

tubIT Live – eine komplette Studierendenumgebung ”to Go”

Thomas Hildmann,¹ Christopher Ritter,² Nico Suhl³

Abstract: Studierende sind heute im Besitz von mehreren mobilen Geräten, die sie mit ins Studium bringen (BYOD). Das Einrichten der für sie geeigneten Lernumgebung stellt sie mit steigender Zahl von Diensten vor eine immer größere Herausforderung. Das hier vorgestellte Linux-System unterstützt das Lernen mit dem PC oder Laptop durch eine automatisierte Personalisierung, die Integration in die Universitätsinfrastruktur, den bereitgestellten Cloudspeicher und ein studienbegleitendes Empfehlungssystem. Das tubIT Live ist sowohl als VM, wie auch in verschiedenen USB-Stickvarianten verfügbar. Ziel ist ferner die Verschmelzung mit der Wartung von PC-Pools.

Keywords: BYOD, Lernumgebung, Infrastruktur, Linux

1 Einleitung

Eine der zentralen Aufgaben der Hochschulen ist die Bereitstellung von Lerninfrastrukturen für Studierende. Waren es vor einigen Jahrzehnten noch vor allem Räumlichkeiten und Literatur, so musste mit zunehmender Digitalisierung und Vernetzung auch entsprechende Computer- und Netzwerkinfrastruktur bereitgestellt werden. Heute ist die zentrale Anforderung der Studierenden nicht mehr die Bereitstellung von Hardware z.B. in Form von PC-Pools, sondern vor allem die Bereitstellung von Bandbreite und Diensten, die zum Lernen benötigt werden. Hardware und ganze digitale Umgebungen, oft bestehend aus diversen Cloud-Diensten werden bereits ins Studium mitgebracht. Mit steigender Zahl der Angebote für Studierende wird auf der anderen Seite das Auffinden der geeigneten Inhalte und Dienste sowie deren Installation, Konfiguration und Verwaltung eine immer zeit- und nervenraubendere Aufgabe für einen Studierenden – ganz besonders zu Studienbeginn. Das hier vorgestellte System bietet daher eine komplette Studierendenumgebung, mit einer Konfiguration durch einmalige Eingabe von Benutzername und Passwort. Es basiert auf einer Linux-Distribution und wird ähnlich dem bekannten Knoppix-Linux als USB-Stick oder wahlweise auch als virtuelle Maschine bereitgestellt. Ziel ist es ein komplett integriertes Ökosystem zu schaffen, wie es Nutzer heute von Anbietern, wie Apple, Google oder Microsoft gewohnt sind und das komplett neben dem bereits etablierten System der Studierenden existieren kann oder aber auch als Ergänzung hierzu genutzt werden kann. Dabei zielen wir bei dieser Lösung nicht auf Smartphones oder Tablets ab, für die entsprechende Apps bereitgestellt werden. Die Lösung adressiert Laptops, PCs zu Hause oder auch in der Universität, die mit der personalisierten Umgebung genutzt werden können.

¹ Technische Universität Berlin, tubIT, Einsteinufer 17, 10587 Berlin, thomas.hildmann@tu-berlin.de

² Technische Universität Berlin, tubIT, Einsteinufer 17, 10587 Berlin, christopher.ritter@tu-berlin.de

³ Technische Universität Berlin, tubIT, Einsteinufer 17, 10587 Berlin, suhl@tubit.tu-berlin.de

2 Funktionen von tubIT Live

Das System in erster Version wurde mit einem Satz von Funktionen implementiert, die zum einen am häufigsten verwendet werden und zum anderen auch die meisten Anfragen im Support des Rechenzentrums ausmachen. Im Folgenden wollen wir kurz darauf eingehen, wie diese Funktionen jeweils realisiert wurden.

