

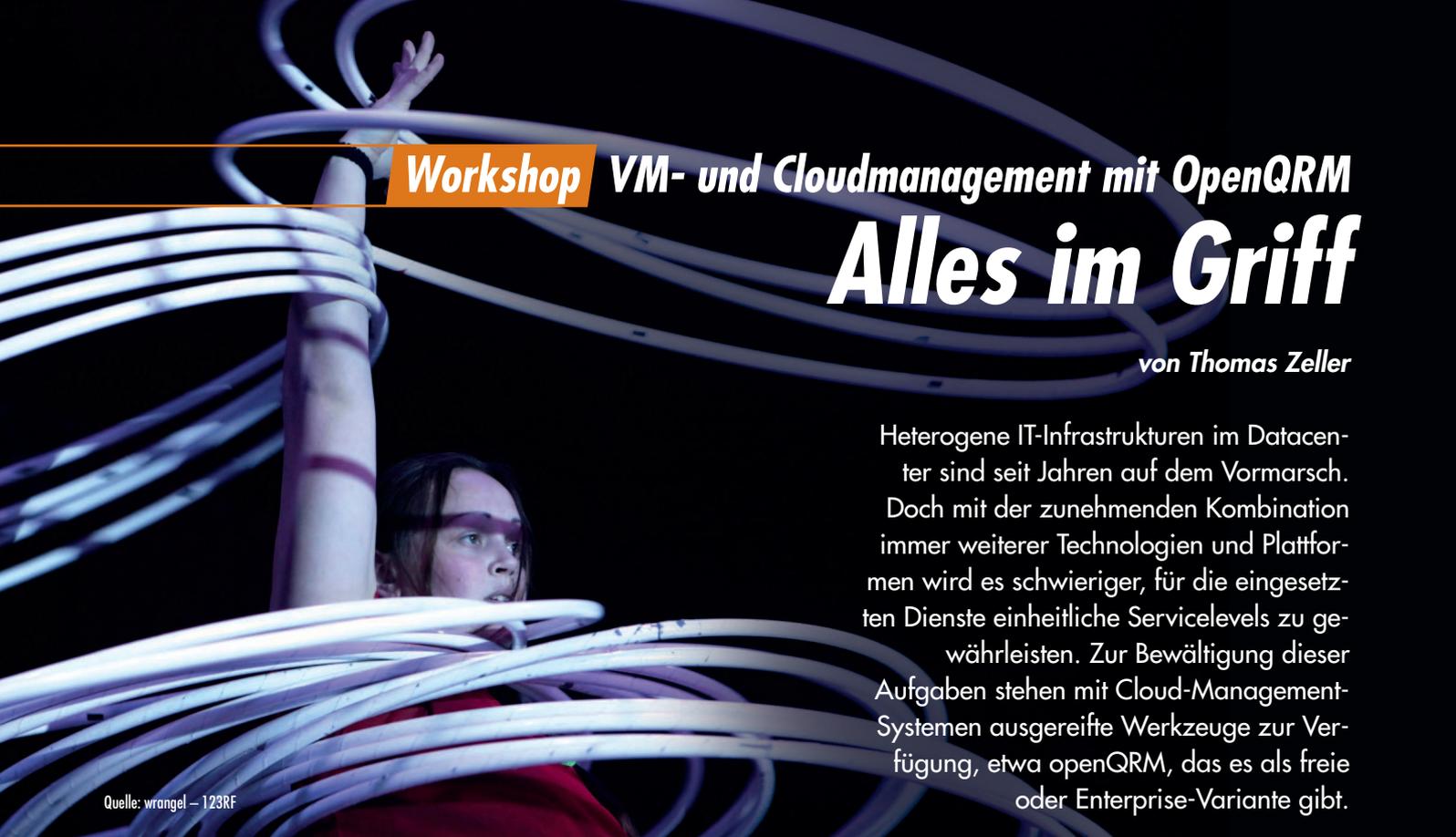
IT Administrator

Das Magazin für professionelle System- und Netzwerkadministration

Workshop:

VM- und Cloudmanagement mit OpenQRM





Workshop VM- und Cloudmanagement mit OpenQRM

Alles im Griff

von Thomas Zeller

Heterogene IT-Infrastrukturen im Datacenter sind seit Jahren auf dem Vormarsch. Doch mit der zunehmenden Kombination immer weiterer Technologien und Plattformen wird es schwieriger, für die eingesetzten Dienste einheitliche Servicelevels zu gewährleisten. Zur Bewältigung dieser Aufgaben stehen mit Cloud-Management-Systemen ausgereifte Werkzeuge zur Verfügung, etwa openQRM, das es als freie oder Enterprise-Variante gibt.

Quelle: wrangel – 123RF

Viele Unternehmen nutzen neben der auf Netzwerk, Server, Storage, Virtualisierung und Clients basierenden "klassischen" IT-Infrastruktur heute auch Cloud-Dienste. In einfachen Fällen ist hier nur der Webserver mit der Unternehmenswebsite zu einem Hostler ausgelagert oder es werden externe IT-Dienste wie Spam- und Virenerkennung als "Managed Service" angemietet. Doch der Druck steigt, den Anwendern immer mehr Dienste mit einer hohen Verfügbarkeit bereitzustellen. Und so werden immer häufiger interne Dienste wie E-Mail, SharePoint oder File-Services in externen Rechenzentren betrieben oder gleich in die Cloud von Amazon, Google, Microsoft & Co verlagert. Dabei ist es praktisch nicht zu vermeiden, dass auch unterschiedliche Technologien, zum Beispiel für die Virtualisierung, zum Einsatz kommen. So setzen viele Unternehmen intern vSphere oder Hyper-V ein, während die Virtualisierung bei Amazon auf Xen basiert. Je mehr Dienste also nach außen verlagert werden, desto eher muss sich das Administrationspersonal auch in neue Technologien und deren Managementwerkzeuge einarbeiten.

Genau hier setzen Cloud-Management-systeme wie OpenQRM an. Es stellt eine

webbasierte Open Source Datacenter Management- und Cloud-Plattform dar, mit deren Hilfe die verschiedenen internen und externen Technologien abstrahiert und wieder in ein gemeinsames Management-Werkzeug überführt werden können. Dieses Managementsystem kümmert sich dann auch gleich um Provisionierung, Hochverfügbarkeit und Monitoring der angebotenen Services. OpenQRM definiert sich dabei selbst als Framework und setzt auf eine offene Plug-in-Architektur. So wird beispielsweise ein bereits vorhandener Hypervisor wie KVM oder Xen einfach als einer von mehreren möglichen "Ressourcen-Provider" eingebunden.

