Durchbruch beim Textverstehen

Künstliche Intelligenz

iX SPECIAL 2019

30 JAHRE iX

IT heute

Strategische Linien der Informationstechnik

Agil und modular

Moderne Softwareentwicklung

Jeder ist hackbar

IT sicherer machen

Smarter produzieren

Industrie 4.0

Zwischen Datenschutz und Kommerz

Die Zukunft der sozialen Netze

Auf Streife im Netz

Kampf gegen Cyberkriminalität

Lightning: Bitcoin light

Kryptowährung für den Alltag

Rechner der Zukunft?

Quantencomputer

Warum die Digitalisierung nicht vorankommt

Meta-Digitalisierung

Verteilt, virtualisiert, automatisiert

Trends bei Server, Storage, Netzwerk

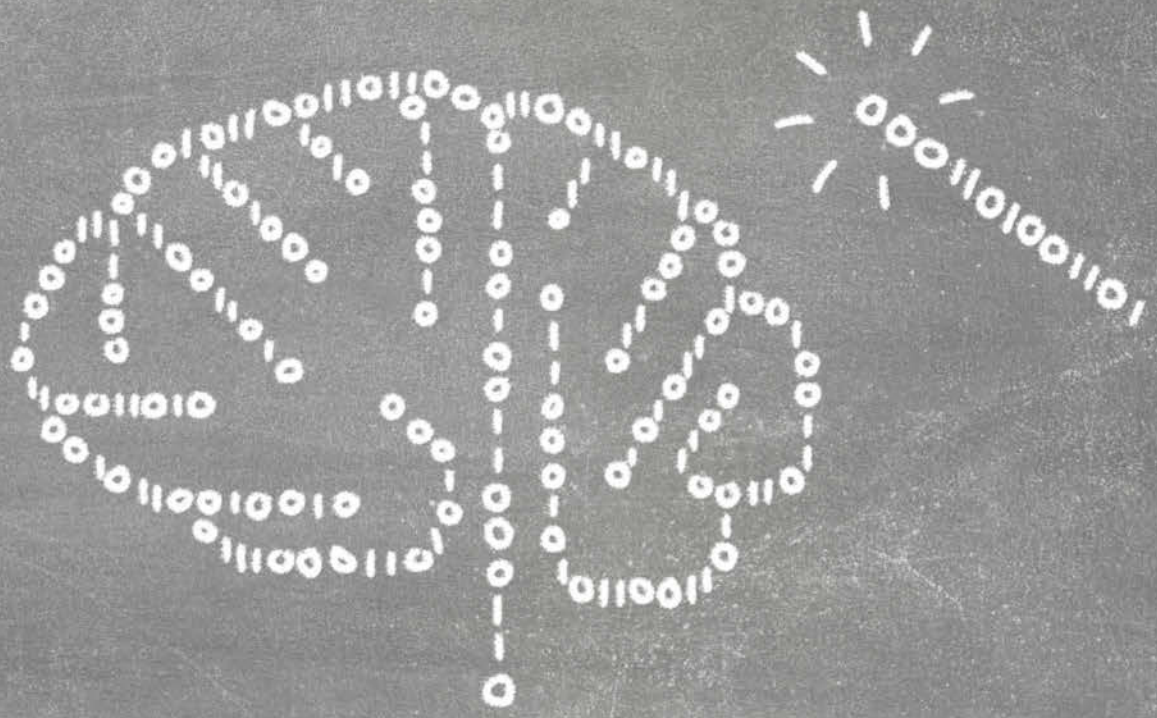
Mit Preisausschreiben

Hosting-Pakete, Drohnen, WLAN-Mesh-Sets u. v. m. zu gewinnen

www.ix.de

Lufthansa
Industry Solutions

Sie sind **Artificial Intelligence Spezialist (m/w/divers)** und erwecken gern Dinge zum Leben?



Artificial Intelligence hat das Potenzial, Unternehmensprozesse und Geschäftsmodelle radikal zu verändern. Als IT-Beratungsunternehmen unterstützen wir unsere Kunden beim sinnvollen Einsatz künstlicher Intelligenz und treiben Prozessoptimierungen voran.

Kommen Sie an Bord: **www.lhind.de/ki-jobs**

LHIND.de

LUFTHANSA GROUP

Be who you want to be
Be-**Lufthansa**.com

IT heute

Informationstechnik nimmt eine immer wichtigere Rolle in allen Lebensbereichen ein: in den Unternehmen, die von der Digitalisierung getrieben sind; in Gesellschaft und Politik, wo die Meinungsbildung zunehmend in sozialen Netzen stattfindet; und in unserem Alltag, der ohne Smartphone und Internet kaum noch vorstellbar ist. Hinter all dem steckt eine gewaltige IT-Infrastruktur aus Rechenzentren, globalen Netzen und immer komplexerer Software, ohne die in der modernen Welt nicht mehr viel funktionieren würde. Haben Sie mal darüber nachgedacht, was alles an Technik nötig ist, damit Sie mit einem Wisch das Netflix-Video von Ihrem Handy auf den Fernseher schicken können?

Mit diesem Heft möchten wir Ihnen einen Überblick geben, welche Themen die IT-Welt derzeit bewegen. Dabei schlagen wir einen großen Bogen: von den aktuellen technischen Entwicklungen in Rechenzentren über Trends bei Softwareentwicklung und IT-Sicherheit bis hin zu gesellschaftlichen Fragen. Wir schauen in die Vergangenheit, wenn der Blick zurück hilft, die heutige Situation einzuordnen; und wo wir uns trauen, wagen wir auch einen Blick in die Zukunft.

Entsprechend finden Sie auf den folgenden Seiten ein breites Spektrum an Themen. Das geht vom Stand der Technik in den Bereichen Server, Storage und Netzwerk über aktuelle Themen wie Edge Computing und Microservices bis zu den Versprechen (und der Realität) der RZ-Automatisierung. Wir analysieren, wie agile Prinzipien die Unternehmenskultur verändern, wo die künstliche Intelligenz derzeit die größten Fortschritte macht und wie weit Quantencomputer noch von der Praxis entfernt sind.

Die dank erhöhter Datenschutzansprüche immer wichtigere Kryptografie ist ebenso Thema wie aktuelle Entwicklungen bei der IT-Sicherheit und im IT-Recht. Mit Progressive Web Apps und Functions as a Service haben wichtige Trends der Softwareentwicklung ihren Platz im Heft gefunden. Wir schauen uns an, wie Facebook und Co. gegen Fake News vorgehen wollen, und Gunter Dueck, Ex-CTO von IBM, sinniert darüber, warum sich so viele Unternehmen so schwer tun mit der Digitalisierung.

Zum Schluss noch ein paar Worte in eigener Sache: *iX* feiert dieses Jahr 30. Geburtstag. Aus diesem Anlass hat Susanne Nolte ein Bilderrätsel gebaut, in dem Sie einen von 30 Preisen gewinnen können (Seite 150). Und Jürgen Seeger, Chefredakteur seit der ersten Ausgabe, erzählt ab Seite 154 einige bemerkenswerte Ereignisse aus der *iX*-Historie – und stellt die Redaktion vor.

Ich wünsche Ihnen eine inspirierende und unterhaltsame Lektüre.

