Handbuch für

MNS+ Debian

von Christian Meyer (c.meyer@rs-einrich.de)

Meine Anpassungen des Debian Betriebssystems zur Nutzung mit dem modularen Netzwerk für Schulen MNS+ – inklusive automatisierter Installation, Anmeldung am MNSplus Server und lokaler Änderungen – dürfen frei verwendet und nach den Regeln der **GPLv3** untersucht, geändert und weitergegeben werden: http://www.gnu.de/documents/gpl.de.html

Verstehen Sie daher meine Arbeit bitte als Geschenk an alle, denen es nützlich ist. Freuen Sie sich darüber, nutzen Sie es, geben Sie es weiter oder löschen Sie es.

Für Kritik und Verbesserungsvorschläge bin ich dankbar, allerdings übernehme ich **keinerlei Garantie** für irgendetwas, sichere keine Fehlerfreiheit zu und komme auch nicht für irgendwelche Schäden (z.B. an durch fehlerhafte Software zerstörten Computern) auf. Dies gilt sowohl für dieses Handbuch, als auch für die Konfiguration des Installationssystems FAI und den Anpassungen des Betriebssystems Debian zum Betrieb mit MNS+.

Bearbeitungsstand: Februar 2019 (MNS+-Debian 0.8)

Inhaltsverzeichnis

0. Vorbemerkung

Meiner persönlichen Meinung nach spielt das Betriebssystem eines Computers im täglichen Gebrauch keine große Rolle. Es gibt zwar persönliche Vorlieben und gewisse Gewöhnung und, aber egal ob Sie Windows, MacOSX, Android, iOS oder was auch immer verwenden, sie finden alles, um es einfach nach Ihren Wünschen und Bedürfnissen zu benutzen.
Das ist auch mit Linux nicht anders. Ich habe mir Mühe gegeben, dass Sie ganz einfach und ohne Linux-Vorwissen oder gar Programmierkenntnisse loslegen können.

Alles was Sie brauchen ist ein ungenutzter Computer, die Installations-CD, einen Internetzugang (sei es im MNS+-Netz oder auch zu Hause) und eine halbe Stunde Geduld (siehe Kapitel „**2.1. Los geht‘s! - Installation kurz und bündig**“).
Den Installationsvorgang habe ich der Betankung im MNS+-Netz nachempfunden, damit Sie nach wenigen einfachen Fragen (wie Zugangsdaten, Computername und Raum) nur noch abwarten müssen um mit dem fertig konfigurierten Arbeitsplatzrechner arbeiten zu können.
Wenn Sie möchten, dann können auch regelmäßig Updates der Konfiguration von meinem Server eingespielt werden, so dass Sie auch ohne tiefer gehende Kenntnisse von Verbesserungen profitieren können. Auch im Verwaltungsnetz oder zu Hause geht das genauso leicht, hier werden einfach die Teile weg gelassen, die nur für MNS+ wichtig sind (Anmeldung am Server, Netzlaufwerke, Fernsteuerung, …). Alles andere bleibt unverändert.

Ansonsten kann auch dieses Handbuch keine erschöpfende Einführung in das Betriebssystem Debian GNU/Linux sein. Es will vielmehr dem erfahrenen und interessierten MNS+-Anwendungsbetreuer die Besonderheiten von GNU/Linux näher bringen (Kapitel 1) und zeigen, wie ein solcher Debian-Rechner automatisiert installiert wird (Kapitel 2).
Außerdem soll es Supportern und Anwendungsbetreuern mit mehr Linux-Erfahrung die Besonderheiten der Installation und die Integration in das MNS+-Netz erläutern. Dadurch können Änderungen und Anpassungen, wie z. B. die Installation neuer Programme, selbst vorgenommen werden. Aber auch spezielle Konfigurationen und schul-, raum- oder rechnerspezifische Skripte können relativ leicht eingebaut werden.
Weiterführenden Informationen, die man benötigt, wenn man das System selbst anpassen möchte sind in Kapiteln beschrieben, die mit der Überschrift „**Hintergrund**“ beginnen. Diese Kapitel sind zum Verständnis der normalen Installation nicht nötig und können ausgelassen werden, wenn man das System erst einmal kennen lernen möchte.

1. Debian GNU/Linux

Auch wenn ich persönlich von Debian GNU/Linux überzeugt bin, möchte ich niemanden dazu drängen oder ihn gar „missionieren“. Andererseits gebe ich meine Erfahrungen und Entwicklungen gerne an Interessierte weiter.

Debian GNU/Linux ist freie Software, das heißt, es darf frei kopiert und weitergegeben werden. Aus den Anfangstagen von Linux hält sich bis heute das Gerücht, dass Linux schwierig zu installieren ist und nur „Freaks“ damit arbeiten.

Falls Sie noch nie (bewusst) mit einem Linux-System gearbeitet haben (Sie haben, da bin ich mir sicher) , dann lesen Sie dieses erste Kapitel und probieren Sie es danach einfach einmal aus – es wird viel einfacher sein, als Sie es erwarten.

Im folgenden Abschnitt möchte ich das Betriebssystem Debian GNU/Linux und seine Besonderheiten kurz vorstellen.

Tiefer gehende Informationen, wie auch den Debian Gesellschaftsvertrag und den Verhaltenskodex finden Sie auf der Homepage des Projekts

- https://www.debian.org/

1.1. GNU / Linux: Freiheit und Geschichte

Als Microsoft in den 1980er Jahren MS-DOS für PCs verkaufte, lief auf den Großrechnern das Betriebssystem UNIX. UNIX wurde zunächst offen entwickelt und kostenlos abgegeben. Als es allerdings stabil genug geworden war, wurde es nur noch kommerziell vermarktet.

Der Programmierer Richard Stallman wünschte sich die offene Entwicklung und die Kultur des gemeinsamen Gebens und Nehmens zurück und gründete deshalb das GNU-Projekt mit dem Ziel ein dauerhaft offenes und freies Betriebssystem zu entwickeln. Wichtige Schritte auf dem Weg dorthin waren die GPL, die GNU Public License, und die FSF, die Free Software Foundation.

Die GPL räumt dem Anwender Rechte ein statt sie zu beschneiden. So dürfen GPL-lizensierte Programme im Quelltext analysiert und angepasst und sogar (kostenlos) kopiert und weitergegeben werden. Die einzige Einschränkung ist: es gibt keine Garantie.

1991 veröffentlichte der Student Linus Torvalds seinen (aus Neugier entwickelten) Kernel „Linux“. Zu diesem Zeitpunkt war das GNU-System weitgehend fertig gestellt, es fehlte aber noch der GNU-Kernel. Nach kurzer Arbeit konnten GNU und Linux kombiniert werden und so ein wirklich freies Betriebssystem bilden. Nach und nach entstand immer mehr freie Software wie z. B. OpenOffice oder Firefox, die ebenfalls mit GNU/Linux kombiniert werden konnte.

Heute läuft GNU/Linux auf 98% aller Supercomputer, auf Internetservern, auf der ISS, beim chinesischen und amerikanischen Militär, in autonom fahrenden Autos, im MNS+-Netz als Server für virtuelle Maschinen und als Skolerouter, und auch bei Ihnen zu Hause auf dem Internetrouter oder als Unterbau für Ihr Smartphone. Einzig auf Desktop-Computern hat sich GNU/Linux noch nicht durchgesetzt.

Die Entwicklung von GNU/Linux wird von verschiedenen Projekten vorangetrieben, denen sowohl einzelne „Hacker“ angehören, als auch große (und konkurrierende) Firmen wie z. B. IBM, Oracle, Intel oder Apple, die Entwickler einstellen oder bezahlen um bestimmte Programme oder Funktionen zu entwickeln, die sie benötigen.

1.2. Debian Projekt

Bei so viel freier Software-Entwicklung wird es schnell unübersichtlich und es kommt zu Problemen im Zusammenspiel von verschiedenen Programmen. Hier kommen die Distributionen auf den Plan. Eine Distribution schnürt das „kreative Chaos“ an Programmen zu einem harmonisch funktionierenden Ganzen zusammen, indem es gut aufeinander abgestimmte Programmpakete mit sinnvollen Voreinstellungen erstellt und verteilt. Einige der mehreren hundert Distributionen sind kommerziell, andere auf ein Spezialgebiet hin ausgerichtet. Bekannte Distributionen sind z. B. Ubuntu, Suse oder RedHat.

Debian wurde 1993 von Ian Murdock gegründet und hat heute über 1000 offizielle Entwickler, die über 43.000 Softwarepakete betreuen. Debian ist basisdemokratisch organisiert und die meisten Entwickler arbeiten ehrenamtlich und in ihrer Freizeit an dem gemeinsamen Projekt, das sich selbst formulierte „Richtlinien für freie Software“ auferlegt hat. Im Gegensatz zu Ubuntu, Suse und RedHat ist Debian auch keine kommerziell orientierte Distribution und wird vielfältig eingesetzt.

Debian gilt als sehr stabil, ausgereift und zuverlässig und ist deshalb die Grundlage für viele andere Distributionen (angeblich über 480) wie z. B. Ubuntu, Knoppix oder Linux-Mint. Diese Tatsache und auch der offene Umgang mit Programmfehlern, bzw. das schnelle Schließen von Sicherheitslücken macht Debian zu meinem persönlichen Favoriten.

Recht interessant ist auch die offizielle „Einführung in Debian“:

- https://www.debian.org/intro/about

1.3. Linux als „~~pädagogische~~ politische Entscheidung“

Es gibt viele gute Argumente, Linux auf Schulcomputern zu installieren. Hier eine kleine Auswahl:

- Linux ist kostenlos – inklusive der kompletten Oberfläche mit Office, Internetprogrammen und Spielen. Mit dabei sind auch sowohl wissenschaftliche Programme, als auch Lernsoftware.
- Linux läuft auch auf alten und schwachen Computern. Dabei ist es trotzdem aktuell, sicher, modern und stabil.
  Haben Sie noch alte Computer im Einsatz, z.B. mit Windows Vista oder gar XP? Microsoft gibt keine (Sicherheits-) Updates mehr heraus und ein aktuelles Windows unterstützt so alte Hardware nicht mehr oder ist schlicht zu langsam? Computerviren und Hacker freuen sich. Bitte probieren Sie Linux!
  Bei uns laufen diese Altrechner (z.B. zur Anzeige des Vertretungsplans) mit einem aktuellen und sicheren Linux. Sicherheitsupdates gibt es übrigens rund um die Uhr (nicht nur am „Patchday“) und sie werden täglich eingespielt.
- Microsoft hat auf dem Betriebssystemmarkt eine Monopolstellung. Auch in der Biologie sind Monokulturen anfällig für Krankheiten (z.B. Viren) und Totalausfälle. Ein Ökosystem wird durch Heterogenität stabiler.

Der Grundgedanke hinter GNU/Linux ist die „Freiheit“. Richard Stallman spricht von "Free as in freedom" und unterscheidet „kostenlos“ von „frei“. Daher definiert er das „free“ in „free Software“ folgendermaßen: "Free software is a matter of liberty, not price. To understand the concept, you should think of free as in free speech, not as in free beer."

Als Lehrer wollen wir freies Denken, Vielfalt und Demokratie vermitteln, wollen kritische Bürger mit MINT-Kentnissen für die Gesellschaft von morgen erziehen.
Was aber leben wir den Schülern von heute vor: Lass dir von großen Konzernen die Freiheiten und Rechte nehmen, sei ein braver Verbraucher und konsumiere, was du vorgesetzt bekommst.
Mit Linux haben wir wenigstens die Chance, Alternativen zu zeigen, Datenschutz ernst zu nehmen und technische Kreativität zu fördern.

1.4. Linux ist anders als Windows

Trotz aller guten Argumente für Linux: Erwarten Sie nicht, dass sich Debian so verhält, wie Windows. Nein, das tut es nicht, und das ist gut so. Es wird auch niemand auf die Idee kommen, den „Internet Explorer“ unter Apples Betriebssystem OS X verwenden zu wollen, oder „Microsoft Office“ auf seinem Android-Smartphone. Trotzdem funktionieren beide tadellos und lassen beide Betriebssysteme nach kurzer Eingewöhnung kaum Funktionen vermissen.

Ähnlich ist es mit Linux. Natürlich unterscheidet sich Debian von Windows und auch die Arbeitsoberfläche „fühlt sich anders an“, sie hat aber vergleichbare Funktionen.

Dass Linux ein Mehrbenutzersystem mit eingeschränkten Benutzerrechten ist, werden Sie nicht merken, weil die Windows-Computer unter MNS+ ebenso konfiguriert sind. Dass die Partition mit dem Betriebssystem nicht „C:“ sondern „/“ (=“root“) heißt, wird Ihnen im MNS+-Netz ebenfalls kaum auffallen, da sie auch unter Linux Ihre MNS+-Verzeichnisse zur Verfügung haben. Und natürlich muss auch ein Linux-System vor dem Ausschalten heruntergefahren werden.

Wie man sich an- und abmeldet bzw. Programme startet, bekommen Sie intuitiv heraus, haben Sie einfach etwas Geduld beim ausprobieren. Die meisten Symbole oder Beschriftungen sind zumindest ähnlich (z. B. Arbeitsplatz, Schreibtisch, Desktop), auch wenn sie manchmal an anderer Stelle zu finden sind. (Wo ist das Startmenü?, Wie schalte ich den Computer aus?)

Der wichtigste Unterschied zu Windows ist das allumfassende Paketsystem von Debian, mit dem man nicht nur den Computer aktuell hält, sondern mit dem man auch gleich über 40.000 Programmpakete installieren bzw. deinstallieren kann. Doch dazu mehr im nächsten Abschnitt.

1.5. „Ersatz“programme unter Linux

Firefox oder Chrome werden Sie von Windows kennen und auch in Debian wiederfinden. Ebenso ist es mit OpenOffice (bzw. LibreOffice), das eine vollwertige Alternative für „MS Office“ ist, auch wenn sich die Bedienung etwas unterscheidet.

Mittlerweile gibt es zwar schon viele Hersteller, die ihre Programme sowohl für Windows als auch für Linux herausgeben (z.B. Stellarium, Promethean ActiveInspire, Geogebra, LibreOffice, Arduino), es wird aber nicht funktionieren, wenn Sie „mal eben so“ ein Programm für Windows unter Linux verwenden möchten.

Mit WINE existiert sogar eine Kompatibilitätsschicht, mit der sich einige (vor allem einfachere) Windowsprogramme auch unter Linux zum Laufen bringen lassen, doch die bessere Herangehensweise ist es, sich ein Ersatzprogramm mit den gewünschten Funktionen auszusuchen. In den allermeisten Fällen brauchen Sie dazu allerdings keine CDs einzulegen und auch nichts von Hand „irgendwo aus dem Internet“ herunterzuladen. Probieren Sie einfach mal das Programm „Synaptic“. Hier suchen und finden Sie über 40.000 Programmpakete für alle Lebenslagen und können diese mit einem Klick (inklusive aller Abhängigkeiten, also aller anderen dafür nötigen Programmpakete) installieren.

Es gibt nur sehr wenige Fälle, bei denen es keine „freie“ Alternative für ein Windowsprogramm gibt. Meist ist dies dann politisch motiviert und vom entsprechenden Hersteller oder Unternehmen nicht erwünscht (wie z. B. bei DRM – Digital Rights Management). Dass es möglich ist, auch unter Linux kommerzielle (oder patent-rechtlich geschützte) Programme zur Verfügung zu stellen zeigen Firmen wie Adobe (mit Flash oder Acrobat) und Promethean (mit Active Inspire).

Eine (sicherlich unvollständige) Liste von Programmen, die meiner Meinung nach für den Schuleinsatz geeignet sind, finden sie im Anhang.

1.6. Oberflächen (GUIs – Graphical User Interfaces)

Anders als bei Windows gibt es bei einem GNU/Linux-System eine Vielzahl von unterschiedlichen (aber gleichberechtigten) grafischen Oberflächen. Sie haben die „Qual der Wahl“. Bei Autos ist das völlig normal und die Frage, ob man VW, Audi, Citroen oder Mercedes fährt, ist eine persönliche Entscheidung.
Wenn Sie die Oberflächen nicht erst ausprobieren möchten und statt dessen (blind) meiner Wahl vertrauen, dann verwenden Sie Gnome für (einigermaßen) moderne Computer und das ressourcenschonende LXDE für leistungsschwache Rechner.

1.7. Sicherheitsüberlegungen: Viren, Rechte, Grub, VTs

Sicherheit ist unter Linux im Allgemeinen kein großes Problem-Thema. Zum einen ist Linux für „normale Cyberkriminelle“ oft noch zu uninteressant, da es nur einen geringen „Marktanteil“ hat. Zum anderen können Schadprogramme durch die abgestufte Rechtevergabe, restriktive Grundeinstellungen und schnelle Sicherheitsaktualisierungen nicht so leicht das ganze System lahmlegen.

Auf der anderen Seite kann man sich selbst natürlich durch zu unüberlegte Einstellungen selbst eine Sicherheitslücke konstruieren. Deshalb bedarf es einer gewissen Sensibilität für potentielle Angriffsszenarien und sinnvolle Sicherheitskonstruktionen.

Im MNS+-Netz befinden sich die Computer hinter einer Firewall und dem MNSplusProxy „Skolerouter“. Dies reduziert die Gefahr für Angriffe „von Außen“. Gegen Angriffe „von Innen“ setzen die Windows-Computer im Netzwerk vor allem auf die mandatorischen Profile der Windows-Domäne.
Unter Linux habe ich dies nachgebildet: Bei der Anmeldung am MNSplusDC werden lokalen Benutzerprofile (mit eingeschränkten Rechten) erstellt, die direkt bei der Abmeldung wieder gelöscht werden. Die Grundeinstellungen dazu werden lokal generiert und per Anmeldeskript weiter angepasst. Zusätzlich werden anschließend (wie auch bei Windows) vorher hinterlegte Profile vom MNSplusDC auf die einzelnen Rechner kopiert.