Statt für die einzelnen Aufgaben wie Konfigurationsmanagement und System Monitoring jeweils eigene Werkzeuge bereitzustellen, bindet openQRM bewährte Open Source Management-Tools wie Nagios, Zabbix und collectd per Plug-in ein. Dank dieser Plug-in-Architektur ist das System auch offen für künftige Technologien und kann im Bedarfsfall vom Kunden sogar selbst angepasst werden. Im Lieferumfang von openQRM Enterprise sind etwa 50 Plug-ins enthalten (Bild 1). In diesem Workshop stellen wir die Basis-Installation, Konfiguration und Funktionsweise der openQRM Enterprise-Edition vor.

Die openQRM-Systemarchitektur besteht aus insgesamt drei Komponenten: Die Basis bildet die "Datacenter Management und Cloud Plattform". Dieses Core-Modul stellt die grundlegenden Funktionen von openQRM bereit und kommuniziert über die Plug-in-API mit ebenfalls im lokalen Netzwerk installierten Datacenter-Ressourcen (Hypervisor, Storage und Netzwerk). Erfreulicherweise bringt openQRM mit VMware ESX, Citrix XenServer, KVM, LXC und OpenVZ gleich Unterstützung für fünf Virtualisierungsumgebungen mit. Ein Plug-in für Virtualbox existiert nach Auskunft der Entwickler ebenfalls, wird aber derzeit noch an die aktuelle openQRM-Version angepasst.

Im Bereich Storage kann openQRM mit LVM, iSCSI, NFS, AOE (ATA over Ethernet) Storage, Sanboot Storage und TMPFS Storage umgehen. Für die Netzwerkkonfiguration integriert openQRM wichtige Netzwerkdienste wie DNS, DHCP, TFTP und Wake-on-Lan. Der mitgelieferte Network-Manager hilft bei der Konfiguration der erforderlichen Network-Bridges für diese Dienste.

Der Hybrid Cloud-Connector sorgt für die Einbindung externer Datacenter-Ressourcen, wie der Amazon Webservices,

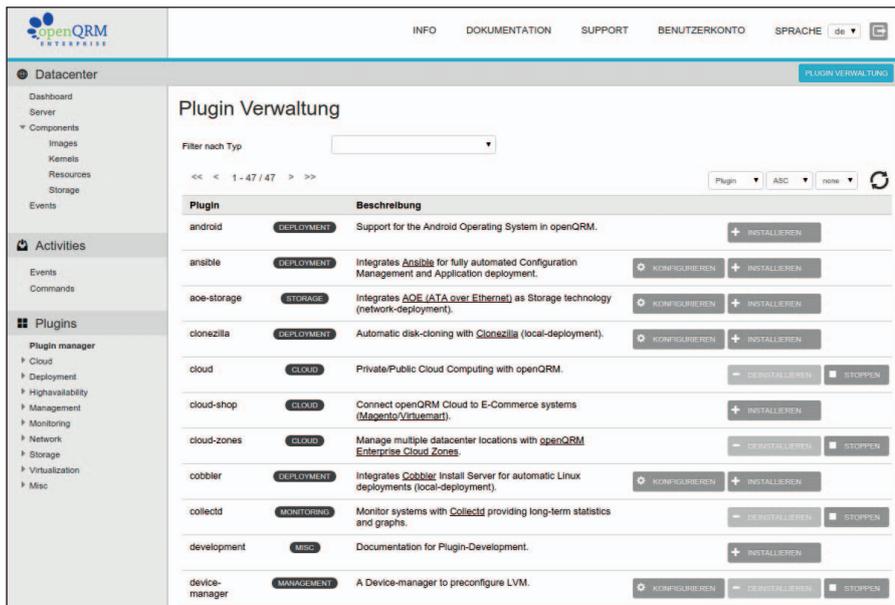


Bild 1: openQRM kommuniziert über zahlreiche Plug-ins mit internen und externen Datacenter-Komponenten. Über 50 Plug-ins werden mitgeliefert und sind sofort einsatzbereit.

Eucalyptus oder OpenStack-Cloud. Dafür werden die APIs des jeweiligen Anbieters verwendet, die über die Plug-in-Architektur an openQRM angedockt werden. Mit dem OpenQRM Cloud-Portal steht schließlich noch eine Webschnittstelle bereit, über die sich interne oder externe Anwender IT-Ressourcen nach Bedarf zusammenstellen können. Bild 2 gibt einen Überblick über die openQRM-Systemarchitektur.

Szenarien und Deployment

Die offene Architektur von openQRM ermöglicht verschiedene Installationsszenarien. Im einfachsten Fall übernimmt openQRM nur die Verwaltung rein lokaler Ressourcen - also das Management einer auf eigener Hardware betriebenen Private Cloud. Dank des Cloud-Connectors lässt sich die Verwaltung im Bedarfsfall aber auch leicht auf externe Ressourcen ausdehnen und bietet dann Management-Funktionen für die Public beziehungsweise Hybrid Cloud.

Über das Cloud-Portal kann openQRM diese Ressourcen mit Hilfe eines fein abgestuften Berechtigungssystems auch Endbenutzern zur Verfügung stellen. Anwender können sich über ein Web-Interface virtuelle Maschinen und Storage also nach Bedarf selbst konfigurieren. Für die Abrechnung dieser Ressourcen bringt das Cloud-Portal ein einfaches Accounting-System mit

einer eigenen virtuellen Cloud-Währung "CCU" (= Cloud Computing Unit) mit. Über eine Anbindung an die Shopsysteme von Magento und Virtuemart kann die virtuelle Währung bei Bedarf auch in echtes Geld umgewandelt werden.

Systemvoraussetzungen und Installation

Die Systemvoraussetzungen hängen stark vom beabsichtigten Einsatzzweck ab. Wer

sich das System nur einmal anschauen möchte, kann es problemlos in einer virtuellen Maschine installieren. Erste Evaluationen unter Praxisbedingungen sind bereits mit einem gut ausgestatteten PC mit Quad-Core-CPU, 1 GByte RAM, einer Netzwerkkarte und 20 GByte freiem Festplattenplatz möglich. Als Minimumkonfiguration für eine kleine Produktivumgebung ohne Hochverfügbarkeit gibt OpenQRM drei physische Systeme (openQRM-Server, Storage-Host, Virtualisierungs-Host) an.

Die Empfehlungen für eine Produktivumgebung mit Hochverfügbarkeit umfassen dann schon sechs (+ n) physische Server (zwei openQRM-Server, zwei Storage-Hosts, zwei Virtualisierungs-Hosts). Als Betriebssystem für den openQRM Server selbst kommt Linux zum Einsatz, der Admin kann zwischen Debian und Ubuntu wählen. Beide Distributionen bringen alle erforderlichen Funktionen out of the box mit. Alternativ bietet OpenQRM aber auch offizielle Unterstützung für die Enterprise-Versionen von Red Hat und SuSE sowie für das kostenfreie CentOS.