OLIVER DIEDRICH

Trends im Rechenzentrum

Wo früher teure Spezialhardware ihren Dienst verrichtete, soll es jetzt Software-defined Anything richten. Server werden zu hoch standardisierten Wegwerf-VMs, Ceph und Co. verwandeln günstige Standardhardware in verteilten Software-defined Storage, SDN abstrahiert die Netzwerke von der Verkabelung und Container machen Dienste unabhängig vom Betriebssystem. Das öffnet Raum für umfassende Automatisierung – zumindest in der Theorie.

ab Seite 6

Moderne Softwareentwicklung

Agiles Arbeiten und DevOps-Ansätze ersetzen in immer mehr Entwicklungsabteilungen umfangreiche Pflichtenhefte und die strikte Trennung zwischen Entwicklern und Admins. Das verändert auch die Art der Zusammenarbeit – und letztlich die Kultur im Unternehmen. Aktuelle technische Entwicklungen wie Functions as a Service, Microservices und KI eröffnen Entwicklern zudem neue Möglichkeiten.

ab Seite 66

Systeme und Netze

Softwareentwicklung

(Rechts-)sichere IT

In den letzten Jahren hat die Regulierung viele Bereiche der IT durchdrungen. Kein Wunder, macht doch die umfassende Vernetzung und das Anhäufen gigantischer Datenmengen gesetzliche Regeln immer nötiger. Ein wirksames IT-Recht und die Verfolgung von Kriminalität im Netz werden immer wichtiger – wie auch der Schutz vor Angreifern durch ein modernes Sicherheitskonzept.

ab Seite 90

Digitalisierung, Wirtschaft und Gesellschaft

Die Digitalisierung, derzeit in aller Munde, verändert schon seit vielen Jahren nicht nur die Unternehmen, sondern auch die Gesellschaft. *iX* wirft einen Blick auf aktuelle Entwicklungen bei sozialen Netzen, Quantencomputing und Kryptowährungen und besucht deutsche KI-Start-ups.

ab Seite 118

Sicherheit und IT-Recht

Wirtschaft und Gesellschaft

30 Jahre iX

Rubriken

Automation im RZ – Wunsch und Wirklichkeit

Das Ziel vor Augen

Martin Gerhard Loschwitz

Heute ist die Automatisierung des Rechenzentrums ein Schlüsselfaktor und entscheidet über Erfolg und Misserfolg. Wunsch und Wirklichkeit unterscheiden sich oft aber radikal.

Wer, wie der Autor, vor fast 20 Jahren in Sachen Rechenzentrum und Serverbetrieb sozialisiert wurde, der erinnert sich noch an Systeme und Workarounds, mit denen junge Administratoren heute gar nichts mehr anfangen können. Wer 2007 etwa einen Hochverfügbarkeitscluster auf Basis von Heartbeat und DRBD konstruierte, musste sich noch Gedanken darüber machen, wie es die DRBD-Konfiguration von einem Clusterknoten auf den anderen schafft. SSH war eine Möglichkeit, doch allzu oft vergaß man das Synchronisieren der Dateien und ärgerte sich später über sich selbst: Ging beim Einrichten etwas schief, ging wegen solcher Divergenzen der Failover in die Hose und gestaltete die Rufbereitschaft anstrengender als nötig.

Der Administrator von heute hingegen lächelt angesichts solcher Aufgabenstellungen nur müde und zeigt auf sein Automationssystem – vielleicht auf Ansible-Basis [1]. Es gewährleistet, dass alle Dateien clusterweit identisch bleiben. Und weil der Administrator schlau war, hat er gleich ein Auditing wie InSpec integriert [2]. Das schaut ebenfalls regelmäßig nach, ob in allen Dateien drinsteht, was drinstehen soll, und schlägt andernfalls Alarm. Schöne neue Welt also, und offensichtlich ist das Thema Automatisierung in modernen Rechenzentren damit ein für alle Mal abgehandelt. Oder doch nicht?

Wer das annimmt, sitzt einem fatalen Irrtum auf. Zwar versprechen die Automation-Hersteller in ihren Broschüren eine perfekte Automatisierung bis in die letzte Ecke des Rechenzentrums, doch Wunsch und Wirklichkeit unterscheiden sich hier fundamental. Schaut man sich in einem beliebigen RZ um, fällt schnell auf: Inseln gibt es allerorten. Oft erfinden Administratoren für jedes Setup, das sie betreiben, das Rad immer wieder neu. Viele Aspekte sind gar nicht automatisiert, etwas das Handling von Hardware.

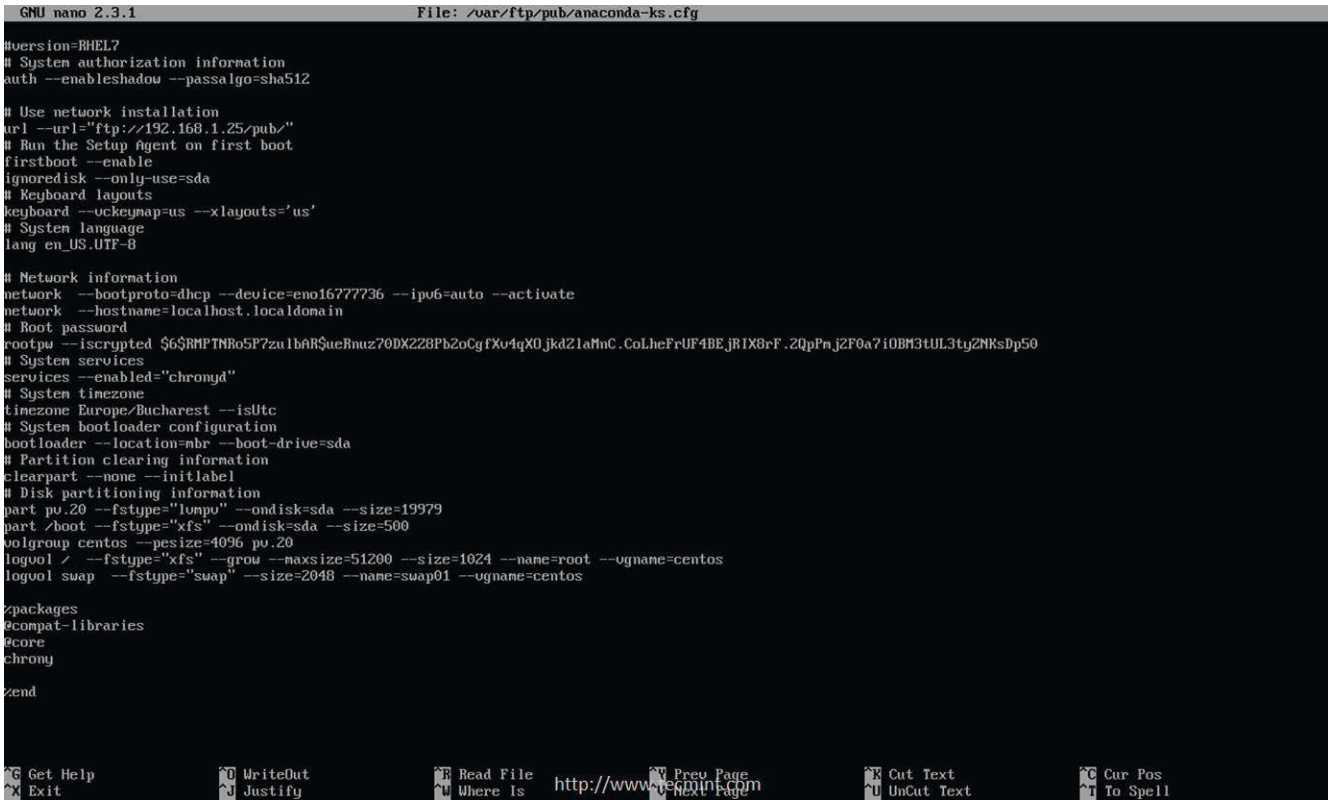
iX-TRACT

- In Rechenzentren lassen sich immer mehr Aufgaben der Serververwaltung automatisieren.
- Für homogene Neuinstallationen wie OpenStack-Clouds gibt es fertige Automationssysteme.
- Bei älteren Umgebungen ist die Automatisierung mit Engagement und Eigenentwicklungen verbunden.

Wie hätte mans denn gern?

Bevor dieser Artikel auf die am Markt verfügbaren Produkte eingeht und zeigt, was mit aktueller Technik möglich ist, stellt sich eine andere Frage: Wie sieht der Idealzustand im Rechenzentrum aus? Welchen Grad an Automation muss ein Setup erreichen, damit sich der Administrator beruhigt um andere Themen als den Betrieb seiner Plattform kümmern kann?