Weitere Angriffspunkte gibt es im Bootvorgang. Deshalb sollte darauf geachtet werden, die Einstellungen im BIOS so zu ändern, das nur von der Festplatte gebootet werden darf, evtl. (bei Netzwerkinstallation) auch noch per PXE. Mit Hilfe von USB-Sticks oder CDs könnte der Festplatteninhalt geändert und so das installierte System angegriffen werden. Gleiches gilt auch, wenn der Linux Bootloader (Grub 2) erlaubt, das System mit speziellen Optionen aufzurufen.

Unkritischer sind Zugänge zum Linux-System während des laufenden Betriebs. Dies ist möglich über virtuelle Konsolen (VT1 – VT 6, erreichbar über Alt+Strg+F1 bis F6) oder auch aus der Ferne mit SSH. In beiden Fällen muss man sich vor dem Zugriff auf das System zunächst mit den MNS+-Zugangsdaten anmelden. Danach hat der potentielle Angreifer nur lokale Benutzerrechte, kann also keine systemrelevanten Einstellungen (also solche, die das System oder andere Benutzer beeinträchtigen) machen. Wem der Anblick eines Terminals (statt grafischer Oberfläche) „zu gefährlich“ aussieht, der kann diese auch deaktivieren.

2. Installation der Debian-Rechner

2.1. Los geht‘s! - Installation kurz und bündig.

Sie wollen nicht viel lesen, sondern erst einmal loslegen und später schauen wie alles funktioniert? Sehr gut, hier ist die Kurzanleitung:

Zuerst sollten Sie sich ein paar Dinge überlegen, nach denen Sie bei der Installation gefragt werden:

- Möchten Sie Updates der Konfiguration einfach und bequem von meinem privaten Server beziehen, oder wollen Sie diese lieber selbst anpassen und pflegen (**lokale Konfiguration**) ?
- Wenn Sie mehrere Rechner im selben Netzwerk installieren möchten, dann sollten Sie sich (mit der selben CD) zuerst einen **FAI-Server** installieren. Der FAI-Server ermöglicht später auch die Installtion über das Netzwerk, er speichert die nötigen Debianpakete, die Konfiguration und die Logdateien der Installation bzw. Updates aller weiteren Debian-Rechner. (Siehe hierzu Kapitel „2.3 Hintergrund: FAI-Server“)
  Wenn Sie (vorerst) nur einen einzelnen oder wenige Rechner mit Debian brauchen (z.B. zum ausprobieren oder zur Verwendung zu Hause), dann brauchen Sie auch keinen FAI-Server.
- Damit die Installationsdateien aus dem Internet geladen werden können empfiehlt es sich, ein paar Internetseiten im Skolerouter freizuschalten. (siehe 3.6 Änderungen am alten Skolerouter: ProxyAllowSite).
  Sollten Sie keinen Zugang dazu haben und auch nicht den Supporter bemühen wollen, dann können Sie (vorerst) auch darauf verzichten: Ihre Zugangsdaten werden dann zum Abrufen der Installationsdateien bei der Installation des FAI-Servers gespeichert.

Genug der Überlegungen, jetzt geht‘s ans Installieren:

1) Besorgen Sie sich das Installationsimage von **https://chbmeyer.de/fai.iso** und schreiben Sie die Datei **fai.iso** auf einen USB-Stick oder eine DVD/CD.
   Wichtig ist, dass Sie die Datei nicht einfach nur auf den Stick kopieren, sondern **als Image** (**Abbild**) „brennen“. Unter Linux funktioniert dies z.B. im Terminal mit dem Befehl
   dd if=/Pfad/zur/fai.iso of=/dev/sdX bs=1M Leider haben sowohl viele USB-Sticks, als auch einige BIOS / UEFI-Versionen immer noch Probleme beim Booten per USB-Stick. Sollte es nicht klappen, probieren Sie bitte einfach einen andern Stick aus, oder verwenden Sie eine DVD.

2) Nach dem Einstecken des Sticks / Einlegen der CD in den betreffenden PC schalten Sie im BIOS (sofern noch nicht geschehen) „Secure Boot“ ab und ändern Sie die Bootreihenfolge, so dass der Rechner von dem USB-Stick / der CD booten kann (Eintrag vor der Festplatte). Starten Sie nach dem Speichern den Rechner neu.

3) Wählen Sie im Bootmenü, welche Konfigurationsvariante Sie bevorzugen (siehe oben). **Achtung:** Falls Sie sich dazu entscheiden, Updates von chbmeyer.de zu beziehen, dann wird Ihr Rechner alle Änderungen erhalten, die ich an der zentralen Konfiguration durchführe. Bei einem FAI-Server werden dabei allerdings Ihre eigenen Änderungen an der Konfiguration regelmäßig überschrieben. Wenn Sie das nicht möchten installieren Sie **lokal**.

4) Nach ein paar Bootmeldungen (bitte ignorieren) werden Sie im MNS+-Netz nach Ihren Zugangsdaten, dem Computernamen und in welchem Raum er stehen soll gefragt. Danach sehen Sie weitere Textmeldungen, die bei Problemen hilfreich sein können, die Sie aber genauso gut ignorieren können. Die Installation der Softwarepakete („task_instsoft“) dauert etwa 15-30 Minuten (abhängig von Ihrer Internetverbindung), anschließend wir der Rechner konfiguriert und neu gestartet.

2.2. FAI – Fully Automatic Installation (Deployment)

FAI (Fully Automatic Installation) ist ein System zur massenhaften Installation des Betriebssystem Linux. Dies wird auch als Deployment oder Rollout bezeichnet. Bei Windows erledigt diese Aufgabe das Windows Deployment System (WDS). Die Installation ist zunächst einmal unabhängig vom Einsatz im MNS+-Netz, allerdings gibt es in der Klasse „MNS“ ein paar Anpassungen für die Verwendung des Computers in einer Windows-Domäne.

Wenn Sie meine Anpassungen einfach übernehmen möchte, dann lassen Sie Ihren FAI-Server die Updates meiner Konfiguration automatisch einspielen (s.o.) – und Sie sind fertig.

Um FAI jedoch an seine Bedürfnisse anpassen zu können muss man wenigstens grob verstehen, wie FAI arbeitet. Die wesentlichen Ideen und Konzepte möchte ich in den kommenden „Hintergrund“ Kapiteln darstellen, eine sehr gute (und ausführlichere) Anleitung finden Sie auf der Homepage des FAI-Projekts unter: ***http://fai-project.org/fai-guide/***

Ausgangspunkt aller Anpassungen ist der FAI-Server. Dort werden Aktualisierungen und Protokolldateien zentral verwaltet. Außerdem hält er die Installationsdateien vor, erstellt angepasste FAI-CD‘s und ermöglicht auch die Installation von Computern per PXE (=Netzwerkboot). Die dazu nötigen Änderungen am MNSplus-Server finden Sie im Kapitel „5.3.3 Wie kann ich FAI per PXE booten?“.

2.3. Hintergrund: FAI-Server (ACNG, Softupdates, CDs)

Der FAI-Server ist optional, aber empfohlen. Der wichtigste Grund dafür ist, das Sie so selbst eigene Anpassungen erstellen und pflegen können. Wird bei der Installation eines Computers ein FAI-Server im Netzwerk gefunden, dann wird statt der Konfiguration des Bootsystems der CD automatisch die angepasste Konfiguration des FAI-Servers verwendet. CDs, die von einem anderen Server erstellt wurden werden zu einer (ansonsten harmlosen) Fehlermeldung führen, dass sich die Identifikation des Servers geändert hat.

Außerdem werden die zur Installation nötigen Paktete auf dem FAI-Server zwischengespeichert. Dazu verwenden wir den **apt-cacher-ng** oder kurz: **ACNG**. Die Datenmenge, die aus dem Internet heruntergeladen werden muss reduziert sich dadurch erheblich und gleichzeitig können neue Computer schneller installiert werden.

Auch für Updates brauchen wir den FAI-Server: Alle Debian-Computer führen täglich ein „**fai softupdate**“ durch und aktualisieren dadurch die installierten Programme sowie die Konfiguration. Unter ***/var/log/fai/*** haben Sie den Überblick über alle Installations- und Update-Logs (Protokolldateien), die mit dem Programm **faiwatch** schnell auf Probleme oder andere Auffälligkeiten hin untersucht werden können.

Schließlich können Sie mit dem FAI-Server auch eigene (angepasste) Installations-CDs erstellen. Üblicherweise läuft auch das (einmal wöchentlich) automatisch, die ISO-Abbilddateien der letzten drei Wochen finden Sie unter */srv/fai* , wie Sie ein Bootmedium erstellen ist im Kapitel „2.1 Los geht‘s“ beschrieben. Benötigen Sie eine „brandaktuelle“-CD, dann erstellen Sie diese mit dem Befehl ***/usr/local/share/fai-server-init***.

Bequemer als die Installation mit DVD bzw. USB-Stick ist eine **Netzwerkinstallation** (per PXE), wie sie auch von Windows beim „Betanken“ verwendet wird. Um dies umzusetzen müssen wir auf dem MNSplus-Server ein paar kleinere Änderungen durchführen (Domänenadmin! → Supporter?). Siehe hierzu Kapitel „5.3.3 **Wie kann ich FAI per PXE booten?**“

Für eine ständige Verfügbarkeit (gerade per PXE) ist es hilfreich, den FAI-Server auf einem 24/7 laufenden (alten?) Computer zu installieren, oder noch besser: in einer virtuellen Maschine neben dem Windows-Server und dem Skolerouter. Siehe hierzu: „5.3.1 **Wie kann ich einen virtuellen FAI-Server installieren?**“

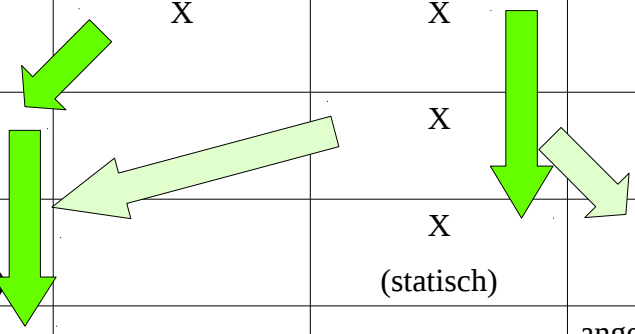
Sie können aber auch ruhig mit einem (normalen) Lehrer-PC als FAI-Server beginnen, und diesen dann später einfach durch einen anderen FAI-Server ersetzen. Dazu erstellen Sie sich einfach eine

Installations-CD mit der aktuellen Konfiguration (siehe oben) und sichern die ISO-Datei auf einem Netzlaufwerk des MNS+Servers. Nun entfernen Sie einfach den bestehenden fai-server in der MNS+-Raumverwaltung aus der Windows-Domäne (Achtung: Alle Konfigurationen werden gelöscht!) und installieren mit dem ISO-Abbild Ihren neuen (virtuellen?) fai-server. (Sie werden bei jeder neuen Installation danach gefragt, ob Sie einen FAI-Server installieren möchten, sofern kein Computer mit dem Namen **fai-server** im Netzwerk gefunden wurde.)

2.4. Hintergrund: Start des FAI-Installationssystems

Der Start des Installationssystems verläuft in 4 Schritten:

	FAI-Server	PXE-Server (MNS+)	CD / USB	Webserver chbmeyer.de
Kernel + Initrd (Booten)		X	X	
NFSROOT (Installationssytem)	X		X	X
Configspace	X (anpassbar)		X (statisch)	X
Paket-Cache, Logfiles	X			angepasste Pakete



Zuerst muss (über den Bootloader) der **Kernel** („Linux“ im engeren Sinn) und eine initiale Ramdisk, deren einzige Aufgabe es ist, das eigentliche Betriebssystem zu starten, geladen werden. Üblicherweise liegen Kernel und Ramdisk auf der zu bootenden Festplatte, bei der Installation kommen aber auch CD/USB oder aber ein Netzwerkboot (PXE) vom MNSplusDC aus in Frage.

Das **NFSROOT** ist das eigentliche Installationssystem, das alle dazu nötigen Werkzeuge mitbringt. Den Namen hat es, weil das „ROOT“-System über „NFS“ (einem Netzwerkdateisystem) vom FAI-Server geladen wird. Das NFSROOT kann aber auch auf CD/USB kopiert werden oder aber auch auf einen Internetserver.

Im **Configspace** (FAI_CONFIG_SRC) wird festgelegt, wie das zu installierende System beschaffen sein soll. Hier wird die Partitionierung festgelegt, die zu installierenden Programme sowie deren Einstellungen bis hin zu eigenen Skripten, die zusätzliche Anpassungen vornehmen. Sprich: der Configspace ist das Herz des ganzen Systems und er wird auch für das tägliche „***fai softupdate***“ der Arbeitsplatzrechner (FAI-Clients) benötigt.
Auf dem FAI-Server finden sie den Configspace unter: ***/srv/fai/config***

Nachdem Sie sich einen eigenen FAI-Server installiert haben, wird dieser als **Cache** für die Debianpakete verwendet. Das entlastet die Internetverbindung und bringt Geschwindigkeitsvorteile. Außerdem werden die **Logdateien** aller Installationen und späteren „Softupdates“ auf den FAI-Server übertragen, damit Sie jederzeit den Überblick über Probleme haben.

2.5. Hintergrund: FAI Klassen: Konzept und Vorgaben

Bei der Installation und Aktualisierung verwendet FAI sogenannte Klassen (CLASSES), um bestimmte Aufgaben und Funktionen in sinnvolle Einheiten aufzuteilen. Je nachdem, welche Aufgaben ein Computer erledigen soll, werden ihm einfach bei der Installation durch die Skripte im Ordner ***/srv/fai/config/class des FAI-Servers*** die entsprechenden Klassen zugewiesen.

Dabei bekommt jeder Rechner mindestens die drei Klassen DEFAULT, seinen eigenen Hostname und LAST zugewiesen. Hier eine Kurzübersicht der von mir definierten Klassen:

DEFAULT ist das minimale „Basis-Paket“, das das Debian-Grundsystem beinhaltet. Es ist zwar schon ein richtiges Debian-System, allerdings ohne grafische Oberfläche oder andere Programme, die nicht von wirklich allen Computern benötigt werden.

MNS beinhaltet meine Anpassungen für das MNS+-Netz. Hierzu gehören vor allem die Integration in die Windows-Domäne mit zentraler Benutzerauthentifizierung und die Netzlaufwerke der Benutzer. Aber auch die Raumzuordnung, Anmeldelog, PCStat sowie Proxy-und Drucker-Einstellungen gehören dazu. Diese Integration wird vor allem durch vielfältige Skripte realisiert.
Siehe hierzu die Detail im Kapitel: „3.1 MNS+-Skripte (Anpassungen für MNS+)“

PROXY installiert einen Proxyserver im MNS+-Netz. Falls die Zugangsdaten eines Domai-Admins eingegeben werden, wird der Skolerouter installiert (MNS+-Steuerung).

CLIENT Alle Computer, die mit Hilfe eines FAI-Servers installiert wurden haben auch die Klasse CLIENT. Hier wird festgelegt, dass ein tägliches Update durchgeführt werden soll und für Softwareinstallationen der Paket-Speicher des FAI-Servers verwendet wird. Das spart Bandbreite und ermöglicht schnellere Updates.

SERVER Der FAI-Server benötigt spezielle Programme wie den Apt-Cacher-NG oder faiwatch zur Überwachung des Clients. Außerdem verteilt er neue Konfigurationen, speichert Logfiles und baut neue Installations-DVDs.

GUI enthält Programme und Einstellungen, die für alle grafischen Oberflächen (Graphical User Interface) gemeinsam benötigt werden.

GNOME heißt die Klasse, die die Programme für die gleichnamige GUI enthält. Gnome 3 ist modern, animiert und benutzerfreundlich. (Siehe http://www.gnome.org)

LXDE heißt die Klasse, die die Programme für die gleichnamige GUI enthält.

LAST wird als letzte Klasse bearbeitet. Hier werden vor allem die Logfiles gespeichert.

2.6. Hintergrund: FAI Tasks: Ablauf von Installation, Update

Ein Installations- oder Updatevorgang besteht aus mehreren Schritten, die Tasks genannt werden und die auf dem FAI-Server unter ***/srv/fai/config*** konfiguriert werden. Die meisten Tasks sind als einfache Shell-Skripte realisiert. Auch hier möchte ich nur die nötigsten Ideen zusammenfassen und ansonsten auf den FAI-Guide verweisen.

defclass — Zunächst müssen die Klassen des Clients definiert werden. Dies geschieht bei der Installation durch die Scripte unter ***/srv/fai/config/class***. Die Klassen werden auf den Clients lokal in der Datei /***var/lib/fai/FAI_CLASSES*** gespeichert und bei Updates einfach wieder ausgelesen.

defvar — Abhängig von den Klassen eines Computers werden nun Variablen definiert. Hierzu gehört z.B. das root-Passwort, aber auch der APT-Proxy oder Informationen zum „Log-User“, der die Logfiles auf dem FAI-Server speichert. Diese Variablen werden in den Dateien ***/srv/fai/config/class/*.var*** hinterlegt. Das“*'“ steht für den jeweiligen Namen der Klasse in Großbuchstaben.

Danach wird $FAI_ACTION ausgewertet. Bei „install“ wird zunächst die Festplatte partitioniert (***/srv/fai/config/disk_config/****) und eingebunden, anschließend wird das Basissystem aus dem NFSROOT extrahiert und auf die Festplatte kopiert.

debconf — Viele Debian-Pakete können schon vor der Installation mit debconf konfiguriert werden. Hier wird z.B. das Tastaturlayout eingestellt. Die Dateien dazu finden sie unter ***/srv/fai/config/debconf/****.

updatebase — Jetzt ist es an der Zeit, die installierten Pakete zu aktualisieren. Dies spielt bei Debian innerhalb des „stable“ Zweiges nur Sicherheitsupdates und sehr wenige äußerst wichtige Updates in das System ein. Da das Debian-Projekt hier sehr sorgfältig und vorsichtig vorgeht, sollte das in aller Regel unproblematisch sein.
Wie auch immer: Falls Ihnen das zu gewagt erscheint können Sie diesen Task (wie auch jeden anderen) überspringen.
Dazu habe ich die eine eigene Konfigurations-Datei erstellt, die bei jedem softupdate ausgelesen wird: ***/srv/fai/config/fai-update.conf***

instsoft — Installiert neue Programmpakete aus den Debian-Repositories. Auch hier werden wieder die Programme installiert, die in der jeweiligen Klassen-Datei im Verzeichnis ***/srv/fai/config/package_config/**** definiert wurden.

configure — Besonders mächtig sind die (nach Klassen sortierten) Skripte in den Unterordnern von ***/srv/fai/config/scripts/**** . Hier lassen sich alle (sonst noch nicht berücksichtigten) individuellen Anpassungen des installierten Systems als Shell-Skript umsetzen. Insbesondere werden hier vorbereitete Konfigurationsdateien kopiert oder bestehende angepasst. Für die Klasse MNS wird hier z. B. der Beitritt zur Windowsdomäne bewerkstelligt, die „abwaschbaren“ Homeverzeichnisse konfiguriert oder die Passwörter für die lokalen Benutzer gesetzt.
Auch die weiteren Skripte, die die MNS+-Integration erreichen werden hier kopiert und für die spätere Verwendung eingerichtet.
Siehe hierzu das separate Kapitel „***3.1. MNS+-Skripte (Anpassungen für MNS+)***“

savelog — Wertet die Logfiles aus und kopiert sie auf den FAI-Server.