Bei unserem Test-Setup haben wir uns für einen physischen Server mit Debian Wheezy entschieden. Während der De-

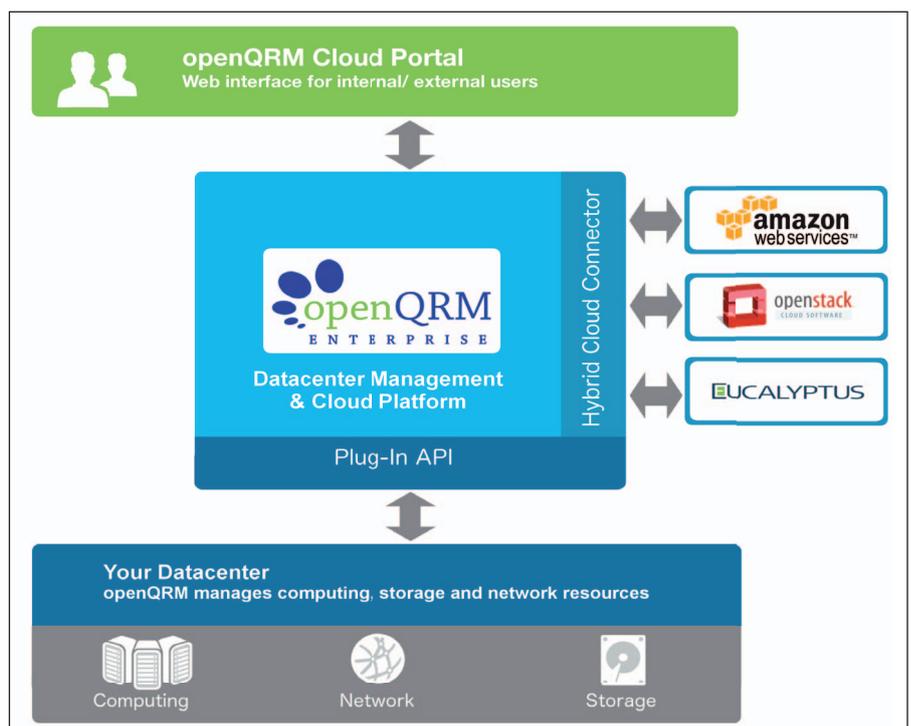


Bild 2: Die Systemarchitektur besteht aus den Core-Komponenten "Datacenter und Management Plattform", "Plug In API" und "Hybrid Cloud Connector".

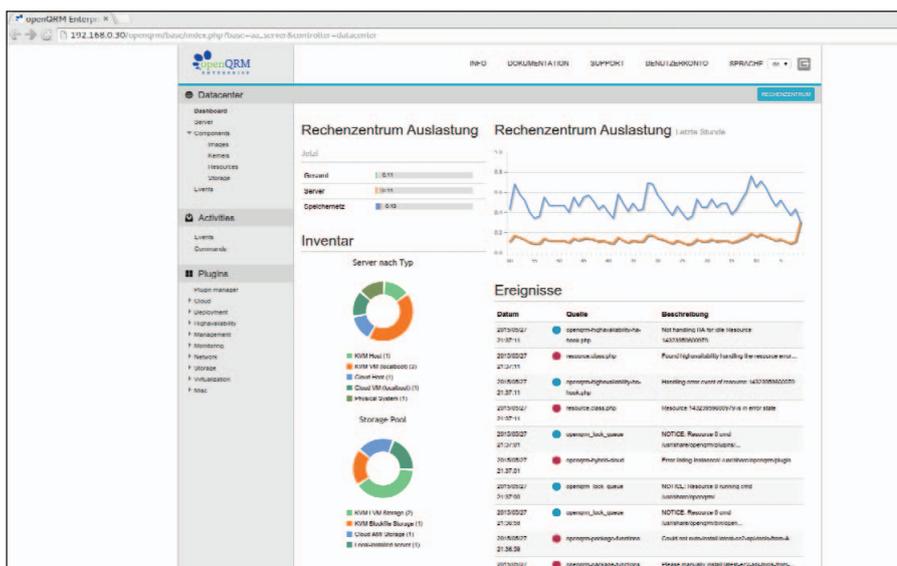


Bild 3: Das Dashboard von openQRM bietet einen guten Überblick über die Auslastung der Datacenter-Ressourcen und Zugriff auf alle wichtigen Funktionen.

ver. Die Installation ist damit bereits abgeschlossen und Sie werden automatisch zum Dashboard von openQRM weitergeleitet (Bild 3).

Überblick und erste Schritte nach der Installation

Sämtliche Arbeiten in openQRM erledigen Sie über das Navigationsmenü auf der linken Seite. Die Funktionen sind dort in drei Hauptbereiche untergliedert: Datacenter, Activities und Plug-in Manager.

Im Bereich "Datacenter" werden Ressourcen wie Server, Images und Storage verwaltet. Der Begriff "Server" repräsentiert dabei keinen Server im klassischen Wortsinn, sondern beschreibt eher ein Master-Objekt beziehungsweise einen Dienst auf der Metaebene, zum Beispiel "Webserver unter Linux virtualisiert unter KVM auf Host 1 mit Netzwerk-Konfiguration xy". Diesem Master Objekt werden nun Komponenten wie Images, Kernels, Ressourcen und Storage zugeordnet. Hintergrund dieser etwas eigenwilligen Abstraktion ist, dass die Sub-Komponenten eines Servers unter OpenQRM jederzeit austauschbar sein sollen.

Unter "Activities" finden Sie Informationen über den Betriebszustand (Events und Commands) und Aktivitäten, die der openQRM-Server gerade ausführt. Über den Plug-in Manager ("Plug Ins") werden die Erweiterungen für openQRM instal-

bian-Installation ist lediglich darauf zu achten, eine statische IP-Adresse zu vergeben und bei der manuellen Partitionierung folgendes Layout anzulegen:

1. Root-Partition
2. Swap-Partition
3. Dedizierte Partition als Speicherplatz für die virtuellen Maschinen

Die dedizierte Partition markieren Sie im Debian-Installer zunächst als "do not use", sie wird erst später im openQRM-Server eingebunden. In der Software-Auswahl des Installers wählen Sie nur den SSH Server aus und führen nach erfolgreicher Installation ein Upgrade aller Pakete durch (`apt-get update && apt-get upgrade`). Installieren Sie dann Apache mit PHP 5:

```
apt-get install apache2 libapache2-mod-php
```

Prüfen Sie anschließend, ob in der Datei `/etc/php5/apache2/php.ini` unter der Direktive "disable_functions" PHP-Funktionen für den Apache-Server deaktiviert wurden. Falls ja, löschen Sie diese. Anschließend starten Sie Apache neu.