Quelle: Tecmint



Kickstart ist Red Hats automatischer Installer, der auf Basis eines simplen Templates Systeme aufsetzt (Abb. 1).

Die Antwort darauf ist verhältnismäßig simpel. Eigentlich hat der Administrator ein Interesse daran, dass alle Aufgaben automatisiert sind, für die er oder seine Kollegen anderenfalls die immer gleichen Handgriffe zu vollziehen hätten. Allerdings verstecken sich hinter dieser Aussage eine Reihe technischer Details – auch und vor allem weil sich das Ziel der Automation in den vergangenen Jahren und Jahrzehnten immer wieder verändert hat, je nachdem, was technisch gerade möglich war.

Ursprüngliches Ziel der Automation war die effizientere Betreuung von Systemen. Administratoren waren es vor 15 Jahren gewohnt, dieselben Aufgaben immer und immer wieder zu erledigen, zumal damals die Zahl der Server, die ein Administrator betreute, weit geringer war als heute. Hatte man einen neuen Server gekauft, ausgepackt und ins Rack eingebaut, folgte zunächst die manuelle Installation eines Betriebssystems. War das vorhanden, installierte man zu Fuß oder mit Shell-Skripten Marke Eigenbau die benötigten Pakete samt den hierfür gebrauchten Konfigurationsdateien. Am Ende des Vorgangs nahm man das betroffene Systeme händisch ins Monitoring auf und führte eine formale Übergabe an den Kunden durch – erst dann war der ganze Vorgang erfolgreich abgeschlossen.

Dass das kein sonderlich effizientes Vorgehen ist, merkten die Linux-Server-Distributoren schon einigermaßen früh. Bei Red Hat existiert Anaconda im Huckepack von Kickstart seit vielen Jahren (siehe Abbildung 1), SUSE lässt sich per AutoYaST im vollautomatischen Modus installieren und Ubuntu erbt das Preseeding, das die Debian-Entwickler schon in den ersten Versionen des Debian-Installers erdacht und sukzessive erweitert haben. Den Fully Automated Installer für Debian gibt es gar noch länger (siehe Abbildung 2).

Wer sich mit dem Thema Automation beschäftigte, konnte sich schon vor vielen Jahren Arbeit vom Hals schaffen. Dass viele Administratoren sich damit nie beschäftigten, hat mindestens zwei Gründe: Zum einen waren konventionelle Setups kleiner als die großen skalierbaren Plattformen der Gegenwart. Zu oft war man deshalb der Meinung, die Arbeit, die man in die Automatisierung investieren müsse, stehe nicht in Relation zum Aufwand der händischen Installation. Zum anderen hielten viele Administratoren die Installation des Betriebssystems für den geringsten Teil. Zeitaufwendig sei das anschließende Einrichten der Systeme.

Die Automation muss also die Installation des Betriebssystems auf Servern umfassen, und sobald ein Server mit den rudimentären Linux-Werkzeugen ausgestattet ist, muss sie die Installation aller benötigten Software-Komponenten übernehmen.

Wartung bitte nicht vergessen

Zu einer leistungsfähigen Automation gehört allerdings viel mehr. Das Out-of-Band-Management etwa ist ein kritischer Faktor bei modernen Servern – denn nur mit ihm kommt man noch remote an den Server, wenn dieser kein funktionierendes Betriebssystem mehr hat. Damit es diese Aufgabe aber erfüllen kann, ist der Management-Port nach der Systeminstallation zu konfigurieren. Beim IPMI-Protokoll ist das keine Herausforderung, denn die Controller sind von der Kommandozeile aus steuerbar. Auch andere Remote-Management-Protokolle wie HPEs iLO oder Dells DRAC bieten entsprechende Schnittstellen. Viel zu oft bleiben sie aber ungenutzt, was im Fall des Falles schiefgehen kann.

Beinahe jeder Administrator wird sich eingestehen müssen, dass es die Fass-mich-nicht-an-Systeme auch in seiner Vergangenheit gegeben hat. Gemeint sind Server, die einmal installiert und konfiguriert und dann unkontrolliert so modifiziert wurden, dass am Ende niemand mehr den Zustand des Systems kannte oder hätte beschreiben können. Solche Systeme bekommen dann keine Updates mehr, weil unklar ist, wie sie sich auswirken.

Heute geht das schon aus Sicherheitsgründen nicht mehr. Eine Aufgabe der Automation besteht darin, die Server der eigenen Flotte regelmäßig zu aktualisieren. Lokale Änderungen entfallen in einer konsequent geplanten Automation, manuelle Änderungen etwa an Ansible vorbei verbieten sich.

Das deutsche IT-Urgestein Kristian Köhntopp postuliert etwa seit Jahren sehr prägnant, dass ein Administrator zwei Dinge tun muss, nachdem er sich per SSH auf einem System eingeloggt

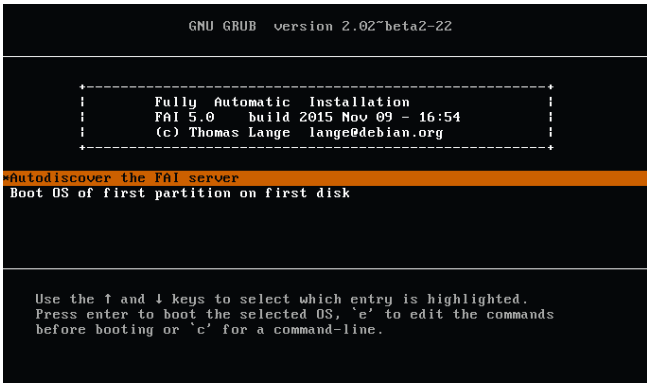
hat. Erstens das System in die – selbstredend völlig automatische – Reinstallation schicken, weil es ab diesem Zeitpunkt als kompromittiert zu gelten hat. Und zweitens ein internes Ticket aufmachen, weil entweder die Automation oder das zentrale Logging nicht gut genug ist, wenn ein manuelles SSH-Login nötig ist. Klingt radikal, trifft aber in Sachen Automation-Wunschszenario den Nagel auf den Kopf.

Und was ist mit der Hardware?

Ein Punkt spielt in den Köpfen vieler Administratoren bisher keine Rolle: die Automation der Hardware. Hilfreich wäre es etwa, wenn es letztlich egal wäre, in welchen Racks sich welche Systeme befinden, weil sie selbst mitteilen können, wo sie sind. Dann wäre höchstens noch auf Faktoren wie Brandschutz- und Verfügbarkeitszonen zu achten – ansonsten könnten die Server in jedes beliebige Rack.

Ganz utopisch ist das nicht: Verschiedene RZ-Ausrüster bieten mittlerweile solche Identifikationssysteme auf RFID-Basis an. Rittal füttert die entsprechenden Informationen in das eigene Asset-Management und spricht gleich einen weiteren wichtigen Punkt der Automation an: Für die automatische Installation muss klar sein, welche Rolle ein Server einnehmen soll. In einer OpenStack-Cloud etwa muss aus ihm ein Compute-Knoten werden. Diese Informationen lassen sich allerdings nicht komplett automatisch herausfinden. Hierfür ist ein DCIM-System (Datacenter Inventory Management) nötig.

