist bei speziellen Zeichen wie griechische und asiatische Buchstaben fehlerhaft, da sie teils unsichtbar werden.

Unten, wird im Okular eine Seite von „The Linux Command Line“ vom William Shotts <http://linuxcommand.org/tlcl.php> angezeigt.

The screenshot shows the Okular PDF viewer interface. The title bar reads "The Linux Command Line — Okular". The menu bar includes "Datei", "Ansicht", "Bearbeiten", "Gehe zu", "Lesezeichen", "Extras", "Einstellungen", and "Hilfe". The toolbar has buttons for "Vorherige Seite", "Nächste Seite", "Seitenbreite", "Verkleinern", "Vergrößern", "Durchsuchen", "Zoom", and "Textauswahl".

The left sidebar shows a table of contents with sections like "Inhalt", "Vorschaubilder", "Rezensionen", and "Lesezeichen".

The main content area displays the "10 - Processes" chapter. It includes a text block explaining the 'top' program and a terminal window showing the output of the 'top' command. The terminal output includes a system summary and a table of processes.

```
top - 14:59:20 up 6:30, 2 users, load average: 0.07, 0.02, 0.00
Tasks: 109 total, 1 running, 106 sleeping, 0 stopped, 2 zombie
Cpu(s): 0.7%us, 1.0%sy, 0.0%ni, 98.3%id, 0.0%wa, 0.0%hi, 0.0%si
Mem: 319496k total, 314860k used, 4636k free, 19392k buff
Swap: 875500k total, 149128k used, 726372k free, 114676k cach
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6244	me	39	19	31752	3124	2188	S	6.3	1.0	16:24.42	trackerd
11071	me	20	0	2304	1092	840	R	1.3	0.3	0:00.14	top
6180	me	20	0	2700	1100	772	S	0.7	0.3	0:03.66	dbus-dae
6321	me	20	0	20944	7248	6560	S	0.7	2.3	2:51.38	multiloa
4955	root	20	0	104m	9668	5776	S	0.3	3.0	2:19.39	Xorg
1	root	20	0	2976	528	476	S	0.0	0.2	0:03.14	init
2	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	RT	-5	0	0	0	S	0.0	0.0	0:00.00	migratio
4	root	15	-5	0	0	0	S	0.0	0.0	0:00.72	ksoftirq
5	root	RT	-5	0	0	0	S	0.0	0.0	0:00.04	watchdog
6	root	15	-5	0	0	0	S	0.0	0.0	0:00.42	events/0
7	root	15	-5	0	0	0	S	0.0	0.0	0:00.06	khelper
41	root	15	-5	0	0	0	S	0.0	0.0	0:01.08	kblockd/
67	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	kseriod
114	root	20	0	0	0	0	S	0.0	0.0	0:01.62	pdflush
116	root	15	-5	0	0	0	S	0.0	0.0	0:02.44	kswapd0

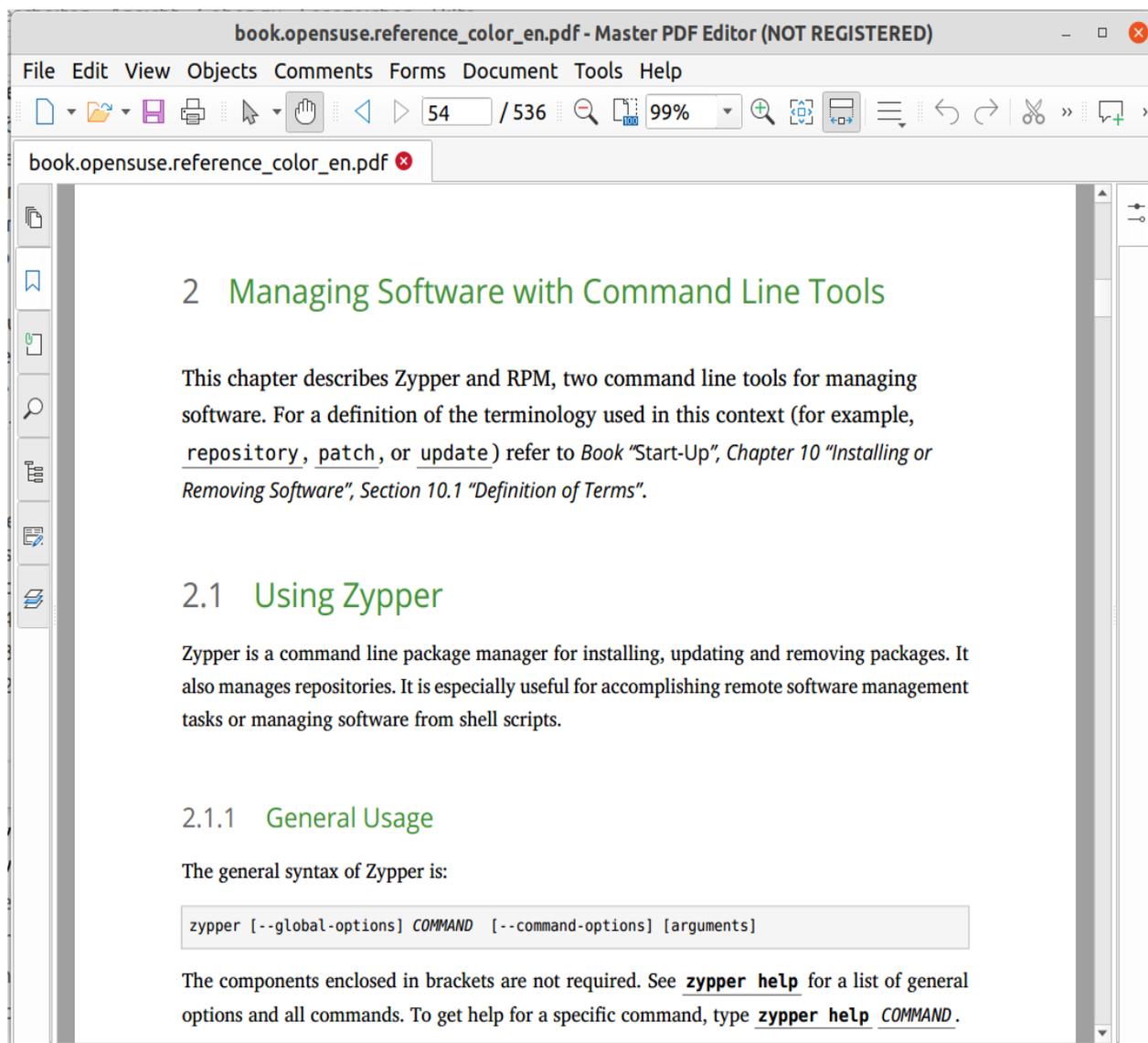
Master-PDF-Editor und Qoppa-PDF-Studio

Zwei nennenswerte gewerbliche Linux-PDF-Editors sind verfügbar. [Master-PDF-Editor \(MPE\)](#), \$69,95 pro Lizenz, das schon im Projekt als Demo getestet wurde und [Qoppa-PDF-Studio Pro](#), \$129 für zwei Geräte, nur als Trial anwendbar ohne es zu kaufen, das teurer als der MPE ist, aber laut Rezensionen auch besser.

Im MPE-Demo, sind alle Funktionen zwar verfügbar, bei den meisten Editorfunktionen wird aber ein Wasserzeichen im Dokument erzeugt.

Die Schreibfunktion ist im Demoverversion jedenfalls freigegeben, aber Akzente wahlweise auf Buchstaben zu setzen, wie in der griechischen Sprache ist nicht möglich.

Ein Bild von MPE folgt, in dem OpenSuse-Reference, erhältlich von <https://doc.opensuse.org/> angezeigt wird.



In einem Fontwiedergabe-Vergleich zwischen Okular und MPE, gewinnt MPE grenzwertig. Es gibt zu verstehen dass MPE eine eigene Rendern-Bibliothek verwendet und nicht das Poppler-Backend.

Weil die PDF-Editors bezüglich Office-Arbeit notwendig sind, um offizielle Formulare auszufüllen unter anderem, es ist zu beachten, dass das MPE-Demo Formen unterstützt im Gegensatz zum Okular, und dass durch deren Bearbeitung kein Wasserzeichen während des Testens entstand.

Linux-Acrobat abgeschafft

Seit 2012, existieren keine Linux PDF-Programme mehr von Adobe.

Foxit

Foxit-PDF-Reader wurde getestet mit dem Ergebnis dass Ubuntu manchmal komplett einfro.

LibreOffice-Draw als PDF-Editor

Eine andere Lösung zur PDF-Problematik unter Linux, ist LibreOffice-Draw.

Anfängliche Probleme

Anfangs werden wir aufgrund von den zwei folgenden Problemen abgeschreckt.

Wenn wir eine größere PDF-Datei mit Draw öffnen, dauert es relativ lang und fängt auch der CPU-Lüfter zu arbeiten.

Dann wenn wir mit der Maus eine Seite ziehen wollen wird vielleicht die PDF-Schicht von der Grundschrift weggezogen, in der Art und Weise von Gimp.

Zudem, eine quadratische Textbegrenzung wird angezeigt, die vom PDF-Text rechts und links überschreitet wird, und störend erscheint.

Konfiguration für PDF-Dokumente

Zuerst die Pdf-Datei mit dem nativen Odg-Format speichern.

[Optionen/LibreOffice/Anwendungsfarben/Textbegrenzungen](#) deaktivieren.

Die Datei soll künftig mit der Odg-Erweiterung verwendet werden und nur wenn sie z.B. per E-mail gesendet sollte, wird wieder als Pdf-Datei exportiert.

Wenn unser Dokument mit einer bestimmten Schrift standardmäßig bearbeitet werden soll, es reicht nicht eine neue Vorlage als Vorgabe zu setzen.

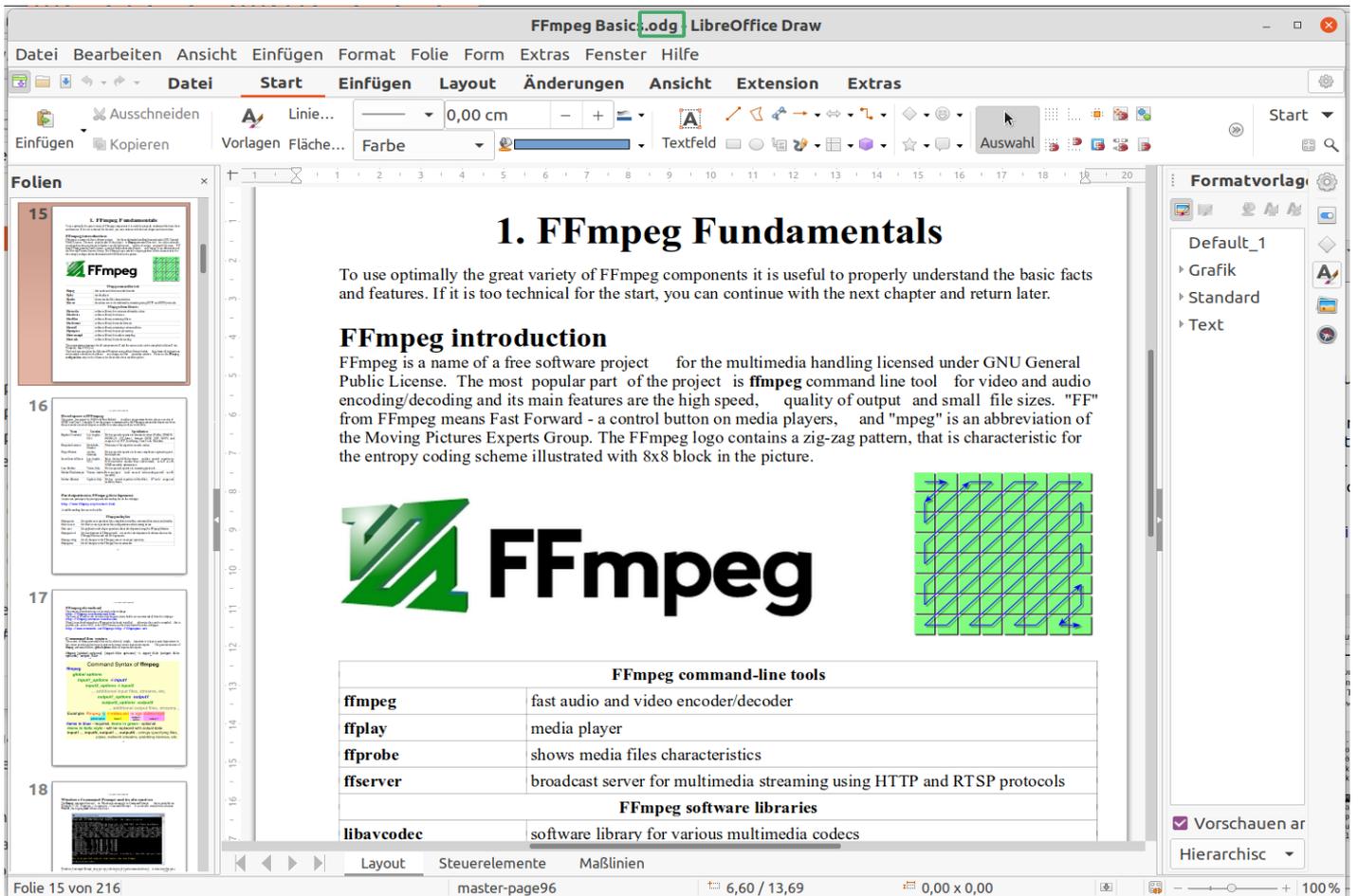
Die Standardschrift ist hardcoded in LibreOffice und kann nicht (auf einfache Weise) geändert werden wie die Dateien [Normal.dot](#) und [Nomal.xlt](#) unter Windows.

Wir müssen also wie im Link [Associating a document with a different template](#) erläutert wird, vorgehen.

[Datei/Dokumentvorlagen/Dokumentvorlage](#) öffnen.

Die gewünschte Vorlage öffnen.

Den gesamten Inhalt unseres Dokuments mit Strg+A und Strg+C kopieren und mit Strg+V in das neue leere Dokument einfügen und speichern.



Jetzt kann endlich die Schreibfunktion durch die Auswahl von „[Textfeld einfügen \(F2\)](#)“ verwendet werden, die schon mit der gewünschten Schrift permanent konfiguriert ist. Rechtschreib-Überprüfung ist auch verfügbar wie nur in den teureren Pdf-Editors.

Noch besser, da LibreOffice-Draw andere Rendern-Bibliotheken einsetzt und nicht das Pdf-Poppler-Backend, die Schriftwiedergabe sieht etwas besser als im Okular aus.

Minuspunkte

Große Minuspunkte gibt es aber trotzdem: Unterschiedliche Abstände zwischen Wörtern im Text, und die Lesezeichen, Hyperlinks etc. gehen jetzt alle verloren. Der Text ist aber noch suchbar.

Pdf2odt

Das ist [ein Skript von Gutschke](#) dass die Daten einer Pdf-Datei in Bildern umwandelt. Die Datei bekommt eine Odt-Erweiterung und kann unter LibreOffice-Writer annotiert werden.

Skript herunterladen, unter `~/bin` - gegebenenfalls Verzeichnis zuerst erstellen - speichern und ausführbar machen mit

```
$ chmod +x pdf2odt
```

```

bi@green:~$ pdf2odt
Usage:
 pdf2odt [ options ] input.{jpg,pdf,png} ... output.{odt,ods}

Options:
 --                Stop scanning for options.
 --format={odt|ods} Select output file format.
 --margin=X{cm|in}  Add extra margin around input files.
 --paper={letter|a4} Select paper size.
 --portrait         Portrait is always the default orientation.
 --landscape        Assume all input files are in landscape format.
 --resolution=RES  Change resolution of embedded images.
                   Only affects PDF files. PDF/PNG files will not
                   be rescaled.
 --trim|--no-trim  Attempt to trim existing margins from input files.
bi@green:~$ █

```

Das Skript scheint einwandfrei zu funktionieren als mit der Konvertierung einer Pdf-Datei von 200 Seiten getestet wurde.

Arranger

Um Dokumente zusammen zu setzen und individuell zu ordnen ist der Open-Source [PDF-Arranger](#) die beste Wahl.

Pdf-Fazit

Das MPE-Demo ist die einzige freie Möglichkeit interaktive Formen auszufüllen. Die Schreibfunktion ist frei, zwar etwas fehlerhaft für manche Sprachen, aber auf Deutsch wird problemlos geschrieben. Okular kann auch Deutsch schreiben, aber griechische und asiatische Buchstaben sind teilweise unsichtbar. Er unterstützt keine interaktiven Formen, wie das MPE-Demo, aber er kann unterstreichen und einfache Formen zeichnen, was beim MPE-Demo nicht freigegeben ist.

Die beide Anwendungen können also ergänzend eingesetzt werden.

Das Pdf2odt-Skript, ist sehr gut wenn wir nicht interaktiv, Formen in mehreren Sprachen und mit Rechtschreibungs-Überprüfung ausfüllen wollen, und auch dementsprechend besser als LibreOffice-Draw da mit dem Pdf2odt, entstehen keine unnatürlichen Abstände zwischen Wörtern.

Wenn wir was professionelleres unbedingt haben wollen, ist der Kauf von Qoppa-PDF-Studio-Pro an erster Stelle oder von Master-PDF-Editor an zweiter Stelle, nötig.

Andererseits, die Linux-Philosophie ist gleichzeitig mit mehreren Tools zu arbeiten die gegenseitig ergänzend sind.

LibreOffice

Vorlagen

Was unter dem Überschrift „LibreOffice-Draw als PDF-Editor“ über Vorlagen erwähnt wurde, gilt für alle LibreOffice-Anwendungen.

Die Auferstehung einer überholten Erweiterung, die „[Template Changer](#)“ heißt, soll das Ändern einer Vorlage erleichtern, die Kompatibilität ist aber laut Entwickler bis LO-Version 6.3 und wurde im Projekt nicht getestet.

Writer - der Mangel der geteilten Ansicht

Insofern können Soft-Links, kurzum Symlinks, verwendet werden, wobei die Symlink-Datei parallel mit der Quelle-Datei, betrachtet werden kann.

Einen Symlink erstellen mit

```
$ ln -s <quelldatei> <link-datei>
```

Um die Linkdatei zu aktualisieren, durch **Alt+F4** schließen, und mit

```
$ libreoffice <link-datei>
```

```
bi@green:german$ libreoffice Projekt-link.odt
```

neu starten.

Bei einer Umbenennung der Quelldatei, die Linkverbindung mit

```
$ ln -sf <quelldatei> <link-datei>
```

aktualisieren.

Die Linkverbindung zur Quelle wird mit ls im Terminal angezeigt.

```
bi@green:german$ ls -lth --color | less
```

```
insgesamt 253M
-rw-rw-r-- 1 bi bi 12M Mär 27 19:34 Projekt-3-27.odt
lrwxrwxrwx 1 bi bi 16 Mär 27 16:20 Projekt-link.odt -> Projekt-3-27.odt
-rw-rw-r-- 1 bi bi 12M Mär 26 22:51 Projekt-3-26.odt
-rw-rw-r-- 1 bi bi 12M Mär 26 22:50 Projekt-3-25.odt
-rw-rw-r-- 1 bi bi 12M Mär 25 22:11 Projekt-3-24.odt
-rw-rw-r-- 1 bi bi 12M Mär 23 21:32 Projekt-3-22.odt
-rw-rw-r-- 1 bi bi 12M Mär 22 22:51 Projekt-3-21.odt
```

Bereinigung unnötiger Dateien

Übriggebliebene Konfigurationsdateien entfernen

Unter <https://help.ubuntu.com/community/AptGet/Howto> wird der untere Befehl empfohlen und kurz beschreibt. Beim Xargs wurde die Option -p oder --interactive aus Sicherheitsgründen zugefügt.