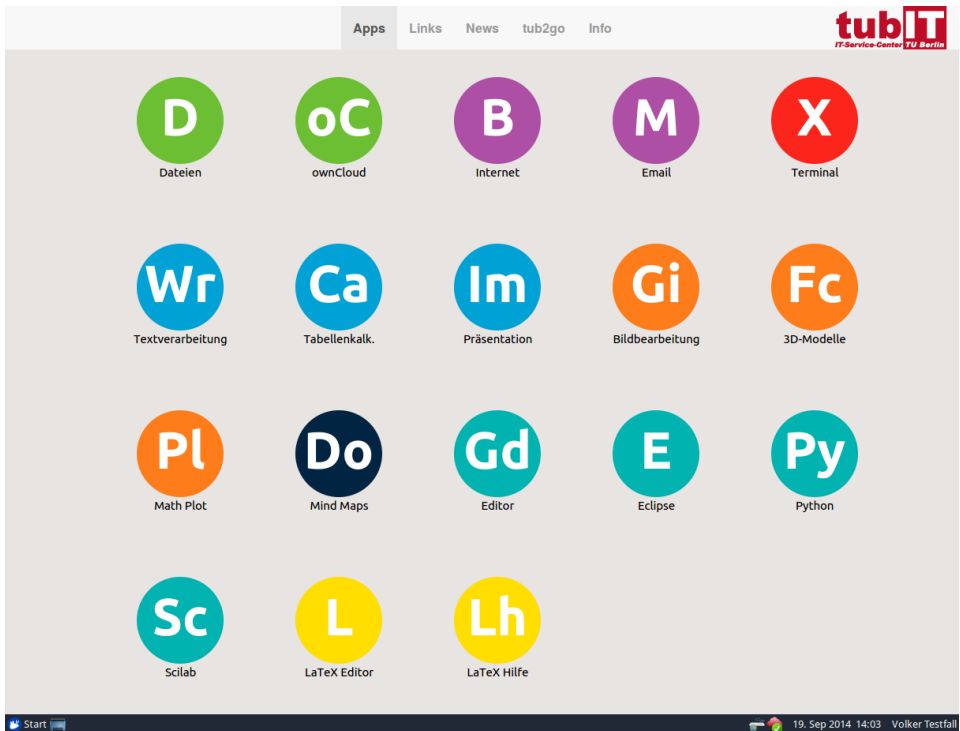


Abb. 1: tubIT Live mit interaktivem Wallpaper

Zentrale Schaltstelle des Systems ist das interaktive Wallpaper, eine HTML-Seite, die den Desktop füllt und Startknöpfe für die wichtigsten Programme enthält sowie weitere über Reiter erreichbare Seiten, die Lesezeichen und ein auf RSS basiertes Mitteilungssystem enthalten.

Die Lesezeichen öffnen jeweils einen Webbrowser und führen auf die zentralen Studierendensysteme für Lehrinhalte, Informationsseiten für Studierende sowie Seiten, die die Funktionalität der mobilen App abbilden (z.B. Indoor-Navigation, Mensaplan, Stundenplan).

Zum Erstellen von Texten, Präsentationen und Kalkulationen steht OpenOffice als lizenzkostenfreies System zur Verfügung. Das Grundsystem setzt komplett auf kostenfreie und großteils quellfrei Software, die aber durch kommerzielle Systeme ergänzt werden könnte.

Auch Programmsysteme wie das vollständig vorinstallierte L^AT_EX werden von Studierenden verschiedener Studienrichtungen sehr begrüßt. Die Installation einer kompletten L^AT_EX-Umgebung inklusive geeignetem Editor stellt für einige Studierende offensichtlich noch immer eine nicht zu unterschätzende Hürde dar. Mit dem System ist auch eine offline verfügbare L^AT_EX-Referenz mit vorinstalliert.

Ein E-Mailprogramm gehört heute wie der Webbrowser zur Standardausstattung. Ebenso wie die Einstiegspunkte für die Webseiten bereits im System enthalten sind, wird das E-Mailkonto bei der Initialisierung des Systems fertig eingerichtet. Mit dem ersten Öffnen des E-Mailprogramms sind alle Einstellungen also bereits gemacht. Weitere bereits vor dem Studium existierende Konten können natürlich hinzugefügt werden, müssen dann jedoch händisch konfiguriert werden.

Der komplette nutzerbezogene Inhalt (Dokumente, Bilder, Multimediadaten) des Systems wird automatisch in den Cloudspeicher (hier ownCloud) synchronisiert. Damit sind die Daten unabhängig vom Speichermedium oder verwendeten Hardwaresystem. Ferner kann auch außerhalb der Plattform über den Cloudspeicher auf die Daten zugegriffen werden.