Darüber hinaus geben „hooks“ („Haken“) die Möglichkeit, sich (klassen- und taskabhängig) an beliebiger Stelle dieses Ablaufs einzuklinken. Die Hooks sind einfache Skripte und liegen im Verzeichnis ***/srv/fai/config/hooks/**** und werden jeweils vor dem gleichnamigen Task ausgeführt. Mit Hilfe von Hooks habe ich z. B. das Schreiben der PCStat(istik) auf den MNSplusDC umgesetzt.

Außerdem ist noch der Ordner ***/srv/fai/config/files/*** zu erwähnen.Hier werden (beliebige) Dateien abgelegt (z. B. Konfigurationen, Skripte, …) , die von einem der Skripte auf das Zielsystem kopiert werden sollen. Allerdings ist der (spätere) Dateiname ein Verzeichnis, in dem verschiedene Versionen der Datei gespeichert sein können – je nachdem für welche Klasse sie verwendet werden soll.

2.7. Hintergrund: Besonderheiten für FAI Skripte

Innerhalb des oben erläuterten FAI Installations- und Updateprozesses sind ein paar Besonderheiten von besonderem Interesse:

Zum einen muss man sich klar machen, das die Befehle beim Update auf dem gerade laufenden lokalen System ausgeführt werden, bei einer Neuinstallation ist aber das gerade laufende System das NFSROOT, das nicht verändert werden soll und auf dem evtl. auch nicht die benötigten Befehle zur Verfügung stehen. Deshalb bringt FAI zwei Variablen mit, die je nach Kontext so gesetzt werden, dass immer das (auf der Festplatte) zu installierende Zielsystem verändert wird.

$target wird Pfadangabe vorangestellt, damit es immer auf das lokale Festplattensystem zeigt: z. B.: ***$target/etc/hostname*** (statt einfach nur „/etc/hostname“)

$ROOTCMD wird Befehlen vorangestellt, die auf dem Zielsystem ausgeführt werden sollen, weil sie z. B. nur dort installiert sind, oder weil durch die Ausführung das Zielsystem verändert werden soll. z. B.: ***$ROOTCMD aptitude -ry install x11vnc***

Zum anderen muss man aufpassen, dass sich durch wiederholtes Ausführen eines Skripts keine Fehler einschleichen. Hier wollen z. B. Fehler abgefangen werden, falls etwa eine Datei bereits gelöscht oder eine Konfigurationsdatei schon einmal geändert wurde. Schwierig ist auch der Umgang mit verschiedenen Klassen, auch im zusammenspiel. Dabei helfen folgende Befehle

ainsl append-if-no-such-line: Einer Datei wird eine weitere Zeile hinzugefügt, aber nur, wenn eine bestimmte Zeile (oder ein Schlüsselwort daraus) noch nicht vorhanden ist.

ifclass kann bei Verzweigungen verwendet werden wenn Befehle nur bei bestimmten Klassen ausgeführt werden sollen.
ifclass MNS ist WAHR, wenn der ausführende Rechner der Klasse MNS angehört.

fcopy kopiert eine Datei aus dem Unterordner ***files/*** des FAI_CONFIG_SRC in das installierte System, sofern es in dem Unterordner eine Datei mit dem Namen der Klasse befindet.

2.8. Hintergrund: FAI Logfiles / faiwatch

FAI erstellt bei jeder Installation bzw. Aktualisierung eine Reihe von Logfiles. Neben den „normalen“ Linux Logfiles (unter /var/log) und Programmen zur Fehlersuche (z. B. dmesg oder journalctl) sind diese FAI-Logs ein guter Ansatzpunkt zur Fehlersuche und -behebung.

Auf jedem einzelnen Computer finden sie die aktuellen FAI-Logs unter: ***/var/log/fai/localhost/last/***

Die Datei **error.log** enthält alle (als wichtig identifizierten) Fehlermeldungen während der Installation bzw. während des Updates, sowie die Logdatei, in der Sie möglicherweise noch weitere Informationen zu dem Problem finden.

Ausführlichere Informationen finden Sie auch in der Datei **fai.log**. Diese enthält die normale Terminalausgabe des Aktualisierungsvorgangs. Zum Analysieren von Fehlern bei den Anpassungsskripten empfiehlt es sich außerdem, die Datei **shell.log** genauer zu betrachten.

Die Logdateien aller Debian-Clients werden außerdem an den FAI-Server übermitelt, der sie ebenfalls unter /var/log/fai/<Computername> speichert und so zentral archiviert.

Die Windows-Computer im MNS+-Netzen verwenden die „PC-Statistik“ um einen Überblick über die Installationen zu bekommen. Auch die Debian-Computer nutzen diese Übersicht (zumindest einen Teil davon). Während eines Updates wird der Hook savelog.MNS ausgeführt und Computer- und Betriebssysteminformationen an den MNSplus-Server übermittelt. Gab es bei einer der letzten FAI-Aufrufe ein Problem (Datei: error.log), dann wird dies ebenfalls festgehalten. Damit nur die aktuellen Einträge in der PC-Statistik auftauchen überprüft der FAI-Server mit dem Skript **40-clear-pcstat** die Datei auf dem MNSplusDC und aktualisiert sie bei Bedarf.

Einen etwas anderen Ansatz verfolgt das Skript ***/usr/local/share/faiwatch/faiwatch***, das nur auf dem FAI-Server vorhanden ist und so aber auf die Log-Dateien aller Clients zugreifen kann. Es gibt aus, wie viele Computer heute schon ein FAI Update erledigt haben. Alle anderen Computer werden zusammen mit dem Zeitpunkt ihrer letzten Aktualisierung aufgelistet. Anschließend werden alle Logdateien (**fai.log**) des „heutigen Tages“ nach Meldungen durchsucht, die nicht eindeutig den Standard-Meldungen von FAI zuzuordnen sind. Die Datei error.log wird dabei nicht berücksichtigt. Faiwatch wird durch die beiden Dateien **ignore** (Rechner, die ignoriert werden sollen) und **patterns** (bekannte FAI-Meldungen) konfiguriert. Damit ermöglicht faiwatch einen genaueren Blick auf mögliche Probleme bzw. auch Problemlösungen.
Interessant ist auch der Aufruf von faiwatch mit einer Datumsangabe, z.B. dem 18. Januar 2017: „***faiwatch -d 2017-01-18***“. Hier werden jeweils die aktuellsten Logdateien aller Rechner ausgewertet, die auch am 18. Januar 2017 aktualisiert wurden.

2.9. Hintergrund: Benutzer und Passwörter

Als Lehrer oder Schüler hat man seine gewohnten **MNS+-Zugangsdaten**. Diese werden vom MNS+-Server verwaltet und überprüft. Dies gilt auch für die Computer mit Debian.

Darüber hinaus gibt es **spezielle MNS+-Benutzer** wie den Notfallbenutzer oder den Lehreradmin. Das Passwort dieser lokalen Benutzer (d.h. kein MNS+-Server nötig) wird bei Windows bei jedem Systemstart vom MNS+-Server aus aktualisiert. Hier klinken sich auch die Debian-Computer ein, legen die entsprechenden lokalen Benutzer an und aktualisieren die Passwörter, die unter Linux (als „verschlüsselter“ Hash) in der Datei /etc/shadow gespeichert werden.

Der **ProfiladminNN** wird unter Windows zum Bearbeiten der Benutzerprofile verwendet. Diese Windows Profile sind nicht kompatibel mit Linux. Spezielle Debian-Profile werden aber unterstützt.

Schließlich gibt es noch den „allmächtigen Administrator“ **root**, dessen Passwort naturgemäß hoch sensibel ist. Im Kontext eines FAI-Servers gibt es drei verschiedene Systeme, die auch verschiedene root-Passwörter haben können:

- Wird ein **FAI-Server** installiert, dann fragt ein Skript nach den künftigen root-Passworten des Servers und der Clients. Das Passwort des Servers wird einmalig gesetzt und kann später jederzeit mit dem Befehl „passwd“ geändert werden.
- Während bei „lokal“ oder mit „git“ installierten Computern das root-Passwort einzeln vergeben werden muss, beziehen alle mit Hilfe eines FAI-Servers installierten **CLIENT-Computer** ihr root Passwort von dem jeweiligen FAI-Server. Dieses wird bei jedem Update aktualisiert und so zurückgesetzt. Um es (für alle Clients) zu ändern müssen Sie auf dem FAI-Server die Datei `/srv/fai/config/class/CLIENT.var` bearbeiten. Dort wird auch erklärt, was Sie dazu tun müssen.
- Mit dem FAI-Server erstellen Sie auch Ihr eigenes **FAI-Installationssystem**, mit dem (per USB, CD oder PXE) neue Computer installiert werden. Das Passwort für dieses NFSROOT-System wird in der Datei `/etc/fai/nfsroot.conf` eingetragen. Um nicht ein drittes (und wenig relevantes) Passwort abfragen zu müssen, verwende ich hier ebenfalls das Passwort der FAI-Clients.

Außerdem gibt es auf dem FAI-Server den Benutzer „FAI“. Dieser hat die Aufgabe, die Logfiles der Clients zu speichern. Er hat kein Passwort, die Authentifizierung wird per SSH mit einem Keyfile erledigt. Die Verteilung dieses Keyfiles wird von FAI selbst erledigt.

2.10. Hintergrund: GIT Versionsverwaltung

Lange Zeit habe ich regelmäßige Backups meines Configspaces durchgeführt, falls ich irgendetwas durch einen Fehler Kaputt gemacht hätte. Im Laufe der Zeit wurde das aber immer unhandlicher (Wann habe ich nochmal was geändert?), als mir die Versionsverwaltung GIT begegnet ist.

Die Idee dahinter ist einfach, die Archivierung und Pflege der Historie eines Projekts einfach, platzsparend und übersichtlich an ein darauf spezialisiertes Programm zu übergeben, das sich selbst im Alltag dezent zurückhält und nur dann kurz in Erscheinung tritt, wenn es benötigt wird.

GIT speichert die komplette Historie eines Projekts, wobei einzelne Änderungen thematisch gruppiert als „Commit“ zusammengefasst werden können. Es lässt sich aber auch jederzeit ein bestimmter Bearbeitungsstand „auschecken“, also wiederherstellen. Dadurch wird der Umgang mit alten Konfigurationen wesentlich vereinfacht.

Besonders schön ist dabei auch, dass mehrere Personen gleichzeitig an einem Projekt arbeiten können und weiterhin dass FAI wiederum seinen Configspace aus einem GIT-Repository beziehen kann. Dadurch lassen sich beispielsweise mehrere Schulen (oder privat genutzte FAI-Installationen) ganz bequem an einer zentralen Stelle pflegen: Die einzelnen FAI-Server holen sich den aktualisierten Configspace einfach regelmäßig von einem Internetserver und verteilen die neue Konfiguration schließlich an die FAI-Clients (Arbeitsplatzrechner).

Auch wenn man nach ***git add***, ***git status*** und ***git commit*** nur selten andere Befehle braucht, gibt es zu GIT gibt es jede Menge gute Dokumentation, auf die ich nur zu gerne verweise:

- CRE Podcast (allgemeinverständlich, nicht zu technisch)
  https://cre.fm/cre130-verteilte-versionskontrollsysteme
- Git Community Book
  https://git-scm.com/book/de/v2
  https://git-scm.com/doc
- Git-Tutorial
  http://schacon.github.io/git/gittutorial.html
- GIT cheat sheets (Die wichtigsten Befehle im Überblick)
  https://services.github.com/on-demand/downloads/github-git-cheat-sheet.pdf
  http://rogerdudler.github.io/git-guide/files/git_cheat_sheet.pdf
  https://www.git-tower.com/blog/git-cheat-sheet/

3. Integration in das MNS+ Netzwerk

Die Debian Rechner sollen sich nahtlos in das MNS+-Netzwerk einfügen und als intuitive Alternative zu den Windowscomputern benutzen lassen. Deshalb sind die Linux-Computer auch Mitglieder der Windowsdomäne.

Darüber hinaus werden die Windowsrechner mit verschiedenen Diensten und Skripten angepasst, um die gewohnten MNS+-Funktionen zu erhalten. Diese Anpassungen habe ich für Debian nachgebaut.

3.1. MNS+-Skripte (Anpassungen für MNS+)

Die MNS+-Computer müssen zu verschiedenen Gelegenheiten bestimmte Aufgaben erledigen. Unter Windows wird beim Hochfahren die PC-Statistik erstellt, beim Herunterfahren werden Updates eingespielt und Software installiert. Aber auch beim An- und Abmelden wird z.B. der Hintergrund eingestellt oder die Netzlaufwerke verbunden. Diese Windows-Skripte liegen auf dem MNSplus-Server in verschiedenen Verzeichnissen versteckt. Die verschiedenen Aufgaben werden hier meist durch „Gruppenrichtlinien (GPO)“ realisiert.

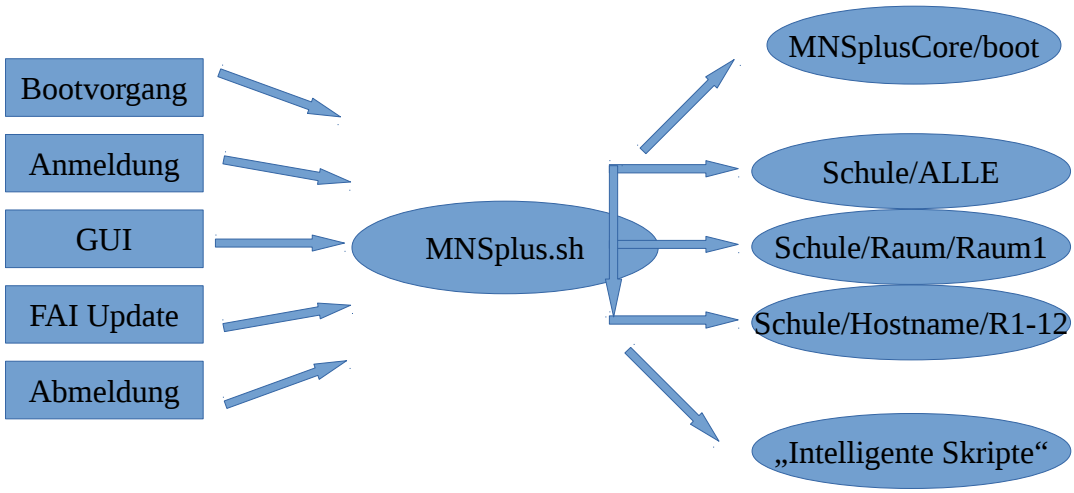
Unter Debian werden die nötigen Skripte beim täglichen Update vom FAI-Server auf die Arbeitplatzrechner kopiert. Die Skripte habe ich so entworfen, dass sie je nach Anlass des Aufrufs eine andere Aufgabe ausführen. Dabei werden zunächst die Grundfunktionen ausgeführt und danach speziellere Anpassungen für verschiedene Schulen, Räume oder auch einzelne Computer. So kann auch unter Linux das Hintergrundbild gesetzt, die richtigen Netzlaufwerke herausgefunden und verbunden, der Drucker gesetzt oder auch spezielle Programme installiert werden. Auch spezielle Konfigurationen (z.B. WLAN) sind möglich.

In den nächsten drei Kapiteln werde ich diese Skripte zunächst allgemein erklären, dann die Basiskonfiguration erläutern und auch individuelle Erweiterungsmöglichkeiten zeigen.

3.1.1. Hintergrund: Besonderheiten beim Aufruf von MNSplus.sh

Die Anpassungsskripte für MNS+ liegen im Verzeichnis ***/usr/local/bin*** der Arbeitsplatzrechner. Änderungen an der Konfiguration werden zentral auf dem FAI-Server unter ***/srv/fai/config/files/usr/local/bin/*** vorgenommen. Damit sie schneller erreichbar sind, ist auf dem FAI-Server der Ordner ***/MNSplus/*** ein Symlink auf den eigentlichen Pfad.

Das Herzstück der MNS+-Skripte besteht aus der Datei **MNSplus.sh**. Dieses Shell-Skript wird bei den verschiedensten Gelegenheiten aufgerufen, und verwaltet die Anpassungen für das MNS+-Netz. Deshalb wird der Kontext des Aufrufs dem Skript durch den ersten Aufrufparameter mitgeteilt (also z. B.: „***MNSplus.sh boot***“). MNSplus.sh wertet den Aufruf aus und ruft seinerseits verschiedene andere Skripte auf, die für die jeweilige Situation gedacht sind und eventuell weitergehende Anpassungen vornehmen.



Zunächst sind die für für die Funktion von MNS+ **essentiellen Skripte** („Core-Scripte“) mit den minimal notwendigen Anpassungen für den jeweiligen Kontext an der Reihe. Diese liegen im Ordner ***/usr/local/bin/MNSplusCore/*** und sind an jeder Schule gleich. Sie sind einfach nach ihrem Aufrufzeitpunkt benannt (also boot, fai, pam_open, ...). In unserem Beispiel wird also das Skript ***MNSplusCore/boot*** aufgerufen, das das System von möglichen „Überresten“ des letzten Herunterfahrens bereinigt.