```
$ dpkg -l | grep '^rc' | awk '{print $2}' | sudo xargs -p dpkg --purge
```

dpkg -l - Paketliste ausdrucken.

grep '^rc' - Drückt Pakete aus, die entfernt wurden, wobei deren Konfigurationsdateien noch vorhanden sind.

awk '{print \$2}' - Drückt die zweite Spalte aus.

xargs - Bereitet Befehlausführungen aus der Liste dass awk ausgedruckt hat, und leitet diese an dpkg weiter.

-p - Interaktiv

--purge - Entfernen samt Konfigurationsdateien

Da xargs kein deutsch versteht auch wenn die BS-Sprache auf Deutsch umgestellt ist, müssen wir bezüglich der -p Fahne, mit **yes** statt **ja** antworten.

Unnötige Pakete mit autoremove löschen

```
$ sudo apt autoremove
```

Daten im Cache löschen

```
$ sudo apt-get clean
```

Löscht alles unter `/var/cache/apt/archives/` und `/var/cache/apt/archives/partial/` bis auf die Lock-Datei.

Daten im Snap-Cache löschen

Alles unter `/var/lib/snapd/cache` löschen, was im folgenden Hyperlink erläutert wird.

<https://askubuntu.com/questions/1075050/how-to-remove-uninstalled-snaps-from-cache>

```
bi@green:~$ sudo -s
root@green:bi# cd /var/lib/snapd/cache
```

```
root@green:cache# pwd
/var/lib/snapd/cache
root@green:cache# du -ah
198M    ./793e3df1f1f1749a93cd7b197bcba3fbf9fe2d2a17efadb15afa82434af06914e7aed6187
bffd7f2a49e9635613de5d0
130M    ./384a8b6c5f085101de0e6d1f6dd172d4585a84a3874e595be64f0ebd0e5088533a6127084
6a9caf0381050c5722e8d42
112M    ./9c361a3c394603b87c44beb039b18503810c0161be0c4be365f4f99661914c3b9eb61fb81
fc23a17834becac513e1ac7
261M    ./127632cc622891a2cd69da70616530d2a15cccf0a9adb50f7d2dca3c1bfc56c0cc9052c39
a8b22cad619a43fd87048f
142M    ./0ee2fdb478d37f304eff574d8a0a31c30560cffac1b1b588c183f823dc0ffd5144c6422e
e6bfaf609860cf10eb60c69
2,3M    ./31274855e5af92b62ad1ce3b7d7af0900556878bfffac0dc2de861b882dffbe15021f145f0
6fa4468256eed44d428650f
119M    ./36244cb7ebc546368e09120688deb5163197ff2dbe3afc2114dd797adf905c96bce87e4ba
e64b714cc4c9aaa42648e9a
962M    .
```

```
root@green:cache# rm *
```

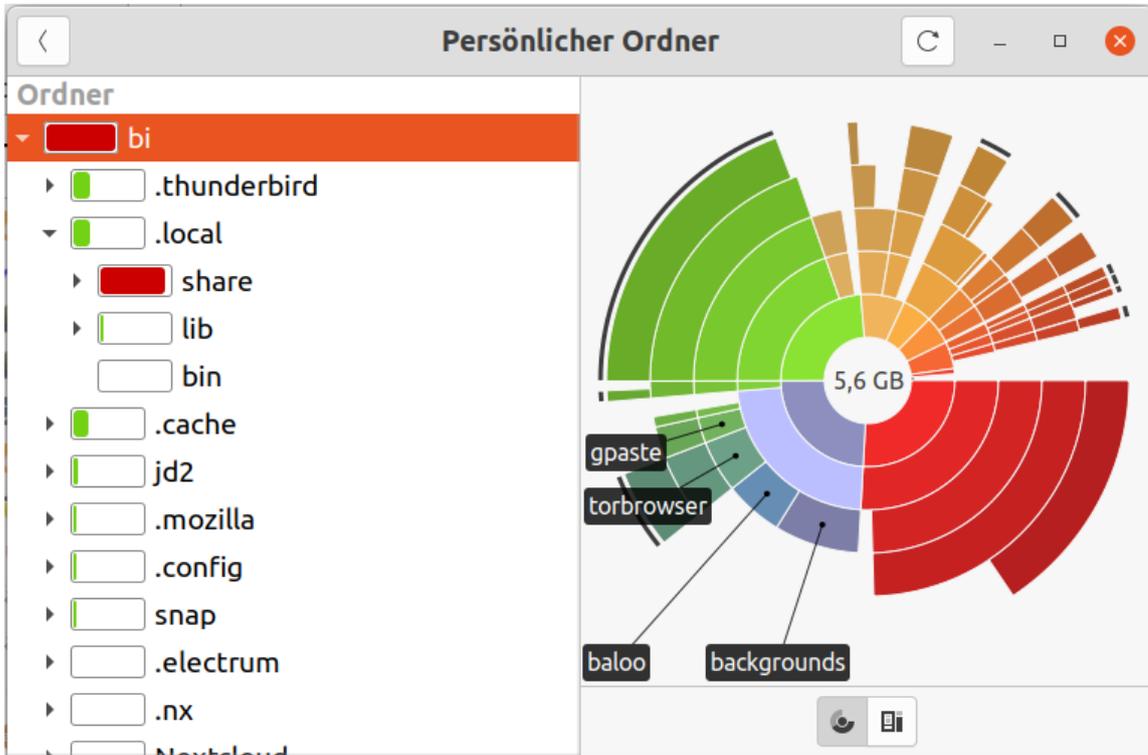
Festplattenbelegungsanalyse

Wenn wir

```
$ sudo apt purge <paketname>
```

verwenden, werden auch die Konfigurationsdateien der Pakete entfernt, aber nicht der versteckten Konf-Dateien im persönlichen Ordner, deren Name mit einem Punkt anfängt, die müssen manuell gelöscht werden.

Die Festplattenbelegungsanalyse ist ein schönes graphisches Tool, um einen Überblick der Verzeichnisse- und Dateigrößen zu gewinnen, um selektiv Daten zu löschen.



Netzwerken

SSH

Zuerst muss überprüft werden ob alle relevanten Pakete und Bibliotheken an beiden Seiten installiert sind. Mit

```
$ ssh benutzer@ip-adresse
```

können wir uns mit dem Server verbinden, und durch

```
$ ssh -X benutzer@ip-adresse
```

ist es möglich GUI-Anwendungen des Servers im Client zu verwenden, kurzum eine Weiterleitung des X11 Fenstermanagers.

Eine private IP-Adresse kann mit

```
$ ifconfig | grep -E 'inet.*192'
```

ermittelt werden. Hier sehen wir IP-Adressen von zwei Netzwerk-Schnittstellen. Das erste Netzwerk, dass 122 einbezieht, ist für die KVMs, und das zweite, dass 31 enthält, für das WLAN.

```

bi@green: ~
┌───(bi@green: ~)───┐
│ bi@green:~$ ifconfig | grep -E inet.*192
│   inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255
│   inet 192.168.31.29 netmask 255.255.255.0 broadcast 192.168.31.255
│ bi@green:~$
└──────────────────┘

```

Rsync über SSH

Hier wird z.B. ein Daten-Backup synchronisiert.

```
$ rsync -aAXsh --delete --preallocate --info=progress2,stats2 -e ssh <quelle> benutzer@ip-  
adresse:<ziel>
```

Im Vergleich mit dem Klonbefehl, darf für ein Daten-Backup die Fahne `-H` fehlen, wenn keine Hardlinks anwesend sind.

`-e` - Weist Rsync an, welches Netzwerkprogramm angewendet wird. Diese Option muss unmittelbar vor SSH stehen. Wenn nicht, funktioniert zwar SSH, aber alle Fahnen zwischen `-e` und `ssh` werden nicht aufgeführt.

Man kann den Rsync-Vorgang mit den Tasten `Strg+Z` unterbrechen. Um fort zu fahren,

```
$ jobs
```

um die Nummer des angehaltenen Job zu zeigen, und

```
$ fg <job-nummer>
```

um Rsync wieder fortzusetzen.

`fg` - Foreground

Remote-Desktop

Die Bildqualität der nativen Remnina-Anwendung von Ubuntu war mit der Vorgabe-Konfiguration nicht überzeugend.

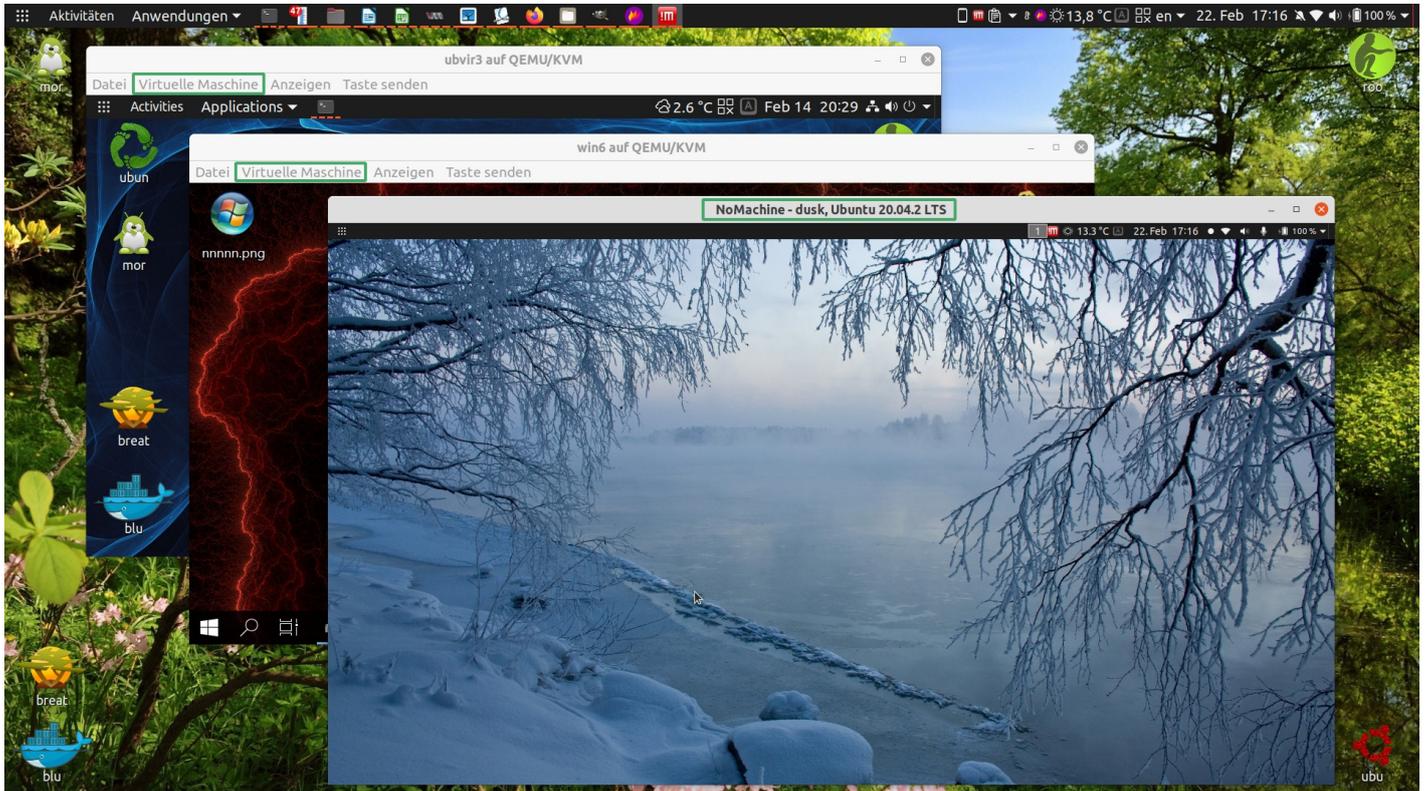
NoMachine ist zwar Closed-Source-Software, aber mit einer freien Version für Linux und insgesamt hat im Projekt gut abgeschnitten.

TeamViewer ist sehr beliebt und bietet auch eine freie Version an, hat aber im Vergleich mit NoMachine keine bessere Bildqualität aufgewiesen.

Sollte vermerkt werden dass TeamViewer 2016 gehackt wurde. Der Südweste Presse nach: [„Einen Anlass, die Kunden über den Angriff zu informieren, sah die Firma damals nicht.“](#)

Remote-Desktop ist zwar sehr praktisch, kann es aber auch gefährlich werden, weil dadurch viele Sicherheitsvorkehrungen umgegangen werden können.

Folgend sehen wir Ubuntu 20.04-LTS, natives BS, ein NoMachine-Client verbunden mit Ubuntu-Server auf einem anderen Laptop mit dem Wintersee-Hintergrund, und auch zwei KVM-Gäste: Windows und Ubuntu.



Remote-Desktop gegen ssh -X

Beim Projekt-Testen mit ssh -X liegt keine Frage von Bildqualität und Auflösung vor. Es ist alles nativ, und muss nach dem Benutzer- und Hostname gesucht werden um heraus zu finden auf welchem Computer gerade gearbeitet wird.

SSH Desktop-Tunneling wäre auch möglich mit VNC, das wurde im Projekt aber nicht getestet.

Beim Remote-Desktop zwischen den zwei Laptops, W530 und T520, war die Bildqualität mit keinem Programm genau so gut wie nativ. Der Grund dafür: Wenn der Server eine viel niedrigere Auflösung als der Client hat, werden die Schriftarten sehr klein angezeigt. Bei einer Vergrößerung des RD-Fensters und Anpassung der Auflösung, wird natürlich alles verschwommen.

Optimal funktioniert RD folglich zwischen Maschinen mit ähnlichen Bildschirmauflösungen.

Alles in allem, SSH mit oder ohne die X Fahne, funktioniert tadellos, und RD wurde weniger eingesetzt.

Unten sehen wir den gleichen NoMachine-Client von oben, mit Ubuntu-Server [ki@dusk](#) im zweiten Laptop.

[ki@dusk](#) weist seine IP-Adresse mit

```
$ ifconfig | grep -E 'inet.*192'
```

vor.

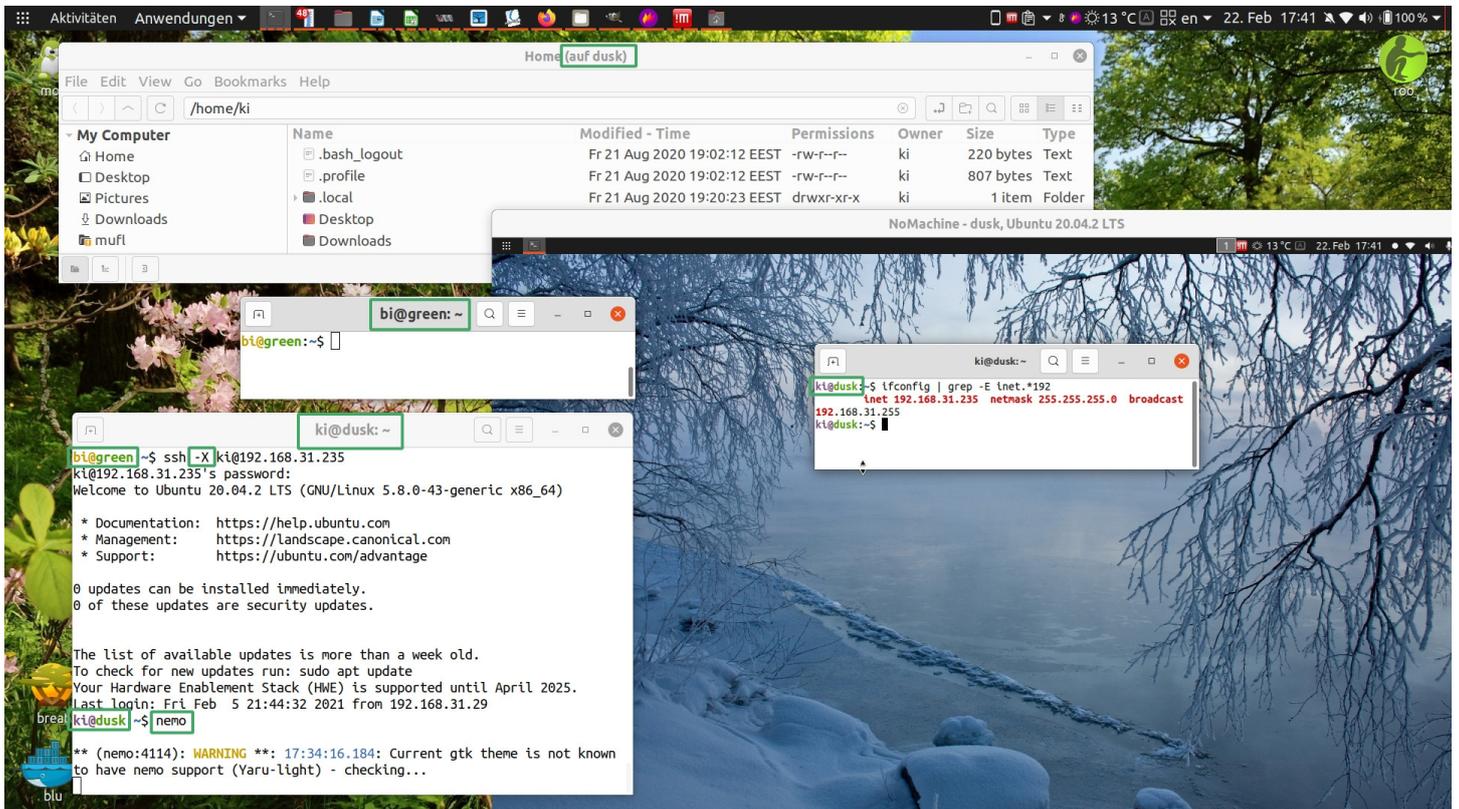
[bi@green](#), das native BS vom Laptop, verbindet sich mit [ki@dusk](#)

```
$ ssh -X ki@192.168.31.235
```

Danach mit

```
$ nemo
```

wird Nemo von [ki@dusk](#), auf dem Desktop von [bi@green](#) aufgerufen.



Im oberen Bild ist es deutlich dass die Schriftarten des Servers sehr klein angezeigt werden, wie weiter oben erwähnt wurde.

BS-Backup und -Klon-Erstellung als Konzept des Verfassers

Von Windows kennen wir nur eine BS-Image-Sicherung. In Linux ist für diesen Zweck möglich, einfach alle Dateien samt verschiedener Eigenschaften zu kopieren.

Rsync

Rsync ist das Terminalprogramm für diese Aufgabe.

Es sind auch Rsync-GUI-Frontends vorhanden, mit geliebtestem das, in Mint vorinstalliert, Timeshift. Es wurde einmal erfolglos damit versucht eine Fedora-Installation wiederherzustellen.

Rsync hat so viele Optionen, die wurden nie in eine GUI reinpassen.

Nach sehr vielen Experimenten wird immer mit Erfolg folgendes verwendet:

```
# rsync -aAHXsh --delete --preallocate --info=progress2,stats2 <quelle>/ <ziel>/
```

-a - Archivierungsmodus, und impliziert, beinhaltet, die Fahnen -rlptgoD, aber nicht -AHX

-r - Eine rekursion in den Ordnern durchführen

-l - Symlinks

-p - Zugriffsrechte

-t - Änderungszeiten

-g - Gruppen

-o - Eigentümer

-D - Geräte- und Spezialdateien

-A - ACLs (impliziert -p)

-H - Harte Links

-X - Erweiterte Attribute

`-s` - oder `--protect-args` - Leerzeichen werden nicht in Namen gespaltet. Das kann uns schützen wenn irgendwo aus versehen Dateinamen mit Leerzeichen vorhanden sind. Diese Option wird eventuell eine neue Standardeinstellung von Rsync.

`-h` oder `--human-readable`

`--delete` - Dadurch werden alle irrelevanten Daten, die nicht in der Quelle vorhanden sind, vom Ziel gelöscht. Auf dieser Weise wird das Ziel ein Spiegelbild der Quelle.

`--preallocate` - Die größeren Dateien werden angrenzend gespeichert, und das Dateisystem wird nicht fragmentiert. Allerdings kopiert Rsync etwas langsamer dadurch.

`--info=progress2,stats2` - Hier wird der Fortschritt angezeigt und zum Schluss Statistiken: wie viele Dateien wurden erstellt, gelöscht usw.

Der Schrägstrich an der Quelle ist sehr wichtig, weil ohne ihn, werden nicht die Inhalte des Ziels aktualisiert, sondern die Quelle wird ins Ziel kopiert, also dadurch werden die Daten durch eine Synchronisierung sich verdoppeln.

Der Befehl darf nur als Root ausgeführt werden.

Woher soll der Befehl ausgeführt werden

Das Quell-BS darf nicht gebootet sein.

Es wurde versucht den Befehl von einer unabhängigen Partition der gleichen Festplatte auszuführen. Der Klon bootet nicht, und wenn wir die Dateien der Quelle und des Ziels mit

```
sudo find <Pfad> | wc -l
```

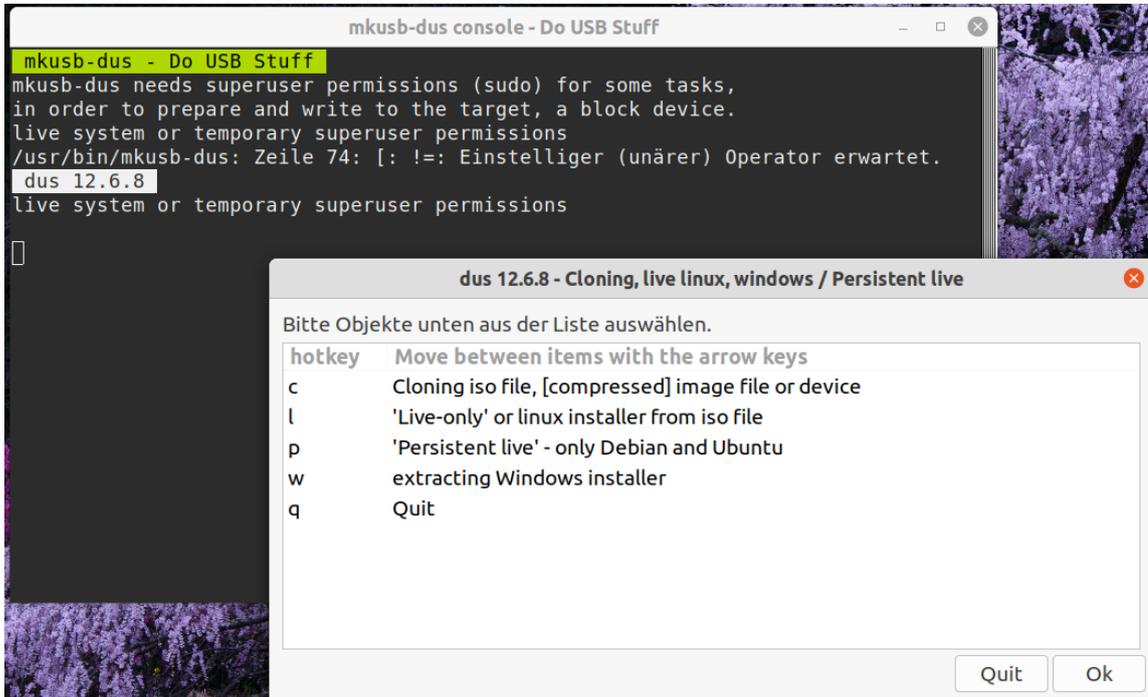
zählen, fehlen vom Klon ungefähr 10 Dateien.

Es ist also notwendig, aus einem externen BS den Befehl auszuführen.

Persistent-Live-USB

Die offensichtliche Lösung wäre vom üblichen Ubuntu-Live-USB zum Rsyncen. Wenn dieser Vorgang aber öfters durchgeführt wird, ist das nicht optimal. Eine bessere Lösung ist ein **Persistent-Live-USB**.

Mit dem Programm **Mkusb**, ist das unter Ubuntu sehr leicht zu bewältigen. Dadurch wird eine zusätzliche **Casper-Partition** erstellt, worauf Programme und Benutzereinstellungen installiert und gespeichert werden. Es wird außerdem eine Partition für Daten erstellt.



Wo sollen die Klone erstellt und aktualisiert

Unten wurde die Partition 14ubuk2 auf die 16ubuk4, von einem Persistent-Live-USB, synchronisiert.

```

root@ubuntu:ubuntu# rsync -aAHXsh --delete --preallocate --info=progress2,stats2 /media
/ubunt/14ubuk2/ /media/ubunt/16ubuk4/
 7.90G 26% 40.45MB/s 0:03:06 (xfr#94302, to-chk=0/436609)
Number of files: 436,609 (reg: 350,063, dir: 49,408, link: 36,932, dev: 9, special: 197
)
Number of created files: 103,275 (reg: 85,329, dir: 16,930, link: 992, special: 24)
Number of deleted files: 99,963 (reg: 82,562, dir: 16,452, link: 939, special: 10)
Number of regular files transferred: 94,302
Total file size: 29.83G bytes
Total transferred file size: 7.90G bytes
Literal data: 7.90G bytes
Matched data: 0 bytes
File list size: 5.77M
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 7.92G
Total bytes received: 10.47M

sent 7.92G bytes received 10.47M bytes 42.32M bytes/sec
total size is 29.83G speedup is 3.76
root@ubuntu:ubuntu#

```

Auf der Notebook-Festplatte sind immer drei Klone vorhanden. Im Fall eines unbeheblichen Fehlers, stehen zwei bootfähige Klone immer parat.

Hierfür ist es wichtig häufig genug die Klone zu synchronisieren, so dass wenn wir einen anderen Klon einsetzen, nicht über viel verlorene Arbeit ärgern müssen.

Bezüglich externer Klone, anfangs wurden sie auf USB-Sticks verschiedener Hersteller erstellt und synchronisiert aber später nur auf Toshiba externe Festplatten.

Die USB-Sticks auch wenn USB3, weisen im Vergleich mit den externen Festplatten, eine viel niedrigere Geschwindigkeit auf, sie werden sehr heiß und auch schnell korrupt. SSDs sind in Bezug darauf auch nicht geeignet dass vom ständigen Schreiben schnell abgenutzt werden.

Auf Verlängerungskabeln muss auch aufgepasst werden, da viele USB3-Kabel, die externen Festplatten nicht richtig erkennen. Die dickere Version von [Oricokabeln](#) haben sich verlässlich erwiesen.

Ferner, die externen Festplatten wurden genau so wie die internen partitioniert und ihre Klone werden auch bootfähig. Die funktionieren aber nur in Verbindung mit demjenigen Computer von dem sie kopiert wurden, die sind nicht portabel. Dazu wird es später verfeinert.

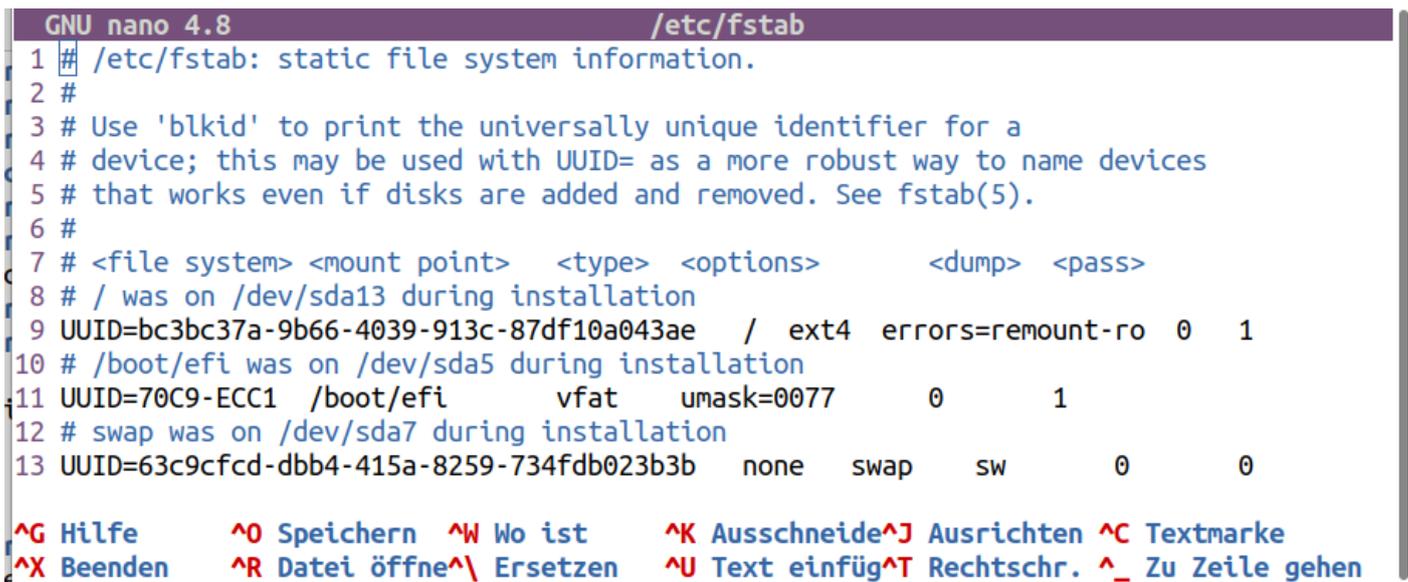
UUIDs in FSTAB ändern

Damit die Klone booten können, müssen in der FSTAB-Datei, die richtigen UUIDs für alle relevanten Partitionen geschrieben werden: also die UUID der Rootpartition brauchen wir immer, und wenn die Klone von einem externen Laufwerk booten sollen, wo wir auch ESP- und SWAP-Partitionen erstellt haben, die sollen auch umgeschrieben werden.

```
$ lsblk -fm
```

Die Ausgabe druckt UUIDs für alle Partitionen aus.

```
$ sudo nano <Klonpfad>/etc/fstab
```



```
GNU nano 4.8 /etc/fstab
1 # /etc/fstab: static file system information.
2 #
3 # Use 'blkid' to print the universally unique identifier for a
4 # device; this may be used with UUID= as a more robust way to name devices
5 # that works even if disks are added and removed. See fstab(5).
6 #
7 # <file system> <mount point> <type> <options> <dump> <pass>
8 # / was on /dev/sda13 during installation
9 UUID=bc3bc37a-9b66-4039-913c-87df10a043ae / ext4 errors=remount-ro 0 1
10 # /boot/efi was on /dev/sda5 during installation
11 UUID=70C9-ECC1 /boot/efi vfat umask=0077 0 1
12 # swap was on /dev/sda7 during installation
13 UUID=63c9cfcd-dbb4-415a-8259-734fdb023b3b none swap sw 0 0

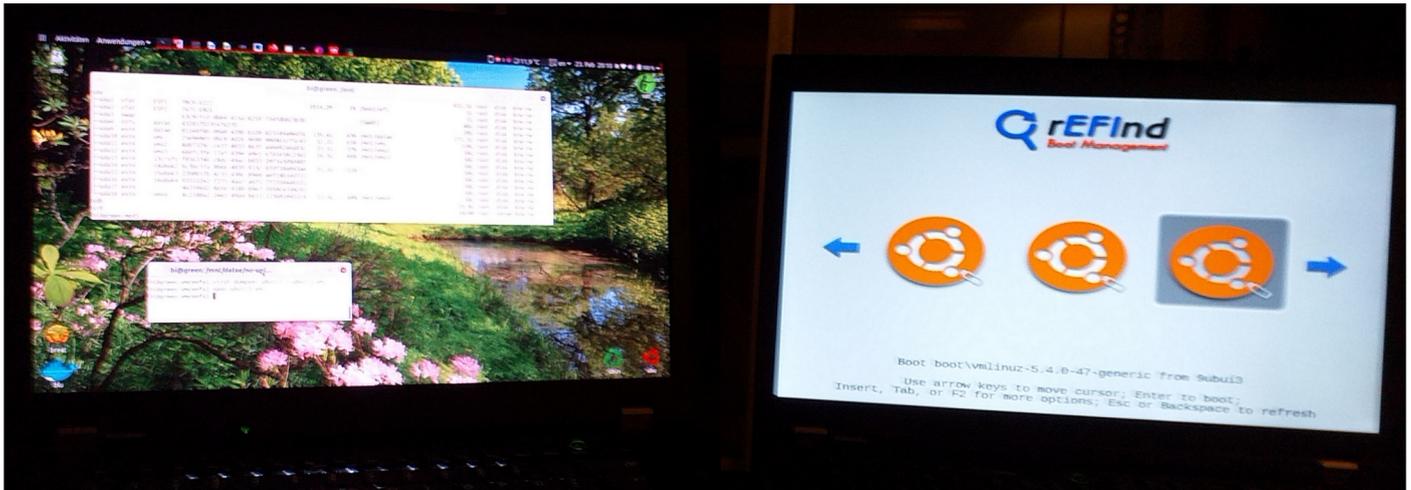
^G Hilfe      ^O Speichern ^W Wo ist    ^K Ausschneide ^J Ausrichten ^C Textmarke
^X Beenden    ^R Datei öffne ^\ Ersetzen  ^U Text einfüge ^T Rechtschr. ^_ Zu Zeile gehen
```

Booten mit dem rEFInd-Bootmanager von Roderick Smith

Nach langer Forschung, der [rEFInd-Bootmanager von Roderick W. Smith](#) hat sich als bester UEFI-Bootloader erwiesen. Der kann auf der Festplatte installiert werden, aber es ist besser ihn von einem USB-Stick zu betreiben.

Alle [GPT/UEFI-Linux-Systeme](#) können damit gebootet werden, ohne dass vom BS ein Bootloader in Anspruch genommen werden muss. Einfach mit **F12** oder ähnliches das Bootmedium auswählen, [rEFInd-Live-USB](#) booten, Klon auswählen, dazu werden die Partitionbezeichnungen aufgelistet, und booten.

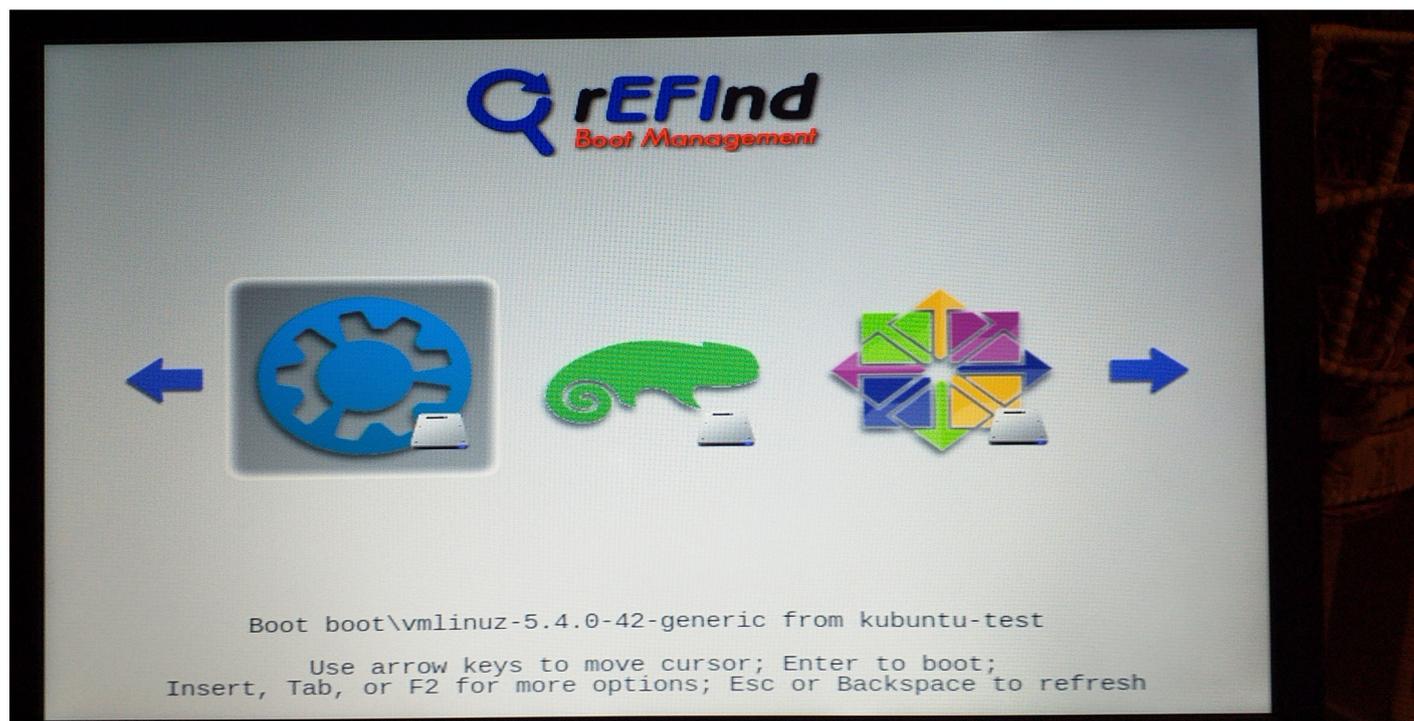
Im nächsten Bild ist das W530-Thinkpad links mit einem Ubuntu-BS und rechts das T520-Thinkpad mit dem rEFInd-Bootloader-Menü.



Folgend sind drei Ubuntu-Klone auf der externen Festplatte von T520-Thinkpad,



und Kubuntu, OpenSuse und CentOS auf der Festplatte des T520-Thinkpads.



Jedes Mal wenn Klone aktualisiert werden, wird es von einem anderen Klon gebootet um sicher zu stellen, dass alles einwandfrei funktioniert.

Bevor ein Klon zum ersten Mal gebootet wird, erkennen wir beim genauen Hinsehen des Dateisystems, dass einige Links als gebrochen aufgelistet sind, die werden aber beim Booten aufgebaut.

Grub-Bootloader für unsere Arbeitspartition installieren

Wenn ein Klon zum ersten Mal geboot wird,

```
$ lsblk -fm
```

um sicher zu stellen dass /, /boot/efi und [SWAP] an den richtigen Partitionen eingehängt sind.

Um GRUB zu installieren

```
$ sudo grub-install /dev/sdx
```

Sdx ist meistens Sda. Dann GRUB aktualisieren mit

```
$ sudo update-grub
```

Wenn wir neu booten steht unsere neue Arbeitspartition schon an der ersten Stelle vom GRUB-Bootmenü.

Portabilitätsbeschränkungen von Ubuntu-Installationen

Ubuntu-Systeme die auf einer internen Festplatte installiert wurden, könnten im anderen Laptop des Projekts nach vielen Fehlermeldungen booten, aber kurz danach frieren immer ein. Kubuntu-Systeme obwohl theoretisch auch Ubuntu sind, könnten beim oberen Versuch gar nicht booten.

Die BS-Klone können problemlos auf einer externen Festplatte eingesetzt werden so lange die externe Festplatte mit dem Klon-Quelle-Laptop verbunden ist, aber nicht wenn die externe Festplatte am anderen Laptop angeschlossen wird.

Installationen externer Festplatten

Installationen die auf der externen Festplatte von einem Notebook durchgeführt wurden, haben am anderen Laptop funktioniert aber anfangs mit folgenden Problemen.

Authentifizierungsfehler an der Bildschirmsperre

Funktionierende Lösungen wurden von <https://askubuntu.com/questions/1032624/ubuntu-18-04-authentication-error-on-logging-after-lock> entnommen.

Am Ende der Dateien

`/etc/sysctl.d/99-sysctl.conf` und `/etc/sysctl.conf`

die Zeile