Bislang war einer der Hauptgründe für die Nutzung von PC-Pools die Bereitstellung von fertig eingerichteten Programmierumgebungen, die für diverse Studiengänge benötigt werden. Viele dieser Programmierumgebungen lassen sich jedoch auch über ein Paket im Linux-System bereitstellen. In einer ersten Version haben wir das System mit gängigen Sprachen und integrierten Entwicklungssystemen ausgestattet. Es ist jedoch vorgesehen, dass abhängig vom Studiengang und ggf. auch besuchter Veranstaltung weitere Pakete installiert werden. Diese können dann auch bereits auf die Anforderungen in der Lehrveranstaltung zugeschnitten sein.

Auch der campusweite Druckdienst steht über die Plattform zur Verfügung und ist vor-konfiguriert. Drucker, die über diesen Dienst angesprochen werden können, sind über den Campus verteilt. So kann sowohl von PC-Pool-Rechnern, als auch vom eigenen Rechner mit tubIT Live aus gedruckt werden.

Eine überschaubare Zahl häufig benötigter Programme wurde bei der Entwicklung des Systems mit Studierenden gemeinsam zusammengestellt und wird mit dem System ausgeliefert. Später ist vorgesehen, in einem vorgegebenen Rahmen eine Auswahl von Programmen oder Alternativen treffen zu können. Grundsätzlich besteht der Mehrwert für die Studierenden jedoch vornehmlich in der redaktionellen Auswahl der Programme, die nicht mehr aus der Vielzahl von Möglichkeiten ausgewählt und evaluiert werden müssen. Auf der anderen Seite bietet die Basis (die Linux-Distribution) für erfahrende Nutzer auch die Möglichkeit weitere Pakete zusätzlich zu installieren. Es ist geplant die Auswahl der vorinstallierten Pakete abhängig vom Studiengang zu variieren.

Eine Liste der im Standard installierten Anwendungen ist im Anhang A aufgeführt.

3 Varianten des Systems

Ein entscheidender Erfolgsfaktor eines Systems, das für die mitgebrachte Hardware der Studierenden (BYOD) gedacht ist, ist die Verfügbarkeit für alle gängigen PC- und Notebook-Modelle und eine möglichst geringe Anforderung an die verfügbaren Ressourcen. Um allen Anforderungen gerecht zu werden, stellen wir tubIT Live in drei Varianten zur Verfügung: zwei USB-Varianten und in einer virtuellen Maschine.

Für die meisten Studierenden ist die Variante USB-Stick die benutzerfreundlichste. Der Stick wird in den Laptop gesteckt, es wird vom Stick gebootet und man kann das komplette System vom Stick benutzen, als wäre es ein eigener Laptop/PC, der nur für das Studium eingerichtet ist. Der Stick kann auch mitgenommen und z.B. im Rechner des/der Kommilitonen/in oder bei den Eltern benutzt werden. Die Bereitstellung muss dabei zwei Hürden überwinden: Die erste ist das Bespielen des USB-Sticks mit dem System und die zweite das Starten von diesem USB-Stick. Für das Bootproblem haben wir entsprechende Anleitungen auf die Webseiten gestellt. Leider gibt es keine wirklich etablierten Standards hierfür.

Um die Herstellung des bootfähigen USB-Sticks jedoch so stark wie möglich zu vereinfachen, haben wir in unserem PC-Pool eine Option hierfür bereitgestellt. Der Nutzer/die Nutzerin wählt aus, dass er/sie sich einen USB-Stick erstellen möchte und wird aufgefordert, diesen in den USB-Port zu stecken. Daraufhin wird das Image auf den Stick kopiert und startet nach einmaliger Initialisierung im eigenen Gerät.

Ferner stehen Anleitungen und Scripte für Windows, OS X und Linux zur Verfügung, um einen Stick zu erstellen. Der kritische Punkt dabei ist, das Image auf das richtige USB-Gerät zu spielen und nicht versehentlich die externe Festplatte zu überspielen.

Leider gibt es mittlerweile auch zwei Firmware-Varianten, die von den Sticks unterstützt werden müssen. Dazu gehört das z.B. von Apple Macs verwendete (U)EFI im Gegensatz zum noch in der PC-Welt verbreiteten „Legacy“ BIOS. Beide Varianten müssen angeboten, beschrieben und gepflegt werden.