Eine Eval-Lizenz beziehungsweise die Vollversion von openQRM Enterprise erhalten Sie unter [1]. Entpacken Sie den Tarball dann im Root-Verzeichnis des Servers. Nach einem Wechsel in das neue Verzeichnis `openQRM-5.2.3-Enterprise-Eval` starten Sie die Installation von

openQRM als root durch Eingabe von `./install-openqrm.sh`.

Der Installer lädt alle erforderlichen Komponenten im Hintergrund herunter und richtet eine Grundkonfiguration ein. Während der Installation werden Sie aufgefordert, ein Root-Passwort für den MySQL-Server und den Nagios-Administrator zu setzen. Damit openQRM später Benachrichtigungen per E-Mail versenden kann, muss außerdem eine funktionsfähige Postfix-Konfiguration erstellt werden, was auch später nachgeholt werden kann.

Die weitere Einrichtung erfolgt dann bereits über die OpenQRM-GUI im Browser. Rufen Sie das Web-GUI unter `http://IP-Adresse/openqrm` auf, loggen Sie sich mit dem Benutzernamen/Passwort "openqrm" ein und bestätigen Sie die Netzwerkkarte, die openQRM verwenden soll (in der Regel eth0). Anschließend legt openQRM die erforderliche MySQL-Datenbank an, dafür wird das während der Installation gesetzte Root-Passwort für MySQL benötigt.

Zum Abschluss ist noch der Upload einer gültigen Lizenz zu erledigen. Die Lizenz besteht aus zwei Teilen: Dem Public-Key und dem eigentlichen Lizenz-File. Laden Sie beide Dateien mithilfe der entsprechenden Buttons im Konfigurationsdialog "Info / Upload License Files" auf den Ser-

openQRM definiert Hochverfügbarkeit auf drei Arten:

1. Hochverfügbarkeit für den openQRM-Server selbst: Da openQRM mit Standardtechnologien (Dateisystem, Apache mit PHP, MySQL/postgreSQL) arbeitet, lässt sich ein HA-System auch mit Linux-Bordmitteln wie `linux-ha`, `pacemaker` oder `corosync` einrichten.
2. Hochverfügbarkeit für Serverobjekte auf dem openQRM-Server: openQRM stellt ein eigenes Plug-in bereit, mit dem sich Server-Objekte einfach per Mausklick hochverfügbar machen lassen. Um die Verteilung auf die vorhandenen Ressourcen kümmert sich openQRM dann selbst.
3. Hochverfügbarkeit für Applikationen: Auch auf Applikationsebene unterstützt openQRM `linux-ha`, `pacemaker` & Co. Für eine einfache Konfiguration hochverfügbarer Applikationen bringt openQRM ein Plug-in für LCMC (Linux Cluster Management Console) mit.

Hochverfügbarkeit



liert und verwaltet. Für eine bessere Übersicht sortiert openQRM die installierten Plug-ins nach Kategorien wie "Cloud", "Deployment", "Monitoring", "Virtualisierung" und so weiter ein.

Sofern Sie den SMTP-Server nicht bereits während der Installation konfiguriert haben, ist jetzt ein guter Zeitpunkt dafür. Ein korrekt konfigurierter SMTP-Server ist Voraussetzung, dass das Plug-in "Event Mailer" Warnungen, Fehlermeldungen und andere Events auch per E-Mail an den Admin versenden kann. Das Cloud-Portal für die Endbenutzer setzt ebenfalls auf dem lokalen SMTP-Server auf. Benutzer, die sich über das Self-Service Portal einen Account generiert haben, erhalten per E-Mail ein Security Token, mit dem sie die Registrierung ihres Accounts abschließen können. Sofern Sie bereits einen Mail- oder Groupware-Server im LAN aktiv haben, wählen Sie im Postfix-Konfigurationsdialog (*dpkg-reconfigure postfix*) einfach "Internet mit Smarthost" und geben dort die IP-Adresse des internen Mailservers als SMTP-Relay-Server an.

Da openQRM sich selbst als zentrale Verwaltungsinstanz im Netzwerk definiert, stellt es grundlegende Dienste wie DNS und DHCP auch selbst bereit. Da diese Dienste aber im Regelfall in einem LAN bereits vorhanden sind (zum Beispiel über ein Active Directory), ist bei der Inbetriebnahme eines openQRM-Servers durchaus Vorsicht geboten. Um nicht zu riskieren, dass auf einmal zwei DHCP-Server im LAN aktiv sind, die unabhängig voneinander IP-Adressen des gleichen Subnetzes im LAN verteilen, sollten Sie im Zweifelsfall den openQRM-Server mit einem eigenen VLAN beziehungsweise Subnetz ausstatten.

Sind diese Voraussetzungen erfüllt, passen Sie mit dem Editor Ihrer Wahl noch den Domain-Namen `OPENQRM_SERVER_DOMAIN="oqnet.org"` in der Datei `/usr/share/openqrm/plugins/dns/etc/openqrm-plugin-dns.conf` an und installieren respektive starten dann im Plug-in Manager die Plugins `dhcpd`, `dns` und `tftpd`. Per Default verteilt der openQRM DHCP-Server die komplette IP-Range des lokalen Subnets. Das lässt sich leicht

ändern, indem Sie den Bereich in der Datei `/usr/share/openqrm/plugins/dhcpd/etc/dhcpd.conf` anpassen. OpenQRM stellt in Kombination mit dem DHCP-Server auch eine PXE/TFTP-Umgebung für das Booten übers Netz bereit. Installieren Sie mit Hilfe des Plug-in Managers daher am besten auch gleich das `tftpd`-Plug-in.

Virtualisierung mit KVM

Mit der Kernel-based Virtual Machine (KVM) bringt openQRM bereits einen sehr leistungsstarken Hypervisor mit, auf dessen Basis wir nun exemplarisch erste virtuelle Maschinen bereitstellen wollen. Bitte beachten Sie, dass für den Betrieb von Windows in einer virtuellen Maschine unter KVM auf dem Hostrechner die Intel VT / VT-x Erweiterung der CPU im BIOS aktiviert werden muss. Sofern noch nicht geschehen, installieren Sie jetzt über den Plug In Manager die folgenden Plug-ins: `dns`, `dhcpd`, `tftpd`, `network-manager`, `local-server`, `device-manager`, `novnc`, `sshterm`, `linuxcoe`, `kvm`.

Die Plug-ins sind zum Teil voneinander abhängig. Achten Sie bei der Installation daher auf kleine gelbe Popups. Diese weisen im Dashboard eher dezent auf unerfüllte Abhängigkeiten hin, installieren Sie die gewünschten Plug-ins dann einfach in der geforderten Reihenfolge.