Ein DCIM verzeichnet alle aktiven Assets eines Rechenzentrums in einem zentralen Dienst. Hier müssten auf Basis der RFID-Informationen sämtliche Server eingetragen sein, sodass der Administrator sie mit validen IP-Adressen für Out-of-Band-Management und LAN sowie Tags für die jeweilige Rolle versehen kann. Und apropos IP-Adressen: Die klassische Excel-



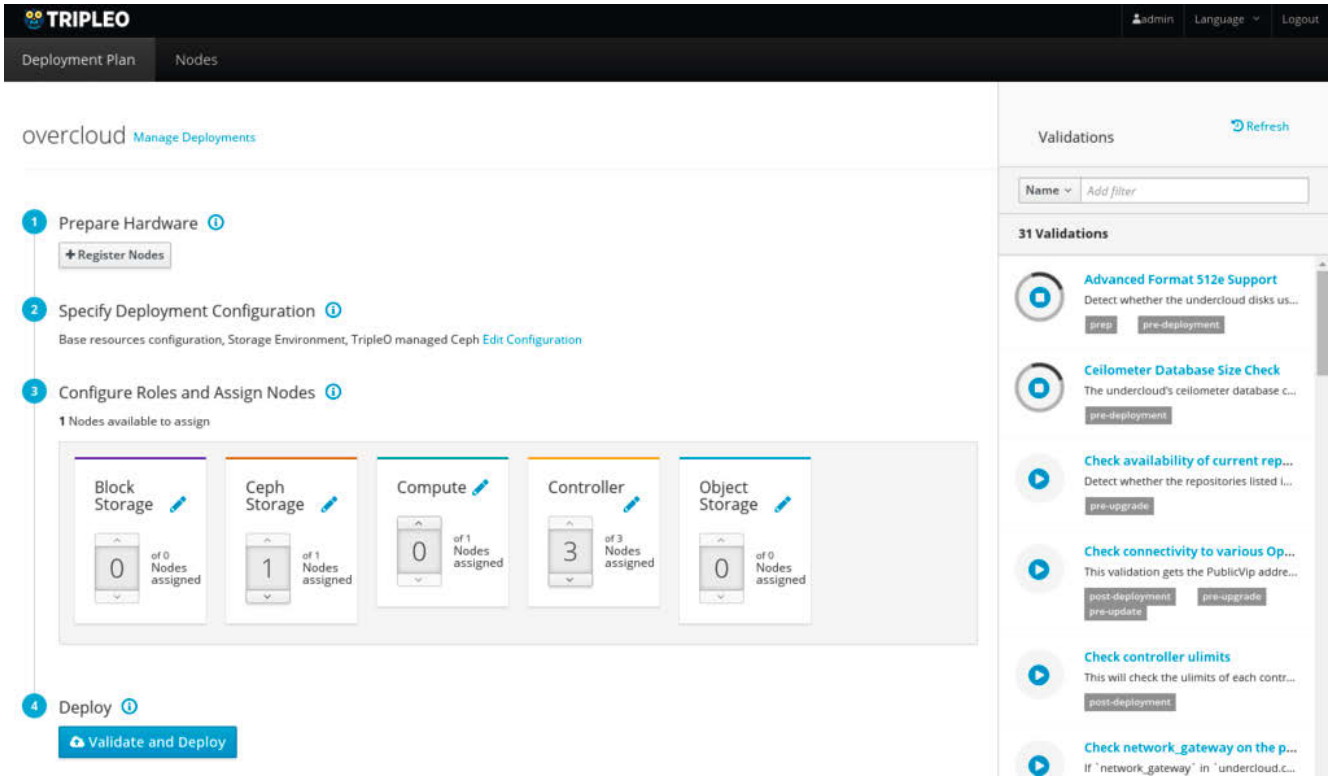
Quelle: FAI

FAI erlaubt es, Systeme auf Debian- und Ubuntu-Basis automatisch und per Preseeding auszurollen (Abb. 2).

Liste, die alle Server samt Adressen verzeichnet, hat in automatisierten Setups ausgedient. Viel eher kommt im besten Fall ein eng mit dem DCIM-System verzahnter IP Address Manager (IPAM) zum Einsatz, der alle relevanten Informationen zentral vorhält.

Last, but not least sollten Administratoren die Infrastruktur im RZ nicht außer Acht lassen. Redundante Stromkreise sind sinnvoll mit den redundanten Netzteilen von Servern und Switches zu koppeln. Die Verkabelung muss also passen.

Eine besondere Herausforderung stellt die Integration der Switches dar, etwa wenn VLANs oder ähnliche Funktionen zum Einsatz kommen, die auf den Switches direkt zu konfigurieren sind. Denn Ansible etwa muss wissen, wie es die zuvor zentral hinterlegte Konfiguration auf den Switches dann auch tatsächlich ausrollt. Sinnvoller wäre freilich ein Ansatz des Software-defined Networking, doch der kommt mit ganz eigener Komplexität und eigenen Ansprüchen daher.



Quelle: OpenStack

TripleO für OpenStack kommt mit einem GUI daher, dem Director, das das Ausrollen von Knoten in OpenStack-Setups signifikant erleichtert (Abb. 3).

Der Idealzustand ist damit skizziert, das Wolkenkuckucksheim, auf das der Administrator hinarbeiten sollte. Stellt sich die Frage, wie viel davon in der Realität umzusetzen ist. Einerseits gibt es die integrierten Plattformen der großen Hersteller wie Red Hats OpenStack Platform oder SUSE Cloud. Die decken bereits viele Aspekte der Automation ab.

Wo geht es jetzt lang?

Daneben existieren die klassischen Umgebungen – und die machen den Löwenanteil der gesamten RZ-IT aus. Wer eine bestehende Installation automatisieren möchte, kann auf die integrierten Fähigkeiten der Fertigprodukte kaum zählen und muss stattdessen selbst Hand anlegen. Dieser Artikel skizziert beide Ansätze, stellt sie aber aufgrund der offensichtlich fehlenden Vergleichbarkeit nicht direkt gegenüber.

Wer eine Containerplattform auf Basis von Kubernetes plant oder sich gar eine OpenStack-Cloud ins Rechenzentrum stellen will, weiß, dass er das Thema Automation gleich beim ersten Versuch sinnvoll angehen muss. Denn wenn das nicht klappt, brennt die schöne neue Cloud im schlimmsten Fall ob ihrer Unwartbarkeit gleich wieder ab. Das haben auch die Hersteller erkannt. Wer deshalb eines der Boxed Products bei SUSE, Red Hat oder einem der anderen Anbieter kauft, bekommt das beinahe größtmögliche Maß an Automation geliefert.

Red Hats OpenStack Platform (RHOP) zeigt beispielhaft, wie das in der Realität aussehen kann: Hier installiert der Administrator zunächst eine „Undercloud“, die Dienste wie PXE, TFTP, DHCP, DNS und weitere basale Services zum Leben erweckt. Bemerkenswert ist, dass diese „Undercloud“ eine vollständig funktionale OpenStack-Installation ist, deren einziger Zweck darin besteht, die „Overcloud“ zu betreiben. In jener laufen die virtuellen Maschinen der Kunden.

Anders, als man es erwarten möchte, basiert die Overcloud selbst nicht auf VMs, sondern auf echtem Blech: Red Hat setzt hier auf die OpenStack-Komponente Ironic, um aus der OpenStack-Komponente Nova heraus echte Server so zu verwalten, wie Nova es auch mit virtuellen Maschinen kann.