```
fs.inotify.max_user_watches = 524288
```

hinzufügen, und dann

```
$ sudo sysctl -p
```

```
$ sudo dpkg-reconfigure gnome-shell
```

SSH - gebrochene Pakete

Alle SSH-Pakete wurden per „`sudo apt purge`“ entfernt und wieder installiert.

Installationen auf USB-Sticks

Das war ein Misserfolg, weil die USB-Sticks fangen, vom ständigen Schreiben, schon nach 10 Tagen, korrupt zu werden an.

Die Option von einem Persistent-Live-USB, die weiter oben diskutiert wurde, kann nicht als Haupt-BS verwendet werden. Es wird schnell deutlich dass nicht alle möglichen Pakete installiert werden können, und wenn ein solches System ständig aktualisiert wird, ist es auch schnell korrupt, nach Forenberichten und -anweisungen.

Den persönlichen Ordner in eine andere Installation kopieren

Da sind meistens Konfigurationen die übertragen werden, die Pakete müssen wieder manuell installiert werden.

Probleme treten hier auf, wenn das BS dass mit dem Home-Verzeichnis synchronisiert wurde, einen anderen Benutzernamen hat. Weiter unten, unter der Überschrift „VM umbenennen“ wird erläutert wie der Benutzername richtig geändert werden kann.

Virtuelle Maschinen - VMs

Unter Linux ist VirtualBox von Oracle sehr beliebt, diesbezüglich gibt es aber einen großen Hacken. Die notwendigen Gasterweiterungen sind Closed-Source, und für den gewerblichen Einsatz nur gegen Entgelt verfügbar.

Weil VMs ein Kernteil von modernen IT-Systemen sind, sollten sie unbedingt native Linuxprogramme sein. Diese Programme sind als KVM oder KVM-Libvirt-Qemu bekannt, inklusiv Bibliotheken und Abhängigkeiten die wir brauchen um VMs zu erstellen und verwalten. Noch dazu, Virtmanager kann als GUI-Frontend installiert werden, und vor allem um die VMs in der graphischen Konsole anzuzeigen.

Leistungsvergleich - KVM gegen VirtualBox

KVM steht für **Kernel-based Virtual Machine**, ist der einzige Hypervisor der im Linux-Kernel eingebaut ist, und konvertiert Linux in einen Hypervisor-Typ-1 (HT1 - Bare-Metal) laut <https://www.redhat.com/de/topics/virtualization/what-is-KVM>.

VirtualBox ist nur ein Hypervisor-Typ-2, d.h. er fungiert als eine zusätzliche Schicht auf dem BS, und kein Teil des Kernels wie KVM, und deswegen sind seine VMs langsamer und brauchen mehr CPU-Zyklen, was auch in Forenthreads berichtet wird.

KVM gegen Xen

Eine Aufklärung zuerst: Viele missverstehen einen Hypervisor-Typ-1 (HT1) als ein eigenständiges BS, weil bei den gewerblichen Lösungen oft nicht offensichtlich wird, dass der HT1 durch ein Host-BS installiert wird, und dass ein Host-BS erforderlich ist, um viele Aufgaben zu bewältigen, wie Datenträgerabbild-Erstellung, Konfigurationsdateien-Bearbeitung, Netzwerk-Einrichtung usw.

Dementsprechend wird auch unter <https://stackoverflow.com/questions/45529207/why-xen-requires-an-os-though-it-is-a-type-i-hypervisor> erläutert.

In dieser Hinsicht muss zuerst Ubuntu, oder eine andere Linux-Distro, installiert werden, und dann die relevanten Pakete für KVM oder Xen, also die Einrichtung zwischen den beiden ist ungefähr gleich aufwendig.

Performance Evaluation of Xen, KVM, and Proxmox Hypervisors

Die umfassendste wissenschaftliche Arbeit die gefunden wurde, die KVM, Xen und dazu auch Proxmox vergleicht, hat den Titel „[Performance Evaluation of Xen, KVM, and Proxmox Hypervisors](#)“ geschrieben 2018 von Algarni, Rafi Ikbal, Alroobaea, Ghiduk und Nadeem, im „International Journal of Open Source Software and Processes“.

Die Ergebnisse sind zugunsten KVM.

„Die Experimente belegen dass KVM in den meisten Testkategorien übertrifft mit Ausnahme der Dateisystem-Leistung, wobei Xen setzt sich durch nur in der Kategorie Dateisystem-Leistung.

Dies bestätigt dass KVM besser für Anwendungen die CPU- und speicherintensiv sind, während Xen wäre besser für Dateisystem- und datenträgerintensive Programme.“ (Umgeschrieben und übersetzt, Seite 52)

Es ist auch nennenswert dass KVM von Google verwendet wird, und dass Amazon einen Umstieg vom XEN auf KVM plant.

KVM-Vorinstallationshandlungsweisen

Unter <https://help.ubuntu.com/community/KVM/Installation> entnehmen wir Einweisungen für die Hauptinstallation und Konfiguration. Unten werden die wichtigsten Punkte auf Deutsch dargestellt und umgeschrieben.

Zuerst müssen wir herausfinden ob der Prozessor Hardware-Virtualität unterstützt, und ob alle Virtualitätsoptionen in den [UEFI-Einstellungen](#) (was früher BIOS war) aktiviert sind.

```
$ egrep -c '(vmx|svm)' /proc/cpuinfo
```

Die Ausgabe 0 bedeutet das der Prozessor keine Virtualität unterstützt, die Ausgabe 1 oder mehr, dass sie unterstützt wird.

```
$ kvm-ok
```

zeigt uns ob KVM-Beschleunigung eingesetzt werden kann.

Wir müssen unbedingt einen 64 Bit Kernel verwenden um mehr als 2GB Arbeitsspeicher pro VM ordnen zu können.

```
$ uname -m
```

zeigt ob wir einen 64-Bit-Kernel verwenden mit der Ausgabe `x86_64`.

Anzeigen ob der CPU, 64 Bit ist mit

```
$ egrep -c `lm` /proc/cpuinfo
```

Die Ausgabe 0 steht für 32 Bit CPU. Eine Ausgabe von 1 oder höher steht für 64 Bit.

KVM-Installation

Die Hauptpakete installieren.

```
$ sudo apt install qemu-kvm libvirt-daemon-system libvirt-clients bridge-utils
```

Der Normalbenutzer des Systems, muss Mitglied der Libvirt- und KVM-Gruppen werden.

```
$ sudo adduser `id -un` libvirt
```

```
Adding user '<username>' to group 'libvirt' ...
```

```
$ sudo adduser `id -un` kvm
```

```
Adding user '<username>' to group 'kvm' ...
```

Es muss dann erneut angemeldet werden. Wenn

```
$ virsh list --all
```

keine Fehlermeldungen ausgibt, ist dann alles in Ordnung.

Um den Virtmanager zu installieren

```
$ sudo apt install virt-manager
```

Standard-Pool-Speicherstelle ändern

Eine Pool enthält die virtuellen Festplatten der VMs. Standardmäßig wird diese Pool unter `/var/lib/libvirt/images`, in der BS-Root-Partition gespeichert, die dadurch schnell voll wird, so wir wollen diese Speicherstelle ändern. Mit

```
$ virsh pool-edit default
```

öffnet sich die Standardpool-Xml-Datei in einem Texteditor, in dem wir die Speicherstelle ändern können.

```
GNU nano 4.8 /tmp/virshJEGHV0.xml
1 <pool type='dir'>
2   <name>default</name>
3   <uuid>9a049575-4697-4ec2-8f7b-f2cfa18c0acd</uuid>
4   <capacity unit='bytes'>0</capacity>
5   <allocation unit='bytes'>0</allocation>
6   <available unit='bytes'>0</available>
7   <source>
8   </source>
9   <target>
10    <path>/mnt/vms/kvm-vms</path>
11  </target>
12 </pool>
13
```

Von den Anweisungen an folgender Adresse

<https://serverfault.com/questions/840519/how-to-change-the-default-storage-pool-from-libvirt>

können wir die Standardpool auch löschen und eine neue erstellen.

VMs-erstellen und -bearbeiten

Mit Virtmanager können neue VMs erstellt werden. Dazu brauchen wir nur ein Installations-ISO. Vom Virtmanager-GUI können auch direkt, entsprechende Teile der XML-Datei konfiguriert werden.

Bei einer bestehenden VM namens „ubvir2“, die XML-Datei kann mit

```
$ virsh edit ubvir2
```

bearbeitet werden.

```
GNU nano 4.8 /tmp/virshWVG000.xml
1 <domain type='kvm'>
2   <name>ubvir2</name>
3   <uuid>111c0615-0502-42a3-bc24-c7d1f11d4a98</uuid>
4   <metadata>
5     <libosinfo:libosinfo xmlns:libosinfo="http://libosinfo.org/x
mlns/libvirt/domain/1.0">
6       <libosinfo:os id="http://ubuntu.com/ubuntu/20.04"/>
7     </libosinfo:libosinfo>
8   </metadata>
9   <memory unit='KiB'>10240000</memory>
10  <currentMemory unit='KiB'>10240000</currentMemory>
11  <vcpu placement='static'>8</vcpu>
12  <os>
13    <type arch='x86_64' machine='pc-q35-4.2'>hvm</type>
14    <loader readonly='yes' type='pflash'>/usr/share/OVMF/OVMF_CO
DE.fd</loader>
15    <nvram>/var/lib/libvirt/qemu/nvram/ubvir2_VARS.fd</nvram>
16    <boot dev='hd'>
17  </os>
18  <features>
19    <acpi/>
20    <apic/>
21    <vmport state='off'>
22  </features>
23  <cpu mode='custom' match='exact' check='partial'>
24    <model fallback='allow'>IvyBridge</model>
25    <topology sockets='1' cores='2' threads='4'>
26    <feature policy='require' name='ibpb'>
27    <feature policy='require' name='md-clear'>
28    <feature policy='require' name='spec-ctrl'>
29    <feature policy='require' name='ssbd'>
30  </cpu>
31  <clock offset='utc'>
32    <timer name='rtc' tickpolicy='delay'>
33    <timer name='pit' tickpolicy='catchup'>
34    <timer name='hpet' present='yes'>
35  </clock>
36  <on_poweroff>destroy</on_poweroff>
[ 167 Zeilen gelesen ]
^G Hilfe      ^O Speichern ^W Wo ist    ^K Ausschneid ^J Ausrichten
^X Beenden    ^R Datei öffn ^\ Ersetzen  ^U Text einfü ^T Rechtschr.
```

Gasterweiterungen

Diese werden automatisch installiert zumindest bei Ubuntu-Gästen die im Projekt eingesetzt wurden.

Die Gasterweiterungen sind die Spice- und VirtIO-Paravirtualitätts-Treiber.

Spicetreiber

Highlights einschließen bessere Bildqualität, automatische Skalierung der Bildauflösung bei Änderungen des VM-Fensters und Zwischenablage-Unterstützung zwischen Host und Gast.

VirtIO-Treiber

Die sind sehr wichtig als Paravirtualitätstreiber für die virtuellen Festplatten, die virtuelle Netzwerkschnittstelle und für andere Hardware.

Gasterweiterungen für Windows-Gäste

VirtIO- und Spicetreiber sind auch für Windows-Gäste vorhanden.

Die VirtIO können von <https://docs.fedoraproject.org/en-US/quick-docs/creating-windows-virtual-machines-using-virtio-drivers/index.html> heruntergeladen werden, und die Spicetreiber von <https://www.spice-space.org/download.html>

Für ihre Installation können die als virtuelle Sata-Cdroms zum Gast präsentiert werden, oder mittels Samba-Share verfügbar werden. Der VirtIO-Datenträgertreiber muss während der Windows-Installation installiert werden.

Hohe CPU

Für die unteren Tests wurde die Hälfte der CPU, also zwei Kerne - vier Threads, an die VMs zugewiesen, was bedeutet dass die CPU am Host ungefähr bis 400% belastet werden kann.

Systemweit CPU-Zyklen

Die Windows-VMs haben sehr hohe CPU-Zyklen aufgewiesen.

Wir können die XML-Datei auch in Virtmanager bearbeiten unter [Anzahl der CPUs:/XML](#). Die unten angezeigten Konfigurationen in den Kästen, sind rechts von einer WVM und links von einer Ubuntu-VM.

Standardmäßig war die Konfiguration, die wir ändern wollen, wie folgend:

```
<timer name="rtc" tickpolicy="catchup"/>
```

```
<timer name="pit" tickpolicy="delay"/>
```

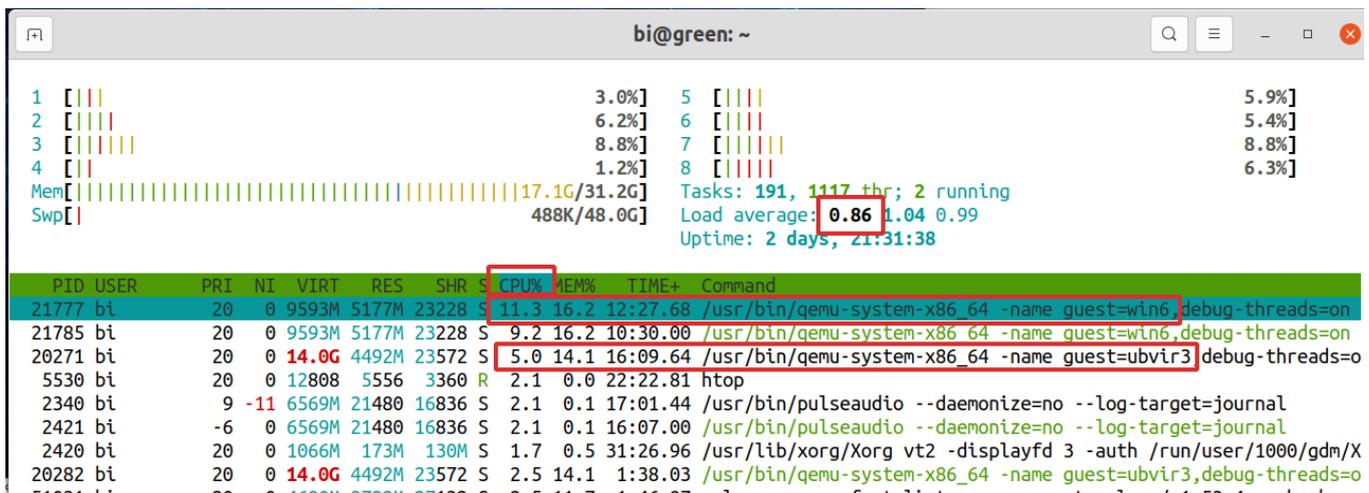
```
<timer name="hpet" present="no"/>
```

Durch die unteren Änderungen haben die CPU-Zyklen in der WVM sehr nachgelassen. In der UVM wurde keine Änderung festgestellt.

```
<feature policy="require" name="ssbd"/>
</cpu>
<clock offset="utc">
  <timer name="rtc" tickpolicy="delay"/>
  <timer name="pit" tickpolicy="catchup"/>
  <timer name="hpet" present="yes"/>
</clock>
<on_poweroff>destroy</on_poweroff>
<on_reboot>restart</on_reboot>
```

```
<feature policy="require" name="ssbd"/>
</cpu>
<clock offset="localtime">
  <timer name="rtc" tickpolicy="delay"/>
  <timer name="pit" tickpolicy="catchup"/>
  <timer name="hpet" present="yes"/>
  <timer name="hypervclock" present="yes"/>
</clock>
```

Unten sehen wir dass die Windows-VM 11.3% und die Ubuntu-VM 5% CPU-Zyklen verwendet. Mit acht CPU-Threads im W530-Thinkpad, das Maximum ist 800% und Load-Average 8.



Videobezügliche CPU-Zyklen

Außer dem Spice-Server,

Spice Server

Typ: SPICE-Server

Listen type: Address

Adresse: Hypervisor-Stand

Anschluss (Port): Auto

TLS Port: Auto

wurde auch der Virgil3d - Virtio-GPU Paravirtualitätstreiber aktiviert.