Danach werden Informationen zum Raum, Computertyp, grafischer Oberfläche, etc. ermittelt und weitere, davon abhängige ***optionale Skripte*** ausgeführt. Dadurch sind eigene Anpassungen und Erweiterungen leichter möglich, ohne das etwas Wichtiges beschädigt wird. Hier lassen sich also schulweite Anpassungen oder auch spezielle Raum- oder Computerskripte getrennt verwalten. Auch zum Testen neuer Skripte oder geänderter Konfiguration auf einem einzelnen Computer eignet sich dieses Vorgehen, bevor sie schließlich ohne Gefahr für alle anderen Rechner freigegeben wird.

Diese optionalen Anpassungsskripte liegen unter ***/usr/local/bin/SCHULE/****.

Innerhalb der Skripte gibt es jeweils eigene Besonderheiten zu beachten, wie z. B. unterschiedliche Rechte, spezielle Umgebungsvariablen oder auch Befehle. Hier eine kurze Übersicht:

- boot — beim Booten des Computers
  (Aufruf als systemd System-Service, mit root-Rechten)
- fai — beim täglichen Update (root-Rechte, FAI-Skripte und -Befehle)
  z.B. $ROOTCMD, $target, fcopy, ainsl, ifclass)
  Siehe Kapitel „2.7 Hintergrund: Besonderheiten für FAI Skripte“
- pam_open — bei der Authentifizierung (PAM-Script, root-Rechte)
  PAM ist der Authentifizierungsdienst von Debian GNU/Linux.
  Er läuft deshalb mit root-Rechten, kennt aber den Benutzer, der sich gerade anmeldet (Variable $PAM_USER).
- systemd_user_start — beim Einloggen des Benutzers
  (Aufruf als systemd User-Service, Benutzer-Rechte)
  Auf Umgebungsvariablen, /home/, LDAP-Daten und Netzwerk-Shares kann mangels Authentifizierung (am MNSplusDC) noch nicht zugegriffen werden.
  Raum-Skripte oder Skripte für Lehrer-/Schülercomputer können also nicht aufgerufen werden.
  Systemd ist allerdings noch recht jung und das Verhalten kann sich ändern.
- gui — beim Start der grafischen Oberfläche. (~/config/autostart, Benutzer-Rechte)
  Intern wird noch weiter unterschieden, welche spezifische Oberfläche (gnome, lxde, …) gerade ausgeführt wird, so das sowohl Anpassungen für alle, als auch nur für einzelne grafischen Oberflächen möglich sind.
- systemd_user_stop — beim Ausloggen des Benutzers (Benutzer-Rechte).
  Es gibt ähnliche Beschränkungen wie bei „systemd_user_start“.
- pam_close — beim Schließen der Sitzung (PAM-Script, root-Rechte)
- cron — regelmäßig alle drei Minuten, erledigt Reparatur- und Wartungsarbeiten.
- shutdown — beim Herunterfahren des Computers (systemd System-Service, root-Rechte)

3.1.2. Hintergrund: MNSplusCore-Skripte

Die MNSplusCore-Skripte erledigen die für alle Rechner wichtigen Basisfunktionen zu verschiedenen Zeitpunkten. Die Skripte haben ihren Aufrufkontext als Dateinamen und liegen auf den Clients im Ordner ***/usr/local/bin/MNSplusCore/***. Gepflegt werden sie auf dem FAI-Server im Ordner ***/MNSplus/MNSplusCore/***

Aufruf bei:	Debian-Rechner (lokal)	Aktion auf MNSplusDC	Aktion auf FAI-Server
Systemstart boot (systemd, root)	Aufräumarbeiten	Autologin? Drucker im Raum?	
Anmeldung pam_open (root)	Veyon konfigurieren Pfade umbiegen	Raumabfrage Anmeldelog Netzlaufwerke mounten Profildateien kopieren	
systemd_user_start	Proxy setzen		
gui	Veyon starten, Menüeinträge		
gnome | lxde | ...	Hintergrundbild, Proxy setzen spezifische Anpass.		
Zeitgesteuert cron (root, 3 Min.)	Aufräumen, Fehlerdiagnose		
Softupdate fai	Aktualisierung von System und Konfiguration	Lokale Benutzer, Drucker abfragen, WLAN Config, PCStat schreiben	Softwareinstallation, Konfiguration, FAI Logs schreiben, Raum-/ Rechnerabh. Drucker installieren
Abmeldung systemd_user_stop	?		
pam_close (root)	Aufräumen	Abmeldelog, Laufwerke trennen	
Shutdown (root)	Aufräumarbeiten		

Weitere Informationen zu den einzelnen Anpassungen finden Sie im Kapitel „5.2. Details der Anpassungen für MNS+“

3.1.3. Hintergrund: optionale MNSplus-Skripte

Unter Windows gibt es die Möglichkeit, Programme oder Einstellungen schulweit, oder nur auf einigen Computern (z. B. den Rechnern eines bestimmten Raumes) einzurichten. Diese Einstellungen finden sich auf dem MNSplusDC im Netzwerkshare „Silent$“. Dies und noch viel mehr ist mit den optionalen Skripten auch unter Debian möglich.

Unter Debian habe ich zunächst die Struktur des Windows-Vorbilds übernommen. Sie findet sich auf dem FAI-Server unter /MNSplus im Ordner mit dem Namen des Schulnetzes, bzw. auf den Arbeitsplatzrechnern unter /usr/local/bin/. Der einzige Unterschied zum „Windows-Vorbild“ ist der, das die Skripte je nach Kontext mit verschiedenen Aufrufparametern gestartet werden (siehe Kapitel 3.1.1.), es können also Installation, Boot- oder Anmeldevorgang thematisch passend in einem Skript zusammen behandelt werden.

Die Skripte unter „ALLE“ werden schulweit ausgeführt, während „MNSRAUM/Raumname“, „MNSTYP/Schueler“ und „MNSHOSTNAME/Rechnername“ in Abhängigkeit der Informationen über den Computer (, die über LDAP vom MNSplusDC bezogen wird,) ausgeführt werden. Der Aufruf erfolgt in dieser Reihenfolge, damit ein einzelner Computer eine Raumkonfiguration auch wieder überschreiben kann. Um die Übersichtlichkeit zu wahren können die Skripte nach ihrer Reihenfolge und ihrem Zweck benannt werden, wie zum Beispiel: /usr/local/bin/RSK3/ALLE/01-automatisch-herunterfahren oder: /usr/local/bin/RSK3/MNSRAUM/Klassenraum/01-kein-easy-control.sh.

Innerhalb eines solchen Skripts können Sie mit dem Aufrufparameter „$1“ abfragen in welcher Situation das Skript gestartet wurde. In entsprechenden Subroutinen können Sie dann z. B. für „fai“ bestimmte Programme installieren, oder mit „boot“ oder „pam_open“ bestimmte Aktionen beim Hochfahren oder bei der Anmeldung eines Benutzers durchführen. Damit kann auch eine komplexere Aufgabe an einem gemeinsamen Ort gepflegt werden und man behält einen besseren Überblick über die Anpassungen.

Bei Tests mit dem gleichen Skript auf zwei verschiedenen Computern bekam ich allerdings recht schnell Synchronisationsprobleme bei den Anpassungen. Das parallele Aktualisieren des Skripts war reichlich umständlich und es störte mich, die selbe Aufgabe an zwei verschiedenen Stellen zu warten. Schließlich sah ich den Bedarf, dies mit „intelligenteren“ Skripten zu vereinfachen:

Alle Skripte, die direkt im Verzeichnis „/usr/local/bin/SCHULE/*“ liegen, werden immer (und auch von jedem Computer) aufgerufen. Ob das Skript aber tatsächlich ausgeführt wird, hängt davon ab, in welchem Raum sich der Computer befindet, welchen Typ er hat und wie er heißt.

Die Informationen darüber werden einfach von MNSplus.sh mit weiteren Aufrufparametern an des

entsprechende optionale Skript übergeben. Im Kopf des Skripts wird (kommagetrennt) in den entsprechenden Variablen angegeben, für welche Räume/Typen/Computer das jeweilige Skript ausgeführt wird. Erfüllt der aufrufende Computer keines der Kriterien, dann wird das Skript wieder beendet, ohne das es abgearbeitet wurde. Ein paar gut dokumentierte Beispielskripte habe ich für die Schule „SCHULE“ beigelegt.

3.2. MNS+-Fernsteuerung / Veyon

Leider funktioniert die MNS+-Fernsteuerung nicht unter Linux. Der Hersteller (FastViewer (?)) bietet das Programm nicht für Linux an und auch unter Wine ist es nicht lauffähig.

Allerdings gibt es freie Alternativen für Linux, ja sogar noch besser: Das Programm Veyon (Virtual Eye on Network) läuft nicht nur unter Linux, sondern auch unter Windows. So können Sie sich entscheiden:

Entweder lassen Sie die MNS+-Fernsteuerung für Windows-Computer und Veyon für Linux-PCs, oder aber Sie steigen (auch für Windows) komplett auf Veyon um und ersetzen so die MNS+-Fernsteuerung für die Windows Computer.
Eine dritte Möglichkeit besteht darin, die MNS+-Fernsteuerung für Windows-Computer zu belassen und Veyon trotzdem zusätzlich auf den Windows-PCs zu installieren. Damit können Sie dann (unverändert) mit der MNS+-Fernsteuerung von einem Windows-Lehrerrechner die Windows-Schülerrechner überwachen bzw. steuern und mit Veyon haben Sie dann Zugriff auf alle (also Windows- und Linux-) Computer.

Der einzige Wermutstropfen dabei ist: Veyon verträgt sich unter Windows nicht mit anderen VNC-Servern (wie z.B. UltraVNC), so dass ich UltraVNC dafür deinstallieren musste.

3.3. Whiteboards: OpenBoard, etc.

Whiteboards gibt es von vielen Herstellern und jeder bringt seine eigene Whiteboard-Software mit. Das ist mitunter schon unter Windows eine Zumutung und nicht sonderlich benutzerfreundlich. Schön wäre eine einheitliche Whiteboard-Software für alle Boards und Betriebssysteme.

Hier bietet sich OpenBoard (einem Fork von Open-Sankoré) an, das auch bereits vom PL für MNS+ (unter Windows) paketiert worden ist. Leider ist es noch kein offizieller Bestandteil von Debian, doch ich habe ein Debian Paket erstellt, das einfach lokal mit FAI installiert werden kann.

Darüber hinaus gab es auch eine Version von Active Inspire für Linux, die allerdings von Promethean nicht mehr unterstützt wird. Sollten Sie MasterTool verwenden (ich tue es nicht), so

können Sie es mit Hilfe von Wine auch unter Debian verwenden (siehe 5.3.6. Wie kann ich ein Windowsprogramm (mit wine) verwenden?).

3.4. MNS+-Schülermodul, Austeilen und Einsammeln, Modi, Fernsteuerung

TODO: Das „MNS+-Schülermodul“ hat lediglich die Aufgabe, Ordner anderer Schüler einzubinden. Dies sollte auch mit Debian gelingen. Ich denke an ein grafisches Shell-Skript, das mit gvfs und Eingabe des Passworts auf den MNSplusDC zugreift.

TODO: „Austeilen und Einsammeln“ ist ein reines Kopierskript. Hier kann die vorhandene History-Datei von Windows ebenfalls verwendet werden. Kein Problem, da Lehrer auf alle Schülerverzeichnisse Zugriff haben.

Der Klassenarbeitsmodus hat bei Windows Auswirkungen auf die Computer des Raums, in dem die Arbeit geschrieben wird und auf die Benutzer, die sich anmelden. Unter Debian konnte ich alle Aufgaben mit den Anmeldeskripten des Benutzers erledigen. Der Schüler kann nicht auf seine Dateien oder Tauschlaufwerke zugreifen, sondern kann nur mit den zur Verfügung gestellten Dateien arbeiten.

Ähnlich ist der Autologin für spezielle Räume. Wird beim Systemstart festgestellt, dass für den zugewiesenen Raum Autologin aktiviert ist, dann wird der entsprechende Benutzer automatisch angemeldet und seine Laufwerke eingebunden. Auch hier wird die Internetfreigabe mit dem MNS+-Kontrollzentrum verwaltet.

3.5. Gedanken zu Updates

Im schulischen Kontext gibt es zwei wichtige Bedürfnisse, die im ersten Moment widersprüchlich zu sein scheinen. Zum einen möchte man Änderungen (neue Programme, geänderte Konfiguration) schnell an alle Computer verteilen, zum anderen sollen stabil laufende Systeme nicht durch unnötige Updates oder fehlerhafte Skripte lahmgelegt werden.

Bei den Windows-Computern im MNS+-Netz wird beim Hochfahren die Konfiguration aktualisiert und Skripte ausgeführt, damit der Rechner bei der Anmeldung auf dem aktuellen Stand ist. Dies führt gelegentlich zu Wartezeiten bevor sich jemand anmelden kann. Aktualisierungen und neue Programme werden dagegen meist beim Herunterfahren installiert, damit der bei Windows nötige Neustart nicht im laufenden Betrieb stattfinden muss. Diese Wartezeit beim Herunterfahren hat in der Vergangenheit bei uns zu Problemen geführt, weil gerade bei Laptops der Deckel schon geschlossen oder das Netzwerkkabel gezogen wurde, bevor sich der Rechner ausgeschaltet hatte.

Auch muss bei Windows der Supporter die zu installierenden Windows-Updates auswählen, da hier immer wieder Probleme nach Updates zu beklagen waren.

Debian ist bekannt für seine Stabilität und konservative Updatepolitik. Während Sicherheitsupdates zeitnah und rund um die Uhr erfolgen gibt es so gut wie keine Updates innerhalb einer stabilen Veröffentlichung, die neue Features hinzufügen. Werden diese gewünscht, so müssen sie manuell (als „Backport“) installiert werden.

Unter Sicherheitsexperten gilt aktuelle Software als viel wichtiger als Antiviren-Programme. Deshalb habe ich mich dazu verleiten lassen, alle Debian-Updates automatisch einspielen zu lassen – bisher ohne Probleme. Eventuell ist hier aber in Zukunft noch einmal Handlungsbedarf. Dann kann das lokale Repository auf einem bestimmten Stand eingefroren werden und erst manuell (nach einem Testlauf auf einem besonderen Rechner) aktualisiert werden.

Der Aufruf eines Updates geschieht über einen Cronjob, der ein „fai softupdate“ (zufällig) innerhalb von 30 Minuten nach einem Systemstart auslöst. Das ist unkritisch, weil bei Linux ein Neustart nur bei einem Kernel-Update nötig ist (und das reicht dann auch am Ende des Schultages). Alle anderen Programme laufen unverändert weiter und werden erst beim erneuten Aufruf in der aktualisierten Version aufgerufen. So ist weder beim Booten, noch beim Herunterfahren eine Wartezeit nötig.

Auf die selbe Weise gelangt auch eine aktualisierte Konfiguration (z.B. Skripte) auf die Computer.

Zum Testen von neuen (oder geänderten) MNS+-Skripten hat sich bei mir der Ansatz bewährt diese erst (als optionales Skript) auf einem einzelnen Rechner ausgiebig zu testen und es anschließend einfach für die anderen Computer „freizugeben“.

3.6. Änderungen am alten Skolerouter: ProxyAllowSite

Der Skolerouter vom MNS+3 basiert auf Debian Wheezy und Squid 3.1. In dieser Version kann Squid seine Benutzer entweder mit NTLM (aus Zeiten von Windows XP) oder mit Kerberos (modernere Windowsversionen oder Linux) nutzen, aber nicht beides gleichzeitig. Leider benutzt auch MNS+3 immer noch NTLM. Deshalb ist (auch für Lehrer!) der Zugriff aufs Internet nur möglich, nachdem die Internetfreigebe für den entsprechenden Raum auf „Volle Freigabe“ gestellt wurde.

Darüber hinaus ist der Skolerouter so konfiguriert, dass bestimmte Seiten immer (also auch bei „keine Freigabe“) aufgerufen werden können. Dies ist z. B. für den Windows-Updateserver nötig. Ebenso ist es sinnvoll, die Debian-Repositories zumindest für den fai-server zugänglich zu machen. Die Konfigurationsdateien dazu finden sich auf dem Skolerouter unter ***/etc/skolerouter.d***.

Entweder Sie ergänzen die Datei ***ProxyAllowSite.extra*** um die Einträge „http.debian.net“, „httpredir.debian.org“, „security.debian.org“ und „chbmeyer.de“ (damit können alle Computer auf diese Seiten zugreifen), oder Sie erlauben (nur) dem FAI-Server, immer in das (voll freigegebene) Internet zu gelangen. Dies erreichen Sie durch einen entsprechenden Eintrag in ***ProxyAllowClientIPs.extra*** bzw. ***ProxyAllowClient.extra***. Nach der Änderung der Konfigurationsdateien müssen Sie den Proxy neu starten, z.B. mit: „/etc/init.d/squid3 restart“.

Sollte Ihnen der Zugriff auf den Skolerouter nicht möglich sein, dann können Sie bei der Installation des FAI-Servers auch Ihr ***Lehrerpasswort speichern*** lassen. Dieses wird in der (vor anderen Augen versteckten) Konfigurationsdatei auf dem FAI-Server gespeichert und nur für die Zwischenspeicherung der Debian-Pakete verwendet (apt-cacher-ng). Andere Benutzer können die Daten nicht auslesen und auch die mit Hilfe des FAI-Servers installierten Computer erhalten das Passwort nicht. Statt dessen holen sich die Clients die Pakete beim FAI-Server ab, der sie seinerseits aus dem Internet herunterlädt.

Sollten Sie jedoch später Ihr Lehrerpasswort ändern, dann kommt es beim FAI softupdate zu Fehlermeldungen wie „Fehlschlag beim Holen von http://http.debian.net“ oder „software.log: 404 Not Found“. Deshalb ist es empfehlenswert, die oben beschriebenen Änderungen am Skolerouter vorzunehmen.

4. Anpassung der Oberflächen

Es gibt für Linux gibt es eine große Vielfalt an grafischen Oberflächen – sie haben die freie Auswahl.

Sämtlichen wichtigen Aufgaben bei der Benutzeranmeldung übernimmt das Skript pam_open. Hier werden die Laufwerke eingebunden und das Anmeldelog geschrieben. Für einen Programmierkurs können Sie also beispielsweise auch völlig auf eine grafische Oberfläche verzichten.

