

Steven Wend und Jan Bredereke*

Automatisierte Auslieferung und Konfiguration einer Linux-Umgebung für Entwickler

DOI 10.1515/pik-2015-0001

Zusammenfassung: Dieser Beitrag beschreibt die Zusammenstellung einer Linux-Dualboot-Infrastruktur für Entwickler in einem mittelständischen Unternehmen, dessen Fokus die Softwareentwicklung ist. Die Entwickler benötigen dabei eine Flexibilität, die von den bekannten Enterprise-Linuxen für Desktop-Umgebungen nicht geboten wird. Dabei laufen Auslieferung und Konfiguration des Linux-Betriebssystems weitestgehend automatisiert ab. Inhalt des Beitrags ist insbesondere eine aktuelle Evaluation von Linux-Distributionen, Konfigurationsmanagement- und Auslieferungswerkzeugen, die dem Leser hilft, eine für seine eigene Situation geeignete Lösung zu finden. Für das betrachtete Unternehmen bestand die mit Erfolg eingesetzte Lösung aus einer Kombination von Ubuntu LTS, FAI und Puppet.

1 Einleitung

Dieser Beitrag beschreibt die Zusammenstellung einer Linux-Dualboot-Infrastruktur für Entwickler in einem mittelständischen Unternehmen, dessen Fokus die Softwareentwicklung ist. Die Entwickler benötigen dabei eine Flexibilität, die von den bekannten Enterprise-Linuxen für Desktop-Umgebungen nicht geboten wird. Die Softwareentwickler müssen auf ihrem Rechner möglichst genau die Versionen der Anwendungsdienste haben, die ihre Kunden verwenden, um ihre Produkte sinnvoll testen zu können. Die Stärken der Enterprise-Linuxe liegen dagegen bei einheitlichen Softwareversionsständen und stabilen, aber auch starren Softwareverteilungsmechanismen. Dies ist sinnvoll z. B. für Verwaltungsabteilungen, in denen vorwiegend mit einer Office-Suite, einem Web-Browser und gegebenenfalls einigen Fachanwendungen gearbeitet wird.

Zunächst betrachten wir die grundlegenden Randbedingungen in dem betrachteten Unternehmen (Kap. 2). In diesem Kontext stellen wir dann Kriterien zusammen und bewerten entlang dieser Kriterien Linux-Distributionen

(Kap. 3) sowie Auslieferungs- und Konfigurationsmanagementwerkzeuge (Kap. 4). Dies soll einem Leser helfen, eine für seine eigene Situation geeignete Lösung zu finden. Am Ende des Beitrags berichten wir auch kurz über die Lösung für das betrachtete Unternehmen (Kap. 5).

2 Relevante Randbedingungen im Unternehmen

Eine Dualboot-Lösung mit einem Linux- und einem Windows-Betriebssystem wurde aus den folgenden Gründen gewählt. Im Unternehmen bestand eine bereits vorhandene, auf Windows basierende Umgebung, welche zur Entwicklung genutzt wird. Es ist nicht geplant, diese zu ersetzen. Parallel dazu wurde eine Linux-Umgebung gewünscht. Viele Kundensysteme basieren auf Linux, da dort Server wie Apache, Wildfly und Tomcat laufen. Abhängig vom Betriebssystem existieren Unterschiede, die bei einer Entwicklung unter Windows und einem Betrieb unter Linux Probleme bereiten können, so zum Beispiel die Zeilenendemarkierung mit CRLF bzw. LF oder auf Applikationsebene eine unterschiedliche Konfigurationsumgebung. Eine Plattformgleichheit kann so die Produktqualität und das Wissen von Entwicklern im Bereich Linux verbessern.

Lösungen mit einer virtuellen Linux-Maschine auf einem Windows-System werden zwar auch genutzt. Sie fordern jedoch den gleichen Verwaltungsaufwand bei der Konfiguration und Wartung wie native Systeme, sind aber durch die Typ-2-Hypervisoren weniger performant. Zusätzlich fallen Lizenzkosten für die Virtualisierungssoftware an. So ist der Hypervisor von VM-Ware kostenpflichtig, und bei VirtualBox wird spätestens beim „Expansion Pack“ um eine Spende gebeten.

Voraussetzungen für jede Lösung sind der Erhalt der bestehenden Windows-Umgebung und ein Anschließen der Linux-Umgebung an die dortigen Dienste. Beispiele für solche Dienste sind die Authentifizierung mit Microsoft Active Directory [Mor+12] via LDAP (Lightweight Directory Access Protocol) und eine Inventarisierung aller Software mit OCSI (Open Computers and Software Inventory) [Ant10]. Die Partitionierung der Windows-Installationen,

Steven Wend: E-Mail: mail@stevenwend.de

* Kontaktperson: Jan Bredereke:

E-Mail: jan.bredereke@hs-bremen.de

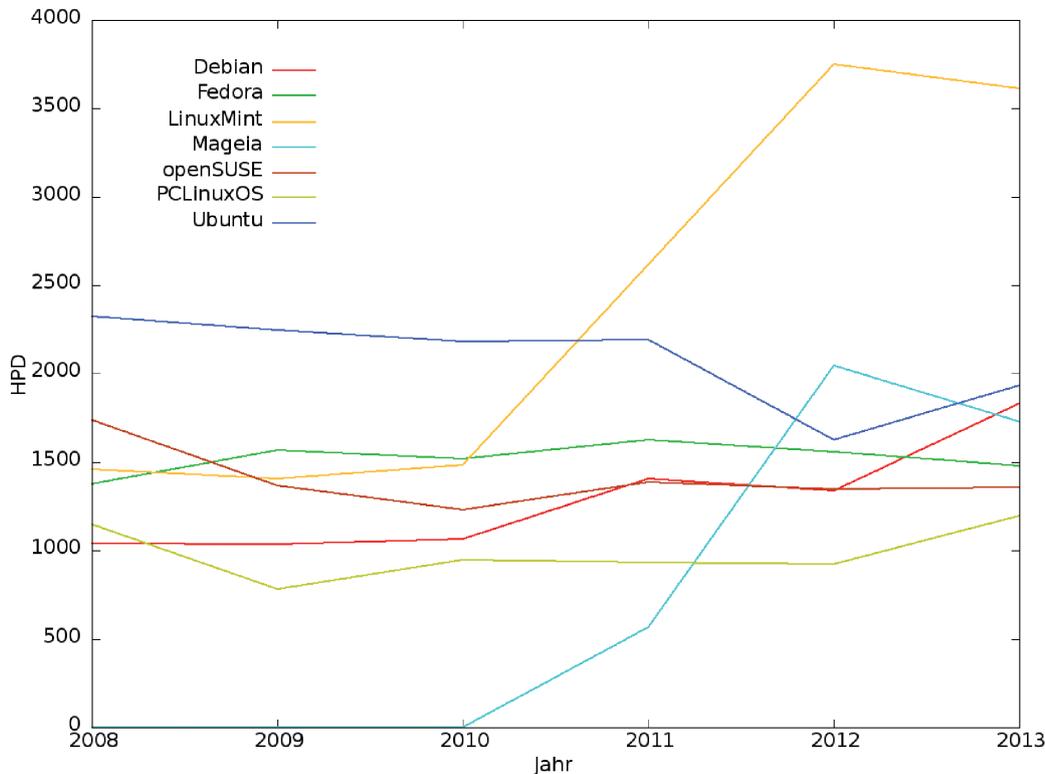


Abbildung 1: HPD der Top-5-Distributionen auf DistroWatch.com von 2008 bis 2013.

welche unternehmensweit standardisiert ist, darf nicht derart verändert werden, dass die Windows-Umgebung Schaden erleidet. Eine geeignete Prüfung während des Änderns der Partitionierung ist daher notwendig. Die Konfiguration des Bootloaders muss beide Systeme zum Betrieb befähigen.

3 Bewertung von Linux-Distributionen

3.1 Vorauswahl

Es gibt hunderte von Linux-Distributionen, so dass eine Vorauswahl nötig ist. Viele dieser Distributionen sind für ganz spezielle Zwecke gedacht. Die folgenden zwei Kriterien schränken den Kreis der Kandidaten bereits auf wenige ein:

Großer, stabiler Benutzerkreis: Es ist zu erwarten, dass die Distribution weiter gepflegt werden wird. Und es gibt eine Community, die bei Problemen Unterstützung leisten kann.

Langzeitunterstützung von mindestens drei Jahren: Sicherheitsupdates werden entsprechend lange ver-

fügar sein, so dass die Personalkosten für einen Umstieg auf eine aktuelle Version nur selten anfallen werden.

Daten über die Größe der Benutzerkreise von Linux-Distributionen sammelt die Internetplattform DistroWatch.com [Dis14]. Sie misst die Zugriffe auf ihre eigenen Beschreibungen der Distributionen, wobei sie pro IP-Adresse maximal einen Treffer pro Tag zählt (HPD, „hits per day“). Damit bietet sie einen Anhaltspunkt für das Interesse an den Distributionen, aber natürlich nicht für den Umfang des Einsatzes oder gar für die Qualität der Distributionen. Trotzdem sollte diese Art der Messung die verbreitetsten Distributionen sicher identifizieren, was für eine Vorauswahl ausreichend ist. Darüber hinaus lässt die Entwicklung des Interesses über die Jahre Trends erkennen, die zur Beurteilung der Stabilität des Benutzerkreises einer Distribution wichtig sind. Die Rohdaten haben wir für die verfügbaren zurückliegenden Jahre abgerufen und zu der Statistik in Abb. 1 verdichtet.

Bei der Auswertung wurden nur diejenigen Distributionen berücksichtigt, die in einem der Jahre unter den ersten fünf Plätzen waren. Auch für die eigene Auswahl sollten lediglich höchstens fünf der Distributionen in die Vorauswahl aufgenommen werden.

Die Distributionen in der ersten Stufe der Vorauswahl waren Debian, Fedora, Linux Mint, openSUSE und Ubuntu.