```
<video>
| <model type="virtio" heads="1" primary="yes">
|   <acceleration accel3d="yes"/>
| </model>
| <address type="pci" domain="0x0000" bus="0x00" slot="0x01" function="0x0"/>
</video>
```

Die UVMs weisen dennoch hohe CPU-Zyklen bei Videoausführungen an. Youtube-Videos laufen mit 210% CPU, VLC-Videos mit 300% CPU.

Eine andere Option wäre Intel-GVT-g, was aber nur ab der fünften CPU-Generation verfügbar ist, die W530-CPU des Projekt-Laptops ist nur dritte Generation - IvyBridge. Dazu müssen zwischen anderem einige Kernparameter geändert werden.

PCI-Durchleitung ist die beliebteste Option, muss aber eine GPU komplett an die VM durchgereicht werden, also der Ubuntu-host braucht eine andere GPU, deswegen ist so was nur möglich wenn die Hardware über zwei GPUs verfügt.

Es muss außerdem unterstrichen werden, dass die Portabilität der VMs im Fall einer GPU-Durchreichung aufwendig ist.

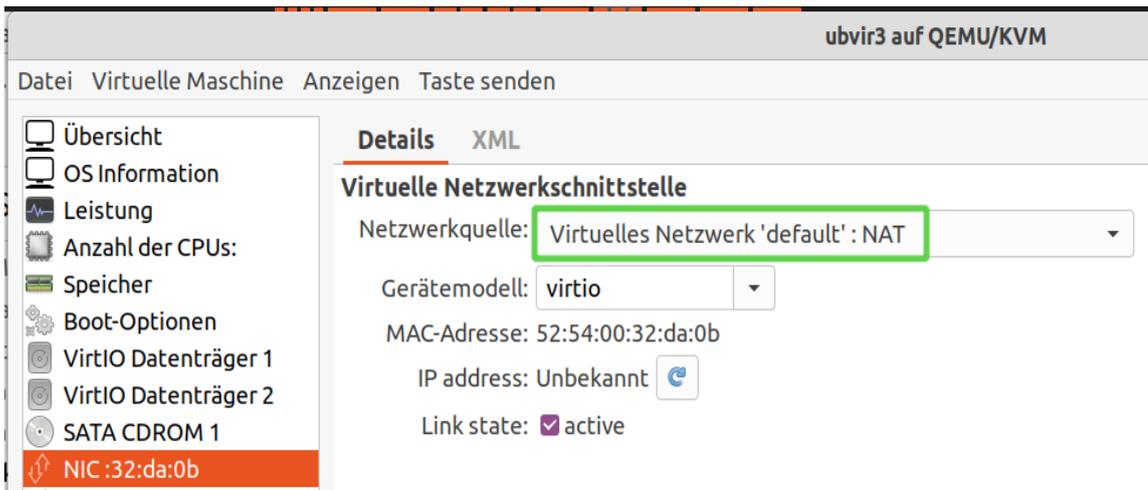
Unter https://wiki.archlinux.org/index.php/QEMU/Guest_graphics_acceleration werden einige dementsprechende Möglichkeiten zusammengefasst.

VM-Netzwerken

Linux

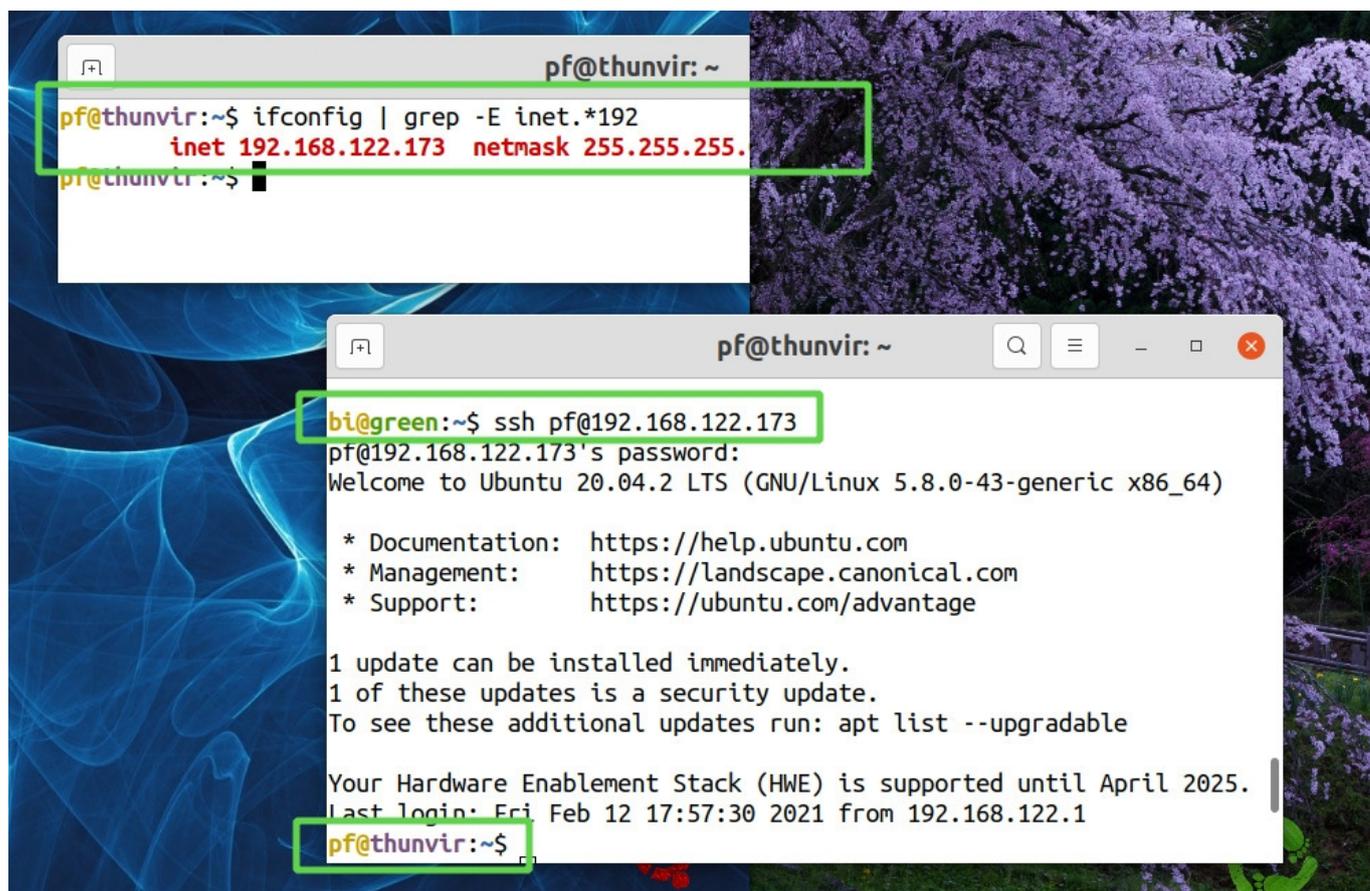
Das **Vorgabe-Nat-Netzwerk** wurde über WLAN verwendet, da unter KVM was anders läuft zwar über Ethernet problemlos, aber über WLAN wird sehr kompliziert. Das Gerätemodell ist Virtio.

In manchen Forenthreads wird berichtet, dass mit NAT, der Host kann sich zwar mit dem Gast verbinden, aber nicht umgekehrt.



Mit SSH kann von beiden Seiten eine Verbindung problemlos hergestellt werden.
Wie unter der Rubrik Netzwerken/SSH erwähnt wurde, eine private IP-Adresse kann mit
`$ ifconfig | grep -E 'inet.*192'`
ermittelt werden.

Unten, ist der Host `bi@green` rechts mit dem lilanen Hintergrund, und der Gast `pf@thunvir` links mit dem blauen Hintergrund. Im Gast wurde die IP-Adresse durch `ifconfig` ermittelt, und `bi@green` hat sich bei `pf@thunvir` angeschlossen, wobei der Prompt von `bi@green` in `pf@thunvir` geändert hat.



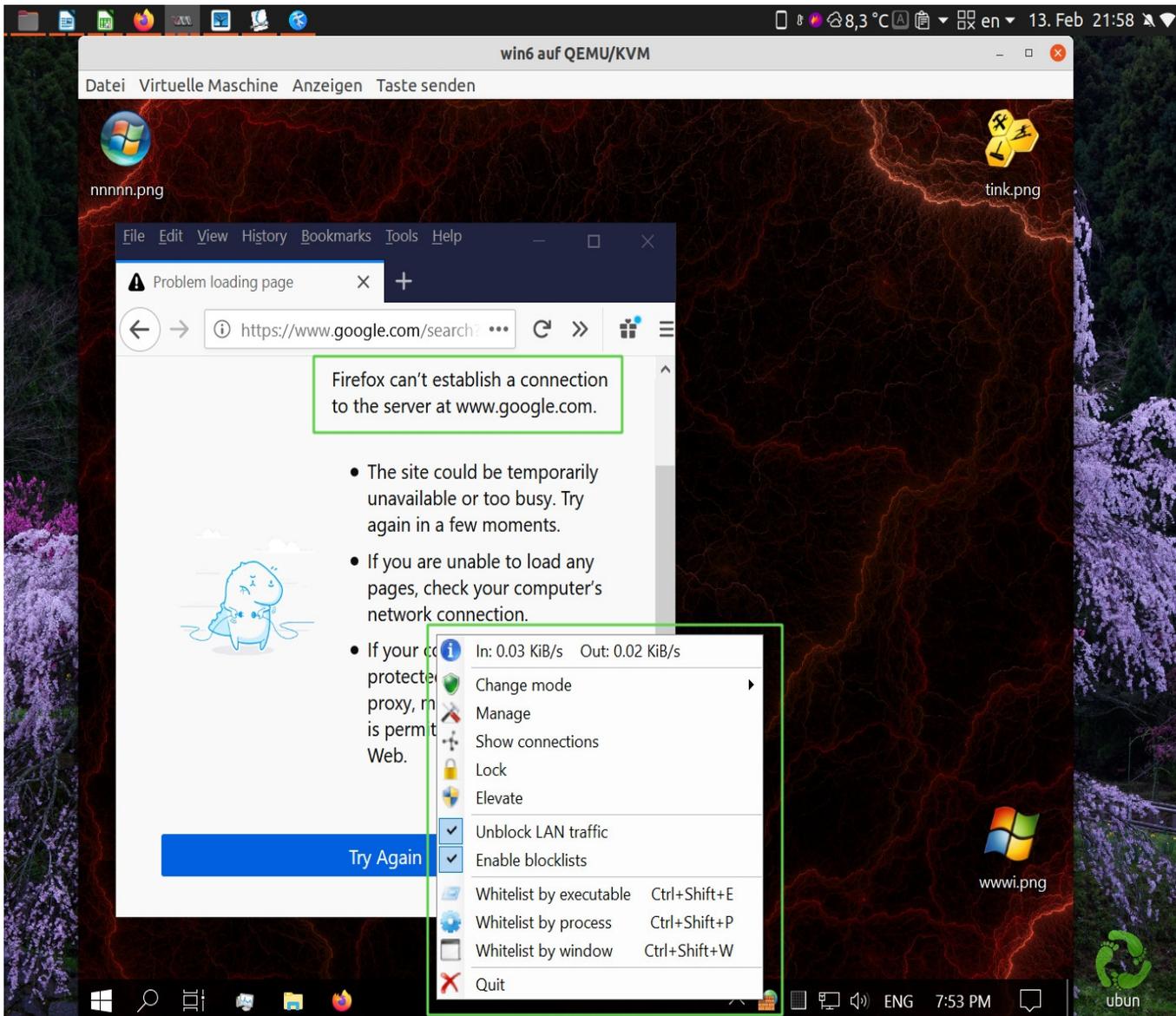
Die gleiche Vorgehensweise wurde ebenfalls ohne Probleme, zwischen zwei Ubuntu-Gästen unternommen.

Windows-VMs

Das Netzwerken mit einer Windows-VM (WVM), wurde anders gedacht. Wenn es unerlässlich sei, bestimmte Programme in einer WVM zu betreiben, dann aber im Sandkasten, so dass die WVM keine Sicherheitsgefahr darstellen kann.

Folgend ist eine WVM mit der [TinyWall-Firewall](#) installiert, die mit den Vorgabe-Einstellungen alles blockiert, also kein Internet. Es wurde aber die Option „Unblock LAN traffic“ getickt, damit die WVM sich mit dem Host verbinden kann.

Aus Sicherheitsgründen wurden die Windows-Linuxtools auch nicht aktiviert.

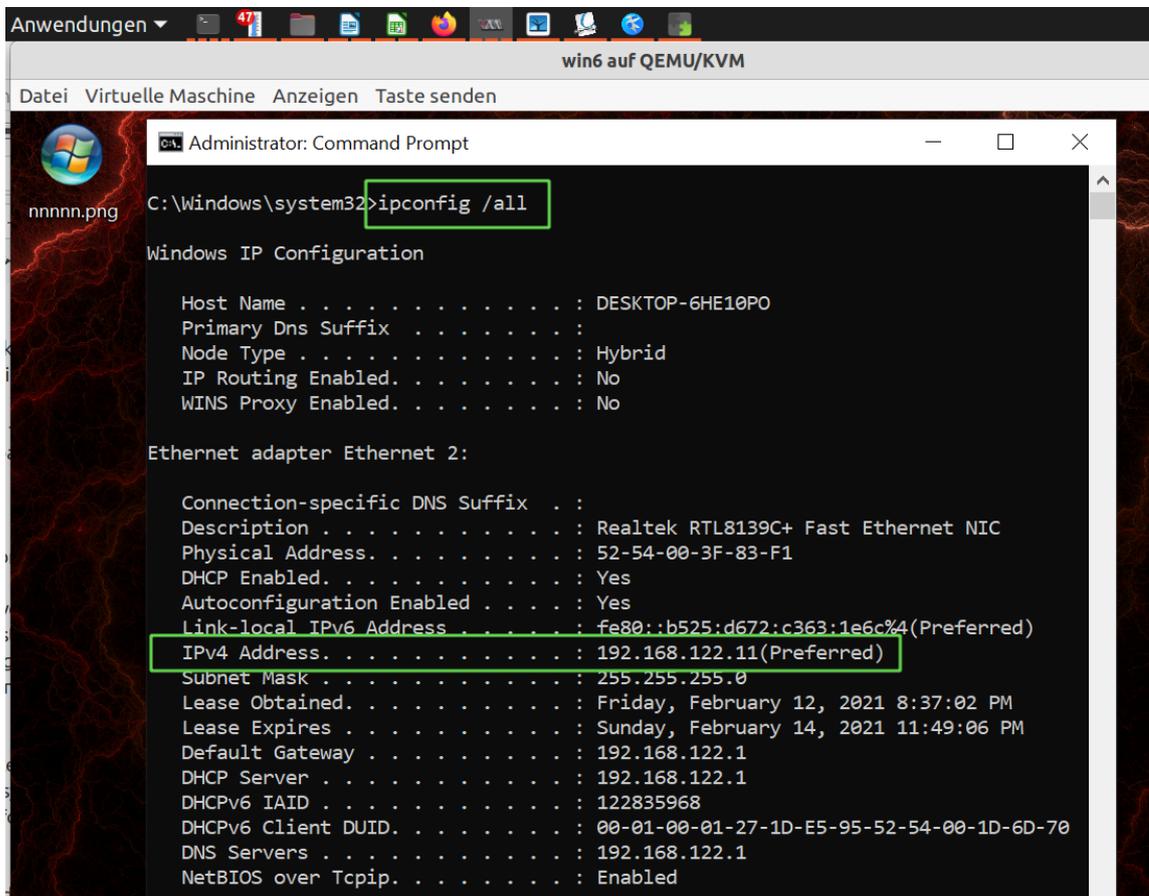


Im WVM einen Ordner namens `wguest-data` teilen.

Unter dem Ubuntu-Host einen Ordner `/mnt/smb` erstellen wo `wguest-data` eingehängt wird.

Die IP-Adresse im WVM ermitteln mit

```
> ipconfig /all
```



Im Ubuntu-Host den geteilten Ordner mit

```
$ sudo mount.cifs //192.168.122.11/wguest-data /mnt/smb -o  
username=<benutzername>,password=<passwd>
```

einhängen.

Weil aber das Passwort jetzt im Terminal gedruckt wurde, sollten wir lieber die Befehlszeile-Geschichte auch gleich löschen mit

```
$ cat /dev/null > ~/.bash_history ; history -c
```

[Virtuelle Maschinen kopieren - updaten - umbenennen](#)

[VM klonen](#)

Einzelne virtuelle Datenträger (VDs) wurden ausschließlich für Daten verwendet, die von mehreren Ubuntu-VMs eingesetzt werden. Wenn aber eine VM kopiert wird, ist es unmöglich den Daten-VD mit zu klonen. Das ist auch nicht gewollt. Also den Daten-VD erst mittels Virtmanager von der VM die wir kopieren wollen entfernen, und die VM dann klonen.

```
$ virt-clone -o <vm-name> -n <klone-name> -f <neuer-virt-datenträger-pfad>
```

Das kopiert den Qcow2 Datenträger und die Konf-Dateien: die Datei `vm_name.xml`, wo die Konfiguration der VM geschrieben ist, und die Datei `vm_name_VARS.fd`, das ist die `Nvram-Datei`. Zusätzlich wird eine Log-Datei kopiert.

Der oben erwähnte Daten-VD, kann zwar gleichzeitig an vielen VMs zugefügt werden, aber Vorsicht! Es dürfen nie zwei VMs gleichzeitig den gleichen Qcow2-VD anhängen und die Daten zugreifen, weil sie dadurch korrupt werden.

VM löschen

```
$ virsh undefine <vm-name> --nvram --storage <vd-pfad>
```

VM aktualisieren

Ein VM-Klon wird aktualisiert mit

```
rsync -aAHXsh --inplace --delete --info=progress2,stats2 <quelle> <ziel>
```

Hier wird also die Qcow2-Datei aktualisiert, und anders als bei der Rsync-Verwendung um ein BS zu kopieren/aktualisieren, für Qcow2-Dateien brauchen wir die Option `--inplace`, wir können aber die Option `--preallocate` die bei der BS-Aktualisierung verwendet haben, hier nicht einsetzen, da Rsync damit zu lange braucht, oder sogar hängt.

VM umbenennen

Wir dürfen nie zwei Klone gleichzeitig im gleichen Netzwerk betreiben, oder wir müssen zuerst einem Klon eine neue Identität geben, was bedeutet: Benutzer- und Hostname sollen geändert werden.

Des weiteren damit wir die VMs besser unterscheiden können, sollte der VM-Name auch geändert werden, falls bei der Klonerstellung kein triftiger Name zugeteilt wurde.

Benutzername ändern

Das Konzept wurde von <https://www.linuxuprising.com/2019/04/how-to-change-username-on-ubuntu-debian.html> entnommen.

Angenommen wir wollen `altname` durch `neuname` ersetzen.

Zuerst erstellen wir einen temporären Benutzer namens `tempuser`, und geben ihm Sudo-Berechtigung.

```
$ sudo adduser tempuser
```

```
$ sudo usermod -aG sudo tempuser
```

Abmelden und als `tempuser` anmelden, um den Benutzernamen, das Heimverzeichnis und die Gruppe zu ändern.

```
$ sudo usermod -l neuname -d /home/neuname -m altname
```

`-l, --login`

`-d, --home` - Das neue Heimverzeichnis des Benutzers

`-m, --move-home` - Verschiebt den Inhalt des alten Heimverzeichnisses des Benutzers ans neue Ziel.

Den Gruppennamen ändern mit

```
$ sudo groupmod -n neuname altname
```

Einen symbolischen Link vom neuen zum alten Home-Verzeichnis erstellen.

```
$ sudo ln -s /home/neuname /home/altname
```

Der letzte Schritt ist sehr wichtig, weil viele Anwendungskonfigurationen an das alte Home-Verzeichnis zeigen, das jetzt nur ein Link zum neuen Heimverzeichnis ist, sonst funktionieren auf einmal einige Programme nicht mehr.

Verschiedene Programme werden nach den obigen Änderungen melden, dass ein anderer Benutzer eingeloggt ist oder ähnliches, aber nach der ersten Abweisung der Meldung läuft alles normal und erscheint die Meldung beim nächsten Mal nicht mehr.

Hostname ändern

Die Vorgehensweise wurde von <https://itsfoss.com/change-hostname-ubuntu/> entnommen.

```
$ sudo hostnamectl set-hostname <neuer-hostname>
```

Es kann sein dass der Hostname auch manuell geändert werden muss im Inhalt zweier Dateien wie folgt:

```
$ sudo nano /etc/hostname
```

```
$ sudo nano /etc/hosts
```

VM-Name ändern

Ein Teil des Konzepts wurde von <https://kifarunix.com/how-to-rename-kvm-virtual-machine-with-virsh-command/> entnommen.

Angenommen wir wollen eine VM namens `alt` in `neu` umbenennen.

Zuerst ändern wir den Namen des VD mit

```
$ sudo mv alt.qcow2 neu.qcow2
```

Die Xml-Datei editieren mit

```
$ virsh dumpxml alt > neu.xml
```

```
$ nano neu.xml
```

Überall wo `alt` steht, muss jetzt `neu` geschrieben werden.

```
$ virsh undefine --nvram alt
```

Wenn unsere VM eine Nvram-Datei hat, der Befehl wird ohne die Fahne `--nvram` scheitern.