Meine bevorzugten beiden GUIs sind Gnome (für moderne Rechner) und LXDE (für leistungsschwächere Geräte). Für diese beiden habe ich (abhängig von RAM und Prozessor) die nötigen Anpassungen bereits vorgenommen. Dabei habe ich mich auf den Anmeldemanager gdm3 festgelegt. Sollten Sie eine andere Oberfläche benutzen wollen, so können Sie gdm3 belassen, müssen aber an drei Orten Änderungen vornehmen:

- **Installation**: Bei der Installation wird für die zu verwendende grafische Oberfläche eine eigene Klasse definiert. Dies wird auf dem FAI-Server in der Datei ***/srv/fai/config/class/70-GUI.source*** festgelegt und bei der Betankung eines neuen Computers für diesem erledigt.
- **Paket-Installation**: Mit Hilfe dieser Klasse können Sie bei der Installation beliebige Schritte für Ihre Oberfläche erledigen lassen. Meist reicht es aber sicher, wenn Sie einfach nur die zu installierenden Pakete auf dem FAI-Server unter ***/srv/fai/config/package_config/OBERFLÄCHE*** eintragen.
- **GUI-Anmeldeskript**:Schließlich benötigen Sie auch noch ein Anmeldeskript für Ihre Lieblings-GUI. Hier wird z. B. das Hintergrundbild eingestellt oder Menü-Einträge angepasst. Dieses Skript wird ebenfalls auf dem FAI-Server eingerichtet und unter ***/MNSplus/MNSplusCore/oberfläche/MNS*** gespeichert. Als Namen verwenden Sie bitte statt „oberfläche“ den von der GUI in der Umgebungsvariablen ***$XDG_CURRENT_DESKTOP*** gespeicherten Wert (in Kleinbuchstaben).

Ansonsten sollten Sie noch wissen, dass die Basiskonfiguration eines neuen Benutzers aus den Dateien des Ordners /etc/skel kopiert wird. Wenn Sie hier eine Datei speichern oder ändern möchten, machen Sie das ebenfalls auf dem FAI-Server unter: ***/srv/fai/config/files/etc/skel/*** Beachten Sie bitte, dass der Dateiname auf dem Server immer ***MNS*** sein muss, weil diese Dateien für die Klasse MNS auf die Clients kopiert wird. Der spätere Dateiname ist der Name des Pfades der Datei.

5. Anhang

5.1. Software für den Schuleinsatz

Diese Liste soll helfen, Alternativen für gewohnte Windowsprogramme zu finden, sie ist aber nur eine kleine Auswahl. Die hier vorgestellten Programme sind meist bereits in Debian enthalten und können in wenigen Sekunden installiert werden.

5.1.1. Allgemein verwendbare Programme

Browser (statt InternetExplorer oder Microsoft Edge):

- Firefox
- Chromium (= Basis von Google Chrome)

Bilder und Bildbearbeitung:

- Shotwell (ordnet Digitalfotos)
- GIMP (Bildeditor, auch für Windows erhältlich)
- Inkscape (vektorbasiertes Zeichenprogramm, auch für Windows erhältlich)
- Blender (3D-Modellierung/-Rendering, auch für Windows erhältlich)

Fernsteuerung:

- Veyon zum Steuern der Schülerrechner (auch für Windows verfügbar)

Flash:

- Flash-Player (Browser Plug-in von Adobe, muss separat installiert werden)

Java:

- OpenJDK 7 Java
- (zur Not sollte auch Oracle Java gehen, muss seperat installiert werden)

Multimedia (statt MediaPlayer?):

- Totem
- VLC (auch für Windows verfügbar)
- Rhythmbox (inspiriert von Apples iTunes)

Mindmapping

- Vym (View Your Mind)
- Freemind (auch für Windows verfügbar)

Office (statt MS Office):

- LibreOffice (Fork von OpenOffice) mit

  Writer als Ersatz für Word
  Calc als Ersatz für Excel
  Impress als Ersatz für Powerpoint

- gedit statt Notepad
- klavaro (Lernprogramm zum Zehn-Finger-Schreiben)

PDF (statt Adobe Acrobat Reader):

- Evince (PDF Betrachter)
- CUPS-PDF (PDF-Drucker)

Whiteboard:

- OpenBoard (auch für Windows verfügbar)
- ActiveInspire? (Linux-Version vom Hersteller)

5.1.2. Schulfächer

Chemie:

- Periodensystem: GPeriodic oder Kalzium

Deutsch:

- Oriolus Lernprogramme (Windowsprogramm, läuft mit WINE)

Englisch:

- Lighthouse Workbook (Windowsprogramm, läuft mit WINE)

Erdkunde:

- Marble (virtueller Globus)
- Gnome-Maps
- GoogleEarth? (auch für Windows verfügbar)

Informatik / TuN:

- Scratch
- Arduino CC - Entwicklungsumgebung für Arduino
- Eclipse (Java-IDE)
- *Fischertechnik Robo Pro (könnte unter WINE laufen)*
- *fischertechnik-school designer (könnte unter WINE laufen)*
- *LEGO Mindstorms (schwierig: siehe http://wiki.ubuntuusers.de/Mindstorms)*
- *Logitron (könnte unter WINE laufen)*
- viele Programmiersprachen / IDEs

Mathematik:

- Geogebra (auch für Windows verfügbar)
- MatheGrafix (Windowsprogramm, läuft in WINE)
- Kopfrechentrainer (Windowsprogramm „KopfRechnenRennen“, läuft in WINE)
- SMILE Programme (z.B. LINEAL, STRAHL, BINOMI, QUADRA, ... laufen unter WINE)

Musik:

- NTed oder Rosegarden (Noteneditor)
- Audacity (Soundeditor, auch für Windows verfügbar)

Physik:

- Stellarium (Planetarium mit realistischem 3D-Himmel)
- Celestia (Weltraumsimulation)

5.2. Details der Anpassungen für MNS+

5.2.1. Hintergrund: LDAP – Sammelstelle für Informationen

LDAP ist ein **Verzeichnisdienst**, der Informationen zu Computern und Benutzern, ihre Berechtigungen und Zuordnungen speichert. Man kann es sich etwa vorstellen wie ein „Telefonbuch“, wo allerlei zusätzliche Informationen gespeichert und auch wieder ausgelesen werden können. Hier wird abgelegt, in welchem Raum ein Computer steht und ob es sich um einen Lehrer- oder Schüler-Rechner handelt. Auch die Benutzer und ihre Gruppen (Lehrer, Schüler, Klassen, Anwendungsbetreuer, …) sind hier gespeichert und bei Bedarf zu ermitteln. Unter Linux gibt es dafür die Programme ***ldapsearch*** und ***ldapmodify***.
Informationen zu den LDAP-Befehlen finden Sie im Internet und in den jeweiligen Handbüchern (man ldapsearch, man ldapmodify bzw. man net)

Die Informationen des „MNS+-Netzes“ über seine Benutzer, Computer und Gruppen werden vom MNS+-Server in einem LDAP-Verzeichnis gespeichert. Ein **Computer-Eintrag** sieht z.B. folgendermaßen aus:

`CN=Name,OU=Raum,OU=Computer_Lehrer,OU=SCHULE,DC=SCHULE,DC=mnsplus`

Sie erkennen darin neben dem Hostname auch den zugewiesenen Raum, als auch den Computertyp (Schüler oder Lehrer). Rechner, die keinem Raum zugewiesen sind, können auch keinen Computertyp haben. Der Pfad sieht dann so aus:

`CN=Name,CN=Computers,DC=SCHULE,DC=mnsplus`

Damit der MNS+-Server die Computer auch bei Neuinstallationen „wiedererkennt“, ist in den Attributen des Eintrags auch eine Computer-ID, die NetbootGUID gespeichert. Diese kann zu Problemen führen, da die Reihenfolge der Bytes unter Umständen (selten) vertauscht ist (Stichwort: Endianess). Dies liegt an der unterschiedlichen Behandlung der Daten von Windows bzw. Linux.

Analog ist die Abfrage bei den **Benutzern**. Der LDAP-Pfad enthält wieder die wichtigsten Informationen: **`CN=Lehrer\, Test (TestLehr),OU=Lehrer,OU=Benutzer,OU=Schule,DC=SCHULE,DC=mnsplus`**

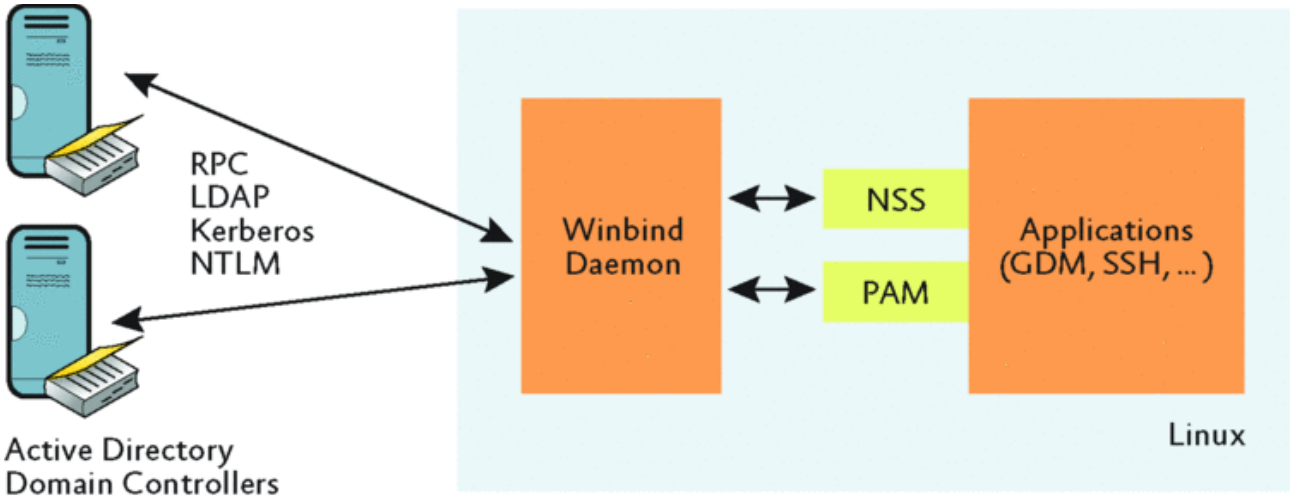
Auch hier sind in den Attributen des Eintrags noch weitere Informationen enthalten, wie z. B. Gruppen- oder Klassenzugehörigkeiten („memberOf“). Diese lassen sich zum Teil aber auch einfacher, z. B. mit dem Befehl **groups** herausfinden.

Ist ein Computer bereits Mitglied der Domäne, dann lassen sich die Informationen auch mit „***net ads status -P***“ (root mit Maschinenaccount) bzw. mit „***net ads status -k***“ (Benutzer) herausfinden.

5.2.2. Hintergrund: Authentifizierung und Autorisierung (Winbind)

„Normalerweise“ (also bei lokalen Benutzern) erfolgt die **Authentifizierung** („Wer möchte Zugang zum System bekommen?“) eines Benutzers anhand eines lokal gespeicherten (verschlüsselten) Passworts oder seines Hashes. Ist die Identität hinreichend überprüft worden, dann erhält der Benutzer die ihm erlaubten Berechtigungen zugeteilt („**Autorisierung**“). Dazu erhält der Benutzer diese Berechtigungen durch Mitgliedschaft in bestimmten Gruppen. Wird später überprüft, ob ein Benutzer eine bestimmte Aktion durchführen darf, so wird einfach systemseitig überprüft, ob er Mitglied in der entsprechenden Gruppe ist.

Die Konzepte von Windows und Linux unterscheiden sich hier in vielen Punkten. Deshalb Übernimmt beim Einbinden von Debian-Computern ins MNS+-Netz der System-Service **winbind** die Übersetzungsarbeit zwischen beiden Welten: Mit der Windows-Domäne „spricht“ winbind RPC, LDAP und Kerberos, auf der Linux-Seite stellt es Kontakte per NSS und PAM zur Verfügung.



Bildquelle und Rechte: http://i.technet.microsoft.com/dd228986.fig03%28de-de%29.gif

5.2.3. Hintergrund: Authentifizierung (PAM)

PAM ist unter Linux eine Softwarebibliothek für Authentifizierungsdienste („Pluggable Authentication Modules“). Mit Hilfe vom PAM kann ein (Login-) Programm relativ einfach überprüfen, ob ein Benutzer tatsächlich derjenige ist, der er zu sein vorgibt. Um dies zu überprüfen gibt es viele Möglichkeiten: vom einfachen Passwort über Fingerabdruck, Iris-Scan bis hin zur kryptografische Chipkarte. Im Fall von MNS+ wird diese Aufgabe an den MNS+-Server abgegeben.

Darüber hinaus werden von PAM auch noch weitere Aufgaben erledigt, die mit allen An- und Abmeldevorgängen zu tun haben. Wir brauchen es deshalb:

- für die Anmeldung von MNS+-Benutzern (**pam_winbind**) bzw. lokalen Benutzern (**pam_unix**)
- zum Anlegen eines (temporären) lokalen Benutzerverzeichnisses (**pam-mkhomedir**).
- zum Schreiben des An- und Abmeldelogs auf dem MNS+-Server und zum Einbinden der Netzklaufwerke des Benutzers (**pam-script**).
- zum Ausführen von Skripten mit Benutzerrechten beim An- und Abmelden (**pam_systemd**). Hierzu gehören das Einstellen eines Proxys oder des Hintergrundbildes.

Wichtiger Hinweis: Es liegt in der Natur der Authentifizierng, das solche gravierenden Aufgaben höchst sensibel sind. So kann eine Fehlkonfiguration von PAM beispielsweise dazu führen, das JEDER (ohne Anmeldung) Administrator-Rechte hat, aber auch dazu, das sich wirklich NIEMAND mehr am System anmelden kann, auch nicht der Administrator „root“ persönlich. Es gibt aber auch eine Reihe von anderen Problemen bei den oben genannten Aufgaben, die auf eine falsche Konfiguration von PAM zurückzuführen sind.

Bevor Sie hier (unter **/etc/pam.d/** bzw. **/usr/share/pam-configs/**) also etwas verändern, lesen Sie bitte unbedingt die (sehr gute) Dokumentation von PAM und seiner Module (**libpam-doc**) und lassen Sie sich Zeit, die Ideen und Konzepte zu verstehen!

Die folgenden Hinweise sind nur eine stark vereinfachte Übersicht, die nicht dazu gedacht ist , eine PAM zu erklären:

PAM kennt vier verschiedene **Typen von Modulen** (erste Spalte der Konfigurationsdateien):

auth	Diese Module dienen der Überprüfung des Benutzers. Sind die Zugangsdaten korrekt? Ist es wirklich derjenige, der er zu sein vorgibt?
account	Hier erfolgt die Authorisierung. Nicht jeder darf Alles. Wer darf sich wann anmelden? Welche Einschränkungen gibt es bei diesem Account?
password	Ändert das Passwort.
session	Verwaltung und Konfiguration von Benutzer-Sessions. Dinge, die vor Einloggen und nach Ausloggen eines Benutzers erledigt werden. Die Module werden bei der Anmeldung außerdem vor und nach der Authentifizierung gestartet, um z. B. Loginversuche zu protokollieren.

Die zweite Spalte enthält die **Kontroll-Flags**, die die Bedeutung des jeweiligen Moduls angibt:

required	Das Modul muss zwingend erfolgreich abgearbeitet werden. Dennoch werden auch beim Fehlschlagen eines required-Moduls noch alle anderen Module dieses Typs abgearbeitet, bevor der Benutzer eine Meldung über das Fehlschlagen erhält.
requisite	Wie die required Module müssen auch alle requisite-Module erfolgreich abgearbeitet werden. Im Fehlerfall wird der Vorgang aber unmittelbar abgebrochen, es werden keine weiteren Module mehr ausgeführt.
sufficient	Gab es bei den vorhergegangenen required-Modulen keine Fehler, dann werden ebenfalls keine weiteren Module mehr gestartet, das aufrufende Programm erhält sofort eine Erfolgsmeldung. Fehler bei sufficient-Modulen haben keine Folgen, statt dessen wird das nächste Modul gestartet.
optional	Erfolg oder Fehlschlag hat keinerlei Auswirkung auf die Authentifizierung.

5.2.4. Hintergrund: Autorisierung (Kerberos)

Kerberos ist ein (kryptografisch sicheres) standardisiertes Verfahren zur Vergabe von Berechtigungen in einer Netzwerk-Domäne. Sowohl Windows, als auch Linux haben dies schon eingebaut. MNS+ verwendet allerdings teilweise auch noch das ältere (und nicht mehr sichere) NTLM.

Nachdem sich ein Benutzer angemeldet hat, möchte er irgendwann bestimmte Dienste des MNS+-Netzes benutzen. Dies sind z. B. der MNS+-Proxy für den Internetzugang, Drucker oder die Netzlaufwerke des MNS+-Servers. Da diese Dienste über das Netzwerk angeboten werden, möchte man die Berechtigungen dazu kryptografisch sicher austauschen. Es ist auch nicht besonders bequem, für jeden Dateizugriff erneut sein Passwort eingeben zu müssen. Deshalb werden für die verschiedenen Dienste des Netzes sogenannte „Tickets“ vergeben, die die Aufgabe von „Berechtigungsscheinen“ übernehmen.

Bei Kerberos bekommt man bei seiner (erfolgreichen) Anmeldung ein TGT, ein „**Ticket Granting Ticket**“ übermittelt. Dieses TGT berechtigt dazu weitere Tickets für die einzelnen Dienste anzufordern. Selbstverständlich läuft dies vom Benutzer unbemerkt im Hintergrund automatisch ab.

Möchte ein Benutzer später z. B. auf das Internet (genau genommen: auf den Proxy-Server) zugreifen, so versucht sich der Browser zunächst völlig unwissend mit der Seite zu verbinden.