Neben den USB-Stick Varianten steht auch eine virtuelle Maschine zur Verfügung, die mit Hilfe des kostenlosen VirtualBox betrieben werden kann. Einige Studierende wollen ihre gewohnte OS-Umgebung nicht verlassen, sondern wünschen sich ein System, das in einem „Fenster“ oder „Screen“ läuft und auf das bei Bedarf umgeschaltet werden kann. Das private System und das Uni-System sollen nebeneinander laufen. Auch diese Variante wird unterstützt, hat jedoch kleinere Einschränkungen. So kann z.B. die WLAN-Umgebung im virtuellen System natürlich nicht fertig eingerichtet werden, da die Netzverbindung vom Hostsystem stammen muss und an das virtuelle System nur durchgereicht werden kann.

4 Integration in die Lehre

Der integrative Aspekt unserer Lösung geht über die Zusammenstellung häufig genutzter Software und deren Individualisierung durch Konfiguration der Benutzerkonten hinaus.

Um den Studierenden eine gezielte Unterstützung im Rahmen ihres Studiums bieten zu können, muss das Softwarepaket vor allem die für den jeweiligen Studiengang benötigte Software enthalten. Aber auch das alleinige zur Verfügung stellen von Software bietet den Studierenden nur zu einem Teil Unterstützung. Auch aktuelle Informationen bezüglich des Studiums oder einzelner Veranstaltungen werden heutzutage primär in digitaler Form verteilt. Neben den klassischen Mailinglisten werden hier immer häufiger RSS-Feeds eingesetzt. Auch Lerninhalte wie Skripte, Literatursammlungen, Multimedia-Material etc. werden in digitaler Form zur Verfügung gestellt. Die Informationen, welche Inhalte wo zu finden sind, werden häufig dezentral und in unterschiedlicher Form verteilt. Zusätzlich sind viele Inhalte mit einem Passwortschutz versehen, so dass die Studierenden eine Vielzahl an Informationen verwalten müssen um an die jeweiligen Inhalte zu gelangen. tubIT Live wurde daher um ein Modul zur Verwaltung von Inhalten erweitert, das neben den Softwarepaketen auch RSS-Feeds und Lerninhalte besitzt.

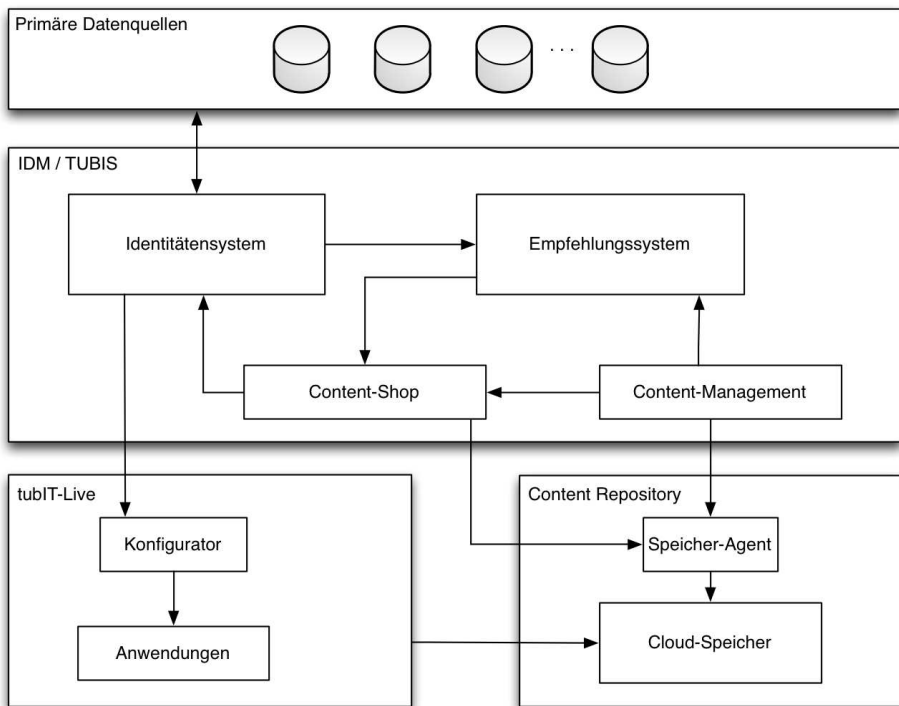


Abb. 2: Zusammenhang: Datenquellen–IDM–tubIT Live

Inhalte, Software und Nachrichtenkanäle können dabei von den Verwaltern der jeweiligen Inhalte mit Metadaten versehen werden, die es dem System ermöglichen diese Komponenten gezielt den gewünschten Empfängern zur Verfügung zu stellen. Mit der direkten Verknüpfung des tubIT Live Systems an das zentrale Identitätsmanagement System existiert zudem die Möglichkeit die Metadaten der Studierenden wie Semesterzahl, Studiengang etc. zu erfassen und Inhalte gezielt zu verteilen. Diese recht statische Zuordnung von Inhal-