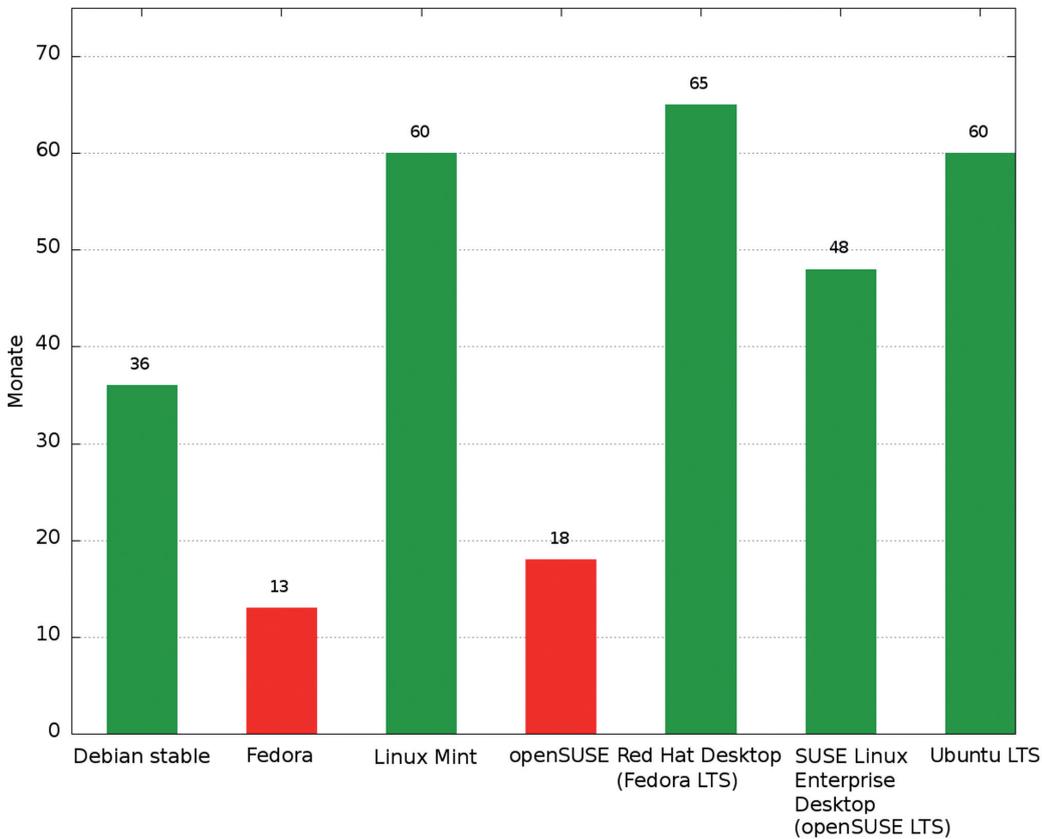


Abbildung 2: Länge des Supports der Distributionen mit großem, stabilem Benutzerkreis und von deren LTS-Versionen.

Sie konnten entweder eine stabile Kurve ohne erhebliche Ausreißer vorweisen oder erfuhren eine deutlich positive Steigung im Laufe der Jahre. Für Mageia konnte noch keine Prognose gemacht werden, da diese Distribution erst im Jahre 2011 erstmalig erschien.

Je länger der Support-Zeitraum einer Distribution ist, desto flexibler kann ein Unternehmen entscheiden, wann ein Versionssprung durchgeführt wird. Die Grenze für den minimalen Zeitraum wurde auf 36 Monate festgelegt, dem vorgegebenen Richtwert im Unternehmen. Abbildung 2 zeigt die Länge des Supports im Vergleich.¹ Dabei haben wir gegebenenfalls vorhandene Versionen von Distributionen mit Langzeitunterstützung (LTS, long term support) zusätzlich aufgenommen.

Folgende Linux-Distributionen sind damit in der zweiten Stufe der Vorauswahl: Debian stable, Linux Mint, Red Hat Desktop (Fedora LTS), SUSE Linux Enterprise Desktop (openSUSE LTS) und Ubuntu LTS. Weitere Informationen zu diesen Distributionen finden sich z. B. jeweils in den

Einzelbeschreibungen [Döll14; Dub14; Lee14a; Lee14b; Die14].

3.2 Relevante Kriterien

Dieser Abschnitt präsentiert Kriterien, die für die weitere Auswahl hilfreich sind, und er bewertet anhand ihrer die nach der Vorauswahl verbliebenen Linux-Distributionen. Zu Beginn der Arbeiten wurde eine große Zahl von Anforderungen zusammengetragen. Es zeigte sich allerdings, dass etliche davon für alle der Kandidaten erfüllt sind. Hier folgen daher nur Kriterien, die etwas zu einer Auswahlentscheidung beitragen können. Wenn ein Kandidat bei einem Kriterium nicht erwähnt wird, gab es jeweils nichts Negatives anzumerken. Leser mit einer anderen Ausgangslage können die Einzelbewertungen hier für eine eigene Abwägung nutzen.

Ein wichtiges Kriterium war, dass die Distribution einen *großen, etablierten Nutzerkreis* hat. Damit kann man erwarten, dass die Unterstützung für einen langen Zeitraum gesichert ist. Aufgrund der Vorauswahl ist dies für alle verbliebenen Distributionen der Fall, außer für die beiden Enterprise-Linexe Red Hat Desktop (Fedora LTS)

¹ Ausgewählte openSUSE-Versionen werden neuerdings vom Evergreen-Team der Community nach Support-Ende weiter mit Sicherheitsupdates versorgt. Dauer und Erfolg des Supports sind noch nicht abzusehen.

und SUSE Linux Enterprise Desktop (openSUSE LTS). Die zugehörigen Community-Linuxe Fedora und openSUSE erfüllen natürlich dieses Kriterium. Aber die Enterprise-Varianten haben nur einen vergleichsweise kleinen Nutzerkreis und hängen von der Unternehmenspolitik ab.