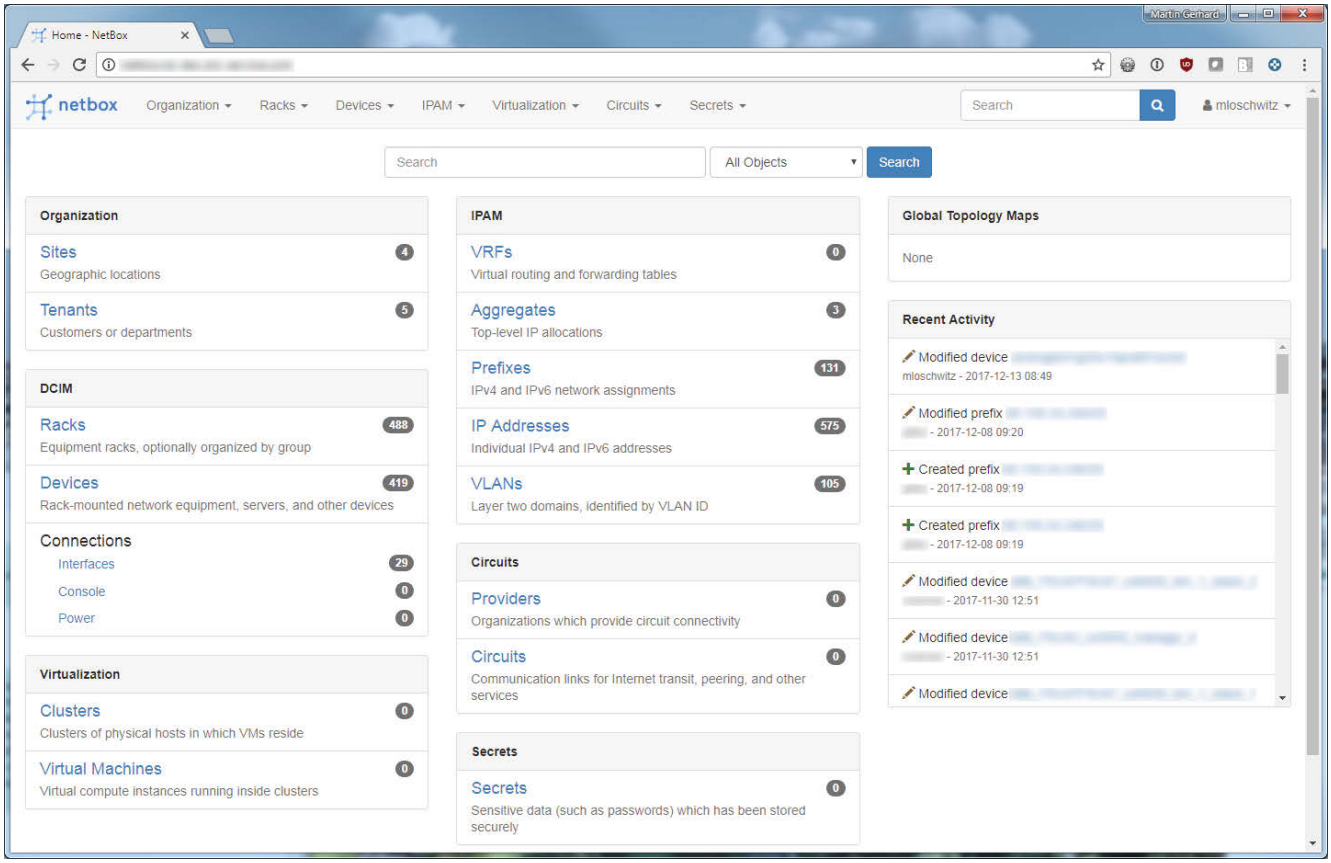
Per OpenStack Director lässt sich in RHOP dann jedem einzelnen Server eine Rolle zuweisen und per Mausklick in eine auf PXE, DHCP und TFPT basierte Installation schicken, an deren Ende ein basales Betriebssystem auf dem Server läuft (siehe Abbildung 3). Der Director ermöglicht ein Flottenmanagement zumindest zum Teil, so ist das Einspielen von Sicherheitspatches möglich.

Viel Licht, wenig Schatten

Nach der Installation des Betriebssystems hört die Automation des Directors übrigens nicht auf, allerdings sieht der Administrator von ihr denkbar wenig. Auf den einzelnen Systemen muss er sich gar nicht einloggen. Deshalb merkt er auch nicht, dass Ansible als Automatisierer zum Einsatz kommt, also Red Hats eigenes Tool für diese Aufgabe. In den aktuellen Releases von RHOP relativiert sich das zum Teil. Mittlerweile bietet die Software dem Administrator ganz offiziell die Möglichkeit, in das Ansible-Geschehen einzugreifen. Davon kann der Administrator Gebrauch machen, muss es aber nicht.

Wer in der luxuriösen Situation ist, ein neues Setup auf der sprichwörtlichen grünen Wiese zu bauen, bekommt in den gängigen Cloud-Distributionen dafür alle nötigen Tools, denn SUSE Cloud oder die Produkte von Canonical unterscheiden sich von

Quelle: NetBox



NetBox bietet sowohl ein Datacenter Inventory Management (DCIM) als auch die Verwaltung von IP-Adressen (Abb. 4).

RHOP zwar in den genutzten Komponenten, nicht aber im Funktionsumfang. Dieser Ansatz geht allerdings auch mit zwei äußerst unangenehmen Nebeneffekten einher.

Hat man keine grüne Wiese zum Neubauen, ist es fast unmöglich, eine bestehende Infrastruktur in die Abläufe eines solchen Frameworks zu integrieren. In den meisten Fällen käme das einer kompletten Neuinstallation aller Dienste gleich, was meist unrealistisch ist. Zudem ist die in Produkten wie RHOP zu findende Automation auf eben jene Produkte eng zugeschnitten. Es ist nutzlos, einen Server als physischen Knoten in RHOP zu verwalten, wenn er nicht eine Rolle in der OpenStack-Cloud haben soll.

Nicht zuletzt decken Automationsprodukte wie RHOP nicht den kompletten Funktionsumfang einer Automation ab. Eine dedizierte Anwendung fürs Asset-Management sucht man etwa vergeblich. Auch das IP Address Management ist nur rudimentär vorhanden. Die Konfiguration der Out-of-Band-Schnittstellen der Server erledigt der Administrator bei RHOP und Co. zwangsläufig selbst, denn diese Verbindung muss stehen, damit die Dienste die Server überhaupt aus der Ferne starten und in eine Betriebssysteminstallation booten können. Wer es flexibler braucht, bastelt deshalb zwangsläufig selbst. Der Grad der erreichbaren Automation hängt stark vom Aufwand ab, den ein Administrator zu treiben bereit ist.

Das teuflische Netz

Alle Komponenten, die für ein Paket wie RHOP nötig sind, braucht der Administrator auch in einer selbst gebauten Automation. DNS, PXE, DHCP und TFTP müssen zentral installiert sein und grundsätzlich funktionieren. Im Idealfall rollt man sie redundant auf Blech aus – automatisiert, damit das Setup zu einem späteren Zeitpunkt reproduzierbar ist. Die Dienste allein helfen aber nicht weiter. Denn wenn Maschinen automatisiert vom Netz booten können, stellt sich immer noch die Frage: Wohin? Und um diese Frage zu beantworten, sind einige grundlegende Designentscheidungen zu treffen.

Grundsätzlich stellt sich etwa die Frage, wie das Netz organisiert sein soll, damit das automatische Booten klappt. Die wenigsten NICs sind überhaupt in der Lage, PXE über getaggte VLANs zu sprechen, und wenn, dann nur mit irgendwelchen obskuren Karten-BIOS-Versionen, die ab Werk garantiert nicht geflasht sind. Man sollte also davon ausgehen, dass das Deployment von Systemen grundsätzlich in einem ungetaggten VLAN passiert und dass jeder Server, der einen PXE-Request absendet, ganz automatisch im Deployment-Netz landet.

Geht man die Sache radikal an, kann das Deployment-Netz sogar mit dem Managementnetz der Plattform identisch sein. Dann sollte man es allerdings hinreichend groß dimensionieren, damit einem nicht die IP-Adressen ausgehen. Möchte man lieber mehrere logische Netzsegmente nutzen, müssen alle Netze mit Ausnahme des Deployment-Netzes getaggt anliegen, und zwar sowohl auf den Servern als auch auf den Switches. Alle Auto-Installer der Distributionen können das auf einem System konfigurieren. Inwieweit die vor Ort vorhandene Netzwerkhardware sich aus der Automation heraus beackern lässt, hängt vom Einzelfall ab.

NetBox als Verbündeter

Als Datenbank für DCIM- und IP-Daten kann im modernen Rechenzentrum NetBox zum Einsatz kommen (siehe Abbildung 4) [3]. Das verfügt praktischerweise auch über eine API nach dem REST-Prinzip, sodass es maschinell auslesbar und bearbeitbar ist. Damit es in einem automatisierten Setup seine volle Wirkung entfaltet, wird der Administrator diverse Skripte schreiben wollen, die Daten von den Systemen in NetBox transferieren und umgekehrt. Die automatische Erkennung von Knoten etwa muss dazu führen, dass die Systeme in NetBox zumindest angelegt sind.