Network-Bridge einrichten

Damit openQRM für die virtuellen Maschinen und andere interne sowie externe Ressourcen später das Netzwerk-Management (DNS/DHCP) übernehmen kann, müssen Sie auf der physischen Netzwerkkarte des Host-Systems noch eine Network-Bridge einrichten. Dazu klicken Sie beim openQRM-Server-Object auf "Edit" und wählen dort unter Plug-ins "Network Manager / Add new network bridge". Im folgenden Dialog vergeben Sie einen Namen für die neue Bridge (`br0`) und tragen unter IP-Adresse die statische IP ein, die Sie dem physischen Server während der Installation gegeben haben (in unserem Szenario `192.168.0.30` auf der Schnittstelle `eth0`). Entsprechend ergänzen Sie dann noch die Felder Subnet (`255.255.255.0`) und Gateway (`192.168.0.254`). OpenQRM richtet im Hintergrund die neue Netzwerkbrücke ein

und installiert auch die dafür erforderlichen Debian-Pakete.

Tipp: Wenn das System bei der Einrichtung der Netzwerkbrücke länger hängen bleibt, liegt das eventuell an einer fehlerhaften Namensauflösung des Servers. Prüfen Sie in diesem Fall, ob in der Datei `/usr/share/openqrm/plugins/dhcpd/etc/dhcpd.conf` im Abschnitt

```
option domain-name-servers
    192.168.0.253 192.168.0.254;
```

der oder die korrekten Nameserver eingetragen sind und korrigieren Sie diese Einstellung gegebenenfalls. Sie können auch verhindern, dass openQRM die Namensauflösung auf Ihrem System selbständig anpasst, indem Sie in der Datei `/usr/share/enqrm/plugins/dns/etc/openqrm-plugin-dns.conf` folgenden Wert auf "no" setzen:

```
OPENQRM_SERVER_ADJUST_RESOLV_CONF="no"
```

LVM-Storage für VMs

Gegenüber dem Speichertyp "Blockfile Storage" bietet der Typ "KVM LVM Storage" eine höhere Flexibilität (Snapshots, Cloning). Wir ordnen daher nun die bei der Debian-Installation vorbereitete (bisher aber nicht zugeordnete) Partition als LVM Storage für die VMs zu. Jede VM erhält später ihr eigenes logisches Laufwerk aus dem LVM-Storage Pool. Auch bei der Aktivierung des KVM LVM Storage kümmert sich openQRM wieder um die Einrichtung der erforderlichen Debian-LVM-Pakete im Hintergrund. Klicken Sie beim Objekt "Storage / KVM LVM Storage" auf "Manage" und dann auf "Add new Volume group". Wählen Sie aus dem Drop-down-Menü die dafür vorgesehene Partition (zum Beispiel `/dev/sda3`), vergeben Sie einen Namen (etwa "kvmg") und bestätigen Sie mit "Submit".

Im nächsten Schritt richten Sie eine virtuelle Maschine unter KVM ein. Prinzipiell verläuft die Installation nach dem Anlegen der VM wie unter anderen Virtualisierungssystemen. Benötigt wird ein Installationsmedium mit dem Betriebssystem - typischerweise ein ISO-Image. Dieses Image binden Sie vor dem Start



The screenshot shows a web form titled "Add Logical Volume to Volume group kvmvg". At the top, there are navigation buttons: "ADD LOGICAL VOLUME", "EDIT VOLUME GROUP", "SELECT VOLUME GROUP", and "SELECT KVM HOST". On the right, there are buttons for "ADD SERVER (3/4)", "SERVER", "ADD IMAGE", and "IMAGES". The form itself has three main input fields: "Name *" with the value "app6992613942" and a "GENERATE NAME" button; "Volume Type" with a dropdown menu showing "lvm2"; and "Size (max. 178120 MB) *" with the value "2000". At the bottom of the form are "SUBMIT" and "CANCEL" buttons.

Bild 4: Die Verwaltung der logischen Volumes lässt sich in openQRM erledigen.

der VM als primäres Boot-Medium ein; die Installation startet dann direkt von diesem virtuellen CD-Laufwerk.

Tipp: openQRM verfügt über ein Plugin für Linux COE (Linux Common Operating Environment) [2]. Linux COE ist ein Open Source-Tool von HP, das ISO-Images für CentOS, Debian/Ubuntu, Fedora und viele andere Linux-Distributionen erzeugt. Das Besondere dabei ist, dass es auch die Auto-Installationsmechanismen dieser Distributionen unterstützt. Auf diese Weise lassen sich sehr leicht vollautomatisch ablaufende Installations-ISOs erstellen. Wie Sie Linux COE innerhalb von openQRM verwenden, erläutert das ausführliche Tutorial auf der openQRM-Website unter [3]. Wenn Sie nicht mit dem COE-Plug-in arbeiten wollen, kopieren Sie stattdessen die gewünschte Linux-Distribution als ISO-Image in ein Verzeichnis auf den openQRM-Server.

Server-Host-Objekt anlegen

Legen Sie jetzt unter "Server / Add Server" ein neues Server-Objekt an. Damit das neue Objekt den Namenskonventionen von openQRM entspricht, klicken Sie auf den Button "generate name" und vergeben eine aussagekräftige Beschreibung im Kommentarfeld. Übernehmen Sie die Voreinstellungen "openQRM Server (Physical system)" im Folgefenster und klicken Sie auf "Submit". In der Liste mit der Serverübersicht editieren Sie nun das soeben angelegte Serverobjekt, indem Sie aus dem Dropdown-Menü "KVM Host" als Virtualisierungstyp auswählen. Anschließend übernehmen Sie die Einstellung mit "Submit".

Auch die virtuelle Maschine benötigt ein Server-Objekt. Sie legen es wie zuvor un-

ter "Server / Add Server" an. Wählen Sie dann im nächsten Fenster "New Resource" und als Typ "KVM (localboot) Virtual Machine". Sie werden automatisch zur Auswahl des KVM-Host-Servers weitergeleitet. Dort klicken Sie auf "VMs" und fügen mit "ADD LOCALVM" eine neue virtuelle Maschine hinzu. Stellen Sie die gewünschten Parameter wie Anzahl der CPUs und Memory ein und achten Sie darauf, dass die erste Netzwerkkarte (Network_0) mit der Network-Bridge (br0) verbunden ist.