ten stellt sich als wenig flexible heraus und bietet den Studierenden keine Möglichkeit auf den Umfang oder die Art der zur Verfügung stehenden Inhalte Einfluss zu nehmen. Um dieses Ziel zu erreichen, wurde tubIT Live zusätzlich um ein Empfehlungssystem erweitert. Über ein online Shop ähnliches System können Studierende alle ihnen empfohlenen Inhalte sehen und aus- bzw. wieder abwählen. Darüber hinaus besitzen sie die Möglichkeit auch auf andere Inhalte zuzugreifen, sofern diese nicht auf bestimmte Zielgruppen beschränkt sind. Änderungen in der Auswahl wirken sich dabei im Fall von Inhalten sofort aus oder im Fall von Software beim nächsten Start des Live Systems. In zukünftigen Versionen werden die Empfehlungen nicht mehr ausschließlich auf statischen Werten wie Studiengang, Semester oder ggf. der Auswahl von Veranstaltungen basieren, sondern auf Grundlage einer breiteren Datenbasis getroffen werden. Eine genauere Beschreibung des Empfehlungssystems wurde in [RBK14] oder mit Bezug auf die Erstellung eines Studienplans in [BRH14] vorgestellt. Mit der Kopplung des tubIT Live an das zentrale Identitätsmanagementsystem war es zudem möglich, die Verteilung des Systems in den Provisionierungsprozess der Studierenden zu integrieren, die jede/r Studierende zu Beginn seines Studiums durchlaufen muss. Mit dem Einschreiben bekommt die/der Studierende einen Brief mit einem temporären Passwort zur Erzeugung eines Nutzerkontos. Nach Wahl des Benutzernamens und eines Passworts werden die Ressourcen für den Nutzer freigeschaltet. Hierzu zählen die Netz- und Portalzugänge wie auch Zugang zum ownCloud-Speicher. Am Ende wird unmittelbar auf den Download des USB- bzw. VM-Images verwiesen bzw. auf die Möglichkeit der Erstellung eines USB-Sticks im PC-Pool. Beim ersten Starten des Systems wird nach dem im Provisionierungsprozess gewählten Benutzernamen und Passwort gefragt. Danach ist das System komplett konfiguriert.

5 Implementierung des Systems

Bei der Implementierung des Systems wurde großen Wert auf die leichte Wart- und Erweiterbarkeit gelegt, so wie auch auf einen hoch verteilten Ansatz. So wurde zu Beginn des Projektes ein GitLab Server [Gi15] aufgesetzt, über den versioniert die gesamte Kontrolle der Konfiguration und der Paketpflege läuft. Ein weiteres Designziel war die Sicherheit des Systems. Es ist selbstverständlich, dass ein USB-Stick entwendet oder verloren werden kann. Befinden sich darauf alle nötigen Zugangsdaten und alle studienrelevanten Dokumente, so ist darauf zu achten, dass ein Zugriff auf diese sensiblen Informationen oder ein Ausspähen der Zugangsdaten nicht möglich ist.

Administriert wird das System über einen Webbrowser. Alle Konfigurationsdateien und Skripte der Bauprozesse werden in git-Repositories verwaltet und dort versioniert. Zur Verwaltung der git-Repositories wird ein GitLab-Server verwendet. Jede Komponente des Systems (Modul) liegt in einem eigenen Repository. Eine Änderung im Repository löst über einen get-Request vom GitLab-Server beim control-Daemon (als Webserver implementiert) ein erneutes Erstellen der Images aus. Der control-Daemon synchronisiert die zum Bau des Livesystems benötigten Module aus den GitLab-Projekten auf den Buildserver und baut neue Images für die unterschiedlichen Varianten des Livesystems mit Hilfe des für den Bauprozess zuständigen Moduls.