Die *Abhängigkeit von einem Herstellerunternehmen* kann man als Nachteil oder als Vorteil werten. Im konkreten Falle sprach gegen die Enterprise-Linuxe vor allem die bereits in der Einleitung diskutierte zu geringe Flexibilität der Enterprise-Linuxe für ein mittelständisches Unternehmen, dessen Fokus die Softwareentwicklung ist. Dies führte letztlich zu deren Ausschluss.

Falls die *Kosten* relevant sind, stehen die Enterprise-Linuxe ebenfalls schlechter da als die anderen Linuxe. Dafür erhält man zwar professionellen Support, aber auch der Support einer großen Community ist oft sehr gut.

Ein weiteres wichtiges Kriterium war entsprechend, ob die *Distribution gut dokumentiert* ist. Dabei sollte auch eine *passende Version* der Dokumentation für die jeweilige Version der Distribution vorhanden sein. Dies war für Debian nicht erfüllt, auch wenn die Debian-Referenz auf der offiziellen Internetpräsenz [Aok13] ansonsten auffallend gut strukturiert war. Die Debian-Referenz hatte den Fokus auf der aktuellen stabilen Version, ohne jedoch explizit Dokumentation für ältere Versionen der Distribution bereitzustellen. Bei Linux Mint war das offizielle Benutzerhandbuch [Mint14] nicht sehr umfangreich. Weiteres konnte man nur der soliden Dokumentation von Ubuntu entnehmen, das technisch die Grundlage für Linux Mint ist.

Das letzte entscheidende Kriterium war, dass die Konfiguration *durch ein Konfigurationsmanagementwerkzeug verwaltet* werden kann. Hier wurde für die sechs verbreiteten Werkzeuge (siehe Kap. 4 unten) Bcfg2, CFEngine, Chef, FAI, m23 und Puppet jeweils geprüft, ob sie genau die jeweilige Distribution unterstützten. Dies ist wichtig, weil sich die Distributionen insbesondere bei der Konfiguration unterscheiden. Linux Mint fiel hier ab, weil es von keinem der Werkzeuge offiziell unterstützt wurde. Lediglich m23 unterstützt (inzwischen) offiziell Linux Mint. Allerdings basiert Linux Mint (in der hier betrachteten Version) technisch stark auf Ubuntu, das unterstützt wird, so dass es wohl nur bei den wenigen Linux Mint-spezifischen Dingen Probleme geben könnte. Zu Bcfg2 macht die offizielle Internetpräsenz lediglich die Aussage, dass alle Systeme auf POSIX-Basis unterstützt werden [Bcf14b]. Spezifische Aussagen zu Distributionen in Verbindung mit deren Versionen finden sich dort nicht. Leichte Abstriche bei der Unterstützung durch die Konfigurationsmanagementwerkzeuge gab es für die beiden Enterprise-Linuxe. Zum Teil wurden von den genannten Werkzeugen nur die Community-Versionen oder andere Enterprise-Versionen

unterstützt. Allerdings bringen beide Distributionen auch eigene Konfigurationsmanagementwerkzeuge mit.

Bei den folgenden beiden Kriterien gab es keine großen Unterschiede zwischen den Distributionen mehr, sondern nur noch leichte.

Ein weiteres Kriterium war, dass die Programme einer Distribution durch offizielle Pakete *regelmäßig auf den aktuellen Funktionsstand gebracht* werden. Hier fiel Debian gegenüber den anderen Distributionen etwas ab. Für eine Produktionsumgebung empfiehlt Debian die jeweilige „Stable“-Version, die bereits besonders lange getestet wurde, auch wenn viele Nutzer mit der Stabilität der jeweiligen „Testing“-Version durchaus zufrieden sind. Zuerst bekommt eine neue Version das Attribut „Unstable“. Im Takt von etwa zwei Jahren reift eine Version um eine Stufe. Durch diesen Prozess fehlen den Programmen der „Stable“-Version die Funktionsverbesserungen der letzten Zeit.

Schließlich war nötig, dass die Distribution durch das Hard- und Software-Inventarisierungswerkzeug *Open Computers and Software Inventory (OCSI)* [Ant10] überwacht werden konnte. Auf der offiziellen Internetpräsenz von OCSI [Ocs14] gab es keine Angaben dazu, welche Versionen der dort angegebenen Distributionen unterstützt wurden, weshalb das jeweilige Vorhandensein von OCSI in den offiziellen Repositories der Distributionen geprüft wurde. Die beiden Enterprise-Linuxe haben OCSI nicht in ihren offiziellen Repositories. Allerdings kann man OCSI selbst aus dem Quellcode kompilieren, und für Red Hat Desktop kann man es auch aus dem inoffiziellen, freien EPEL-Repository (Extra Packages for Enterprise Linux) bekommen.

Bei der konkreten Entscheidung im Unternehmen fielen zuerst die beiden Enterprise-Linuxe aus den oben beschriebenen Gründen heraus. Linux Mint basiert (in der hier betrachteten Version) technisch stark auf Ubuntu, fiel aber wie gerade diskutiert bei mehreren der hier relevanten Kriterien deutlich gegen Ubuntu LTS ab. Ubuntu LTS basiert technisch wiederum auf Debian. Im Vergleich mit Debian Stable sprachen für Ubuntu LTS der aktuellere Softwarestand und die noch bessere Dokumentation. Daher wurde im konkreten Falle Ubuntu LTS gewählt.