NetBox beherrscht die Abbildung von RZ-Räumen, Rackreihen und Racks (siehe Abbildung 5). Wer über ein RFID-System zur Erkennung der Systeme verfügt, könnte darüber auch ihre exakte Position in NetBox eintragen. Das wäre allerdings eine komplette Eigenentwicklung, die insbesondere davon abhängt, ob der Hersteller der RFID-Lösung eine API in seiner Software hat und ob diese offen und gut dokumentiert ist.

Bleibt die Frage, was ein Server denn booten soll, wenn der Administrator ihn ausgepackt, im Rack verschraubt und das erste Mal eingeschaltet hat. Hier eignet sich nur ein eigens für diesen Zweck gebautes Inventarisierungsimage. Es muss die lokal

auf der Maschine befindlichen Netzwerkschnittstellen inventarisieren, sich aus dem Out-of-Band-Netz in NetBox eine freie IP-Adresse heraussuchen, diese dem Management-Port des Systems zuweisen und den root-Account mit einem Passwort versehen. Am Ende der Inventarisierung bootet das System dann idealerweise gleich in den Installer des jeweiligen Betriebssystems, der den restlichen Prozess startet.

NetBox spielt in einer Umgebung dieser Art eine viel größere Rolle, als es zunächst den Anschein hat. In NetBox sollte etwa für jeden Tag verzeichnet sein, in welchem Zustand sich eine Maschine gerade befindet. Ist sie installiert, verharrt sie im Leerlauf oder läuft auf ihr ein valides Betriebssystem?

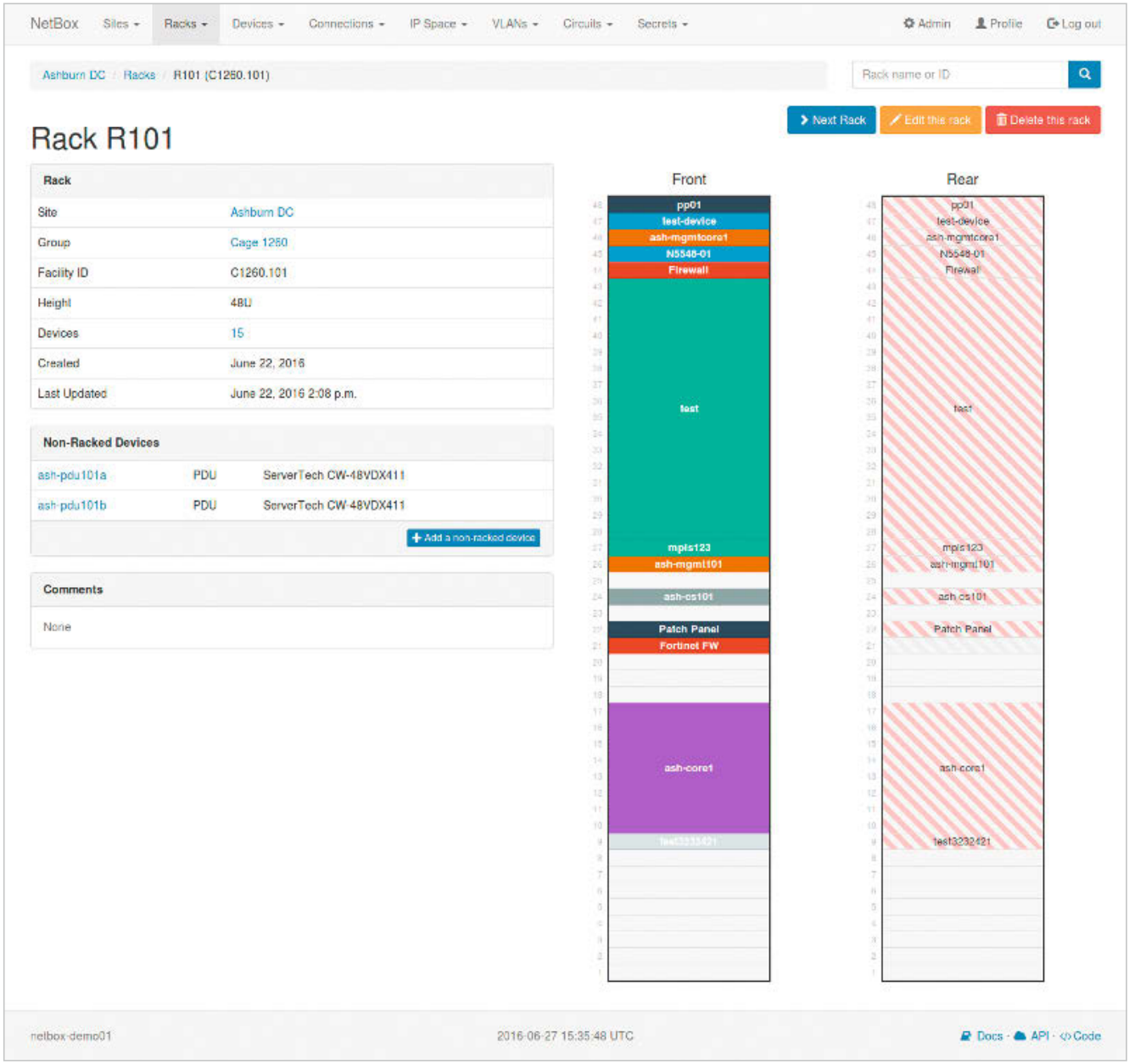
Auch andere Dienste brauchen NetBox

In einer vielschichtigen Umgebung ist es unwahrscheinlich, dass der Administrator alle Systeme mit dem gleichen Betriebssystem ausstatten möchte. Deshalb sollte es in NetBox ein entsprechendes Tag geben, das Auskunft darüber gibt, welche Distribution auf dem System landen soll.

Damit PXE dem jeweiligen System aber das richtige Boot-Image für die Installation zuschanzen kann, muss man die Konfigurationsdateien für PXE aus NetBox heraus generieren. Dank der NetBox-API und des Python-Moduls pynetbox ist das aber nicht schwierig.

Ähnliches gilt für DHCP. Ideal ist es immer, wenn sich IP-Adressen dynamisch zuweisen lassen. Wünschenswerter für das Monitoring und andere Dienste sind aber pseudodynamische Adressen. Das bedeutet, die Systeme bekommen die IP-Adresse zwar vom DHCP-Server, dort sind sie aber an die jeweilige Mac-Adresse gebunden.

Nicht zuletzt spielt NetBox auch bei der Konfiguration der Dateien eine Rolle, die der Betriebssystem-Installer für seine Arbeit benötigt. Pro Host lassen sich aus Kickstart heraus etwa Dateien generieren, die die spezifische Netzwerkkonfiguration



Quelle: NetBox

Die Ansicht von RZ-Raum, Rackreihen und Racks im IP Address Management (IPAM) von NetBox muss der Administrator händisch befüllen (Abb. 5).

Ganz fette

Glückwünsche zu 30 Jahren iX!

Vor über fünf Jahren habe ich bei Rheinwerk angefangen – damals noch Galileo. Und Ihr habt vom Start weg über meine Programmierabenteuer berichtet. Da staunte auch der alte Bossingen nicht schlecht, wie groß Euer Interesse an meinen Code-Experimenten war. Hat mir sehr imponiert. Die iX lese ich ja regelmäßig, in der Kaffeepause oder auch schon mal zum Schmökern in meiner Werkstatt. Klar, Ihr seid ja auch die Top-Adresse, wenn man in der IT-Welt auf dem Laufenden bleiben will. Da wünsche ich Euch gleich noch einmal 30 Jahre treue Leser. Auf mich könnt Ihr auf alle Fälle zählen!

Euer Schrödinger
Ach ja: Die Rheinwerker lassen auch schön grüßen!

Über meine neuesten Programmierabenteuer kann man sich hier aufschlauen:
rheinwerk-verlag.de/schroedinger-programmiert

vorgeben. Nach der Installation sind dann alle vorgesehenen Ports konfiguriert.