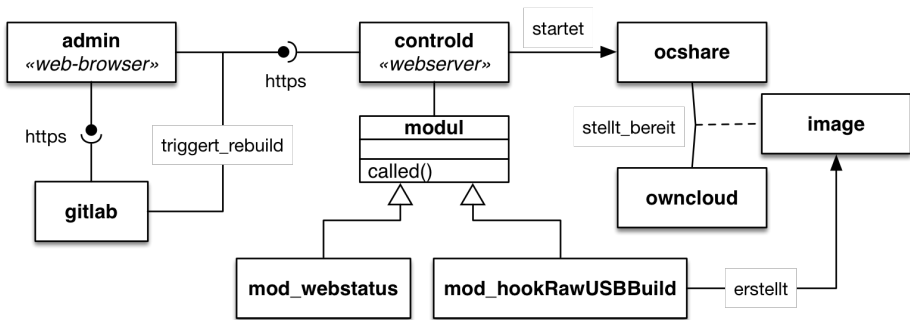


Abb. 3: Klassendiagramm Bauprozess

Die Weboberfläche des control-Daemons erlaubt Administratoren die Übersicht des aktuellen Status, die Ansicht der zuletzt gebauten Images, das manuelle Triggern der Bauprozesse, sowie das Vornehmen von Änderungen an den Einstellungen des Daemons selbst.

Ist ein Image erstellt und getestet, kann es über das ocshare-Script via ownCloud bereitgestellt werden. Somit ist der gesamte Life-Cycle in dem System abgebildet. Änderungen können über die Logs im GitLab nachvollzogen werden. Durch die Versionierung kann jederzeit auf einen älteren Stand zurückgegangen werden, sollten beispielsweise Änderungen den Bauprozess stören. Das gesamte System baut auf dem GitLab-Server und dem build-Server auf, auf dem der control-Daemon läuft. Auf Grund der Modularisierung der verschiedenen Projekte im GitLab ist ein verteiltes Arbeiten in verschiedenen Teams möglich. So kann die Oberfläche z.B. vom Kundendienst gestaltet werden, wohingegen neue Softwarepakete vom UNIX-Team erstellt und hinzugefügt werden.

6 Aufbau des USB-Sticks

Initial, also vor der ersten Nutzung, existiert auf dem USB-Stick eine logische Partition. Auf dieser sind das Tinycoresystem [Ti15], die Konfigurationsskripte, sowie das Basissystem gespeichert. Das Basissystem liegt in komprimierter Form vor und ist nur lesbar.

Zum Start der Systeme wird je nach Variante des Sticks extlinux [ES15] für BIOS-System oder rEFInd [Sm15] für (U)EFI-Systeme als Bootloader genutzt.

Nachdem die Konfiguration des Livesystems über das Konfigurationsskript abgeschlossen wurde, befinden sich zwei weitere Partitionen auf dem USB-Stick. Eine FAT32-Partition soll dafür sorgen, dass der USB-Stick nicht versehentlich unter Windows formatiert wird, da die anderen Partition nicht genutzt und geöffnet werden können. Eine entsprechende Textdatei in dieser Partition erklärt den Sachverhalt zusätzlich. Die dritte Partition füllt den kompletten restlichen Speicher des USB-Sticks aus und wird zur Speicherung aller Daten aus dem Livesystem genutzt. Die Partition ist über cryptsetup verschlüsselt, ein darüberliegendes ext4-Dateisystem wird mittels OverlayFS eingehangen, sodass alle am Basissystem vorgenommenen Änderungen in dieser Partition gespeichert werden.

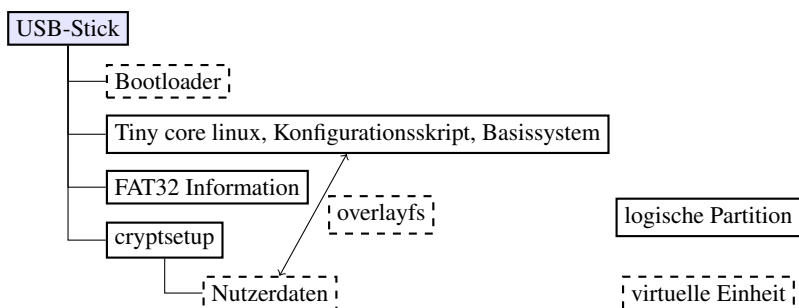


Abb. 4: Aufbau des USB-Sticks

Durch diesen Aufbau wird auch die minimale Größe nutzbarer USB-Sticks festgelegt. Die Größe der initialen Partition, die dem herausgegebenen Image des Livesystems entspricht, gibt die Minimalanforderung an Speicherplatz eines USB-Sticks vor. Der restliche Speicherplatz wird für Änderungen am System und für vom Nutzer erstellte Dateien genutzt. Für das aktuelle Livesystem wird ein USB-Stick mit einer Größe von mindestens acht Gigabyte empfohlen.