In der Sektion "Boot from" wählen sie "ISO" und navigieren mit dem Browse-Button zu dem Verzeichnis, in dem Sie zuvor das ISO-Image abgelegt haben (in unserem Beispiel ist das /root). Vergeben Sie ein VNC-Passwort, damit Sie später über NoVNC auf die VM zugreifen und die Betriebssysteminstallation anstoßen können. Schließen Sie diesen Schritt wieder mit "Submit" ab. Etwas verwirrend ist, dass Sie nun wieder im Wizard landen und die soeben erzeugte Ressource nochmals mit "Submit" bestätigen müssen.

Image und logisches Laufwerk erzeugen

Ist auch dieser Schritt erledigt, erscheint im Wizard der Button "New Image". Als Image bezeichnet openQRM ein logisches Objekt, das später dem physischen Storage zugewiesen wird. Klicken Sie auf "New Image / LVM deployment for KVM" und wählen Sie in der KVM-Host-Auswahl "LVM" aus. Es erscheint die Volume Group "kvmvg", die Sie zuvor als LVM Storage angelegt haben (Bild 4). Mit "Select / Add new logical volume" fügen Sie ein neues logisches Volume hinzu. Legen Sie die Größe des logischen Volumes in MBytes fest und übernehmen

Sie die restlichen Voreinstellungen. Mit einem Klick auf "Submit" wird das Image daraufhin im Bearbeiten-Modus geöffnet. Legen Sie hier den Betriebssystem-Typ (Windows, Linux oder Other) fest. Im Fall einer automatischen Installation mit Linux COE könnten Sie dem Installer hier auch gleich das Root-Passwort mitgeben, dieser Schritt entfällt natürlich in unserem manuellen Installations-Szenario. "Submit" bringt Sie wieder zurück zur Serverliste, an deren Ende Sie Ihre neue VM finden. Ein Klick auf den Start-Button erfüllt die VM jetzt mit Leben.

Betriebssystem installieren

Nun sind Sie bereit, um das Betriebssystem in der VM zu installieren. Während des Startvorgangs hat die VM bereits das Installations-ISO gebootet und wartet jetzt auf Tastatureingaben. Um die Installation fortzusetzen, greifen Sie über den "Novnc"-Button auf die VM zu, wofür Sie das zuvor gesetzte VNC-Passwort benötigen. Hat alles geklappt, sehen Sie jetzt den Start-Screen des Debian-Installers und führen die Installation des Betriebssystems wie gewohnt durch.

Nach Abschluß der Installation des neuen Betriebssystems in der VM müssen Sie jetzt noch das ISO-Image als Bootmedium entfernen und die VM auf "local boot" umstellen. Rufen Sie dafür die Serverübersicht im Dashboard auf und halten Sie den soeben eingerichteten Server mit dem "Stop"-Button an. Rufen Sie die Übersicht Ihrer VMs über den Dialog "Plugin Manager / Virtualization / KVM / Hosts" auf, dort wird die dem Server-Objekt zugeordnete VM jetzt mit dem Status "Idle" angezeigt. Mit dem "Update"-Button gelangen Sie wieder in den Einstellungsdialog für die VM und können dort die Bootreihenfolge von "ISO" auf "Local" ändern. Beim nächsten Start bootet der Server von seinem lokalen Bootmedium.

Lokale VMs ins System Management einbinden

Abschließend sollten Sie nun noch den openQRM-Client in der neuen VM installieren. Er stellt eine Schnittstelle zwischen dem Betriebssystem in der VM und dem openQRM-Host-Server zur Verfü-



gung. Auf diese Weise können lokale VMs problemlos in das System-Management von openQRM eingebunden und so beispielsweise das Monitoring der VM per Nagios, das Konfigurationsmanagement mit Puppet oder der Shell-Zugriff per SSH auf die virtuelle Maschine aktiviert werden. Laden Sie den openQRM-Client unter "Plugins / Misc / Local-server / Local VMs" herunter und kopieren Sie ihn in die virtuelle Maschine:

```
scp openqrm-local-vm-client
root@IP-der-VM:/root
```

Zur Installation des Clients loggen Sie sich per SSH in die virtuelle Maschine ein, wechseln ins Verzeichnis `/root` und geben dem Client Ausführungsrechte:

```
chmod +x openqrm-local-vm-client
```

Die Installation starten Sie dann durch Eingabe von `./openqrm-local-vm-client`.

Fehlende Pakete werden wieder im Hintergrund automatisch heruntergeladen und installiert. Nach Abschluss der Client-Installation können Sie beispielsweise mit SSH-Term (Secure Shell im Browserfenster) auf Linux-VMs zugreifen oder die VMs ins Systemmanagement mit Nagios aufnehmen. Dazu klicken Sie in der Serverliste im Dashboard auf die entsprechenden Buttons mit dem Pluszeichen.

Tipp: Auf die gleiche Weise können Sie auch andere bereits im lokalen Netzwerk vorhandene physische oder virtuelle (Linux) Server in das openQRM-Management aufnehmen. Sie erscheinen dann als neue Ressource im Dashboard und können zum Beispiel als zusätzliche Storage-Server verwendet werden. Kopieren Sie dafür den Client unter `/usr/share/openqrm/plugins/local-server/bin/openqrm-local-server` auf den bestehenden Linux-Host und führen Sie das Programm aus:

```
scp /usr/share/openqrm/plugins/
local-server/bin/openqrm-local-
server root@VM:/tmp
ssh VM /tmp/openqrm-local-server
integrate -u openqrm -p openqrm -q
192.168.0.30 -i eth0 [-s
http/https]
```

Die IP-Adresse 192.168.0.30 gehört hier dem openQRM-Server.

Windows virtualisieren unter KVM

Natürlich können Sie in einer virtuellen KVM-Maschine auch Windows installieren. Dafür muss am physischen openQRM-Host im BIOS die Virtualisierungserweiterung (VT-X) aktiviert sein. Haben Sie beim Anlegen der VM die Default-Einstellung "virtio" übernommen, benötigt Windows zudem weitere Treiber für den VirtIO-SCSI-Controller, damit es während der Installation einen lokalen Datenträger erkennt.

Der Treiber befindet sich zusammen mit dem ebenfalls erforderlichen Treiber für den VirtIO-Ethernet-Adapter auf einer Treibersammlung des Fedora-Projekts. Laden Sie das neueste ISO der Sammlung von [4] herunter und binden Sie es in Ihre VM-Konfiguration als zweites CD-ROM-Laufwerk ein. Die Windows VM bootet dann vom ersten CD-ROM Laufwerk, vom zweiten laden Sie die Treiber, wenn diese vom Windows Setup angefordert werden.

Ist Windows in der virtuellen Maschine installiert, sollten Sie unbedingt auch das openQRM Enterprise Windows Plug-in installieren. Es enthält die erforderlichen Werkzeuge, um das Windows-Betriebssystem für Clone- und Snapshot-Aktionen in openQRM vorzubereiten. Diese mit Sysprep erstellten Windows Master-Images lassen sich dann beliebig oft vervielfältigen und zum Beispiel für Cloud-Deployments einsetzen.