4 Bewertung von Auslieferungs- und Konfigurationsmanagementwerkzeugen

Auslieferungs- und Konfigurationsmanagementwerkzeuge dienen dazu, eine vorkonfigurierte Arbeitsumgebung möglichst automatisch auf eine große Zahl von Arbeitsrechnern zu bringen und dabei nötige Anpassungen ebenfalls

Tabelle 1: Rangfolge von Open-Source-Konfigurationsmanagementwerkzeugen im Linux Journal.

Puppet	andere	Chef	OpenQRM	Spacewalk	CFEngine	RHN Satellite	Bcfg2
42,5%	15,7%	8,9%	8,9%	8,7%	8,1%	4%	3,2%

automatisch nach vordefinierten Regeln vorzunehmen. Es gibt Werkzeuge, die beide Aspekte abdecken, aber auch spezialisierte Werkzeuge für nur jeweils einen der Aspekte. Daher werden beide zunächst zusammen betrachtet.

Die gewählte Linux-Distribution Ubuntu LTS bietet bereits die zwei Auslieferungswerkzeuge Kickstart und Preseedings, die deshalb in die Vorauswahl aufgenommen werden. Allerdings sind sie weniger für ein komplexes Konfigurieren gedacht, so dass weitere Werkzeuge über diese Bordmittel hinaus im betrachteten Falle nötig sind.

4.1 Vorauswahl von Konfigurationsmanagementwerkzeugen

Nachdem Auslieferungswerkzeuge bereits grundsätzlich zur Verfügung stehen, sind noch Kandidaten für ein Konfigurationsmanagementwerkzeug nötig. Einige Werkzeuge beherrschen dabei sogar beide Aspekte, so dass sie daher auch in beiderlei Hinsicht bewertet werden sollen.

Die Auswahl von Kandidaten begann wiederum mit der Suche nach den meistgenutzten Konfigurationsmanagementwerkzeugen. So detaillierte Informationen wie bei den Linux-Distributionen standen hier leider nicht zur Verfügung.

Delaet et al. [DJB10] berücksichtigen in ihrer Untersuchung von Konfigurationsmanagementwerkzeugen die folgenden Open-Source-Werkzeuge: Bcfg2, CFEngine 3, Chef, Netomata, Puppet und Lcfg. Dabei fallen zwei allerdings gleich für die Zwecke hier wieder heraus: Netomata zielt auf Netzwerkgeräte und -dienste und scheint außerdem nach kurzer Lebensdauer bereits im Jahre 2012 wieder eingeschlafen zu sein. Lcfg (Local ConFiGuration system) unterstützt nur die Linux-Distributionen Fedora Core und Scientific Linux [LCF07].

Weiterhin veröffentlichte das Linux Journal im Dezember 2012 einer Leserumfrage u. a. zum besten Open-Source-Konfigurationsmanagementwerkzeug. Dabei ergab sich die Rangfolge in Tabelle 1 [Pow12].²

² Seit Durchführung der Auswahl ist im Linux Journal eine weitere Leserumfrage erschienen [Pow13, S. 74]. Die Rangfolge dort war: 38,8% apt, 21,1% Puppet, 9,2% Subversion, 11,6% Yast, 6,6% SaltStack, 9,6% andere (beliebt dabei: Ansible, Chef und git), 3,1% CFEngine. (Der Autor S. Powers vermutete hinter dem unerwarteten

Dies erweiterte potentiell die Liste der Kandidaten. Allerdings fielen alle zusätzlichen Kandidaten für die Zwecke hier gleich wieder heraus. OpenQRM ist für die Verwaltung von Cloud-Infrastrukturen gedacht [Mei+11, S. 61f.]. RHN Satellite gehört zur Enterprise-Umgebung von Red Hat [Lan11, S. 4], und Spacewalk ist die Community-Upstream-Version von RHN Satellite [Spa14].

Damit verblieben bei den Konfigurationsmanagementwerkzeugen, von zwei Seiten bestätigt, als Kandidaten Bcfg2 [Bcf14a], CFEngine [CFE14], Chef [Che14] und Puppet [Kru+13; Pup14].

4.2 Vorauswahl von kombinierten Werkzeugen

Im Bereich der kombinierten Auslieferungs- und Konfigurationsmanagementwerkzeuge identifizierte eine Studie über existierende Lösungen [ALP08] FAI (Fully Automatic Installation) [LG11], m23 [Goo14] und Lcfg, die daher hier in Betracht gezogen wurden. Sowohl FAI als auch m23 unterstützten Ubuntu 14.04 LTS. Lcfg unterstützte leider nicht Ubuntu, siehe Abschnitt 4.1 oben, daher fiel es aus der Vorauswahl gleich wieder heraus.

4.3 Relevante Kriterien

Analog zur Bewertung der Distributionen oben präsentiert auch dieser Abschnitt Kriterien, die für die weitere Auswahl von Auslieferungs- und Konfigurationsmanagementwerkzeugen hilfreich sind, und er bewertet anhand ihrer die Werkzeuge in der Vorauswahl. Wieder werden hier nur Kriterien angeführt, die etwas zu einer Auswahlentscheidung beitragen können, und wenn ein Kandidat bei einem Kriterium nicht erwähnt wird, gab es jeweils nichts Negatives anzumerken. Die Einzelbewertungen können ggf. wieder bei einer anderen Ausgangslage für eine eigene Abwägung genutzt werden. Die Werkzeuge in Tabelle 2 werden in der dort genannten Hinsicht untersucht.