```
$ virsh define neu.xml
```

VM-Sicherungskopien

Wenn wir eine VM migrieren wollen, oder auch für den Fall eines Hardware-Ausfalles, wir benötigen die Qcow2- und die Xml-Datei zu sichern.

Die XML-Dateien können wir mit

```
$ rsync -aAHXsh -delete --info=stats2 /etc/libvirt/qemu/ /backup-ziel/
```

sichern.

Die folgenden Migrations-Anweisungen sind zwar ergänzend, wurden im Projekt aber nicht getestet.

Wenn beide Hosts laufen und über die gleichen BSs und KVM-Libvirt-Qemu verfügen, wir können die VM mit

```
$ virsh migrate <vm-name> <ziel>
```

migrieren.

Wir sollten sonst, und vor allem im Fall eines Hardware-Ausfalles, wie unter dem Überschrift „`VM-Name ändern`“ vorgehen.

Die Xml-Datei editieren mit

```
$ virsh edit <vm-name>
```

Hier ist es wichtig das neue VD-Pfad anzugeben, und gegebenenfalls die Netzwerk-Konfiguration zu ändern. Anschließend

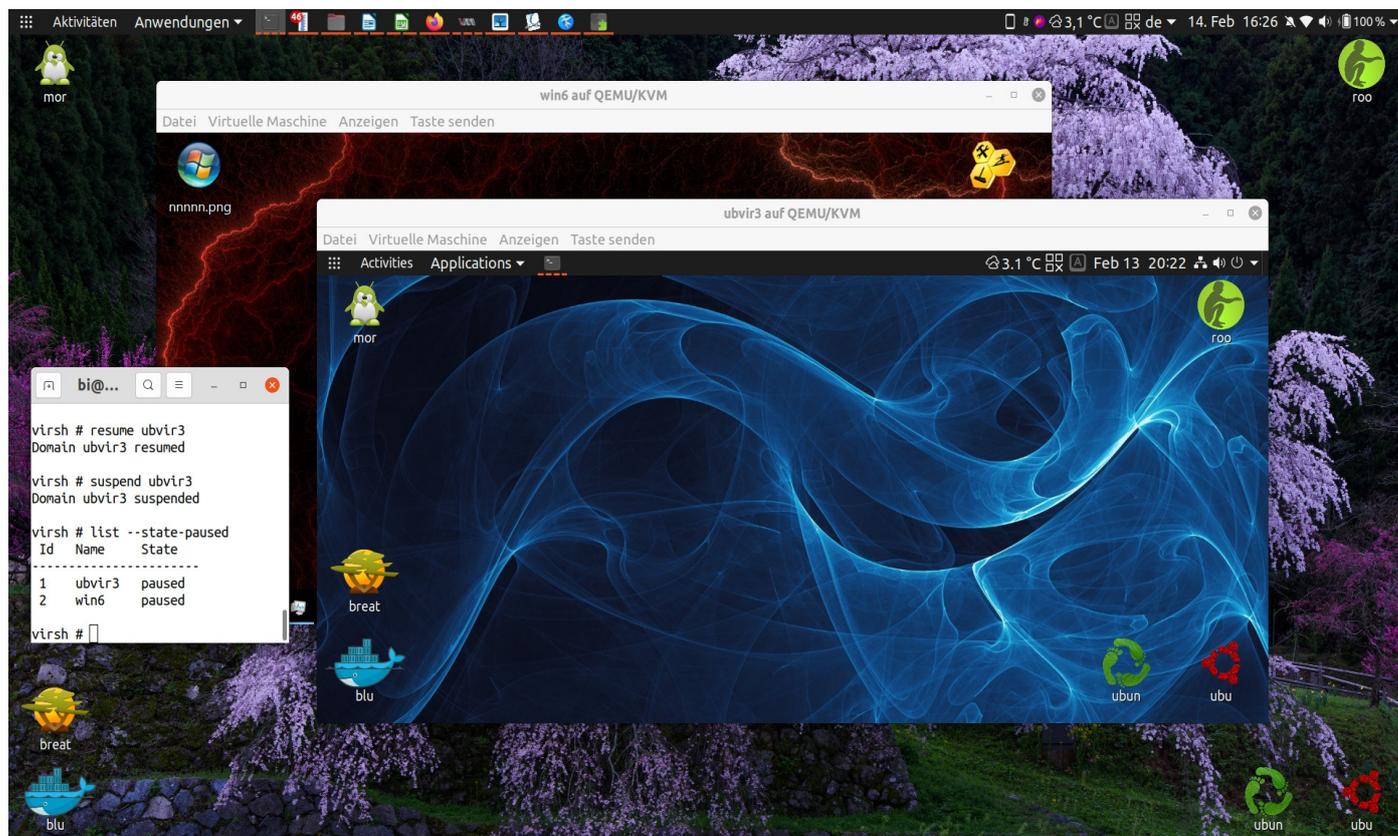
```
$ virsh define <vm-name>.xml
```

VM-Verwaltung und -Ressourcen

Die VMs können durch den Virtmanager-GUI verwaltet werden, aber auch mit vielen Kommandozeile-Programmen, wie weiter oben erwähnt.

Alternativ und ergänzend zum Virtmanager ist das Befehlszeile-Programm Virsh. Damit können wir die VMs unter vielen anderem: starten, pausieren, wiederaufnehmen usw.

Unten ist ein Ubuntu-Host mit zwei Gästen, einmal Ubuntu und einmal Windows. Im Terminal-Fenster, wurden ein paar Virsh-Befehle ausgeführt.



Oben sehen wir, dass beide VMs pausiert sind, und deswegen sie keine CPU-Zyklen in Anspruch nehmen. Speicher wird aber weiterhin verwendet.

Probleme mit Windows-VMs

Wenn wir Pools löschen und andere erstellen, VMs umbenennen und verschieben, die Ubuntu-Gäste booten weiterhin ohne Probleme. Nicht so mit den WVMs, in einem solchen Fall, die booten nicht mehr.

Voran genau das liegt, wurde nicht mehr recherchiert.

Anfänglich wurde angenommen dass WVMs für spezifische Programme nötig seien. Im Laufe des Projekts ist aber klar geworden dass dieser Ansatz sehr wenig zutrifft, wie z.B. unter der Rubrik: Pdf-Fazit, weiter oben dargestellt wurde.

Die WVMs offenbaren ein sehr hohes Umständlichkeit-Leistungs-Verhältnis zumal unter Linux eine Vielzahl von Lösungen für jede Aufgabe verfügbar ist.

Speicher

Um bequem VMs einzusetzen, sollte ein Minimum von 32GB Speicher im Rechner verbaut sein. Das W530-Thinkpad hat ursprünglich nur 8GB Speicher gehabt, das auf 32GB aufgerüstet wurde.

*-memory

Beschreibung: Systemspeicher

Physische ID: 7

Steckplatz: Systemplatine oder Hauptplatine

Größe: 32GiB

*-bank:0

Beschreibung: SODIMM DDR3 Synchron 1600 MHz (0,6 ns)

Produkt: CT102464BF160B.M16

Hersteller: 859B

Physische ID: 0

Seriennummer: [REDACTED]

Steckplatz: ChannelA-DIMM0

Größe: 8GiB

Breite: 64 bits

Takt: 1600MHz (0.6ns)

Die Wahl fiel auf [Crucial CT102464BF160B - 8GB DDR3L-1600 SODIMM x 4 = 32GB](#) aufgrund guten Preis-Leistungs-Verhältnisses die sich auch bewiesen hat.



VMs als Haupt-BS gegen Distroerstellung

Ubuntu-Installationen sind nicht portabel oder nur in Spezialfällen wie es unter dem Überschrift: Portabilitätsbeschränkungen von Ubuntu-Systemen weiter oben, erläutert wurde.

Es wurde auf die Frage eingegangen welche ist ein leichter und praktischer Weg eine bestehende Installation überall, als Spiegelbild einzusetzen. Gedanken über eine mögliche Lösung sind über zwei allgemeine Überschriften geordnet, mit Virtualitätseinsetzung und ohne, oder mit einem Mix von beiden.

Eine Einsetzung der Ansätze unter den zwei folgenden Rubriken sind für ein künftiges Projekt geplant.

Distroerstellung

Wenn das BS installiert wird, werden hardware-spezifische Treiber installiert und entstehen dazugehörige Konf-Dateien. Im Rahmen des Projekts wurde recherchiert ob und wie wäre es möglich eine bestehende Installation auf anderer Hardware einzusetzen.

Eine leicht einsetzbare Funktion die nach neuen Geräten aller Art sucht, und dann die alten Treiber durch neue ersetzt, existiert nicht.

Manche Distros wie MX und Manjaro, verkünden dass die Möglichkeit gibt, das BS anzupassen und danach eine eigene Distro daraus zu erzeugen. Bei den meisten von diesen Lösungen, was erreicht wird, ist neue Pakete in die Vorgabeinstallations-ISO hinzufügen, alias [Remastering](#), aber kein Spiegelbild einer bestehenden Installation samt aller Konfigurationen die der Benutzer unternommen hat.

Das kann aber erreicht werden durch einen mehrstufigen Prozess bekannt als „Live CD Customization From Scratch“ der unter <https://help.ubuntu.com/community/LiveCDCustomizationFromScratch> ausführlich dokumentiert ist.

Virtualisierungslösung

Virtualisierung einer physischen Installation

Wir können eine physische Installation in eine KVM-VM umwandeln, mit [Virt-p2v](#), was zusammen mit den anderen Virt-Paketen installiert wird, muss aber als Live-USB eingesetzt werden.

Das kann entweder mit [virt-p2v-make-disk](#) oder mit [virt-p2v-make-kickstart](#) erreicht werden. Die erstellte VM wird dann auf anderer Hardware mittels KVM oder Xen eingesetzt.

Netzwerk-Synchronisierung der VMs

Die VMs können durch das Netzwerk mit

```
# rsync -aAHXsh --inplace --delete --info=progress2,stats2 \  
e- ssh <quelle> \  
benutzer@ip-adresse:<ziel>
```

synchronisiert werden.

Fazit: Distroerstellung gegen Virtualisierung

VMs brauchen immer mehr CPU-Zyklen im Vergleich mit einer physische Installation für die gleiche Aufgabe. Dieser Unterschied fällt nicht gleich auf mit Befehlszeile-Programmen oder LibreOffice-Anwendungen.

Wie es unter der Rubrik „[Videobezügliche CPU-Zyklen](#)“ erläutert wurde, Verbesserungsmöglichkeiten benötigen Kernkonfigurationen, neuste Hardware und/oder GPU-Durchreichung. Wenn die VMs portabel werden sollen, die GPU-Durchreichung wird sehr aufwendig.

Eine angepasste Distro zu erstellen ist die andere Option, die auch kompatibel mit älterer Hardware ist.

Ein VM-Masseneinsatz ist einfach wenn die VMs als Server oder als Test-BSs verwendet werden, aber ihre Verwendung für normale Büroarbeit, die z.B. Videokonferenzen einschließt, wird mit anderer Hardware in einem künftiges Projekt recherchiert werden.

Rsync-Zusammenfassung

Rsync wurde im Projekt mehrmals eingesetzt. Hier folgt eine Zusammenfassung einiger wichtigen Fälle.

BS synchronisieren

```
rsync -aAHXsh --delete --preallocate --info=progress2,stats2 <quelle> <ziel>
```

EXT-Daten synchronisieren

```
rsync -aAXsh --delete --preallocate --info=progress2,stats2 <quelle> <ziel>
```

In diesem Fall, fällt nur die Fahne -H für harte Links, die aber eingesetzt werden muss falls welche vorhanden sind.

EXT-DATEN über SSH synchronisieren

```
rsync -aAXsh --preallocate --info=progress2,stats2 -e ssh <quelle>  
benutzer@<ip-adresse>:/<ziel>
```

QCOW2-VD-Datei synchronisieren

```
rsync -aAHXsh --inplace --delete --info=progress2,stats2 <quelle> <ziel>
```

NTFS-Daten synchronisieren

```
rsync -rtsh --modify-window=1 --delete --info=progress2,stats2 <quelle> <ziel>
```

NTFS und EXT weisen Unterschiede an den Zeitstempeln auf. Die Option `--modify-window=1` sorgt dass wenn eine Datei entweder neuer oder älter um eine Sekunde ist, der Zeitstempel als gleich angesehen wird.

Zudem, von der Fanne `-a=rlptgoD`, bleibt nur `-rt` übrig, weil alles andere unter NTFS nicht unterstützt wird, wie auch `-AHX`.

Ein Beispiel mit EXT-Daten

Eine externe Festplatte anschließen, die betreffende Sicherungspartition einhängen, Quell- und Zielpartition drucken, und das No-upload-Verzeichnis synchronisieren.

```
bi@green:mnt$ mount dataei
bi@green:mnt$ lsblk | grep -E 'datae|dataei'
├─sda9   8:9   0 271,5G 0 part /mnt/datae
├─sdc5   8:37  0 219,5G 0 part /mnt/dataei
bi@green:mnt$ rsync -aAXsh --delete --preallocate --info=progress2,stats2 \
> /mnt/datae/no-upload/ /mnt/dataei/no-upload/
 4.93M 17% 20.19MB/s 0:00:00 (xfr#18, to-chk=0/108)

Number of files: 108 (reg: 96, dir: 12)
Number of created files: 15 (reg: 15)
Number of deleted files: 22 (reg: 16, dir: 6)
Number of regular files transferred: 18
Total file size: 28.55M bytes
Total transferred file size: 4.93M bytes
Literal data: 4.93M bytes
Matched data: 0 bytes
File list size: 0
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 4.94M
Total bytes received: 1.30K

sent 4.94M bytes received 1.30K bytes 3.29M bytes/sec
total size is 28.55M speedup is 5.78
bi@green:mnt$
```

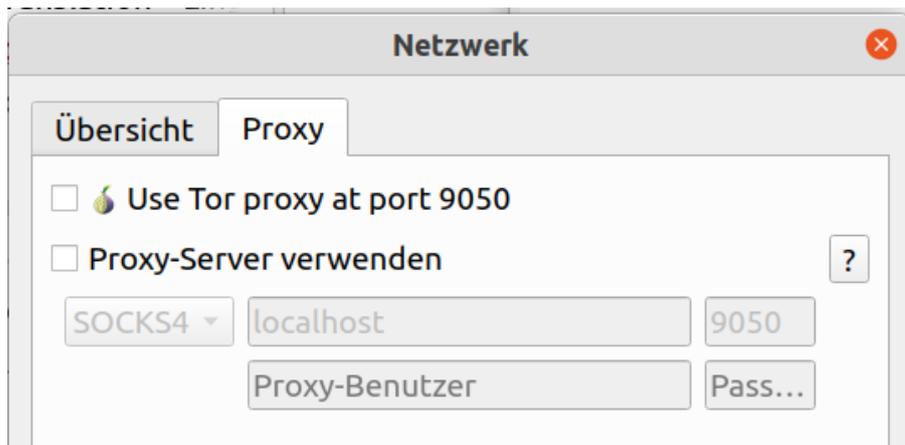
Anschließend, die Zielpartition aushängen und, mittels `udisksctl` das externe Laufwerk ausschalten.

```
bi@green:mnt$ umount dataei
bi@green:mnt$ sudo udisksctl power-off -b /dev/sdc
bi@green:mnt$
```

Electrum-Wallet für Bitcoins

Mit einem Crypto-Wallet können wir Münzen an andere Wallets überweisen, und falls das Wallet am Netzwerk durch einen Proxy verbunden wird, können die Überweisungen einigermaßen privat sein.

Die Verwendung von Proxys wurde im Projekt nicht getestet.



Electrum wurde für das Projekt ausgewählt, weil so gut wie das älteste und sicherste Open-Source Bitcoin-Wallet ist.

Überprüfen der Schlüsselsignaturen

Wir laden die Programmdateien von <https://electrum.org/#download> herunter und die ASC-Datei, die wir brauchen um die Schlüsselunterschriften zu überprüfen.

Zuerst

```
$ gpg --verify Electrum-4.0.9.tar.gz.asc
```

```
gpg: assuming signed data in 'Electrum-4.0.4.tar.gz'
```

```
gpg: Signature made Do 15 Okt 2020 21:21:40 EEST
```

```
gpg:          using RSA key 6694D8DE7BE8EE5631BED9502BD5824B7F9470E6
```

```
gpg: Can't check signature: No public key
```

Wir werden jetzt den oberen RSA-Schlüssel verwenden um den öffentlichen Schlüssel zu importieren.

```
$ gpg --keyserver pool.sks-keyservers.net --receive-keys  
6694D8DE7BE8EE5631BED9502BD5824B7F9470E6
```

```
gpg: key 2BD5824B7F9470E6: public key "Thomas Voegtlin (https://electrum.org)  
<thomasv@electrum.org>" imported
```

```
gpg: Total number processed: 1
```

```
gpg:          imported: 1
```

Jetzt die ASC-Datei verifizieren, die die Signaturdatei ist.

```
$ gpg --verify Electrum-4.0.4.tar.gz.asc
```

```
gpg: assuming signed data in 'Electrum-4.0.4.tar.gz'
```

```
gpg: Signature made Do 15 Okt 2020 21:21:40 EEST
```

```
gpg:          using RSA key 6694D8DE7BE8EE5631BED9502BD5824B7F9470E6
```

```
gpg: Good signature from "Thomas Voegtlin (https://electrum.org) <thomasv@electrum.org>" [unknown]
gpg:      aka "ThomasV <thomasv1@gmx.de>" [unknown]
gpg:      aka "Thomas Voegtlin <thomasv1@gmx.de>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:      There is no indication that the signature belongs to the owner.
Primary key fingerprint: 6694 D8DE 7BE8 EE56 31BE D950 2BD5 824B 7F94 70E6
```

Wir berücksichtigen „Good signature“ an der Ausgabezeile vier, also alles trotz Warnungen in Ordnung.

Das Programm wird mit PIP „Perl Installation Program“ installiert.

```
$ python3 -m pip install --user Electrum-4.0.9.tar.gz
```

Electrum vom Terminal ausführen mit

```
$ electrum
```

Optionale Abhängigkeiten fehlen

Cryptodomex

Es kann sein dass wir die Warnung bekommen:

```
W | crypto | found module 'Cryptodome' but it is too old: 3.6.1<3.7
```

Lösung umgeschrieben von <https://bitcointalk.org/index.php?topic=5310339.msg56127626#msg56127626>

Es geht um eine optionale Abhängigkeit, Cryptodomex, und die wird installiert mit

```
$ python3 -m pip install -U pycryptodomex
```

```
Collecting pycryptodomex
```

```
  Downloading pycryptodomex-3.9.9-cp38-cp38-manylinux1_x86_64.whl (13.7 MB)
```

```
    |████████████████████████████████████████████████████████████████████████████████| 13.7 MB 5.2 MB/s
```

```
Installing collected packages: pycryptodomex
```

```
Successfully installed pycryptodomex-3.9.9
```

Zbar-tools

Die Meldung

```
failed to load zbar. exceptions:
```

bedeutet es fehlt uns die Möglichkeit einen QR-Code zu scannen. Wir müssen nur

```
$ sudo apt install zbar-tools
```

Startwert oder Seed

Das sind zwölf Wörter, mit denen wir unser Wallet in einem anderen Rechner wiederherstellen können.

Privater Hauptschlüssel oder Master Private Key (MPK)

Der Seed ist vom MPK erzeugt worden, und letzterer kann alle unseren privaten Schlüssel generieren.

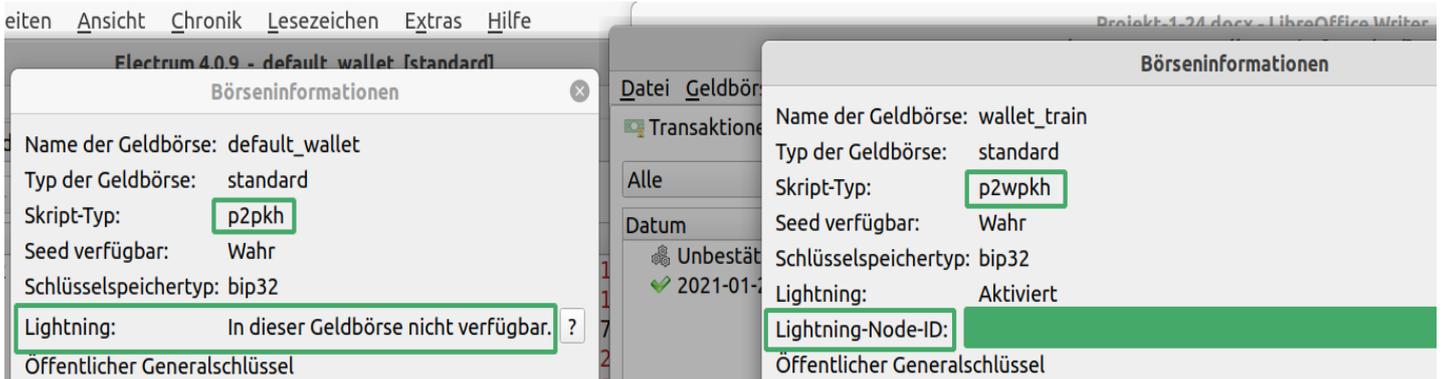
Der MPK wird extrahiert mit