- Der Proxy lehnt die Verbindung ab und fragt nach dem „Passierschein“.
- Der Browser gibt diese Anfrage an das Betriebssystem weiter. Dieses hat zwar kein Ticket für den Dienst „Proxy-Server“, dafür verfügt es aber über das „Ticket Granting Ticket“.
- Also baut der Computer, an dem der Benutzer angemeldet ist, eine Verbindung zum MNS+-Server auf, zeigt das TGT zusammen mit der Anfrage des Proxys vor und erhält ein persönliches und unterschriebenes „Proxy-Ticket“ zurück.
- Dieses „Proxy-Ticket“ wird jetzt zunächst für weitere Anfragen gespeichert und im zweiten Versuch an den Proxy mitgeschickt.
- Dieser überprüft zuerst die Unterschrift des MNS+-Servers und anschließend ob das Ticket auch tatsächlich zur Nutzung des Dienstes berechtigt. (Lehrer, Schüler, Internet gesperrt?) Danach wird der Auftrag entsprechend der Berechtigung ausgeführt.

Unter Debian sorgen verschiedene „***krb5***-Pakete“ für den reibungslosen Ablauf. Die Konfigurationsdatei ist ***/etc/krb5.conf*** .

5.2.5. Hintergrund: Vernetzung mit Windows (Samba, smb)

„Samba is opening windows to a wider world“, es ermöglicht die Vernetzung mit Windows-Servern, insbesondere mit den Netzwerkfreigaben (Shares) und den Druckern.

Über die persönlichen, Raum- und Tauschlauwerke hinaus werden weitere Shares von MNSplus auch unter Debian verwendet. Neben dem „mounten“ der Netzlaufwerke ist hier der Befehl smbclient sehr hilfreich, der mit verschiedenen Berechtigungen (Schüler-Rechte, Lehrer, Machine-Account, …) ausgeführt wird. Benutzt werden:

- Lokale Benutzer (z.B. Notfallbenutzer) oder WLAN-Zugangsdaten (Kap. 5.3.4)
- Anmeldelog und PCStatistik
- Drucklog (noch nicht unterstützt, kann aber ausgelesen werden, Kap. 5.3.5)
- smb://mnsplusdc/profiles/Debian

  Die Debian-Profile sind ähnlich aufgebaut, wie bei Windows. Sie werden einfach in das gerade frisch erstellte home-Verzeichnis des Benutzers kopiert. Allerdings sind diese ja nur für schulspezifische Änderungen sinnvoll. Deshalb verwende ich sie bevorzugt, um:

  - unerwünschte Programme auszublenden
  - Verknüpfungen zu Webseiten (z.B. elektronisches Klassenbuch) zu hinterlegen
  - Windowsprogramme von smb://mnsplusdc/Programme einzubinden (Kap. 5.3.6)

  In allen Fällen geschieht das durch einen Programmstarter („.desktop“-Textdatei), die unter ***Lehrer/.local/share/applications/*** abgelegt wird (siehe 5.3.6 Integration von Windows-programmen).

  Gemäß der „Desktop Entry Specification“ von freedesktop.org können Sie mit einem solchen Starter aber auch Programme ausblenden lassen (auch Linux-Programme), indem Sie dem Starter eine zusätzliche Zeile hinzufügen:

  - „Hidden=true“ bedeutet, dass das Programm für den speziellen Benutzer komplett gelöscht wird (obwohl es an übergeordneter Stelle weiterhin existiert).
  - „NoDisplay“ bedeutet: Das Programm existiert, wird aber nicht im Menü angezeigt.

  Autostart-Programme realisieren Sie mit einer solchen „.desktop“-Datei im Unterverzeichnis ***~/.config/autostart/***

5.2.6. „Unveränderbare“ Benutzerverzeichnisse

Unter Windows wird bei jeder Anmeldung das vorbereitete „mandatorische Profil“ für Lehrer bzw. Schüler vom MNSplus-Server geladen und beim Abmelden einfach verworfen. Dadurch können lokal durch Schüler (oder Lehrer) keine bleibenden Änderungen vorgenommen werden.

Mandatorische Profile gibt es unter Debian nicht und ich wollte sie auch nicht implementieren. Trotzdem sollen auch hier lokale Änderungen beim Abmelden wieder verworfen werden.
Wenn sich ein Domänenbenutzer anmeldet wird von PAM mit „mkhomedir“ sein home-Verzeichnis lokal neu erstellt. Die Vorlage dazu ist unter /etc/skel auf jedem Computer gespeichert. Hier lassen sich auch allerlei Anpassungen (z.B. Standardeinstellungen für bestimmte Programme, etc.) für alle Benutzer hinterlegen. Darüber hinaus fand ich es sinnvoll, wenn für jede der vier Benutzergruppen (Schüler, Lehrer, Anonym und Klassenarbeit) ein anpassbares Profil (z.B. vorbereitete Konfigurationsdateien) auf dem MNSplusDC bereit stünde, das bei jeder Anmeldung im MNS+-Netz auf die Debian Computer kopiert wird. Hier kann zwar noch einiges vereinfacht werden, fürs Erste finden (und bearbeiten) Sie diese Profile einfach unter: ***smb://mnsplusdc/profiles$/Debian/***

Sollten weitere Anpassungen nötig sein, können diese noch durch die Autostart-Skripte /usr/local/bin/MNSplusCore/gui oder spezieller für den jeweiligen Desktop z.B. /usr/local/bin/MNSplusCore/gnome vorgenommen werden. Siehe hierzu Kapitel „3.1 MNS+-Skripte (Anpassungen für MNS+)“ und die folgenden Hintergrund-Kapitel.

Beim Abmelden, oder (falls dies fehlschlägt) spätestens beim Neustart wird das gesamte home-Verzeichnis einfach gelöscht und bei Bedarf wieder neu angelegt.

5.2.7. Einbinden der Netzwerkshares

Unter Linux gibt es eine Vielzahl von Möglichkeiten, Laufwerke (und auch Netzwerkshares) einzubinden. Technisch gesehen stellt der MNSplus-Server diese als SMB- bzw. CIFS-Freigabe zur Verfügung. Um dies unter Linux zu unterstützen, benötigen wir das Samba-Paket.
Um das Einbinden zu erleichtern gibt es verschiedene „Erleichterungen“:

- gvfs-mount kann schnell und unkompliziert Laufwerke einhängen. Dabei ist es aber abhängig von den Gnome-GVFS-Bibliotheken, die (je nach grafischer Oberfläche) nicht immer vorhanden sind. Außerdem ist es im Terminal schwierig, auf das „Laufwerk“ zuzugreifen.
- PAM-mount ist im Anmeldeprozess verankert und kann auch mit Gruppenzugehörigkeiten der Benutzer (z.B. Lehrer) umgehen. Um auch Klassen-, Kurs- oder AG-

Laufwerke verwenden zu können, müssen diese von einem Skript als „User-Share“ angelegt werden. Dabei gab es jedoch ein Rechte-Problem mit dem Anmeldelog. Außerdem hat pam-mount im Moment keinen Debian-Maintainer, weshalb es nicht sonderlich gut unterstützt wird.

- AutoFS kann auch automatisch Benutzer-Laufwerke einhängen. Ich kenne es nicht.
- mount.cifs nach einigen Problemen mit den verschiedenen „Erleichterungen“ habe ich mich dazu entschieden, die Laufwerke „klassisch“ mit dem Linux-typischen „mount“-Befehl einzubinden. Dies erledigt ein Script (pam-open), das bei der Anmeldung als „root“ ausgeführt wird. Das Aushängen erledigt in der Regel pam_close.

Bei der Verwendung der MNSplus Netzwerkfreigaben gibt es noch zwei weitere Probleme:

1. Timeout: Wird nicht auf das Netzlaufwerk zugegriffen, so wird die Verbindung nach etwa einer viertel Stunde auf „DISCONNECTED“ gesetzt. Kontrollieren kann man dies in der Datei /proc/fs/cifs/DebugData. In der Folge kommt es zu verschiedensten „CIFS VFS: Send error in SessSetup“ Fehlermeldungen.

   Um dies zu vermeiden wird ein Cron-Script alle drei Minuten ausgeführt, das sich um wiederkehrende Reparatur- und Wartungsaufgaben kümmert. Hier werden also auch regelmäßig die bestehenden Verbindungen auffrischt.

2. Aktive Verbindungen: können (und sollten) beim Abmelden nicht abgebrochen werden. Wird z.B. gerade noch eine Datei geschrieben, so bleiben die Netzlaufwerke auch nach der Abmeldung weiter verbunden. Dies führt einerseits wieder zu Timeouts, andererseits zu der Möglichkeit des unberechtigten Datenzugriffs durch Fremde.

   Das Problem wird wieder durch das Cron-Script gelöst, das als root alle drei Minuten die Netzwerkfreigaben überprüft und gegebenenfalls aushängt.

5.3. Wie kann ich …

5.3.1. … einen virtuellen FAI-Server installieren?

Lassen Sie das den Supporter erledigen. Ein virtueller FAI-Server ist nur dann möglich, wenn Sie bereits über eine XEN- oder KVM-Virtualisierung des MNSplus-Servers verfügen. Haben sie diese (noch) nicht, dann verwenden Sie einfach einen Lehrer-Rechner oder einen ausgedienten älteren Computer, den Sie sich z. B. in den Serverschrank stellen..

Falls Sie in Besitz der (root-) Zugangsdaten des XEN- bzw. KVM-Servers sind und einen Zugang als Domänenadministrator zum MNSplusDC haben, dann können Sie sich per „Betrachter für entfernte Bildschirme“ mit dem Protokoll RDP auf der Admin-Machine anmelden und die Installation selbst vornehmen. Dazu müssen Sie dem Virtualisierungs-Server das Abbild der FAI-CD zuerst zur Verfügung stellen. Möglicherweise möchten Sie aber statt dessen auch eine reale DVD brennen und diese vom realen DVD-Laufwerks Ihres Servers in der virtuellen Maschine installieren.

Wenn Sie auf eine grafische Oberfläche verzichten, dann reicht für den FAI-Server eine 20 GB-Festplatte und ein 64-bit Prozessor. Lediglich die „Memory“ Einstellungen sollten bei 2 GB liegen. Dies beschleunigt die Installation bzw. Updates der weiteren Rechner, da so alle nötigen Debian-Pakete im RAM gehalten werden können. Den genauen Wert für Ihre Installation erfahren Sie aus den Dateien fai.log der einzelnen Installationen. Hier wird unter „rx_bytes“ das Download-Volumen für die Installation angezeigt.

Ansonsten wird die virtuelle Maschine „ganz normal“ (über das Citrix XEN-Center oder den virt-manager) eingerichtet und installiert. Verwenden Sie die FAI-ISO-Datei als „CD-ROM“, wählen Sie die im Bootmenü „von DVD/USB installieren“.

Außerdem kam es schon einmal zu Problemen mit der Netzwerkverbindung einer virtuellen Debian-Maschine auf einem XEN-Server. Die konnten behoben werden, indem für diese VM „Offloading“ aktiviert wurde.

5.3.3. … FAI vom Netzwerk (per PXE) booten?

Der FAI-Server unterstützt von Hause aus das Booten per PXE. Um nicht mit der MNS+-Windows-Betankung in Konflikt zu kommen, müssen wir jedoch dem MNS+-Server beibringen, wie und von wo aus das Debian-System installiert werden soll. Dazu benötigen Sie (/der Supporter) die Zugangsdaten eines Domänen-Administrators (nicht: Lehreradmin!). Die folgende Anleitung gilt für den Windows Server 2008 R2, klappt aber möglicherweise auch mit neueren Versionen.

"Eigentlich" würde es reichen, das PXE-Bootmenü des MNSplusDC um einen Eintrag für "Linux" zu ergänzen. Dies ist technisch aber leider nicht möglich. Dafür können wir jedoch den Bootloader PXELINUX auf dem MNSplusDC verwenden und so konfigurieren, das die Abfrage "Windows oder Linux" dem ursprünglichen Bootmenü vorgeschaltet wird.

Damit es (für unsere x64-BIOS-Computer) schneller von der Hand geht, habe ich mit einem Skript bereits (fast) alles vorbereitet, dennoch ist noch etwas Handarbeit nötig:

1. Wir benötigen die vorbereiteten Dateien aus dem Ordner ***/srv/mnsplusdc/*** des FAI-Servers:

   Dazu loggen Sie sich (als Domänen-Administrator) auf dem MNSplus-Server (mnsplusdc) ein, das geht auch aus der Ferne per RDP, z.B. mit dem „Betrachter für entfernte Bildschirme“. Öffnen Sie dort den Windows Datei-Explorer und klicken unter „Netzwerk“ auf den Computer „fai-server“. Hier finden Sie die Netzwerkfreigabe „***PXE-Dateien***“. Markieren Sie den Inhalt dieser Freigabe einfach mit allen Dateien und Unterordnern und kopieren sie in den Ordner ***F:\RemoteInstall\Boot\x64*** des Windows-Servers.

2. Damit PXELINUX auch weiterhin die Windows-Betankung aufrufen kann, brauchen wir noch zwei KOPIEN von Windows-Dateien aus dem Ordner ***F:\RemoteInstall\Boot\x64***

   Kopieren Sie die Dateien ***pxeboot.n12*** und ***abortpxe.exe*** in den eben erstellten Unterordner **./pxelinux.cfg/boot/** und nennen Sie sie: ***pxeboot.0*** bzw. ***abortpxe.0***

   Achtung: Windows blendet standardmäßig die Dateinamenerweiterung aus! Hier hilft es, wenn Sie diese vor dem Umbenennen kurzzeitig anzeigen lassen. Wenn es weiterhin zu Problemen kommt, dann kopieren Sie die Dateien in eines Ihrer Netzlaufwerke und benennen Sie sie von einem Debian-Computer aus um.

3. Nun müssen Sie nur noch dem Windows Deployment Server beibringen, künftig PXELINUX zu laden. Dies tun Sie (als Administrator: cmd.exe) mit den folgenden beiden Befehlen:

- ***wdsutil /set-server /bootprogram:boot\x64\pxelinux.com /architecture:x64***
- ***wdsutil /set-server /N12bootprogram:boot\x64\pxelinux.com /architecture:x64***

Den Erfolg kontrollieren Sie mit: ***wdsutil /get-server /show:config***

Standardstartprogramme:	x64	- boot\x64\pxelinux.com
Standard-N12-Startprogramme:	x86	- boot\x86\pxeboot.n12

Sollten Sie später einmal PXELINUX wieder löschen und die ursprüngliche Konfiguration verwenden wollen, dann kommen Sie hiermit ans Ziel:

- ***wdsutil /set-server /bootprogram:boot\x64\pxeboot.com /architecture:x64***
- ***wdsutil /set-server /N12bootprogram:boot\x64\pxeboot.n12 /architecture:x64***

Achtung:

Es ist normalerweise nicht nötig, das Installationssystem regelmäßig zu aktualisieren. Falls Sie aber den FAI-Server neu installieren oder das Installationssystem manuell aktualisieren (mit ***/usr/local/share/fai-server-init setup***), dann **MÜSSEN(!)** Sie die geänderten Dateien auch auf dem MNSplusDC aktualisieren.
Ansonsten werden Sie bei einer neuen Installation ein unfertiges System (z.B. mit leerer Festplatte oder kritischen Fehlern: „Kernelpanic“) erhalten.
Zum Aktualisieren der Boot-Dateien auf dem MNS+-Server wiederholen Sie einfach (nur) den ersten Schritt der Anleitung. Danach sollte wieder alles funktionieren.

5.3.4. … ein WLAN konfigurieren?

Die meisten WLAN-Adapter sollten problemlos mit Debian funktionieren. Dazu habe ich schon viele Treiber und die benötigte Firmware in die Basiskonfiguration eingepflegt.

MNS+ 3.1 verwendet den ***'ct-WLAN-Kloner*** zum verteilen der MNS+-Netzwerkschlüssel. Diese Konfigurationsdateien lesen auch die Debain-Systeme aus.

Bei späteren MNS+-Versionen kommt RADIUS zum Einsatz. Debian kann damit umgehen, allerdings habe ich mich noch nicht damit beschäftigt und folglich noch nicht eingebaut.

Während der Installation und auch bei FAI softupdates werden die zutreffenden Unterordner (Alle, Raum, …) des Silent$-Shares des MNSplusDC nach *.wlan-Dateien mit Netzwerkkonfigurationen durchsucht. Die dort gefundenen Zugangsdaten werden (bei vorhandener WLAN-Hardware) mittels wpa_supplicant eingerichtet und per systemd_networkd in den Netzwerkstack integriert. Der grafische und flexible Network-Manager bietet zu viele Manipulationsmöglichkeiten (z.B. durch Schüler) und wird daher über ein Skript der Klasse DEFAULT deaktiviert. Außerdem steht die WLAN-Funktionalität so auch schon beim Systemstart (Anmeldung) und für nicht-grafische Arbeitsplätze zur Verfügung.

Bei Problemen mit dem WLAN helfen folgende Befehle, die als root auf dem betroffenen Computer ausgeführt werden müssen:

iw dev listet die am Computer angeschlossenen (und erkannten) WLAN-Adapter auf.

iw dev wlan0 scan | grep SSID zeigt die empfangenen WLAN SSIDs an.

iw dev wlan0 link gibt an, ob der WLAN-Adapter mit einem Netzwerk verbunden ist.

ip addr show zeigt, welche Interfaces aktiv sind und welche IP-Adresse sie haben.

networkctl ist das systemd-Diagnoseprogramm für Netzwerke

/etc/wpa_supplicant/ hier werden die eingerichteten Netzwerke abgelegt.

Scheitert schon die Erkennung des WLAN-Adapters, dann wird Ihnen bei der Problemlösung bzw. Recherche die Ausgabe von ***lspci -vv*** (für fest eingebaute Karten) bzw. ***lsusb -vv*** (für USB-Geräte) weiterhelfen, ebenso wie z.B. https://wiki.debianforum.de/WLAN_Einrichten

5.3.5. … einen Drucker verwenden?

Linux unterstützt das Drucken auf Windows-Netzwerkdruckern über die Programmpakete Samba und CUPS (Common Unix Printing System).

Unter Windows lassen Sie die Installation eines Druckers vom Supporter erledigen. Bei Linux wird die Installation eines Druckers leider nicht leichter. Das liegt daran, das einige Druckerhersteller keinen Support – und damit auch keine Treiber - für Linux anbieten. Andere Hersteller veröffentlichen zwar Treiber für Linux, allerdings unter einer Lizenz, die eine Weiterverteilung seitens Debian verhindern.