Nach der Installation des Windows-Plugins über den Plug-in-Manager finden Sie den Client `openqrm-client.exe` zum Download unter "Plug In / Deployment / Windows". Am einfachsten rufen Sie das openQRM Webinterface gleich mit dem Browser in der virtuellen Windows-Maschine auf. Dazu verbinden Sie sich wieder über das NoVNC-Plug-in in die VM. Speichern Sie den Client im Wurzelverzeichnis `C:\` und rufen Sie den Gruppenrichtlinienditor `gpedit.msc` auf. Dort fügen Sie die `openqrm-client.exe`-Datei jetzt als Start-Skript hinzu: "Richtlinien für Lokaler

Computer / Computerkonfiguration / Softwareeinstellungen / Windowseinstellungen / Skripts (Start/Herunterfahren) / Starten". Als Start-Parameter geben Sie wiederum die IP-Adresse des openQRM Servers an.

Sysprep standardisiert Images

Damit das so erstellte Windows-Image später auch automatisiert für andere VMs zur Verfügung gestellt werden kann, erzeugen Sie mit `sysprep.exe` nun noch ein standardisiertes Windows-Image. Dazu laden Sie die Konfigurationsdatei `unattend.xml` unter "Plug In / Deployment / Windows" auf den Desktop des Administrators herunter und tragen in die "AdministratorPassword / Value Sektion" ein komplexes Passwort ein. Dieses Passwort ist das spätere Administrator-Passwort der automatischen Windows-Installation.

Bevor Sie Sysprep starten, sollten Sie allerdings die Festplatte der VM defragmentieren. Verwenden Sie dazu das Windows-Bordmittel `defrag.exe` und öffnen Sie nach Abschluß der Defragmentierung eine Windows Command Shell mit Administrator-Rechten. In der Shell führen Sie nun den folgenden Befehl aus:

```
C:\windows\system32\sysprep\
sysprep.exe /generalize /oobe
/shutdown /unattend:C:\Users\
Administrator\Desktop\unattend.xml
```

Nach Abschluß der Sysprep-Aktion fährt das Windows System herunter und das Windows-Image ist bereit zum Clonen beziehungsweise Snapshot, speziell für weitere Cloud Deployments.

Cloud-Frontend für die Benutzer einrichten

Das openQRM Cloud Plug-in stellt einen vollautomatisierten Serverprovisionierungs-Workflow mit einem eigenen Webportal zur Verfügung. Externe Benutzer können sich an diesem Portal entweder per Self-Service registrieren oder werden vom Administrator manuell hinzugefügt. In Abhängigkeit von ihrer Gruppenzugehörigkeit können die Benutzer dann Cloud-Bestellungen aufgeben. Je nach Konfiguration lassen sich diese Bestellungen entweder manuell oder auch auto-



matisch freigeben – openQRM kümmert sich dann selbständig um den kompletten Provisionierungsablauf.

Die Einrichtung eines Cloud-Frontends ist in einem How-To unter [5] beschrieben. Die grundlegende Einrichtung des Dienstes erfolgt im Plug-in Manager unter "Cloud / Configuration / Main Config". Dort müssen Sie mindestens die folgenden Einstellungen vornehmen:

```
cloud_admin_email - E-Mail Adresse
                   des openQRM Administrators
auto_provision - true
show_private_image - true
```

Optional passen Sie bei dieser Gelegenheit gleich noch die folgenden Parameter an:

- external_portal_url: Hier legen Sie einen externen DNS/Domainnamen für das Cloud Portal fest.
- cloud_currency: Wählen Sie, ob die Cloud-Währung für die Benutzer in US-Dollar oder Euro dargestellt werden soll.
- cloud_1000_ccus: Zuordnung der virtuellen Cloud-Währung CCU zu realem Geld.
- vm_loadbalance_algorithm: Loadbalancing-Algorithmus für virtuelle Maschinen, zum Beispiel 0 = Load, 1 = Memory, 2 = Random, 3 = Erster verfügbarer Host, bis das VM-Limit erreicht wurde.

In der weiteren Konfiguration erstellen Sie dann unter anderem die Cloud-Produkte, die die Benutzer später mit ihren Zugangsdaten buchen können, und legen die Benutzergruppen, Preise beziehungsweise den Abrechnungsmodus für die über das Cloud Portal gebuchten Ressourcen fest. Die Endbenutzer erreichen das Cloud-Portal über <http://IP-Adresse-des-Servers/cloud-portal/>.

Backup und Restore der OpenQRM Umgebung

"Keep it simple" ist auch beim Backup und Restore einer openQRM-Umgebung die Maxime. Da eine openQRM-Installation lediglich aus Dateien und einer Datenbank besteht, ist es ausreichend, diese beiden Komponenten für den Disaster-Fall zu sichern. Prinzipiell genügt es also, alle Dateien unter `/usr/share/openqrm` zu-

sammen mit einem Dump der MySQL-Datenbank von openQRM an einen sicheren Ort zu kopieren. Dies erledigen die folgenden, mit Root-Rechten ausgeführten Befehle:

```
cp -aR /usr/share/openqrm /usr/share/openqrm-5.2.before-update
mysqldump openqrm -p >
  /usr/share/openqrm-5.2.before-update/openqrm-db-5.2.sql
```

Für den MySQL-Dump wird das Root-Passwort für die Datenbank interaktiv abgefragt. Der Restore verläuft dann einfach durch Verschieben des Originalverzeichnis und Zurückkopieren der zuvor angelegten Sicherungsdateien beziehungsweise des Datenbank-Dumps mit abschließendem Neustart der openQRM-Dienste.