```
$ electrum getmasterprivate --offline -w <wallet-pfad>
```

Native-Segwit gegen Legacy

Ältere (legacy) BTC-Adressen fangen mit „1“, der Skript-Typ ist P2PKH, die sind überall kompatibel, aber die Überweisungskosten sind hoch. Zudem unterstützen sie nicht das neue Lightning-Transaktionsystem.

Bei Native-Segwit, von „Segregated Witness“, die BTC-Adressen fangen mit „bc1“ an und Skript-Typ ist P2WPKH. Die sind auch als BECH32 bekannt. Hier sind die Transaktionsgebühren niedriger, Lightning wird unterstützt, die sind aber noch nicht überall eingesetzt.

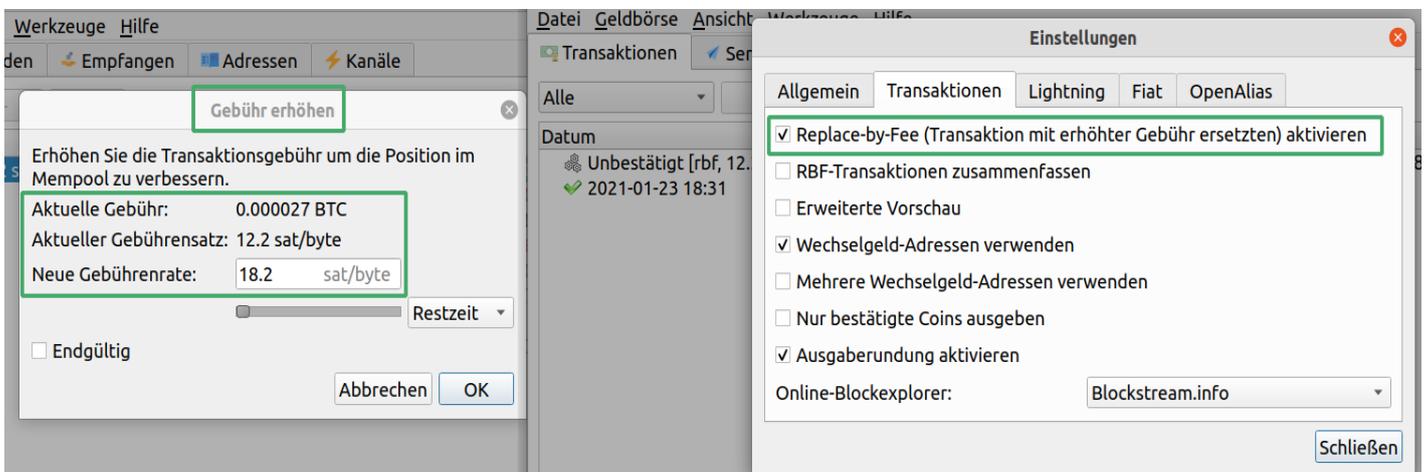


Eingenommen wir wollen aus Sicherheitsgründen unsere BTCs in ein neues Wallet überweisen, weil wir einen neuen Startwert haben wollen. Wir können den Startwert im bestehenden Wallet nicht ändern, das würde die Adressen ändern.

Transaktionsgebühren dynamisch gestalten

Da die Blockchain von Transaktionen oft sehr belastet ist, müssen wir mehr an Transaktionsgebühren zahlen, oder länger warten. Die Gebühren können insofern dynamisch gestaltet werden, kurzum Replace-By-Fee (RBF). Nach der Absendung der Überweisung und falls die Transaktionsbestätigung länger dauert als wir wollen, können die Gebühren manuell gesteigert werden.

Mit der Auswahl „Transaktion anzeigen“ sehen wir „Position im Mempool: XX MB von oben“. Wenn die MBs steigen es bedeutet dass wir mehr Zeit bis zur Bestätigung brauchen und vice versa.



Es ist auch möglich dass der Empfänger eine ergänzende Gebühr zahlt, alias Parent-Pays-For-Child (PPFC).

Wallets verschiedener Software erstellen

Im folgenden Link <https://bitcointalk.org/index.php?topic=5324164.msg56576204#msg56576204> wird erläutert dass die privaten Schlüssel nie in eine andere Wallet-Software importiert werden sollten, weil dadurch die Gefahr steigt die Münzen von möglichen Bugs oder anderen Problemen zu verlieren. (Umgeschrieben und übersetzt.)

Die Münzen sollten stattdessen immer überwiesen werden, aber im Fall einer gesamten Überweisung, können die Transaktionskosten nicht mehr gestiegen werden, wenn keine Münzen im Wallet übrig bleiben.

Diesbezüglich am besten zuerst recherchieren oder eine Testüberweisung durchführen um die Equilibrium-Kosten für eine Überweisung die nicht länger als 48 Stunden dauert herauszufinden. Derzeitig (März 2021) liegen die entsprechenden Gebühren bei 20Sat/Byte.

BTC Börsenplätze als Nebenbemerkung

Bezüglich Segwit- und Legacy-Adressen, müssen wir merken dass es noch Börsenplätze gibt die nicht mit SegWit kompatibel sind.

Ein guter Börsenplatz für die europäische Union, wäre [Kraken](#), dass mit der [Fidor Bank](#) mit Sitz in München zusammen arbeitet, als Vorgabe-Bank in der EU.

Kraken ist auch der einzige Crypto-Börsenplatz dass jemals in Reddit als bester Bitcoin-Börsenplatz des Jahres, 2014 gewählt wurde.

Des weiteren die Wartezeit für einen live Kundenservice nach Erfahrungen des Verfassers, war höchstens 15 Minuten und die KYC (Know Your Customer) Verifizierung hat nur einen Tag gedauert. Bei Binance hat die gleiche KYC-Verifizierung zwei Wochen gedauert.

Die Tradingspreads sind nicht so niedrig wie die von Binance, aber dort müssen wir nach Forenberichten auch stundenlang auf Hilfe warten.

Sicherheitsmaßnahmen und Datenverschlüsselung

Rclone

Rclone ist eine sehr gute Wahl um unsere Daten in der Cloud zu speichern und synchronisieren. Das Programm ist ausgereift, kompatibel mit den meisten [Cloud-Backends](#), bietet unzählige Optionen und wahlweise Verschlüsselung. Die folgenden Beispiele wurden mit dem [Google-Drive-Backend](#) durchgeführt.

Zuerst müssen Remotes erstellt werden. Wenn verschiedene Google-Konten vorhanden sind, kann ein einzelner Remote für jedes Konto erstellt werden. Für die Verschlüsselung erstellen wir einen [Crypt-Remote](#) innerhalb eines normalen Remotes.

Das soll ein Ordner im Google-Drive sein. Alles was mit diesem Ordner synchronisiert wird, ist verschlüsselt. Darüber hinaus die Remotes können lokal, mit dem Fuse-Dateisystem unverschlüsselt einhängt werden.

Client-ID erstellen

Rclone verwendet eine eigene Client-ID für den Google-Drive, die ist aber langsam, weil sie viel eingesetzt und auch deswegen gedrosselt wird. Als erstes müssen wir deshalb, eine eigene erstellen mittels Anweisungen von <https://rclone.org/drive/#making-your-own-client-id>.

Folgend im Google-Drive nachdem wir eine Client-ID erstellt haben.

The screenshot shows the Google Cloud Platform console interface. At the top, there is a blue header with the text 'Google Cloud Platform', 'My Project 1', and a search bar. Below the header, the left sidebar is titled 'APIs & Services' and contains a list of navigation items: Dashboard, Library, Credentials, OAuth consent screen (which is highlighted), Domain verification, and Page usage agreeme... The main content area is titled 'OAuth consent screen' and displays the configuration for the 'rclone' application. It includes an 'EDIT APP' link, a 'Verification status' section indicating that verification is not required, a 'Publishing status' section showing the app is 'In production' with a 'BACK TO TESTING' button, a 'User type' section set to 'External', and an 'OAuth rate limits' section with a 'Your user cap'.

Unter der Rubrik Credentials, oben links, haben wir schon unsere Client-ID und das Client-Geheimnis.

My Project 1 Search products and resources

Client ID for Desktop DOWNLOAD JSON RESET SECRET DELETE

Name *
Desktop client 1
The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

Client ID
Client secret
Creation date 30 January 2021 at 15:47:03 GMT+2

SAVE CANCEL

Remotes

Mit dem Befehl

```
$ rclone config
```

können Remotes verwaltet werden: konfiguriert, erstellt, gelöscht usw., durch die Auswahl von Dialog-Optionen, die meistens selbsterklärend sind.

Bei der Erstellung eines neuen Remotes, kann es zwischen vielen Backends ausgewählt werden. Google-Drive ist die Nummer 13. Unten wird die Konfiguration von [Bkp-gdrive](#) als Beispiel angezeigt.

```
bi@green:~$ rclone config
Current remotes:
Name                Type
====                ==
bkp-gdrive          drive
my-gdrive           drive
secret-gdrive       crypt

e) Edit existing remote
n) New remote
d) Delete remote
r) Rename remote
c) Copy remote
s) Set configuration password
q) Quit config
e/n/d/r/c/s/q> e
Choose a number from below, or type in an existing value
 1 > bkp-gdrive
 2 > my-gdrive
 3 > secret-gdrive
remote> 1
-----
[bkp-gdrive]
type = drive
client_id = [REDACTED]apps.googleus
ercontent.com
client_secret = [REDACTED]
scope = drive
token = {"access_token": [REDACTED]}
```


Wenn ein neues Remote erstellt wird, soll die Option [Refresh](#) ausgewählt werden, so dass wir unsere App (Client-ID), von einem Google-Konto genehmigen: Daraufhin öffnet sich der Browser, dann Konto auswählen und genehmigen. Weil die App die wir erstellt haben, (noch) nicht verifiziert ist, kommt eine Sicherheitswarnung, die wir abweisen müssen.

Crypt

Die Konfiguration von Crypt ist anders. Der Wert vom [Crypt-Remote](#) ist, wie unten angezeigt, der Pfad [bkp-gdrive:/gdata](#), also Crypt ist ein verschlüsseltes Verzeichnis innerhalb des Remotes [Bkp-gdrive](#). Was in diesem Ordner kopiert wird, ist verschlüsselt, was außerhalb von [Gdata](#) befindet, aber unter [Bkp-gdrive](#), ist unverschlüsselt.

```
e) Edit existing remote
n) New remote
d) Delete remote
r) Rename remote
c) Copy remote
s) Set configuration password
q) Quit config
e/n/d/r/c/s/q> e
Choose a number from below, or type in an existing value
 1 > bkp-gdrive
 2 > my-gdrive
 3 > secret-gdrive
remote> 3
-----
[secret-gdrive]
type = crypt
remote = bkp-gdrive:/gdata
filename_encryption = obfuscate
directory_name_encryption = true
password = *** ENCRYPTED ***
password2 = *** ENCRYPTED ***
-----
Edit remote
** See help for crypt backend at: https://rclone.org/crypt/ **
Value "remote" = "bkp-gdrive:/gdata"
Edit? (y/n)>
y) Yes
n) No (default)
y/n> 
```

Folgend ein lokales Verzeichnis wird mit Crypt synchronisiert. Crypt bezieht sich auf den Ordner Gdata.

```
bi@green:datae$ rclone sync --fast-list --user-agent="rclone/v1.53.4" --modify-window=1s --checkers=16 --transfers=20 --tpslimit=8 --tpslimit-burst=10 --contimeout=60s --drive-chunk-size=512M -P /mnt/datae/muf secret-gdrive:/gdata/muf
Transferred:      32.351M / 1.416 GBytes, 2%, 1.687 MBytes/s, ETA 14m
Checks:           52075 / 52075, 100%
Transferred:      25 / 5689, 0%
Elapsed time:     57.4s
Transferring:
* linux-help/linux-old/l...isk_hybrid_2.02s10.iso: 22% /19.637M, 235.651k/s, 1m5s
* linux-help/edx/other e...ate-Handbook-v1.17.pdf:100% /278.453k, 14.353k/s, -
* linux-help/edx/other e...n_guide_August2016.pdf:100% /1.380M, 72.780k/s, -
* linux-help/linux-old/u...-Installer-1.9.8.6.exe:100% /1.701M, 89.756k/s, -
* linux-help/linux-old/u.../YUMI-UEFI-0.0.1.2.exe:100% /4.859M, 256.354k/s, -
* linux-help/linux-old/u...t tools/rufus-3.4p.exe:100% /1008.055k, 51.938k/s, -
* linux-help/linux-old/u...bootin-windows-661.exe:100% /4.608M, 243.122k/s, -
* linux-help/linux-old/u...ager-1.0.0-install.exe: 5% /11.985M, 37.750k/s, 5m6s
* linux-help/linux-old/u... prepare/rufus-3.8.exe:100% /1.086M, 57.277k/s, -
* linux-help/linux-old/u... prepare/rufus-3.9.exe:100% /1.138M, 60.006k/s, -
* linux-help/edx/Course ... edX_files/commons.js:100% /1.884M, 99.348k/s, -
* linux-help/linux-old/u...rw32-setup-1.0.7.0.exe:100% /1.393M, 73.487k/s, -
* linux-help/linux-old/u...ager-1.0.0-install.exe: 5% /11.985M, 37.867k/s, 5m5s
* linux-help/linux-old/u...4/LiLi USB Creator.exe:100% /1.358M, 71.624k/s, -
* linux-help/linux-old/c...0191024-eoan-amd64.iso: 0% /313M, 39.550k/s, 2h14m44s
* linux-help/linux-old/u...soft.VC90.CRT.manifest:104% /1.025k, 0/s, -
* linux-help/linux-old/u...soft.VC90.MFC.manifest:105% /884, 0/s, -
* linux-help/linux-old/u...b-creator/PyQt5.Qt.pyd:100% /15.500k, 0/s, -
* linux-help/linux-old/u...eator/PyQt5.QtCore.pyd:100% /2.507M, 2.507M/s, -
* linux-help/linux-old/u...reator/PyQt5.QtGui.pyd:100% /2.531M, 0/s, -^Z
[2]+  Angehalten          rclone sync --fast-list --user-agent="rclone/v1.53.4" --modify-win
dow=1s --checkers=16 --transfers=20 --tpslimit=8 --tpslimit-burst=10 --contimeout=60s --drive-ch
unk-size=512M -P /mnt/datae/muf secret-gdrive:/gdata/muf
bi@green:datae$
```

--fast-list - Es kann damit schneller gehen, verbrauchen wir aber mehr Speicher.

--user-agent - Damit das Cloud-Backend unser Programm richtig versteht.

--modify-window=1s - Zwischen unterschiedlichen BSs können die Zeitmarken leicht abweichen. Durch diese Option geben wir eine Toleranz von einer Sekunde.

--checkers - Wie viele Checkers parallel laufen sollten.

--transfers - Wie viele parallele Transfers.

--tpslimit - Ein Limit in der Transaktionen pro Sekunde setzen, damit uns das Cloud-Backend, die Geschwindigkeit nicht drosselt.

--tpslimit-burst - Einen Stoß geben für die Zeit die Transaktionen unter dem Limit oder inaktiv waren, aber den langfristigen Durchschnitt nicht über den Wert des --tpslimit erhöhen.

--contimeout - Verbindungs-Timeout setzen.

--drive-chunk - Diese Option kann leicht den verwendet Speicher multiplizieren bis zum BS-Einfrierung. 128M ist eine sicherere Konfiguration.

-P - Damit wir den Fortschritt sehen.

Je nachdem wie viele Dateien synchronisiert werden und wie groß sie sind, kann die Geschwindigkeit sehr variieren. Am Anfang steigt die Geschwindigkeit sehr, wird sie aber allmählich viel niedriger.

Um Teiluploads nicht wieder von Anfang uploaden zu müssen, können wir mit **Strg+Z** pausieren, statt mit **Strg+C** abzubrechen, und dann mit **\$ jobs** und **\$ fg <job-nummer>** wiederaufnehmen genau so wie mit Rsync. Es kann aber sein, dass nachher die Geschwindigkeit sehr gering ist, dann müssen wir das Programm zwangsläufig beenden und neu anfangen.

Duplikate

Manchmal könnte es vorübergehende Fertigstellungen von Upload-Fehlern geben, wobei der Upload erfolgreich war, aber aufgrund von Fehlern eine Datei erneut synchronisiert wird, was Duplikate erzeugen könnte. (Übersetztes Zitat vom Benutzer eharris <https://github.com/rclone/rclone/issues/4429>.)

Wenn es Duplikatmeldungen gibt, der Sync-Unterbefehl kann sie nicht mehr synchronisieren. Dazu muss der Unterbefehl **Dedupe** mit der Fahne **--dedupe-mode=first** z.B., womit die erste Datei erhalten wird und alle anderen Duplikate gelöscht werden.

Hier sehen wir dass ein Duplikat gefunden wurde.

```
bi@green:datae$ rclone sync --fast-list --user-agent="rclone/v1.53.4" --modify-window=1s --checkers=16 --transfers=20 --tpslimit=8 --tpslimit-burst=10 --contimeout=60s --drive-chunk-size=256M -P /mnt/datae/muf secret-gdrive:/gdata/muf
2021-02-21 19:16:41 NOTICE: linux-help/edx/edx chapters/linux foundation grade.JPG: Duplicate object found in destination - ignoring
Transferred:      1.260G / 1.260 GBytes, 100%, 312.807 kBytes/s, ETA 0s
Checks:           57990 / 57990, 100%
Deleted:          5812
Transferred:      5586 / 5586, 100%
Elapsed time:    1h14m24.4s
```

Daher, statt dem Unterbefehl **Sync**, verwenden wir hier **Dedupe** wie es oben erwähnt wurde. Außerdem, in diesem Fall existiert keine Quelle, nur der Zielordner.

```
bi@green:datae$ rclone dedupe --dedupe-mode=first --fast-list --user-agent="rclone/v1.53.4" --modify-window=1s --checkers=16 --transfers=20 --tpslimit=8 --tpslimit-burst=10 --contimeout=60s --drive-chunk-size=256M -P secret-gdrive:/gdata/muf
2021-02-21 20:36:27 NOTICE: linux-help/edx/edx chapters/linux foundation grade.JPG: Found
2 files with duplicate names
2021-02-21 20:36:27 NOTICE: linux-help/edx/edx chapters/linux foundation grade.JPG: Delete
d 1 extra copies
Transferred:      0 / 0 Bytes, -, 0 Bytes/s, ETA -
Checks:           1 / 1, 100%
Deleted:          1
Elapsed time:     2m11.4s
bi@green:datae$
```

Gocryptfs

Gocryptfs <https://nuetzlich.net/gocryptfs/> ist ein gestapeltes kryptographisches Dateisystem. Gestapelt weil es über das normale Dateisystem funktioniert https://wiki.archlinux.org/index.php/Data-at-rest_encryption#Stacked_filesystem_encryption, und damit können wir einzelne Ordner verschlüsseln.

Gestapelte Dateisysteme geben uns die Möglichkeit im laufendem BS empfindliche Daten verschlüsselt zu bewahren im Gegensatz zur Gesamt-Datenträger-Verschlüsselung, die

„Corolla servicing“ in das Mountfs-Verzeichnis kopieren wo die Dateien verarbeitet werden können, aber gleichzeitig erscheinen sie verschlüsselt im Ordner Encrypted.

```
bi@green:crypt$ cp -a 'corolla servicing' mountfs/
bi@green:crypt$ ls encrypted/
44ofSIyX32cnENZ2o8_je5iouSe0oFhpScCe8nKH9Sc  gocryptfs.diriv
bi@green:crypt$ ls encrypted/44ofSIyX32cnENZ2o8_je5iouSe0oFhpScCe8nKH9Sc/
1tezSb9X-Sz0tkez3LAgofRaNNkGJy_KzkEbunx4wyw
3WHz13Zwz9tkcg0IRpZrGA
8kr6Tzf8g9YmMqgEbpjMrzcsrR4Ps34h9sqXxeISbnE
DZjLaCR93fNQTx0HkkKLUKAXfwQ2-Zu3NbKkbJtCwGY
e6xPpP0P1Zo2TuhV1XuS0A
EKcPglWixaqtMuPa9p9arA
gocryptfs.diriv
gsV5n8lGdpQ26TrLKCSb1iBavmnAqebD6nTsNY40eFc
h31980bEqi12zI2Mjfgsw
iovSFyecYG_BqmWnRUB5A69zwqqsoY_em3WlusQfo_Y
KstAA1eJMHlje4ziFKw35w
LC5vUkJXgZ0iKwBgX08tNw
lfL0u-7yIPZoQWfcREdJ4jCqMbslSMcHSD3xi0BksYeDZhUiopc9hdlCgrfLckiW
rQTa1FRQr7LeAeg5FLMTUyoyDVA65ILKBQRssbTp7v8
Sbyzc-Ds0okMdaA1tb7VSA
XjJbvq_YAKQ8RRF850me_iS6mEvcs3YuPiqddMaG9d8
```

Es ist interessant dass Gocryptfs eine [Fsck-Option](#) um nach Korruption zu suchen bietet, was die meisten Programme der Kategorie nicht tun. Allerdings bei einigen GBs dauert es sehr lang mit der Fsck-Subroutine.