Die Variante des Livesystems unter Nutzung einer virtuellen Maschine funktioniert analog. Anstelle eines Images für USB-Sticks wird eine virtuelle Festplatte zur Verfügung gestellt, deren Inhalt dem der USB-Images entspricht.

7 Konfigurationsskript

Im Tinycore-Teil des Systems befindet sich als grundlegender Bestandteil das Konfigurationsskript. Tiny Core bietet ein minimales und kleines Linuxsystem (ca. 100MB). Es wird beim ersten Start des Livesystems (Setup) und bei einer Neuinstallation (Recovery) ausgeführt.

Aufgabe des Konfigurationsskriptes ist es, das Livesystem vorzubereiten und zu personalisieren. Zunächst werden über eine graphische Benutzeroberfläche die nötigen Benutzerdaten wie Benutzername, Kontoname und Passwort abgefragt.

Zur weiteren Personalisierung wird dann die Partitionierung des Datenträgers erweitert sowie Dateisysteme und deren Verschlüsselung mit dem Benutzerpasswort eingerichtet. Die anschließende Anpassung des Basissystems zur Personalisierung erfolgt über das Overlayfs in den verschlüsselten Bereich. Dabei werden Konfigurationsdateien diverser Programme und Systemtools mit dem Benutzerkonto und -passwort angepasst.

Nach erstmaligem Durchlauf der Konfiguration wird das Startmenü im Bootloader angepasst, sodass das eigentliche Livesystem zum Starten ausgewählt werden kann und das System neugestartet.

Das Konfigurationsskript kann jederzeit über den Bootloader ausgeführt werden und das System somit neu aufgesetzt werden.

tubIT-live Installation
Schritt 2: tubIT Login

Bitte geben Sie zur Konfiguration Ihres tubIT-live Systems Ihre tubIT-Benutzerkennung und Ihr Passwort ein.

Benutzername:

Passwort:

Passwort wiederholen:

vollständiger Name:

Abb. 5: Graphischer Installationsassistent

8 Cloudspeicher als „Enabling Technology“

Die Kombination des Livesystems mit der Cloudspeicher-Technologie ist von großer Wichtigkeit und macht das System in dieser Form erst so möglich. Der Cloudspeicher erfüllt in diesem Zusammenhang drei wichtige Aufgaben:

1. **Unabhängigkeit von der Hardware:** Die Studierenden können im Wechsel sowohl mit ihrem Stick, der virtuellen Maschine oder z.B. im PC-Pool arbeiten und haben jeweils Zugriff auf den aktuellen Stand ihrer Dokumente, da diese in die Cloud synchronisiert werden. Ein USB-Stick kann genauso wie jede andere Hardware leicht ersetzt werden. Die Daten werden automatisch auf das neue System synchronisiert.
2. **Der systemübergreifende Zugriff auf die Daten** bietet eine Schnittstelle zu anderen Plattformen. So steht das Livesystem nicht für sich allein, sondern kann in Kombination mit dem eigenen Windows-System oder einer mobilen Plattform genutzt werden.
3. **Das Teilen von Daten über den Cloudspeicher** ermöglicht nicht nur eine effiziente Zusammenarbeit mit den Kommilitonen, sondern ist auch eine gute Möglichkeit zur Integration in andere Software. Sie stellt das Recommender-System Lehrinhalte aus einem zentralen Pool über die Teilenfunktion den Studierenden zur Verfügung.

9 Ausblick

Neben der Bereitstellung des tubIT Live Systems wurde an der TU Berlin in die PC-Pools investiert. Ein Linux-basierter Lader kann entweder ein Linux-System lokal starten oder einen Client für die VDI-Umgebung bereitstellen. Weitere Funktionen sind ebenfalls möglich. So ist z.B. das Erstellen der USB-Sticks für das Live-System ebenfalls hierüber realisiert. Ziel ist es nun, das Linux-System der PC-Pools mit dem Live-System zu vereinen, so dass nur noch eine Distribution gewartet werden muss.