Gewinner apt dabei den Wunsch, auf ein Konfigurationsmanagementwerkzeug ganz zu verzichten.) Diese neue Kandidatenliste konnte für die hier beschriebenen Arbeiten nicht mehr berücksichtigt werden.

Tabelle 2

	Bcfg2	Chef	CFEngine	FAI	Kickstart	m23	Preseedings	Puppet
Auslieferung				X	X	X	X	
Konfig.-Mgmt.	X	X	X	X		X		X

4.3.1 Kriterien für Auslieferungswerkzeuge

Bei den Auslieferungswerkzeugen gab es nur wenige wesentliche Unterschiede. Ein wichtiges Kriterium war, dass die *Anpassungen der schon bestehenden Umgebung möglichst klein* sein sollten. Hier passte nur m23 nicht gut. Der m23-Server stellt selbst verschiedene Dienste zur Verfügung, unter anderem DHCP (Dynamic Host Configuration Protocol) und TFTP (Trivial File Transfer Protocol). Dies kollidierte mit den bereits laufenden Servern des Unternehmens, und eine m23-Serverinstallation ohne diese Dienste war zwar erwähnt, jedoch nicht dokumentiert. Hier wäre bei m23 also einige Handarbeit nötig.

Ein weiteres wichtiges Kriterium speziell im betrachteten Unternehmen war, dass der *Server des Auslieferungswerkzeugs auch unter der Linux-Distribution SUSE Linux Enterprise Server (SLES)* läuft, die aktuell für mehrere Server genutzt wird. Hier passten Kickstart und Preseedings nicht gut, da sie von SLES nicht unterstützt wurden.

Bei den folgenden Kriterien gab es keine großen Unterschiede zwischen den Werkzeugen mehr, sondern nur noch leichte.

Eine *bestehende Partitionierung sollte verändert werden können*. Preseedings konnte dies zwar nicht selbst [Can14, Kap. B.4.6], aber man konnte Partitionierungsrezepte für das Partitionierungsprogramm partman verwenden [Can14, Kap. B.5.1].

Eine *umfangreiche und gepflegte Dokumentation für die jeweilige Version* des Werkzeugs sollte verfügbar sein. Hier fehlte bei FAI [LG11] und m23 [Goo14] jeweils die Differenzierung nach den Versionen in der ansonsten wie bei den anderen Werkzeugen auch gut strukturierten und umfassenden Dokumentation.

Damit das Werkzeug langfristig weiter gepflegt wird, sollte es nicht nur von einer Gemeinschaft getragen werden, sondern auch *durch ein Unternehmen gepflegt* werden. Dies war zwar für alle Werkzeuge der Fall. Aber bei m23 schien das Unternehmen im wesentlichen von einer Person getragen zu sein. Und Kickstart wird zwar von dem großen Unternehmen Red Hat für deren Linux-Distributionen gepflegt, aber hier kommt es darauf an, dass das Unternehmen Canonical hinter der Unterstützung der Version für Ubuntu steht. Die Dokumentation der Ubuntu-Version verweist allerdings auf die von Red Hat und beschreibt selbst

nur Basisfunktionen. Hinzu kam ein Katalog an Funktionen, die unter Ubuntu nicht nutzbar sind [Can14, Kap. 4.6.2], sowie eine Reihe von Erweiterungen seitens Canonical.

In Hinsicht auf die Bewertung für das konkrete Unternehmen bleibt zusammenfassend festzuhalten, dass FAI [LG11] bei den Auslieferungswerkzeugen das einzige war, das alle Kriterien gut erfüllte.

4.3.2 Kriterien für Konfigurationsmanagementwerkzeuge

Bei den Konfigurationsmanagementwerkzeugen erfüllte längst nicht jedes Werkzeug alle Kriterien, die für das Unternehmen relevant waren. Auch hier sollten die Anpassungen der schon bestehenden Umgebung möglichst klein sein, weshalb m23, wie schon oben in Abschnitt 4.3.1 beschrieben, nicht gut passte.

Die *Anzahl der konfigurierbaren Clients* sollte *mindestens 1000* betragen, um auch langfristig genug Raum für ein Wachstum des Unternehmens zu haben. Für Bcfg2 machte die offizielle Internetpräsenz keine Aussage, und in der Literatur fand sich ein Feldrechercheergebnis, dass weniger als 1000 Clients unterstützt würden [DJB10, Kap. 3.2.1]. Die Anzahl der von m23 konfigurierbaren Clients fand sich weder in der offiziellen Dokumentation noch in Veröffentlichungen.

Sämtliche Kommunikation beim Konfigurieren sollte durch *Verschlüsselung und gegenseitige Authentifizierung* gesichert sein. Bei FAI fand sich dazu nichts in der Dokumentation, außer dass die Übertragung der Logfiles bei der Auslieferung per Ssh gesichert werden kann. Nach der Auslieferung kann man natürlich die Kommunikation von Hand z. B. per Ssh und Rsync über Ssh oder evtl. per Ssh und NFS 4 mit Kerberos absichern. Bei m23 machte die Dokumentation [Goo14] nur im Kontext von Backups Aussagen zu Verschlüsselung und Authentifizierung, sonst nicht.