```
bi@green:crypt$ gocryptfs -config ./backup-conf/gocryptfs.conf -fsck encrypted
Using config file at custom location /mnt/datae/crypt/backup-conf/gocryptfs.conf
Password:
Decrypting master key
fsck summary: no problems found
```

Wir listen jetzt den Inhalt vom Mountfs der die verschlüsselten Daten unverschlüsselt enthält, und mittels `Fusermount`, Mountfs anschließend aushängen. Der Inhalt von Encrypted ist immer verschlüsselt, und enthält auch eine Datei die `Gocryptfs.diriv` heißt.

```
bi@green:crypt$ ls mountfs/  
'corolla servicing'  
bi@green:crypt$ ls mountfs/'corolla servicing'/  
'~$rolla servicing 2018.docx'      'forum pics'  
'~$rolla servicing 2020.docx'      'oil filter.PNG'  
'car oil.txt'                      'procedure notes'  
'corolla servicing 2016.docx'      'repair manuals'  
'corolla servicing 2018.docx'      shops  
'corolla servicing 2020.docx'      'spare parts 2014'  
'corolla servicing.docx'           tires  
corolla_tech_specs_tcm273-538949.pdf  
bi@green:crypt$ fusermount -u mountfs  
bi@green:crypt$ ls mountfs/  
bi@green:crypt$ ls encrypted/  
44ofSIyX32cnENZ2o8_je5iouSe0oFhpScCe8nKH9Sc  gocryptfs.diriv
```

Die `Gocryptfs.diriv`-Datei sorgt für die Verschlüsselung der Ordner- und Dateinamen. Jedes Verzeichnis und Unterverzeichnis enthält eine solche Datei. Wenn eine `Diriv`-Datei gelöscht wird, alles in dem Ordner und unterhalb verschwindet.

Es sind Anweisungen vorhanden um die Daten zurück zu holen, und ihnen neue Namen zu zuweisen, die kurzen Tests die im Projekt unternommen wurden, waren entsprechend aber nicht gelungen.

Es heißt also zumindest von der `Diriv`-Datei des Root-Verzeichnisses auch eine Sicherungskopie zu erstellen. Falls wir dort einen Namen ändern sollten, müssen wir die alte Sicherungskopie durch eine neue ersetzen.

Bei verschlüsselten Daten ist es noch wichtiger ständig Sicherungskopien zu erstellen.

Deswegen die Diriv-Datei sollte auch in das Verzeichnis Backup-conf kopiert werden, und noch dazu eine Kopie des verschlüsselten Verzeichnisses Encrypted erstellen mit dem Namen Encrypted1, und daraufhin encrypted1 auf Mountfs einhängen um zu testen ob alles funktioniert.

```
bi@green:~$ cp -a encrypted/gocryptfs.diriv backup-conf/gocryptfs.diriv
bi@green:~$ ls backup-conf/
gocryptfs.conf  gocryptfs.diriv
bi@green:~$ cp -a encrypted encrypted1
bi@green:~$ gocryptfs -config backup-conf/gocryptfs.conf encrypted1 \
|> mountfs
Using config file at custom location /mnt/datae/crypt/backup-conf/gocryptfs.conf
Password:
Decrypting master key
Filesystem mounted and ready.
bi@green:~$ ls mountfs/'corolla servicing'/
'~$rolla servicing 2018.docx'      'forum pics'
'~$rolla servicing 2020.docx'      'oil filter.PNG'
'car oil.txt'                     'procedure notes'
'corolla servicing 2016.docx'      'repair manuals'
'corolla servicing 2018.docx'      shops
'corolla servicing 2020.docx'      'spare parts 2014'
'corolla servicing.docx'          tires
corolla_tech_specs_tcm273-538949.pdf
bi@green:~$ fusermount -u mountfs
bi@green:~$ ls mountfs/
bi@green:~$ cp -a backup-conf backup-conf1
bi@green:~$ ls
backup-conf  'corolla servicing'  encrypted1
backup-conf1 encrypted            mountfs
```

Schließlich „Corolla servicing“ löschen, wobei der Ordner jetzt nur verschlüsselt in zwei Kopien existiert, Encrypted und Encrypted1. Die Ordner Encrypted1 und Backup-conf1 auf zwei externe Festplatten kopieren oder verschieben und jegliche Änderungen mit Rsync synchronisieren wie unter der Rubrik EXT-Daten synchronisieren erläutert wird, weiter oben.

```
bi@green:~$ rm -rf 'corolla servicing'
bi@green:~$ ls
backup-conf  backup-conf1  encrypted  encrypted1  mountfs
bi@green:~$
```

Daten-Korruption

Es sind vielerorts Meldungen vorhanden, dass mit verschiedenen Verschlüsselungs-Programmen, Daten-Korruption während der Synchronisierung vorkommen kann.

Ein Programm wie Rclone unterliegt keiner solchen Gefahr weil Synchronisierung und Verschlüsselung, Teile der gleichen Anwendung sind.

Wenn aber ein kryptographisches Dateisystem mit einer Anwendung synchronisiert wird, dass von den Programmentwicklern nicht ausführlich getestet wurde, Korruption könnte entstehen.

Der Gocryptfs-Entwickler, [rfjakob](#), empfiehlt Rsync, aber wenn die Daten nur mit externen Festplatten synchronisiert werden, ist es gegen Korruption besser zwischen Oberverzeichnissen zu synchronisieren, oder die verschlüsselten Verzeichnisse ausgehängt mittels Rsync zu updaten.

Tar mit GPG

Wichtige Daten sollten durch einen zweiten Weg auch verschlüsselt und gesichert werden. Vom Oberverzeichnis des Gocryptfs können wir die Dateien kopieren und Gpg einsetzen.

Gpg ist ein gutes Verschlüsselungs-Programm, kann aber nur einzelne Dateien verschlüsseln. Deswegen, archivieren wir unsere Daten zuerst mit Tar, und die Ausgabe ins Gpg pfeifen.

Umgekehrt, mit Gpg entschlüsseln und die Ausgabe ins Tar pfeifen.

```
bi@green:test$ ls
noup0
bi@green:test$ tar cf - noup0 | gpg -c > no0.tar.gpg
bi@green:test$ ls
no0.tar.gpg  noup0
bi@green:test$ rm -rf noup0
bi@green:test$ gpg -d no0.tar.gpg | tar xf -
gpg: AES256 verschlüsselte Daten
gpg: Verschlüsselt mit einer Passphrase
bi@green:test$ ls
no0.tar.gpg  noup0
bi@green:test$ █
```

7z

Falls es nötig wird, verschlüsselte Dateien in einem Windows-System zu entschlüsseln, kann dementsprechend 7z eingesetzt werden.

Verschlüsseln wir mit

```
7z <archiv-name> <dateien> -p
```

Entschlüsseln mit

```
7z x <archiv-name>
```

Stapel-Umbenennung von Dateien

Wenn wir Daten klonen und synchronisieren, wollen wir unter Umständen auch mehrere Dateien gleichzeitig umbenennen.

Das folgende Beispiel verwendet einen einfachen zusammengesetzten Befehl von Bash, im ersten Fall um Buchstaben zu ersetzen, und im zweiten, Buchstaben zu löschen.

```
bi@green: /mnt/datae/test
bi@green:test$ touch ubuntu-kvm-{1..9}
bi@green:test$ ls
ubuntu-kvm-1  ubuntu-kvm-3  ubuntu-kvm-5  ubuntu-kvm-7  ubuntu-kvm-9
ubuntu-kvm-2  ubuntu-kvm-4  ubuntu-kvm-6  ubuntu-kvm-8
bi@green:test$ for i in ubuntu-kvm-*; do mv "$i" "${i//kvm/xen}"; done
bi@green:test$ ls
ubuntu-xen-1  ubuntu-xen-3  ubuntu-xen-5  ubuntu-xen-7  ubuntu-xen-9
ubuntu-xen-2  ubuntu-xen-4  ubuntu-xen-6  ubuntu-xen-8
bi@green:test$ for i in ubuntu-xen-*; do mv "$i" "${i//-xen-/}"; done
bi@green:test$ ls
ubuntu1  ubuntu3  ubuntu5  ubuntu7  ubuntu9
ubuntu2  ubuntu4  ubuntu6  ubuntu8
bi@green:test$
```

Schlussfolgerungen

Beweggrund

Open-Source-Software verleiht uns das Motiv neues zum Lernen, kreativ zu werden, und unsere Systeme leichter an unsere Bedürfnisse anzupassen.

Befehlszeile-Lernen

Wir lernen schnell dass für Aufgaben wo wir früher nach speziellen GUI-Programmen suchen mussten, reicht jetzt eine halbe Befehlszeile, daher Zeit muss am Befehlszeilelernen investiert werden.

Closed-Source-BSs vermeiden

Der Inhalt von Closed-Source-BS ist nur zum eigenen Entwickler bekannt und sollte möglichst vermieden werden. Ein solches Konzept wird dann unerlässlich wenn Missionskritische Daten verwaltet werden.

Ubuntu-Zuverlässigkeit hat sich bewährt

Täglich getestet für länger als ein Jahr. Pakete wurden auch durch PPAs, DEB-Dateien und Snaps installiert und KVMs laufen ständig.

Kurz nach der Ubuntu-Installation werden manchmal Fehlermeldungen angezeigt, deren Fenster, Infos zu der Quelle des Problems enthält. Es wurde immer hinsichtlich nachgegangen, wobei die Lösungs-Vorgehensweisen waren entweder selbsterklärend oder musste zuerst gegoogelt werden.

Alle solchen Probleme wurden dank der großen Ubuntu-Gemeinde leicht gelöscht und kam seit einem Jahr keine weitere Fehlermeldung.

Paketverwaltung

Die Paketverwaltung hat nie Fehler und Instabilität hervorgerufen und bietet mehrfache Optionen. Die Repos sind so gut organisiert, dass es selten woanders nach Programmen gesucht werden musste.

Das Look-and-Feel wurde durch Erweiterungen sehr angepasst und die Vielfalt von verfügbaren Programmen ist viel größer als anfangs gedacht.

PDF-Bearbeitung

Durch eine ergänzende Einsetzung von Programmen und Tools können Pdf-Dokumente hinreichend editiert werden. Das [Pdf2odt-Skript von Gutschke](#) beweist was durch Kreativität anhand eines kurzen Skripts erreicht werden kann.

Für einen professionelleren Gebrauch sollte das Qoppa-Pdf-Studio gekauft werden.

LibreOffice

Vor allem Writer und Calc wurden im Projekt ein ganzes Jahr weitgehend verwendet. Am seltenen Fall dass eine Funktion nicht vorhanden ist, kann eine Umgehungslösung angewendet werden, wie die Verwendung einer Symlink-Datei um die fehlende geteilte Ansicht unter Writer zu ersetzen.

Des Weiteren, es ist viel effektiver wenn die native ODF-Formatierung eingesetzt wird. Es ist vorgekommen dass am Anfang manche DOC- und XLSX-Dateien nicht richtig funktionierten wenn sie als ODT- und ODS-Dateien gespeichert wurden.

In einem solchen Fall, erst alles aktualisieren und dann mit dem Befehl `apt search` nach fehlenden und zusätzlichen Paketen suchen und nachinstallieren.

Vernetzung

SSH (-X) und NoMachine-RD haben überall einwandfrei funktioniert.

BS klonen und aktualisieren

Der Klon-Ansatz wurde ausgiebig getestet, weil drei mal im Monat ein aktualisierter Klon als Vorgabe-BS ohne jegliche Probleme verwendet wird.

Es sollten immer mechanische externe Festplatten aber keine USB-Sticks oder SSD-Datenträger.

BS-Portabilität

Klone sind aber nicht portabel. Nur Installationen auf externen Festplatten können mit einigen Problemen, von einer anderen Maschine gebootet werden.

Die Portabilitätsbeschränkungen der physischen BS-Installationen können vermindert werden durch eine „[Live CD Customization From Scratch](#)“, also eine Distro-Erstellung aus einem jeweiligen Zustand des BS, was für ein künftiges Projekt geplant ist.

KVMs

KVM Virtuelle Maschinen (VMs) werden am Besten mit Qcow2 virtuellen Datenträgern eingesetzt, können anhand von SSH (-X) vernetzt werden und CPU-Zyklen für Befehlszeile- und LibreOffice-Programme sind niedrig. Einige Klone können gleichzeitig betrieben werden gemäß dem verfügbaren RAM. VMs beanspruchen keine CPU wenn die pausiert sind und können auf diese Weise im Host parat stehen.

Klon-Erstellung und -Updaten werden anhand von Virt-clone und Rsync jeweils unternommen. Rsync wird hierbei mit der Fahne --inplace verwendet, die die einzige betreffende Möglichkeit für Qcow2-Dateien ist, die aber leider eher kopiert als updatet.

Windows-VMs

Umständlichkeits- und Boot-Probleme der WVMs wurden weiter oben erwähnt.

Andererseits Linux erweist sich kompetent für alle Aufgaben, solange wir etwas Zeit nehmen um die Aufgabe zu recherchieren. Ergänzend können auch Linux-Closed-Source-Anwendungen eingesetzt werden.

WVMs haben nicht nur ein hohes Umständlichkeit-Leistungs-Verhältnis, weil wenn die weitgehend und mit Internet-Zugang eingesetzt werden, ist das Sicherheits- und Virenrisiko so hoch dass die Anwendung von z.B. Crypto-Hot-Wallets praktisch unbrauchbar gemacht wird.

KVM-GPU-Beschleunigung

Ubuntu-Vms können alltäglich verwendet werden solange keine Video-Anwendungen eingesetzt werden, infolge erhöhter CPU-Zyklen. Außer der üblichen Konfiguration mit QXL-Treiber und Spice-Client, der Virgil3d Paravirtualitätstreiber wurde auch eingesetzt, aber ohne betreffende Verbesserung.

Andererseits, eine GPU-Durchreichungslösung ist aufwendig und beschränkt die Portabilität der VM.

Ein künftiger Versuch soll durch die [Intel-GVT-g-Erweiterung](#) erfolgen. Dazu wird aber neuere Hardware benötigt. Wenn die Video-CPU-Zyklen sich erniedrigen lassen dann macht auch mehr Sinn die Möglichkeit der Virtualisierung einer physischen Installation mithilfe des [Virt-p2v](#).

Viren - Crypto-Wallets

Durch die Einbehaltung der Vorgabe-Regeln für die Verwendung des BSs, nach vielseitigem täglichen Gebrauch von Ubuntu für länger als ein Jahr, sind Viren und Co. ein fremdes Konzept.

Eine perfekte Umgebung für die Verwendung von Crypto-Wallets. Electrum, ein der ältesten und sichersten Wallets, wurde problemlos eingesetzt.

Verschlüsselung

Rclone

Daten werden in der Cloud mit Rclone synchronisiert und verschlüsselt. Rclone ist kompatibel mit sehr vielen Cloud-Backends und bietet unzählige Optionen um die Verbindung mit dem jeweiligen Backend zu optimieren.

Um die bestmögliche Geschwindigkeit zu erreichen muss zuerst eine eigene Client-ID für das entsprechende Cloud-Backend erstellt werden.

Gocryptfs - GPG

Das gestalte kryptographische Gocryptfs-Dateisystem wird seit einiger Zeit eingesetzt ohne Daten-Korruption oder andere Probleme.

Eine Gesamt-Datenträger-Verschlüsselung birgt vielmehr das Risiko eines totalen Datenverlustes und eines Versagens des BS.

Gocryptfs ist einfach, stabil, und verfügt über eine Integritäts-Subroutine. Eine Zusammenführung von Tar und Gpg wird ergänzend angewendet.

Literaturquellen

1. alphr.com
2. appuals.com
3. ark.intel.com
4. ask.fedoraproject.org
5. ask.libreoffice.org
6. askubuntu.com
7. astroa.physics.metu.edu.tr
8. bhphotovideo.com
9. bitcoin.stackexchange.com
10. bitcoinops.org
11. bitcointalk.org
12. bugs.launchpad.net
13. bugzilla.redhat.com
14. buybitcoinworldwide.com
15. bytesized-hosting.com
16. centos.org
17. citrix.com
18. coderwall.com
19. coindesk.com
20. community.kde.org
21. community.spiceworks.com
22. computingforgeeks.com
23. console.cloud.google.com
24. creativecommons.org
25. crucial.com
26. cyberciti.biz
27. debian.org
28. deepin.org
29. defuse.ca
30. deonsworld.co.za
31. distrowatch.com
32. docs.fedoraproject.org
33. docs.gimp.org
34. documentation.libreoffice.org
35. documentation.suse.com
36. elearn.ellak.gr
37. electrum.org
38. electrum.readthedocs.io
39. en.bitcoin.it
40. extensions.gnome.org
41. extensions.libreoffice.org
42. forums.thinkbroadband.com
43. fosslinux.com
44. freecodecamp.org
45. geeksforgeeks.org
46. gemini.com
47. github.com
48. gnu.org
49. groups.google.com
50. help.cryptocoin.com
51. help.libreoffice.org
52. help.ubuntu.com
53. howtoforge.com
54. howtogeek.com

55. hpc.uni.lu
56. <http://ftp.riken.jp/Linux>
57. <http://linuxcommand.org>
58. ibm.com
59. imagemagick.org
60. itsfoss.com
61. kde.org
62. kifarunix.com
63. kraken.com
64. kraxel.org
65. kubuntu.org
66. libguestfs.org
67. libreoffice.org
68. libreofficehelp.com
69. libvirt.org
70. [Linux Handbuchseiten](#)
71. linux-community.de
72. linux-kvm.org
73. linux-magazine.com
74. linux.com
75. linux.die.net
76. linuxconfig.org
77. linuxhint.com
78. linuxinsider.com
79. linuxize.com
80. linuxjournal.com
81. linuxprograms.wordpress.com
82. linuxuprising.com
83. manjaro.org
84. medium.com
85. mondorescue.org
86. mxlinux.org
87. namecheap.com
88. networkinterview.com
89. nuetzlich.net/gocryptfs
90. onapp.com
91. openoffice.blogs.com
92. opensource.com
93. openstack.org
94. opensuse.org
95. ostechnix.com
96. pediaa.com
97. peteris.rocks
98. phoenixnap.com
99. phoronix.com
100. praxistipps.chip.de
101. proxmox.com
102. putorius.net
103. rclone.org
104. reddit.com
105. redhat.com
106. researchgate.net
107. rodsbooks.com
108. searchservervirtualization.techtarget.com
109. section.io
110. security.stackexchange.com
111. serverfault.com
112. snapcraft.io

113. sourcedigit.com
114. spice-space.org
115. ss64.com
116. stackoverflow.com
117. superuser.com
118. systutorials.com
119. techmint.com
120. techrepublic.com
121. thetinhhat.com
122. ubuntu.com
123. ubuntuforums.org
124. ufsexplorer.com
125. unix.stackexchange.com
126. userbase.kde.org
127. virgil3d.github.io
128. vitux.com
129. wiki.alpinelinux.org
130. wiki.archlinux.org
131. wiki.documentfoundation.org
132. wiki.ubuntu.com
133. wiki.xenproject.org
134. wikipedia.org
135. windowcentral.com
136. xenproject.org
137. youtube.com