Noch einfacher geht es allerdings mit dem "state-backup"-Feature, das alle Konfigurationsdateien und den Inhalt der Datenbank mit einem einzigen Befehl automatisch sichert beziehungsweise wieder einspielt:

```
/usr/share/openqrm/bin/openqrm state
backup -n 5.2.before-update
```

Der folgende Befehl listet alle auf diese Weise angefertigten Backups auf:

```
/usr/share/openqrm/bin/openqrm state
list
5.2.3.before-update-05-06-
15_09.14.20
```

Test-State-Backup-05-24-15_13.09.05

Um nun beispielsweise die Systemkonfiguration aus der Sicherung "Test-State-Backup-05-24-15_13.09.05" wiederherzustellen, genügt der folgende Befehl:

```
/usr/share/openqrm/bin/openqrm state
restore -n Test-State-Backup-05-
24-15_13.09.05
```

```
openQRM-server: Restoring server
state from /var/spool/openqrm-
state/Test-State-Backup-05-24-
15_13.09.05.tgz
```

```
Restoring the openQRM-database from
/usr/share/openqrm/database.sql
```

Verbindung zur Amazon Public Cloud aufnehmen

Die Amazon Web Services (AWS) bestehen aus Diensten, die etwa virtuelle Server (EC2), Speicherplatz (S3) und Netzwerk (CloudFront, VPC) bereitstellen, die über ein Webinterface administriert und nach Nutzung pro Stunde oder Ressource abgerechnet werden. Neben dem Webfrontend von AWS gibt es auch eine Programmierschnittstelle, die die Steuerung der AWS-Dienste über ein Commandline-Interface ermöglicht. Das openQRM-Plug-in "Hybrid Cloud" kann ein bestehendes AWS-Konto über diese Schnittstelle einbinden und steuern. Auch für dieses Szenario hält die openQRM-Website unter [6] ein ausführliches HowTo bereit, wir geben daher nachfolgend nur einen kurzen Überblick über die grundsätzlich erforderlichen Schritte.

Nach der Installation des Hybrid-Cloud Plug-ins müssen zunächst die aktuellen API- und AMI-Tools auf dem openQRM Server installiert werden. Dazu loggen Sie sich als root auf dem openQRM Server ein und führen die folgenden Befehle aus:

```
wget s3.amazonaws.com/ec2-down-
loads/ec2-api-tools.zip
wget s3.amazonaws.com/
ec2-downloads/ec2-ami-tools.zip
unzip ec2-api-tools.zip
unzip ec2-ami-tools.zip
mkdir /usr/local/ec2
cp -r ec2-ami-tools-1.5.7/* /usr/
local/ec2/
cp -r ec2-api-tools-1.7.3.2/*
/usr/local/ec2/
apt-get update && apt-get install
default-jdk
```

Eventuell weichen die Versionsnummern bei Ihnen ab, die obigen waren zum Redaktionsschluss aktuell. Den Link zum Download der AMI/API-Tools finden Sie übrigens auch unter "Cloud / Hybrid-Cloud / About / Plugin Typ". Der dort angegebene Link führt zu umfangreicher Dokumentation der Tools. Fügen Sie die beiden Programme jetzt noch Ihrem Systempfad hinzu, indem Sie die folgenden Umgebungsvariablen in die Datei `/etc/profile` eintragen:

```
export EC2_HOME=/usr/local/ec2
```



```
export PATH=$PATH:$EC2_HOME/bin
export JAVA_HOME=/usr
```

Um die Umgebungsvariablen zu aktivieren, loggen Sie sich einmal aus und wieder ein. Starten Sie danach mit `service openqrm restart` die openQRM-Dienste neu.

Die Lizenz der Enterprise-Edition von openQRM basiert auf dem "Perpetual"-Modell, Lizenzen lassen sich also dauerhaft (das heißt auch nach Ablauf der Maintenance) nutzen. Eine Server-Lizenz besteht immer aus einer Lizenz für den openQRM-Host und zwei Client-Lizenzen für das Management je eines Clients (= physische Hosts). Im Preis ist Maintenance (Updates und Bugfixes) und Support für 12 Monate enthalten. Client-Lizenzen zur Integration weiterer Hosts können jederzeit zusätzlich erworben werden. Derzeit wird openQRM Enterprise in den folgenden Editionen angeboten:

- SMB Edition Datacenter and IaaS MultiCloud 5.2, für kleinere und mittlere Unternehmen mit einem oder mehreren Systemadministratoren, 2000 Euro.

- Large Edition Datacenter and IaaS MultiCloud 5.2, für Organisationen mit einem Team von Systemadministratoren, mit zentraler Benutzerverwaltung (LDAP/AD), 4000 Euro.

- Enterprise Edition Datacenter and IaaS MultiCloud 5.2, für Unternehmen mit einem oder mehreren Datacentern, einem Team von Systemadministratoren und mit Anbindung an E-Commerce-Plattformen, mit zentraler Benutzerverwaltung (LDAP/AD), 7000 Euro.

Im deutschsprachigen Raum bietet openQRM darüber hinaus auch ein Hardware-Bundle mit vorinstallierter Supermicro-Hardware und Einrichtungssupport an. Details zu den Bundles finden Sie unter [7].

Lizenzierung und Preise



Am besten legen Sie in Ihrem AWS-Account im Bereich IAM einen eigenen Benutzer für openQRM an. Anschließend testen Sie mit dem folgenden Befehl auf der Kommandozeile, ob die Verbindung zu AWS klappt:

ec2-describe-regions

```
-O AWS-Access-Key -w
AWS-Secret-Key
```

Tragen Sie anstelle der Platzhalter Ihren AWS Access Key und Ihren Secret Key ein. Das Ergebnis sollte dann in etwa wie folgt aussehen:

```
REGION    eu-central-1    ec2.eu-
           central-1.amazonaws.com
```

```
REGION    sa-east-1      ec2.sa-east-
           1.amazonaws.com
```

```
REGION    ap-northeast-1 ec2.ap-
           northeast-1.amazonaws.com
```

...

Sie sind nun prinzipiell startbereit, um Instanzen in der Amazon Cloud anzulegen und zu verwalten.

Fazit

Um ein openQRM-Projekt in einem überschaubaren Zeitrahmen an den Start zu bekommen, sollte der Admin entweder selbst über solide Grundkenntnisse im Bereich Systemarchitektur und Open Source-Werkzeuge verfügen oder besser gleich die professionelle Unterstützung des Herstellers für Planung, Installation und

Konfiguration in Anspruch nehmen. Denn die Bedienlogik von openQRM erschließt sich (wie bei den meisten Cloud-Managementsystemen) leider nicht intuitiv. Teilweise erschwert fehlende oder veraltete Dokumentation die Einarbeitung noch zusätzlich.

Open Source- und Linux-affine Administratoren werden an openQRM aber ihre Freude haben. Denn dank der offenen, auf Plug-ins basierenden Systemarchitektur stehen viele bekannte und populäre Systemwerkzeuge aus der Linux-Welt ohne größeren Konfigurationsaufwand zur Verfügung. Darüber hinaus bietet der deutsche Hersteller mit Sitz in Bonn ein äußerst günstiges Lizenzmodell sowie kompetenten und deutschsprachigen Support. (of) 

[1] OpenQRM Enterprise Evaluation
F8P51

[2] Linux COE
F8P52

[3] Tutorial: OpenQRM auf Wheezy
F8P53

[4] Virtio für Windows
F8P54

[5] Cloud Computing mit OpenQRM
F8P55

[6] Amazon AWS mit OpenQRM
F8P56

[7] OpenQRM Hardware-Bundle
F8P57

Link-Codes