Als Letztes wäre da noch der Automatisierer, der nach der Basisinstallation einen Host zu dem macht, was er werden soll. Auch er bezieht seine Inventory-Liste idealerweise aus NetBox. Für Ansible gibt es gleich mehrere Skripte, die das exemplarisch ermöglichen und das für Ansible benötigte Inventory im Flug aus den Daten in NetBox generieren.

Wer NetBox in der beschriebenen Art und Weise einsetzt, hat zwar am Anfang viel Arbeit für Eigenentwicklungen vor sich, bekommt dafür aber einen komplett automatisierten Workflow für das Deployment von Systemen – zumindest ab dem Zeitpunkt, zu dem die Maschine im Rack hängt. Wenn das System installiert und konfiguriert ist und alle beteiligten Services laufen, ist die Arbeit aber noch nicht vorbei.

Das neue System muss etwa ins Monitoring, und auch dieser Prozess ist idealerweise automatisiert. Das geht mit manchen Monitoringwerkzeugen allerdings deutlich besser als mit anderen. Immerhin: Im Grunde kann mittlerweile jedes Monitoringsystem Hosts aus einer Datenbank auslesen. Im simpelsten Fall baut man ein Skript, das alle aktiven Hosts aus NetBox ausliest und sie – samt Art des Systems – direkt im Monitoringsystem hinterlegt. Sonderlich elegant ist das allerdings nicht.

Cooler wäre es, clusterweit einen kleinen Zusatzdienst auszurollen: Consul als Cluster-Konsens-Algorithmus braucht so gut wie keine Ressourcen, ermöglicht aber das Definieren von „Diensten“, die auf einem Host laufen. Hat das Monitoringsystem wie Prometheus oder InfluxDB eine Schnittstelle zu Consul, kann es die zu überwachenden Hosts automatisch erkennen, sobald darauf Consul läuft. Ein ähnliches Vorgehen empfiehlt sich für ein zentralisiertes Logging, so dieses gewünscht ist. Wie üblich gilt: Je weniger Arbeit „zu Fuß“ zu erledigen ist, desto besser.

Wer NetBox auf diese Weise benutzt, merkt schnell, dass die Skripte und Erweiterungen zum Teil Funktionen nachrüsten, die es woanders schon gibt. Lifecycle-Management-Werkzeuge wie Foreman funktionieren nachweislich gut, decken nach der Erfahrung des Autors aber stets nur Teilaspekte der Automation ab. Zudem ist es komplizierter, die fehlenden Funktionen auf Basis von NetBox und Co. um bestehende Managementanwendungen herumzubauen, als mit den NetBox-Basics zu beginnen und die benötigten Funktionen mit Bordmitteln nachzurüsten. Wer sein Glück trotzdem versuchen möchte, sollte sich von diesem Artikel aber nicht abhalten lassen.

Die ideale Entwicklungsumgebung

Da das Bauen von Images viel Zeit frisst und diese dann auch noch regelmäßig zu aktualisieren sind, tut der Administrator gut daran, sich eine geeignete Entwicklungsumgebung zu schaffen, in der er die Betriebssystemabbilder automatisiert bauen kann. Auch diese zusätzliche Zeitinvestition zahlt sich auf die Dauer aus. Die „Economoy of Scale“ funktioniert bekanntlich am besten, wenn die einmal automatisierten Aufgaben möglichst immer und immer wieder durchzuführen sind.

Große Umgebungen lassen sich ohne ein gewisses Maß an Automation gar nicht sinnvoll betreiben. Zudem ist es unwirtschaftlich, gut ausgebildete Systemverwalter tumb die immer selben Arbeiten erledigen zu lassen. Denn einerseits erhöht stumpfe Routine in vielen Situationen die Zahl der Fehler statt sie zu senken. Andererseits gestaltet das systematische Reduzieren von Fehlern eine Umgebung stabiler – so ein Kernprinzip des „DevOps“-Ansatzes. Das jedoch schließt den Faktor Mensch quasi aus, zumindest was den Betrieb im Alltag angeht.

So logisch das aus technischer Sicht auch sein mag, die Forderung in einer Firma, mehr Automation umzusetzen, führt immer zu Sorgen innerhalb der Belegschaft. Schnell macht das Schreckgespenst des Personalabbaus die Runde, obwohl sich in Zeiten des Fachkräftemangels einmal rausgeworfene Leute nicht einfach ersetzen lassen. Das Ziel besteht nicht darin, Arbeit zu automatisieren, um Personal zu sparen, sondern darin, die vorhandenen gut ausgebildeten Menschen mit sinnvollen Dinge zu beschäftigen. Wer Automation in ein Unternehmen bringen möchte, sollte die dabei aufkommenden Sorgen trotzdem nicht unterschätzen. Hier hilft nur ausgiebige, nachvollziehbare Kommunikation.

Zudem lohnt es sich, sich rechtzeitig Verbündete zu suchen. Wer etwa mit dem Hardwarelieferanten abspricht, dass die Server bei der Übergabe so konfiguriert sein sollen, dass sie gleich ins PXE booten, erspart sich viel Arbeit beim Umkonfigurieren.

Hilfreich kann es auch sein, wenn der Hersteller vorab eine Liste der Systeme mit zentralen Werten wie den MAC-Adressen liefert. Dann lässt sich der beschriebene Prozess umdrehen: Per Parser lassen sich die Server dann in NetBox eintragen, sodass beim ersten Boot alle Infos bekannt sind und die Inventarisierung entsprechend kürzer ausfällt. Viele Administratoren fragen bei ihren Lieferanten wegen solcher Dinge oft gar nicht nach, obwohl durchaus die Option dazu bestünde. Deshalb gilt: Fragen kostet nichts.

Fazit

Sowohl geplante als auch ältere Installationen lassen sich automatisieren. Wer auf der grünen Wiese beginnt und sich ein Box-Produkt eines Herstellers organisiert, hat es am leichtesten: Die von den Anbietern geschnürten Pakete liefern in aller Regel alle jeweils benötigten Funktionen für ein bequem wartbares Setup – wenn auch mit leichten Abstrichen in der B-Note.

Anders sieht es bei nicht standardisierten Umgebungen aus. Hier ist viel Eigeninitiative und einiges an Entwicklungsaufwand nötig, die sich letztlich aber auszahlen.

An ihre Grenzen stößt die Automation im Augenblick noch bei der Hardware. Systeme, die Auskunft darüber geben, wo sich welche Server befinden, existieren zwar, sind aber im Massenmarkt noch nicht angekommen und etwas kantig in der Handhabung. Auch das automatisierte Einbauen von Systemen an ihren Zielstandort ist noch eine Zukunftsvision – in Zukunft dürfte damit allerdings zu rechnen sein. (sun@ix.de)

Quellen

[1] Sebastian Meyer, André Nähring; Systemverwaltung; Gut abgemischt; Konfigurationsmanagement ohne Client-Installation; *iX* 10/2014, S. 72

[2] Rudolf Grötz, Nebojsa Nikolic; Compliance; Schnell abgehakt; Sicherheitsregeln automatisch testen; *iX* 1/2018, S. 114

[3] Martin Gerhard Loschwitz; RZ-Verwaltung; Verkabelte Kisten; DCIM & IPAM mit NetBox; *iX* 3/2018, S. 58

Martin Gerhard Loschwitz

ist Senior Cloud Architect bei Mirantis und beschäftigt sich vorrangig mit OpenStack, Ceph und Kubernetes.