Das Werkzeug sollte einen „*Trockenlauf*“ eines Konfigurierlaufs erlauben. FAI erlaubte dies nicht. Es können aber Tests geschrieben werden, die einen Lauf nach Abschluss prüfen. Auch für m23 fand sich im offiziellen Handbuch [Goo14] kein Trockenlauf.

Das Werkzeug sollte *widersprüchliche Konfigurationen erkennen* können, bei denen z. B. eine erste Regel eine Netzwerkkarte einschaltet und eine zweite Regel sie ausschaltet. FAI erlaubte dies nicht. Dies ergab sich schon daraus, dass FAI die Flexibilität von mehr als einer Skriptsprache bietet, mit der zugehörigen logischen Komplexität. Auch für m23 fand sich hierzu nichts in der Dokumentation [Goo14].

Das Werkzeug sollte per *LDAP* Anmeldeinformationen und weitere Informationen aus dem Unternehmens-Active-Directory abrufen können. Bei Bcfg2 war dies nur über ein Plugin möglich, das aber noch experimentell war. Bei FAI fand sich dazu nichts in der Dokumentation. Bei Chef und CFEngine ist diese Funktion der kommerziellen Version vorbehalten. Anmeldeinformationen kann m23 laut Dokumentation per LDAP nutzen; ob auch weitergehend z. B. Druckerinformationen, blieb unklar.

Bei den folgenden Kriterien gab es keine großen Unterschiede zwischen den Werkzeugen mehr, sondern nur noch leichte.

Eine *umfangreiche und gepflegte Dokumentation für die jeweilige Version* des Werkzeugs sollte verfügbar sein. Bei CFEngine fehlte die Differenzierung nach der kommerziellen oder kostenlosen Version, und es wurde nur grob in Version 2 und 3 unterschieden. Bei Bcfg2 und m23 fehlte jeweils die Differenzierung nach Versionen ganz.

Der *Status der Clients* sollte *überwacht* werden können. Bei Chef musste man hier ein externes Programm nehmen, z. B. Nagios.

Damit das Konfigurationsmanagementwerkzeug langfristig weiter gepflegt wird, sollte es nicht nur von einer Gemeinschaft getragen werden, sondern auch *durch ein Unternehmen gepflegt* werden, genau wie das Auslieferungswerkzeug, s.o. Hier gab es nur bei m23 wieder dieselben leichten Bedenken wie oben in Abschnitt 4.3.1.

Die erstellten *Konfigurationen* sollten *als Textdatei* gespeichert werden, damit sie von einem Versionsverwaltungssystem gut handhabbar sind. Bei m23 ist dies zwar wohl der Fall, aber es blieb in der Dokumentation [Goo14] offen, wie ein laufender m23-Server auf eine externe Veränderung dieser Dateien durch ein Versionsverwaltungssystem reagiert.

In Hinsicht auf die Bewertung für das konkrete Unternehmen bleibt zusammenfassend festzuhalten, dass nur Puppet [Kru+13; Pup14], Chef [Che14] und CFEngine [CFE14] alle Kriterien gut erfüllten. Chef und CFEngine erlaubten LDAP allerdings nur in der kommerziellen Version, Chef benötigte ein externes Programm, um den Status der Clients zu überwachen, und bei CFEngine war die Differenzierung der Dokumentation nach konkreten Versionen nicht so wie erhofft, s.o.

Ein kombiniertes Auslieferungs- und Konfigurationsmanagementwerkzeug bot sich nach der Evaluierung für den konkreten Zweck nicht an. FAI war zwar das einzige Auslieferungswerkzeug, das alle Kriterien gut erfüllte, aber es passte nicht gut als Konfigurationsmanagementwerkzeug. Daher fiel die Wahl auf FAI als Auslieferungswerkzeug und Puppet als Konfigurationsmanagementwerkzeug. Weitere Informationen zu diesen beiden Werkzeugen finden sich in der zugehörigen Dokumentation [LG11] und [Kru+13; Pup14].

5 Ergebnis und Erfahrungen

Für das betrachtete Unternehmen bestand die mit Erfolg eingesetzte Lösung aus einer Kombination der Linux-Distribution Ubuntu LTS [Die14], des Auslieferungswerkzeugs Fully Automatic Installation (FAI) [LG11] und des Konfigurationsmanagementwerkzeugs Puppet [Kru+13; Pup14]. Für Leser mit einer anderen als der beschriebenen Ausgangslage sind die Kriterien und Einzelbewertungen in den vorangegangenen Abschnitten hoffentlich für eine eigene Abwägung nützlich.

Vorab, bei der grundsätzlichen Entscheidung für eine zusätzliche Betriebssystemumgebung, sollte berücksichtigt werden, dass zwar die beschriebene Software kostenfrei erhältlich und effizient bedienbar ist, dass aber natürlich trotzdem mit nennenswertem zusätzlichem Support-Aufwand für die Endnutzer zu rechnen ist, einfach, weil ein Betriebssystem mehr zu unterstützen ist.

Im betrachteten Unternehmen änderten sich während des Projekts die Anforderungen insofern, dass die Dualboot-Fähigkeit vorläufig nicht mehr verlangt war, sondern dass für die an Linux interessierten Entwickler ein vollständiger Umstieg auf Linux reichte.