Für die meisten Drucker gibt es aber freie Treiber, die automatisch mit CUPS installiert werden.

1. Installieren Sie den gewünschten Drucker zunächst „von Hand“ an einem beliebigen PC. Am einfachsten ist die Druckerinstallation mit dem Browser. Geben Sie dazu ***http://localhost:631/*** in die Adresszeile ein.

   Wichtig ist hier vor allem, welchen Treiber sie verwenden. Drucken Sie eine Testseite, um sicher zu gehen, dass der gewünschte Treiber korrekt ist. Wenn Sie schon etwas Erfahrung haben, können Sie mit dem Befehl ***lpinfo*** ***-m*** verfügbaren Treiber anzeigen lassen und die Liste mit grep oder less schnell durchsuchen.

   Beispiel für einen Treiber: drv:///hpcups.drv/hp-officejet_pro_8100.ppd

2. Zum Abschluss sollen die Drucker für alle Debian-Computer automatisch erkannt und installiert werden. Dies erledigt ein Skript (/usr/local/bin/printer-config-MNS), dass beim Booten und beim Softupdate aufgerufen wird.
   Da die Konfiguration schulspezifisch ist, habe ich eine Subroutine in die Datei ***smb://mnsplusdc/profiles$/Debian/printer.conf*** ausgelagert. Diese Datei müssen Sie selbst anlegen und für jeden Drucker den zu verwendenden Treiber eintragen. Eine Beispielkonfiguration der Subroutine treiber() befindet sich am Ende der Datei /usr/local/bin/printer-config-MNS.

3. Nachdem der Drucker „prinzipiell“ funktioniert, möchten wir nun den Druckerserver des MNSplusDC verwenden. Dadurch wird die Druckersteuerung von MNS+ verwendet und außerdem werden alle Ausdrucke im Drucklog protokolliert.
   TODO: Das funktioniert noch nicht! → Druckerserver des MNSplusDC verwenden!

5.3.6. … ein Windowsprogramm (mit wine) verwenden?

Wine („Wine Is Not an Emulator", eine Windows-kompatible Laufzeitumgebung) ist bereits standardmäßig installiert, ebenso wie WineTricks und PlayOnLinux. Damit lassen sich recht viele (aber bei weitem nicht alle) Windowsprogramme auch unter Linux zum laufen bringen. Im Terminal können Sie dies mit ***wine /pfad/ordner/programm.exe*** ausprobieren.

Unter Windows gibt es im MNS+-Netz zwei Möglichkeiten, wie ein Programm aufgerufen werden kann. Im einfachsten Fall ist das Programm (ohne Installation) im Netzlaufwerk „Programme“ abgelegt und wird einfach über eine Verknüpfung im Startmenü aufgerufen.

Unter Linux müssen Sie dafür einen sogenannten Starter erstellen. Dies ist eine Textdatei mit der Endung .desktop. Die Möglichkeiten und Spezifikationen folgen der „Desktop Entry Specification“: https://specifications.freedesktop.org/desktop-entry-spec/desktop-entry-spec-latest.html

Angenommen, Sie möchten diesen Starter für das Programm „rechentrainer.exe“ erstellen, das im Programme-Ordner des MNSplus-Servers liegt, genauer gesagt im Unterordner /Programme/Medien/Kopfrechnen/ .

- Testen Sie im Terminal, ob das Programm unter wine läuft:
  wine /media/Programme/Medien/Kopfrechnen/rechentrainer.exe
- Erstellen Sie mit einem Texteditor die Datei
  ~/.local/share/applications/kopfrechnenrennen.desktop

```
[Desktop Entry]
Name=Kopfrechentrainer
Comment=Mathematik Kopfrechnentrainer
Keywords=Mathematik;Kopfrechnen;Rechnen;Trainer
Exec=wine "/media/Programme/Medien/Kopfrechnen/rechentrainer.exe"
Path=
Type=Application
Terminal=false
Categories=Education;
StartupNotify=true
Icon=/media/Programme/Medien/Kopfrechnen/rechentrainer.png
X-GNOME-UsesNotifications=true
```

- Sollte das Programm kein separates Icon mitbringen, dann erstellen Sie doch einfach eines, z.B. durch einen Screenshot. Testen Sie anschließend den Starter.
- Wenn dies funktioniert müssen Sie jetzt nur noch dafür sorgen, dass der Starter beim Anmelden in das Benutzerverzeichnis kopiert wird. In Anlehnung an die MNS+-Windowsprofile unterstützt auch MNS+-Debian Benutzerprofile. Rufen Sie (als Anwendungsbetreuer) im Dateimanager die Adresse ***smb://mnsplusdc/profiles$*** auf und erstellen (sofern noch nicht geschehen) den Ordner ***Debian***, mit den Unterordnern ***Anonym***,

Klassenarbeit, ***Lehrer*** und ***Schueler***. Diese werden beim Anmelden in das aktuelle Benutzerverzeichnis kopiert.
Kopieren Sie deshalb den eben erstellten Starter (hier also: ***~/.local/share/applications/kopfrechnenrennen.desktop***) in das gewünschte Profil. Dabei müssen Sie darauf achten, die Ordnerstruktur ebenfalls zu übernehmen. (Cave: Groß- und Kleinschreibung beachten!)

- Es ist müßig, hier alle Konfigurationen aller Windowsprogramme pflegen zu wollen. Deshalb nur eine kleine und unvollständige Auflistung, was bei uns problemlos unter wine läuft:

  Kopfrechentrainer, Mathegrafix,
  SMILE, Oriolus,
  Crocodile Physics,
  MasterTool,
  Lighthouse Englisch-Workbook

TODO: Idee: /opt/wine_drive_c/ für alle Benutzer, chmod ohne write, also: 755

system.reg, userdef.reg, user.reg evtl. über /etc/skel verteilen

dosdevices beim Anmelden anpassen (~~nach wine ping zum Erstellen der Config~~?)

https://wiki.winehq.org/Winecfg

https://wiki.debianforum.de/Wine

https://ubuntuforums.org/showthread.php?t=1516693

https://www.gutefrage.net/frage/linux-wine-installationsverzeichnis-aendern

https://forum.winehq.org/viewtopic.php?t=15329

https://forums.opensuse.org/showthread.php/474275-wine-how-to-enable-multiuser

https://ubuntuforums.org/showthread.php?t=917422

5.3.7. … die vorbereitete dconf-Datenbank ändern?

Gnome und verwandte Desktops (Cinnamon, Mate) speichern ihre Einstellungen in einer eigenen Benutzer-Datenbank unter: ~/.config/dconf/user.
Diese Datenbank wird vom FAI-Server aus in die Ordner /etc/skel/.config/dconf der Clients kopiert und von dort aus bei jeder neuen Anmeldung in das jeweilige Benutzerverzeichnis. Wenn Sie also eine Änderung festlegen wollen, dann nehmen Sie ganz normal (mit dem Programm dconf-editor) Ihre Änderungen vor und kopieren die Datei ~/.config/dconf/user auf den FAI-Server in die Datei: /srv/fai/config/files/etc/skel/.config/dconf/user/MNS

Beachten Sie bitte, dass diese Datei für alle Benutzer verwendet wird. Entfernen Sie deshalb bitte vor dem Kopieren z.B. das benutzerabhängige Hintergrundbild. Diese (und andere) individuelle Einstellungen werden beim Anmelden von einem Skript vorgenommen (z.B. /srv/fai/config/files/usr/local/bin/MNSplusCore/lxde/MNS oder etwa einem Raumskript) .

5.3.8. … eine andere GUI verwenden?

Tun Sie‘s einfach. Unter config/class/60-GUI.sh wird festgelegt, welche GUI (als Klasse) installiert werden soll. Entscheidendes Kriterium hierbei ist die RAM-Größe, ab 1024 MB (zugegeben: recht wenig, besser sind hier 2 GB) wird Gnome installiert, darunter LXDE.

Ändern Sie diese Grenze ab, installieren immer LXDE, oder definieren einfach eine neue Klasse. Mit Cinnamon und Mate habe ich recht gute Erfahrungen gemacht, die Klassen sind vorhanden und (zumindest grob) konfiguriert.

5.3.9. ?

5.4. Fortgeschrittene Themen

In diesem Abschnitt ist nicht für Einsteiger gedacht. Vielmehr möchte ich Ihnen ein paar Ideen vorstellen, die mir im Laufe der Zeit das Leben leichter gemacht haben. Neben Erfahrung auf der Linux Kommandozeile (z.B. mit bash) brauchen Sie auf jeden Fall auch den root-Zugang zum Skolerouter Ihres MNS+-Netzes (oder zum XEN-Server).

An dieser Stelle darf natürlich die obligatorische Warnung nicht fehlen: „**aus großer Macht folgt große Verantwortung**“. Linux gibt Ihnen auch die Freiheit, Ihr System lahmzulegen.

5.4.1. Zugang zum MNS+-Netz aus dem Internet (von zu Hause)

Die Firewall des Routers, über den Ihr MNS+-Netz mit dem Internet verbunden ist, blockiert fast alle eingehenden Verbindungen, lediglich ein paar Ports sind z.B. für Wartungsarbeiten (Supporter) oder den Zugang zur schuleigenen OwnCloud geöffnet.

Zunächst benötigen Sie den „Hostname“, unter dem Ihr MNS+-Netz aus dem Internet erreichbar ist. Das ist der Name unter dem Sie auch Ihre OwnCloud aus dem Internet erreichen, sagen wir: rpl-meine-schule.de.

Als nächstes ist der Port interessant, unter dem der Skolerouter aus dem Internet erreicht werden kann. Sie finden ihn heraus, indem Sie sich z.B. mit einem Laptop oder Tablet mit dem (W)-LAN des Routers verbinden. Loggen Sie sich ein und schauen nach, welcher (externe) Port zum Skolerouter (172.16.0.1) weitergeleitet wird. Als Beispiel: Port 33022

Mit diesen Angaben können Sie sich aus der Ferne auf dem Skolerouter anmelden:

ssh -l root rpl-meine-schule.de -p 33022

Wenn Sie sich ohne Passwort einloggen möchten, erstellen Sie sich einen SSH-Schlüssel (z.B. mit ***ssh-keygen -t rsa -b 4096 -f ~/.ssh/rsa_mns***) und hinterlegen den öffentlichen SSH-Schlüssel mit auf dem Skolerouter: ***ssh-copy-id -i ~/.ssh/rsa_mns root@rpl-meine-schule.de -p 33022***.

Bleibt noch das Problem, das SSH bei fast jedem neuen Verbindungsaufbau nachfragt, ob man dem Server „rpl-meine-schule.de“ (mit IP und Fingerprint) vertraut und die Verbindung hergestellt werden soll. Dies kommt daher das sich die IP des schulischen DSL-Anschlusses (und damit des Skolerouters) täglich ändert. Erstellen Sie deshalb die Datei ***~/.ssh/config*** mit folgenden 3 Zeilen:

Host rpl-meine-schule.de
CheckHostIP no
StrictHostKeyChecking no

5.4.2. Zugang zum FAI-Server aus dem Internet

Im nächsten Schritt können Sie sich vom Skolerouter aus im MNS+-Netz bewegen. Am einfachsten geht das, wenn Sie den Skolerouter als ProxyCommand mit eingeben:
ssh -o ProxyCommand="ssh -l root -W %h:%p rpl-meine-schule.de -p 33022" fai-server -l root

Sie werden sich nicht als root anmelden können, da dies aus Sicherheitsgründen deaktiviert wurde. Loggen Sie sich daher zunächst als MNS+-Benutzer ein, wechseln mit ***su*** zum root-Benutzer und hinterlegen Ihren öffentlichen SSH-Schlüssel wieder in der Datei ***/root/.ssh/authorized_keys***.

Natürlich können Sie auch mit einem Dateimanager (z.B. Nautilus) direkt auf die Dateien des FAI-Servers zugreifen. Dazu wollen Sie wahrscheinlich die SSH-Konfiguration weiter anpassen (siehe ***man ssh_config***).

5.4.3. FAI-Server im Verwaltungsnetz

Meine Anpassung von FAI installiert auch ohne MNS+-Netz Debian auf Computer. Sollte kein FAI-Server erkannt werden, wird die Installation eines solchen angeboten. So können Sie überall einen fertig konfigurierten FAI-Server betreiben, auch im Verwaltungsnetz.

Angenommen, Sie haben einen FAI-Server im MNS+-Netz und einen im Verwaltungsnetz. Es wäre viel einfacher, wenn man den Configspace beider Server an nur einer Stelle pflegen bräuchte. Heute nutze ich dazu die Möglichkeit eines ***GIT-Configspace***s bei der Installation. Die einzelnen FAI-Server holen sich täglich den aktuellen Configspace aus dem GIT-Repository und verteilen ihn dann weiter an ihre Clients. Dank GIT lassen sich auch verschiedene „branches“ für verschiedene FAI-Server verwenden. Auch werden verschiedene Passwörter bzw. ssh-keys und IP-Adressen vor dem Update gesichert und nach der Aktualisierung wieder hergestellt.

Meine vorherige (kleine) Lösung:
Der FAI-Server im Verwaltungsnetz holt sich täglich (z.B. per cron und ssh) den Configspace vom Server im MNS+-Netz und aktualisiert sich damit. Das anschließende initialisieren des FAI-Servers korrigiert die SSH Schlüssel wieder:

- ***scp -o ProxyCommand="ssh -p 33022 root@rpl-meine-schule.de -W %h:%p -i /root/.ssh/rsa_mns" -i /root/.ssh/rsa_mns root@fai-server:/srv/fai/mns.tar.gz /tmp***
- ***rm -rf /srv/fai/config/* ; cd /srv/fai/config/; tar -x -f /tmp/mns.tar.gz***
- ***/usr/local/share/fai-server-init setup***
- ***fai softupdate***

5.4.4. Zugang zum FAI-Server im Verwaltungsnetz

Wie bereits beschrieben möchte ich das das Verwaltungsnetz nicht aus dem Internet erreichbar ist. Trotzdem wäre es schön, wenn zumindest der FAI-Server überwacht bzw. repariert werden könnte.

Hier hilft ein „reverse SSH Tunnel“. Das ist eine SSH-Verbindung, die vom FAI-Server des Verwaltungsnetzes aus aufgebaut wird und in umgekehrter Richtung benutzt werden kann. Außerdem hift das Programm ***autossh***, das bei Verbindungsabbrüchen dafür sorgt, das automatisch eine neue Verbindung aufgebaut wird.

- Reverse Tunnel vom Verwaltungsnetz FAI-Server aus (zum Skolerouter) aufbauen. (Am Besten in der Datei /etc/rc.local eintragen). Dadurch wird der FAI-Server des Verwaltungsnetzes vom Skolerouter aus erreichbar:
  autossh -N root@rpl-meine-schule.de -p 33022 -R 33044:127.0.0.1:22 -i ~/.ssh/rsa_mns &
- Nun können Sie den Skolerouter wieder als Proxy verwenden, diesmal als Zugang zum Verwaltungsnetz. Die IP Adresse 127.0.0.1 darf nicht geändert werden, sie gibt an, das das Ende des SSH-Tunnels lokal (auf dem Skolerouter) erreicht werden kann.:
  ssh -o ProxyCommand="ssh -l root -W %h:%p rpl-meine-schule.de -p 33022" 127.0.0.1 -l root -p 33044

Falls Sie z.B. für die Vertretungsplanmonitore VNC eingerichtet haben, können Sie natürlich auch den VNC-Port (5900) zum Skolerouter tunneln:

- ***autossh -N root@rpl-meine-schule.de -p 33022 -R 33590:127.0.0.1:5900 -i ~/.ssh/rsa_mns***
- Um den Bildschirm nun über das Internet betrachten zu können, tragen Sie im VNC Viewer (z.B. für Gnome: Vinagre, „Betrachter für entfernte Bildschirme“) als Rechner (VNC-Server) ein: ***127.0.0.1:33590*** und als SSH-Tunnel ***root@rpl-meine-schule.de:33022***

Die Arbeit mit einem VNC-Tunnel macht natürlich nur dann Sinn, wenn Sie auch über eine entsprechend schnelle Internetverbindung verfügen. Die Bilddaten müssen insgesamt dreimal durch die DSL-Leitung (raus aus dem Verwaltungsnetz, rein ins MNS+-Netz und wieder raus aus dem MNS+-Netz) und sind trotz Kompression immer noch größer als ein reiner SSH Datenstrom.

Bitte beachten Sie außerdem, dass Sie für jede Verbindungsart einen neuen Zielport auf dem Skolerouter verwenden müssen.

5.4.5. Edoo.sys RLP Server unter Debian installieren

Da bei uns der Edoo.sys Server mittlerweile extern gehostet wird (mit Zugriff per VPN), sind die kommenden beiden Kapitel eigentlich hinfällig geworden und nur noch als Information enthalten. Interessant (und aktuell) ist aber weiterhin die Installation eines Edoo.sys Clients. Siehe Kapitel 5.4.7.

Edoo.sys ist eine Java Anwendung, die unter Windows ebenso läuft wie unter OS X oder Linux. Die Installation ist im edoo.sys Handbuch auch für Linux beschrieben und auch die Installationsskripte werden regulär für Linux mitgeliefert.

Zunächst habe ich im Verwaltungsnetz (mit der FAI-CD) einen eigenen Rechner als künftigen edoo.sys-Server installiert. Ansonsten reicht auch eine reguläre Debianinstallation. Da edoo.sys auch als Server eine graphische Oberfläche benötigt, installieren Sie bitte eine. Sollten Sie den edoo.sys Server ohne Bildschirm und Tastatur betreiben wollen ist evtl. auch noch ein SSH-Server oder auch eine Zugriffsmöglichkeit mittels VNC sinnvoll.