Ferner suchen wir andere Hochschulen, mit denen wir das Projekt gemeinsam fortführen können. Aus unserer Sicht bietet es sich an, dass Live-System in Module aufzuteilen,

die an verschiedenen Hochschulen gepflegt werden. Das Live-System lässt sich leicht an die jeweiligen Gegebenheiten an anderen Hochschulen anpassen und kann so als lokales Ökosystem dort bereitgestellt werden.

10 Fazit

Linux-Livesysteme gibt es schon lange. Sie haben jeweils ihre Nischen z.B. als Recovery- oder als Forensikwerkzeug gefunden. Die Verwendung verschiedener mobiler Geräte, wie Smartphones, Tablets und Laptops haben die Nutzer zu einer Multiplattform-Strategie gebracht. Es fällt heute einfacher, mit einem weiteren System zu hantieren, auch wenn es nicht das „Heimatbetriebssystem“ besitzt. Deshalb kommt tubIT Live auch mit einer Oberfläche, die denen der Mobilgeräte nachempfunden ist. Für die Nutzer bietet ein solches Livesystem alles an einem Ort. Für das Rechenzentrum bietet es die Möglichkeit den Support-Aufwand zu senken. Die Universität hat mit einem solchen System die Möglichkeit ein eigenes Ökosystem zu schaffen, wie dies von Apple, Google oder Microsoft bekannt ist mit diversen ineinandergreifenden Diensten, die hier in einer Plattform zusammengefasst werden können, die aber jeweils auch offen für die Nutzung durch andere Plattformen sind. Ohne diese Offenheit würde die Lösung nicht angenommen werden. Ein zentraler Punkt für die Realisierung ist der Einsatz des Cloudspeichers, der die Daten zusammenführt und auf der anderen Seite die Schnittstellen in die anderen Systeme aber auch zu anderen Benutzern bereitstellt.

Literatur

- [RBK14] Christopher Ritter, Patrick Bittner, and Odej Kao. Vom BYOE zu GYSE. In Paul Müller, Bernhard Neumair, Helmut Reiser, and Gabi Dreo Rodosek, editors, *7. DFN-Forum - Kommunikationstechnologien, 16.-17. Juni 2014, Fulda, Germany*, volume 231 of *LNI*, pages 67–74. GI, 2014.
- [BRH14] Patrick Bittner, Christopher Ritter, and Thomas Hildmann. Get your study plan. In Erhard Plödereder, Lars Grunske, Eric Schneider, and Dominik Ull, editors, *44. Jahrestagung der Gesellschaft für Informatik, Informatik 2014, Big Data - Komplexität meistern, 22.-26. September 2014 in Stuttgart, Deutschland*, volume 232 of *LNI*, pages 1861–1872. GI, 2014.
- [Gi15] GitLab Community Edition, GitLab B.V., <https://about.gitlab.com/>, Stand: 19.04.2015
- [Ti15] Tiny Core Linux, <http://www.tinycorelinux.net/>, Stand: 19.04.2015
- [ES15] Extlinux, Syslinux, <http://www.syslinux.org/wiki/index.php/EXTLINUX>, Stand: 19.04.2015
- [Sm15] The rEFInd Boot Manager, Roderick W. Smith, <http://www.rodsbooks.com/refind/>, Stand: 19.04.2015

A Installierte Anwendungen

Die folgende Liste zeigt die im Standard installierten Anwendungen:

Firefox: Safe and easy web browser from Mozilla

Evolution Mail: groupware suite with mail client and organizer

Libre Office: office productivity suite

Gimp: GIMP is an advanced picture editor.

FreeCAD: Extensible Open Source CAx program

KmPlot: mathematical function plotter for KDE

Docear: Docear is a unique solution to academic literature management, i.e. it helps you organizing, creating, and discovering academic literature.

Scilab: Scientific software package for numerical computations

Gummi: GTK+ based LaTeX editor with live preview

Eclipse: Extensible Tool Platform and Java IDE

Pycharm: Python IDE & Django IDE for Web developers

gitg: gitg is a fast git repository browser.