secunet gratuliert der **iX** ganz herzlich zum **30-jährigen Jubiläum!**

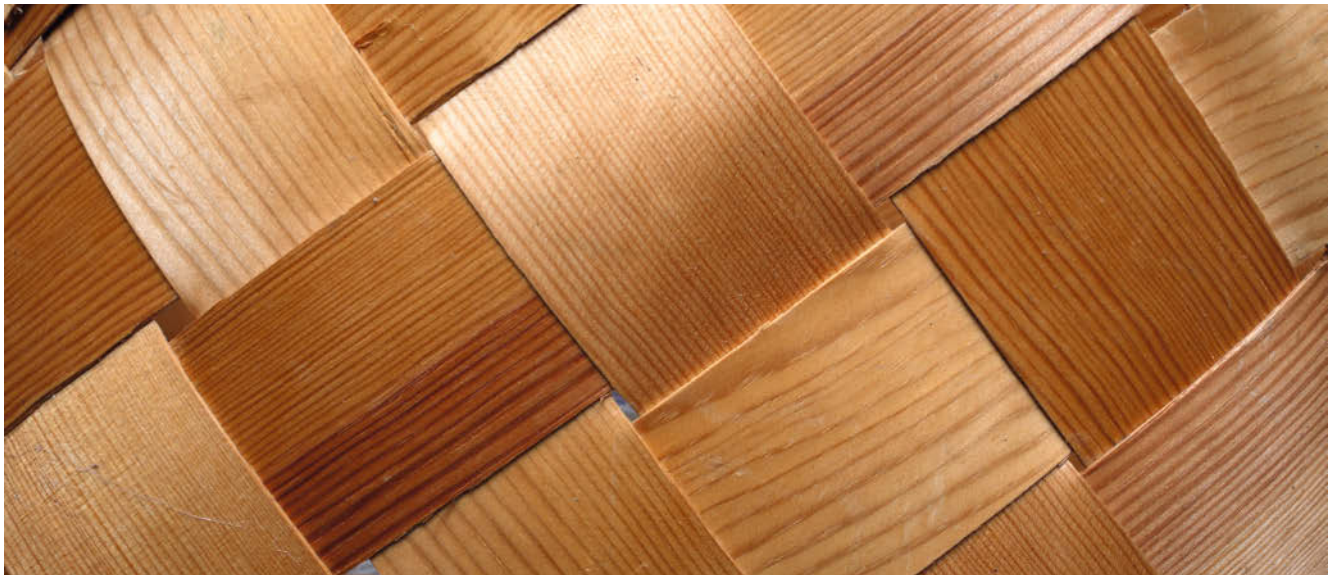
Was tun Sie bei einem Hackerangriff?

Entspannt bleiben – denn mit secunet sind Daten und Infrastruktur premiumsicher.

Wo Daten und IT-Infrastrukturen vor Cyberangriffen geschützt werden müssen, steht secunet bereit. Als IT-Sicherheitspartner der Bundesrepublik Deutschland bieten wir Behörden und Unternehmen Expertenberatung und premiumsichere Lösungen zum Schutz von Kommunikation und Daten.

secunet – Ihr Partner für IT-Premiumsicherheit.

secunet



Von verteilten Systemen zum Pervasive Computing

Verwoben

Ulf Troppens

Dem Wandel von Clients und Datentypen müssen sich auch die Speichermethoden stellen – eine Bestandsaufnahme.

Was auch immer es an großen Fortschritten bei den Speichermedien gab, ihre technischen Verbesserungen sind Peanuts im Vergleich zu der Einführung von Speichernetzen und dem Pervasive Computing. In einem 2003 veröffentlichten Artikel über Speichernetze war noch zu lesen: „Durchschnittlich verdoppelt sich in einem Unternehmen jedes Jahr die installierte Speicherkapazität: Wer heute mit 250 GByte Festplattenspeicher auskommt, wird in fünf Jahren um die 8 Terabyte verwalten und sichern. Solche Datenmengen benötigen eine professionelle Speicherverwaltung." [1]

Unter professioneller Speicherverwaltung verstand man damals die Einführung von Speichernetzen. Davor war die serverzentrierte IT-Architektur die vorherrschende Architektur für verteilte Systeme. In ihr sind Speichergeräte an einen einzelnen Server angeschlossen, zur Erhöhung der Verfügbarkeit per sogenanntem Twin-tailed Cabling auch an zwei, wobei stets nur ein Server das Speichergerät nutzen kann (siehe Abbildung 1). In beiden Fällen existiert der Speicher immer nur in Abhängigkeit des Servers, an den er angeschlossen ist. Andere Server müssen immer über diesen Server gehen, wollen sie auf die Daten zugreifen.

Fallen die zuständigen Server aus und existieren keine weiteren Kopien, gibt es keinen Zugang mehr zu den Daten. Zudem sieht dieses Konzept keine dynamische Zuteilung von Speicherressourcen vor. Wenn etwa eine Datenbank mehr Speicher benötigt als lokal vorhanden, nützt es überhaupt nichts, dass andere Server noch freie Kapazität haben.

Speichernetze heben diese Einschränkungen auf. Darüber hinaus bieten sie neue Möglichkeiten, Daten zu verwalten. Die Idee: Man ersetzt die SCSI-Kabel zwischen Server und Speicher durch ein Netz, das man zusätzlich zum existierenden LAN installiert und das ausschließlich dem Datenaustausch zwischen Servern und Speichergeräten dient (siehe Abbildung 2).

Die Emanzipation der Speichersysteme

In Speichernetzen existieren Speichersysteme als unabhängige Einheiten, auf die mehrere Server direkt zugreifen können, ohne dass zwangsläufig ein anderer Server involviert ist. Speichergeräte rücken damit in das Zentrum der IT-Architektur; Server hingegen werden zum Anhängsel an die Speichergeräte, die die Daten verarbeiten. Deshalb heißen IT-Architekturen mit Speichernetzen auch speicherzentrierte IT-Architekturen. Zudem erlauben Speichernetze Storage-Systemen, dass sie ihre Daten untereinander spiegeln und replizieren können, ohne dass Server involviert sind, und gewährleisten damit die Hochverfügbarkeit konsistenter Daten.

Mit der Einführung eines Speichernetzes konsolidierte man in der Regel auch den Speicher, indem man die vielen kleinen lokalen Festplatten durch wenige große Disksysteme ersetzte. Sie können heute eine Speicherkapazität von mehreren PByte haben. Zugleich finden sich in ihnen zunehmend Flashspeicher statt Festplatten.

iX-TRACT

- Nicht nur Speichermedien haben sich im Laufe der Jahrzehnte weiterentwickelt, auch die Speicherorganisation und -anbindung.
- Nachdem die Jahrhundertwende den Übergang von der serverzentrierten zur speicherzentrierten Architektur einleitete, vollzieht sich nun der Wechsel zum Pervasive Computing.
- Mit dem Pervasive Computing ziehen moderne Speicherarchitekturen wie Cloud-Computing und Objekt-Storage ein, außerdem Unmengen unstrukturierter Daten sowie ganz neue Anforderungen.

DIE EIERLEGENDE WOLLMILCHSAU GRATULIERT DER IX ZUM 30. GEBURTSTAG

- 20 eigene .de-Domains
- 2 x 250 GB SSD Speicherplatz
- NodeJS, Ruby, PHP, Perl, Python, Git, MySQL, u.v.m.
- 500 E-Mail-Postfächer
- Kostenlose SSL-Zertifikate

nur **4,76 €** pro Monat

Zum 30. Geburtstag der IX gibts

JEDE 30. BESTELLUNG GRATIS!*

BESTER PREIS GARANTIERT!
Wir unterbieten vergleichbare Tarife um 10%

www.netcup.de/sau

- Serverstandort Deutschland
- DDoS-Schutz inklusive
- Zufriedenheits-Garantie

Readers' Choice HOSTING & SERVICE PROVIDER AWARDS 2018 PLATIN

HOSTtest Webhoster des Jahres 2018 netcup

HOSTtest Webhoster des Jahres 2018 netcup

HOSTtest Webhoster des Jahres 2018 netcup

HOSTtest Webhoster des Jahres 2018 netcup

netcup® Quality Webhosting

www.netcup.de

Die genannten Preise verstehen sich inkl. MwSt. Alle Details zu diesen Angeboten finden Sie auf www.netcup.de
* Genauere Infos zur Gratis-Bestellung finden Sie auf www.netcup.de/sau. Der Rechtsweg ist ausgeschlossen.