1. Datenbank installieren.

   Edoo.sys baut auf der Datenbank postgresql auf. Sie brauchen diese nicht umständlich herunterladen, sondern installieren sie einfach (als root) aus den Debian Repositories:
   apt-get install postgresql

2. Edoo.sys installieren.

   Spätestens hier brauchen Sie die graphische Oberfläche. Melden Sie sich als Benutzer an, speichern und entpacken Sie die edoo.sys Installationsdateien z.B. unter ~/Downloads und öffnen ein Terminal:

   cd ~/Downloads/edoosys; su; chmod +x ./install.sh; ./install.sh

3. Benutzer zum Ausführen des Programms anlegen und Rechte zuweisen:

   adduser --home /opt/edoosys/server --no-create-home --disabled-password –disabled-login --gecos "" --no-create-home svpdss
   # adduser svpdss
   chmod 700 /opt/edoosys/
   chown -R svpdss:users /opt/edoosys/
   chmod +x /opt/edoosys/server/edoosys-server

4. Nun sorgen wir dafür, dass die Datenbank und der edoo.sys Server automatisch gestartet werden. Erstellen (oder ergänzen) Sie die Datei ***/etc/rc.local*** :

#!/bin/bash

service postgresql start

su - svpdss -c "cd /opt/edoosys/server; /opt/edoosys/server/edoosys-server &"

und rufen Sie sie auf:

chmod +x /etc/rc.local; /etc/rc.local

Fertig. Ihr edoo.sys Server ist nun im Verwaltungsnetz erreichbar. Tragen Sie bei der Installation von Clients im Verwaltungsnetz nun die IP Adresse des Servers und den Port 8765 ein.

5. Zu guter Letzt fehlte mir noch eine automatische Datensicherung für den Katastrophenfall. Zwar wird der Datenbestand ja (gelegentlich) mit dem Landesserver repliziert, eine lokale Sicherung der ASV-Datenbank auf einer zweiten (internen) physischen Festplatte ist aber immer sinnvoll. Bei mir tat es folgender crontab-Eintrag:

   00 04,10,16 * * * su -l postgres -c "/usr/bin/pg_dump -Fc asv | gzip" > /backup/edoosys-asvDB-$(date +\%Y-\%m-\%d_\%H:\%M).backup.gz

Siehe auch:

http://www.zdnet.de/39136063/backups-und-wiederherstellung-von-postgresql-datenbanken/2/?inf_by=54f0e88c5937dc0c06463f69

https://stackoverflow.com/questions/12720967/how-to-change-postgresql-user-password

6. Falls Sie auch noch eine UPS verwenden wollen, helfen Ihnen folgende Hinweise:

   https://wiki.debian.org/apcupsd

   http://www.seismo.ethz.ch/static/linux/apc_usv.html

   apt-get install apcupsd

   apctest

   apcaccess status
   service apcupsd start

5.4.6. Edoo.sys RLP Server über das Internet erreichen

Ist es sinnvoll, dass der edoo.sys RLP Server aus dem Internet erreicht werden kann?

Mit Sicherheit nicht. Daher möchte ich Ihnen aus Datenschutzgründen und mit Blick auf die rechtliche Verantwortung bzw. Konsequenzen bei Angriffen dringend davon abraten. Um einen Server oder Dienst sicher im Internet anbieten zu können, braucht es weit mehr Wissen und Können, als dies durch einen „Schul-Administrator“ geleistet werden kann. Überlassen Sie das Ihrem Supporter (oder jemand anderem, der die Risiken im Internet einschätzen kann und im Zweifel haftbar zu machen ist).

Wie auch immer. In Ausnahmefällen kann es hilfreich sein, kurzfristig über das Internet auf den edoo.sys Server zugreifen zu können. Die folgende Lösung ist zwar in vielerlei Hinsicht nicht optimal, dafür ist sie aber durch Verschlüsselung und obligatorischem Zugang zum Skolerouter akzeptabel sicher. Wenn Sie auf Ihrem Client Computer nur Windows installiert haben, brauchen Sie zusätzlich ein (kostenloses) SSH Programm für Windows, z.B. PuTTY:

1. Richten Sie vom edoo.sys Server aus einen root-Zugriff zum Skolerouter ein. (Siehe Abschnitt 5.4.1: SSH Schlüssel erstellen und dem Skolerouter übergeben)

2. Tunneln Sie den Port 8765 des edoo.sys Servers im Verwaltungsnetz zum Skolerouter Ihres MNS+-Netzes. Dies geht am einfachsten mit einem „reverse SSH Tunnel“:
   ssh -N root@rpl-meine-schule.de -p 33022 -R 48765:127.0.0.1:8765 -i /root/.ssh/rsa_mns

3. Jetzt brauchen Sie nur noch den Datenverkehr auf Ihren (sicheren!) PC weiterzuleiten. Bauen Sie dazu von einem beliebigen Internetanschluss aus ebenfalls einen SSH-Tunnel zum Skolerouter auf:
   ssh -N root@rpl-meine-schule.de -p 33022 -L 8765:127.0.0.1:48765 -i /root/.ssh/rsa_mns

4. Durch die beiden SSH-Tunnel haben Sie sich – bildlich gesprochen – ein verschlüsseltes Verlängerungskabel von Ihrem edoo.sys Server zu Ihrem PC verlegt.
   Wenn Sie nun den edoo.sys RLP Client auf Ihrem PC installieren, lassen Sie die Adressangaben für den DSS Server unverändert auf ***localhost:8765*** stehen, denn dort liegt der Eingang zu Ihrem SSH Tunnel (bildlich: die Steckdose, an die der edoo.sys Client angeschlossen wird).

Der SSH Tunnel wird (konstruktionsbedingt absichtlich) gelegentlich abbrechen. Mit ***autossh*** ließe sich das zwar vermeiden, doch die Lösung soll ja „eigentlich“ auch nur kurzfristig funktionieren. … Bei einem Verbindungsabbruch überprüfen Sie Ihre beiden SSH-Tunnel und starten Sie sie neu.

5.4.7. Edoo.sys RLP Client unter Debian installieren

Da es leider keinen „offiziellen“ Hersteller-Support für die Installation unter Linux gibt, möchte ich hier eine kleine Anleitung dokumentieren, bei Problemen mit der Installation hilft Ihnen aber trotzdem auch das SVP-RLP TopDesk Helpcenter.:

Bei uns läuft die Noteneingabe vor den Zeugnissen über zwei nur für diesen Zweck vorgesehenen Rechner mit einem Benutzeraccount „Lehrer“. Deshalb habe ich den edoo.sys-Client auch nur mit Benutzerrechten installiert. Sie benötigen dazu einen Debian-Computer im Verwaltungsnetz (z.B. mit der FAI-CD installiert) und die offizielle Installationsdatei von edoo.sys.

1. Entpacken Sie die Installationsdatei in einem beliebigen Ordner (z.B. /home/lehrer/)
2. Öffnen Sie ein Terminalfenster starten den Installer:

   cd /home/lehrer

   chmod +x install.sh

   ./install.sh

3. Achtung: Der edoo.sys installer fragt Sie nach einem Installationspfad für das Programm. Geben Sie hier bitte /***home/lehrer/edoosys/client*** ein, da es ansonsten beim Starten von edoo.sys zu Problemen mit der Pfadangabe kommt.
   Achten Sie auch auf die korrekte Angabe der Server-IP und des Ports.
4. Machen Sie das Programm ausführbar:

   chmod +x /home/lehrer/edoosys/client/edoosys-client

5. Leider ist der grafische Starter für edoosys auch fehlerhaft. Bitte löschen Sie den mitgelieferten Starter und erstellen einen neuen:
   rm /home/lehrer/.local/share/applicantions/edoosys*.desktop
   Erstellen Sie die Datei /home/lehrer/.local/share/applications/edoosys.desktop:

   ```
   [Desktop Entry]
   Name=edoo.sys
   Comment=Startet edoo.sys RLP
   Keywords=edoosys;edoo.sys;SVP;Schulverwaltung;
   Exec=/home/lehrer/edoosys/client/edoosys-client
   Path=/home/lehrer/edoosys/client
   Icon=/home/lehrer/edoosys/client/icon.xpm
   Terminal=false
   Type=Application
   ```

5.4.8. Eine KVM-Virtualisierungs-Umgebung installieren

Zur Virtualisierung der Server-Infrastruktur stehen unter Debian zwei verschiedene Möglichkeiten zur Verfügung: Xen und KVM. Xen war der Vorreiter, doch KVM hat ihm mittlerweile den Rang abgelaufen. Andere Lösungen (wie xenserver.org, Proxmox, Turnkey o.a.) setzen meist ebenfalls auf Xen oder KVM auf.
Hinter KVM steht der „Enterprise-Linux“-Konzern Red Hat, wo es auch eine hervorragende Dokumentation zur Virtualisierung mit KVM gibt: *https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/6/html/virtualization_administration_guide/*

Die FAI-Klasse KVM installiert ein Basis-KVM-Wirtssystem (ohne GUI) und wird definiert, wenn der Hostname mit „KVM“ beginnt. Als Supporter haben Sie höchst wahrscheinlich andere Vorstellungen, Bedürfnisse oder Gegebenheiten als ich. Daher möchte ich hier die Einrichtung meines privaten Servers skizzieren. Fühlen Sie sich frei, die Konfiguration abzuändern, bevor Sie die Virtualisierungs-Gäste hinzufügen.

- RAID

  Auch wenn mein Server über einen „echten“ Hardware RAID-Controller verfügt, habe ich mich nach der Lektüre von *https://skrypuch.com/raid/* dazu entschieden, meine insgesamt 4 Festplatten mit einem Linux-Software-RAID (mit mdadm) zu verwenden. Den Hauptvorteil sehe ich darin, dass das Betriebssystem die einzelnen Festplatten selbst und direkt verwenden kann – inklusive Fehlerdiagnose und möglicher Änderungen.
  Das KVM Wirtssystem „wohnt“ auf einem 10 GB großen RAID-1 der ersten beiden Festplatten (also: md0 auf sda1 und sdb1).
  Der Rest dieser beiden Platten wird wieder zu einem RAID-1 (md1 auf sda3 und sdb3) zusammen geschaltet, ebenso die beiden verbleibenden Festplatten (md2 auf sdc1 und sdd1).

- LVM

  Auf den beiden RAID-Verbünden (md1 und md2) setzt der „Logical Volume Manager“ auf, der diese „Physical Voumes“ zu einer „Volume Group“ zusammenschließt und sie in flexiblere (aber auch abstraktere) „Logical Volumes“ aufsplitet. Diese beherbergen (später) die verschiedenen virtuellen Maschinen („Gäste“). Dazu werden bei der Installation schon ein paar „Logical Volumes“ angelegt.
  Sowohl Festplatten, als auch RAID, LVM und Logical Volumes werden durch die Datei /var/lib/fai/config/*disk_config/KVM* festgelegt. Sie können diese Datei einfach bei der Installation an Ihre Bedürfnisse anpassen (Alt-Strg-F2).

- Konfiguration des Wirtssystems

  Der Host braucht keine GUI und auch nur sehr wenige Softwarepakete, die von FAI (bzw. der Klasse KVM) gleich mit installiert (*package_config/KVM*) und eingerichtet (*scripts/KVM/**) werden.

  Zunächst wird der Benutzer „user“ berechtigt, den Wirt (mittels libvirt) zu verwalten.

  Danach wird das LVM als ***Storagepool*** für die virtuellen Maschinen eingerichtet. Außerdem werden beim ersten Systemstart beide ***Netzwerkkarten*** mit systemd-networkd als ***Bridge*** eingerichtet, auf die auch die virtuellen Maschinen zugreifen können.

  Damit aus der Ferne schnell Systeminformationen abgerufen werden können (etwa ob eine RAID-Festplatte „degraded“ ist), habe ich das Programm „***Cockpit***“ installiert, das auf Port 9090 von einem Webbrowser aus aufgerufen werden kann.

  Im Fall des degradierten RAID-Verbundes lässt sich dieser auch direkt im Browser unter dem Punkt „Terminal“ reparieren. Hier können Sie direkt Befehle an das System absetzen, etwa diese: https://www.thomas-krenn.com/en/wiki/Mdadm_recover_degraded_Array

Manuelle Anpassungen:

Als Supporter können Sie natürlich noch eigene Skripte in die FAI Installation einbauen (etwa andere Benutzer / Passwörter, SSH-Schlüssel, announce-Scripte uvm., ja Sie könnten sogar das Anlegen von virtuellen Maschinen automatisieren). Schauen Sie sich dazu dazu auf dem FAI-Server den Ordner /srv/fai/config/scripts/KVM/ an und ergänzen, was Sie benötigen.

Um komfortabel mit dem KVM-Host arbeiten zu können, sollten Sie aber zumindest auf Ihrem Arbeits-PC / in Ihrer Admin-Maschine:

- einen eigenen SSH-Schlüssel erstellen (falls noch nicht vorhanden):

  ssh-keygen -t dsa -C "Mein_KVM-Key" -f ~/.ssh/kvm_key

- diesen SSH-Schlüssel auf dem KVM-Wirt installieren:

  ssh-copy-id user@<IP-Adresse-des-KVM-Wirts> -i ~/.ssh/kvm_key

  CAVE: für den Benutzer „root“ funktioniert das nicht, das Passwort-Login ist deaktiviert.

- eventuell möchten Sie Ihren KVM-Host später auch noch so einrichten, dass er sich bei einem noch zu installierenden FAI-Server regelmäßig aktualisiert.

  Dies können Sie durch Anpassungen in der Datei ***/etc/fai/fai.conf*** (FAI_CONFIG_SRC, LOGUSER und FAI_LOGPROTO) und einen Eintrag in ***/etc/crontab*** erreichen.

  Beachten Sie auch, dass der KVM-Host dadurch nicht automatisch zu einem CLIENT wird und die Passwörter der Benutzer demzufolge nicht geändert werden.

Reinstallieren des KVM-Hosts:

Dadurch, dass das System des KVM-Wirts in einem anderen RAID-Verbund als die virtuellen Maschinen liegt, lässt sich dieses gegebenenfalls neu installieren, ohne die virtuellen Festplatten der Gäste zu zerstören.

Die Installationsprozedur versucht (class/90-KVM-data.sh), auf einem eventuell bestehenden RAID-Verbund

- eine frühere FAI-Installation zu erkennen (anhand der Logs der Festplattenkonfiguration),
- die Partitionierung sowie das LVM zu übernehmen und auch
- das Konfigurationsverzeichnis /etc/libvirt zu kopieren,
- ebenso wie frühere „authorized_keys“ des Benutzers „user“.

Im Erfolgsfall wird (statt aller Platten) nur der erste RAID-Verbund neu installiert, die virtuellen Maschinen (auf dem LVM auf weiteren RAID-Verbünden) sollten erhalten bleiben. Diese Prozedur ist noch nicht wirklich ausgiebig getestet und Sie sollten sich daher der damit verbundenen Gefahren bewusst sein und z.B. durch ein Backup Vorsorge treffen.

Laut KVM-Dokumentation ist die sauberste Lösung, im alten System die Einstellungen (Storagepools, Netzwerke, virtuelle Maschinen) als xml-Datei zu exportieren, extern zu speichern und nach erfolgreicher Installation wieder zu importieren.

Dokumentation zu KVM und libvirt:

- https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/6/html/virtualization_administration_guide/
- https://wiki.debian.org/KVM
- https://wiki.debian.org/libvirt
- https://wiki.hetzner.de/index.php/KVM_mit_libvirt
- https://wiki.libvirt.org/page/Main_Page

5.4.9. Den KVM-Host einrichten und verwalten

Nachdem das Wirtssystem installiert ist, müssen Sie es anpassen und weiter einrichten. Wir wollen den Server „headless“ betreiben können, also müssen Sie zunächst per SSH ein paar Kleinigkeiten einrichten:

Auf der KVM Maschine ist der Benutzer „user“ (Default-Passwort: Debian) bereits zur Verwaltung der virtuellen Maschinen konfiguriert. Damit Sie auch ohne Passwort (und dem nicht installierten Programm ssh-askpass) damit arbeiten können sollten Sie sich zunächst auf der KVM-Maschine Ihren SSH Schlüssel einrichten. Dies ist im vorherigen Abschnitt unter „Manuelle Anpassungen“ beschrieben, kurz: Sie müssen den öffentlichen SSH-Schlüssel in der Datei *~/.ssh/authorized_keys* hinterlegen. Ändern Sie bei dieser Gelegenheit auch gleich das Passwort des Benutzers (*passwd*). Möglicherweise möchten Sie als Supporter auch noch gleich ein paar Cronjobs, Wartungsscripte (z.B. announce.sh) oder ähnliches einrichten.

Der Rest geschieht grafisch. Unter Debian ist der ebenfalls von Red Hat stammende **virt-manager** („**Virtual Machine Manager**“) das Programm der Wahl um die virtuellen Maschinen einzurichten und zu verwalten. Dieser übernimmt die Kommunikation mit libvirt und erspart so die Arbeit auf der Kommandozeile mit virsh.

Zuerst müssen sie von Ihrer Administrations-Maschine eine Verbindung zum Wirtssystem herstellen:

Verbindung hinzufügen -> „entfernter Rechner“.
Der Hypervisor ist QEMU/KVM, verbinden Sie sich per SSH als Benutzer user mit dem Host.

Kontrollieren Sie zunächst die Netzwerke (2 Bridges: bridge_extern und bridge_MNSplus), sowie die beiden Storagepools (Speicher). Hier sollte neben dem „iso_storage“ auch die LVM-Datenträgergruppe „vg_guests“ erscheinen. Ebenfalls sollte im Storagepool „iso_storage“ bereits eine Datei fai.iso liegen (oder gerade heruntergeladen werden) – zum Installieren von Gästen.

Für normale ***Systemwartungsaufgaben*** habe ich auf dem KVM-Server das Programm ***Cockpit*** (ebenfalls von RedHat) installiert. Sie erreichen es von einem Webbrowser aus auf Port 9090, also z.B. https://kvm-server:9090. Hier haben Sie einen Überblick über Datenträger, Statusmeldungen, Fehlerberichte, ...

Wenn Sie eine neue ***virtuelle Maschine erstellen***, dann wählen Sie die vorhandene ISO-Datei aus und weisen der Maschine ein Logical Volume des LVM-Verbundes zu. Ansonsten achten Sie einfach darauf, welches „Netzwerk“ Sie verwenden. Falls Sie für Datenträger oder Netzwerkkarten einen Treiber auswählen können: Die beste Performance bietet der empfohlene ***virtio-Treiber***.