Literatur

- ALP08 Ivan Augé, Vincent Leligeour und Olivier Pons. *A Distribution and System Compiler for Handling Heterogeneous Computer Site*. Techn. Ber. CEDRIC-08-1549. Paris: CEDRIC-CNAM, 1. Jan. 2008. Kap. 2.3. URL: <http://cedric.cnam.fr/fichiers/RC1549.pdf>.
- Ant10 Barzan „Tony“ Antal. *IT Inventory and Resource Management with OCS Inventory NG 1.02*. Packt Publishing, 13. Mai 2010.
- Aok13 Osamu Aoki. *Debian Referenz*. Version 2. 3. Dez. 2013. URL: <https://www.debian.org/doc/manuals/debian-reference/index.de.html> (besucht am 17. 9. 2014).
- Bcfg14a *Bcfg2 Documentation*. url: <http://docs.bcfg2.org/> (besucht am 18. 9. 2014).

- Bcf14b *Bcfg2 Documentation – OS Support*. 2014. URL: <http://docs.bcfg2.org/introduction/os-support.html> (besucht am 22. 9. 2014).
- Can14 *Ubuntu Installation Guide*. Version 14.04 LTS, i386. Canonical. URL: <https://help.ubuntu.com/lts/installation-guide/i386/index.html> (besucht am 19. 9. 2014).
- CFE14 *CFEngine Guide*. URL: <https://docs.cfengine.com/latest/guide.html> (besucht am 18. 9. 2014).
- Che14 *An Overview of Chef*. URL: http://docs.getchef.com/chef_overview.html (besucht am 18. 9. 2014).
- Die14 Oliver Diedrich. „Ubuntu 14.04 LTS“. In: c't (19 23. Aug. 2014), S. 124–125.
- Dis14 *Distrowatch.com*. Unsigned Integer Ltd., Hong Kong. URL: www.distrowatch.com (besucht am 4. 5. 2014).
- DJB10 Thomas Delaet, Wouter Joosen und Bart Van Brabant. „A Survey of System Configuration Tools“. In: *Proc. 24th Large Installation System Administration Conference, LISA 2010*. (San Jose, CA, USA). Hrsg. von Rudi van Drunen. USENIX Association, 7.–12. Nov. 2010. URL: <https://www.usenix.org/conference/lisa10/survey-system-configuration-tools>.
- Döl14 Mirco Dölle. „Debian GNU Linux 7.6 Wheezy“. In: c't (19 23. Aug. 2014), S. 126–127.
- Dub14 Liane M. Dubowy. „Linux Mint 17 Qiana“. In: c't (19 23. Aug. 2014), S. 134.
- Goo14 Hauke Goos-Habermann. *m23-Benutzerhandbuch*. Version rock 14.2. URL: <http://m23.sourceforge.net/docs/manual/html/de/manual.html> (besucht am 17. 9. 2014).
- Kru+13 Spencer Krum u. a. *Pro Puppet*. 2. Aufl. Apress, 9. Dez. 2013.
- Lan11 R. Landmann. *Red Hat Network Satellite 5.5 Benutzerhandbuch*. 2. Aufl. Red Hat, Inc., 2011.
- LCF07 *LCFG: FAQ*. 6. Nov. 2007. URL: <http://www.lcfg.org/faq/> (besucht am 18. 9. 2014).
- Lee14a Thorsten Leemhuis. „Fedora 20“. In: c't (19 23. Aug. 2014), S. 130–131.
- Lee14b Thorsten Leemhuis. „OpenSuse 13.1“. In: c't (19 23. Aug. 2014), S. 128–129.
- LG11 Thomas Lange und Henning Glawe. *FAI Guide (Fully Automatic Installation)*. Version 3.0.6. 21. Juli 2011.
- Mei+11 Christoph Meinel u. a. *Virtualisierung und Cloud Computing: Konzepte, Technologiestudie, Marktübersicht*. Techn. Ber. 44. Universitätsverband Potsdam, 2011.
- Mint14 *Official User Guide Linux Mint Cinnamon Edition*. URL: http://www.linuxmint.com/documentation/user-guide/Cinnamon/english_17.0.pdf (besucht am 17. 9. 2014).
- Mor+12 Rand Morimoto u. a. *Windows Server 2012 Unleashed*. Sams, 13. Sep. 2012.
- Ocs14 *OCS Inventory NG*. URL: <http://www.ocsinventory-ng.org/en/> (besucht am 26. 9. 2014).
- Pow12 Shawn Powers. „Readers' Choice Awards 2012“. In: *Linux Journal* (224 Dez. 2012), S. 70ff.
- Pow13 Shawn Powers. „Readers' Choice Awards 2013“. In: *Linux Journal* (236 Dez. 2013), S. 56–81.
- Pup14 *Puppet 3.7 Reference Manual*. Puppet Labs. URL: <https://docs.puppetlabs.com/puppet/3.7/reference/> (besucht am 18. 9. 2014).
- Spa14 *What's the difference between Satellite and Spacewalk?* Spacewalk Project. 2014. URL: <http://www.spacewalkproject.org/faq.html#compare> (besucht am 18. 9. 2014).



Steven Wend: Hochschule Bremen –
Fak. 4, Flughafenallee 10, 28199 Bremen



Jan Bredereke: Hochschule Bremen –
Fak. 4, Flughafenallee 10, 28199 Bremen