

Middleware - Cloud Computing – Übung

Tobias Distler, Klaus Stengel,
Timo Hönig, Christopher Eibel

Friedrich-Alexander-Universität Erlangen-Nürnberg
Lehrstuhl Informatik 4 (Verteilte Systeme und Betriebssysteme)

www4.cs.fau.de

Wintersemester 2015/16



Cloud-Computing-Infrastruktur

Physikalische Infrastruktur

Software-Infrastruktur

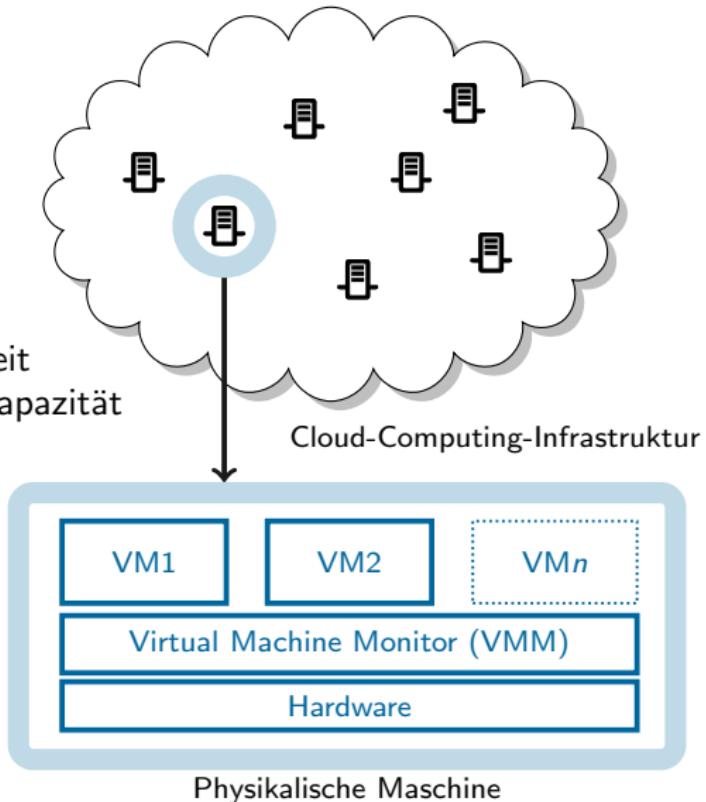
Aufbereiten des Abbilds

Betrieb der virtuellen Maschine



Physikalische Infrastruktur

- Standard-Hardware
 - „commodity hardware“
 - billig
 - **fehleranfällig**
- Charakteristik
 - hoher Grad der Verteiltheit
 - „unbegrenzte“ Speicherkapazität



- **Glance:** Bereitstellung von Abbildern
 - Registry: Metadaten für Images
 - API unterstützt verschiedene Speichersysteme
- **Cinder:** Bereitstellung von Volumes
 - Volume-Service: Lokale Datenhaltung
 - Scheduler: Verteilung der Daten(-transfers) auf Rechner
- **Nova:** Verwaltung virtueller Maschinen
 - Compute: Steuerung von VMs (QEMU/Xen/...) auf Rechnern
 - Scheduler: Verteilung auf verfügbare Hardware
- **Neutron:** Netzwerkmanagement und virtuelle Router
 - Server: Steuerung und Zustandsverwaltung
 - Agents: Helfer für DHCP, Open vSwitch, Metadaten
- **Horizon** (Dashboard): Weboberfläche für Anwender



- API-Dienst je Komponente für REST-Anwenderschnittstelle
→ Kommandozeilentools
- Kommunikation der Dienste intern via RabbitMQ
 - Gruppenkommunikation über Nachrichtenbus
 - Standardisiertes Protokoll: AMQP
- Speicher-Backend für Glance und Cinder: **Ceph**
 - Block-Storage oder Dateisystem verteilt auf Rechner-Cluster
 - Flexible Konfiguration von Replikationseigenschaften
 - Transaktionen über Paxos-Protokoll



Installation der Java-Anwendung

- Automatisches Starten der Dienste
- Beim Systemstart führt init(8) die Init-Skripte aus
 - Kopieren der Projektdateien nach /opt/mwcc
 - Modifikation von /etc/rc.local

```
JAVA="/usr/bin/java"

cd /opt/mwcc
$JAVA -cp <classpath> mw.cache.MWCache <parameter> &
$JAVA -cp <classpath> mw.path.MWPathServer <parameter> &
$JAVA -cp <classpath> mw.MWRegistryAccess register <parameter>
```

- Hilfestellung und Debugging
 - Hinzufügen von echo-Anweisungen
 - Ausführen von /etc/rc.local in der chroot-Umgebung
 - Starten von /etc/rc.local mit bash -x



Shell Quoting

■ Doppelte Anführungszeichen

- Übergabe als einzelnes Argument
- Expansion von Variablen möglich, „Escaping“ mittels Backslash (\)

```
$ argc() { echo "$@ has $# args"; }  
$ argc Mein Home ist \"${HOME}\"  
Mein Home ist "/home/cloud" has 4 args  
$ argc "Mein Home ist \"${HOME}\""  
Mein Home ist "/home/cloud" has 1 args
```

■ Einfache Anführungszeichen (keine Expansion, kein Escaping):

```
$ argc 'Mein Home ist \"${HOME}\"'  
Mein Home ist \"${HOME}\" has 1 args
```

■ „Backticks“ (Gravis)

- Führt Inhalt als Befehl aus
- Ausgaben werden als Argument eingesetzt

```
$ argc Es ist `date`  
Es ist Fri Nov 11 11:11:00 CET 2011 has 8 args  
$ argc "Es ist `date`"  
Es ist Fri Nov 11 11:11:00 CET 2011 has 1 args
```



- Speicherarten
 - Volume: Änderungen persistent, nur in einer Instanz
 - Image(Abbild): Änderungen flüchtig, Basis für viele Instanzen
- Abbild innerhalb von OpenStack erzeugen
 - Starten einer Grml-Instanz
 - Neues Volume anlegen und einhängen
 - Befüllen mit Daten (Betriebssystem)
 - Umwandeln in Image
- Web-Frontend
 - Dashboard: <http://i4cloud.informatik.uni-erlangen.de>
 - Zugangsdaten: siehe E-Mail mit SVN-Zugangsdaten
- Kommandozeile
 - OpenStack-Client-Programme: nova, cinder, glance, neutron
 - **Vor Verwendung:** openrc-Datei sourcen



Grml-Instanz starten

Launch Instance

Details Access & Security Networking

Availability Zone:
nova

Instance Name: **grml-instance**

Flavor: **m1.tiny**

Instance Count: 1

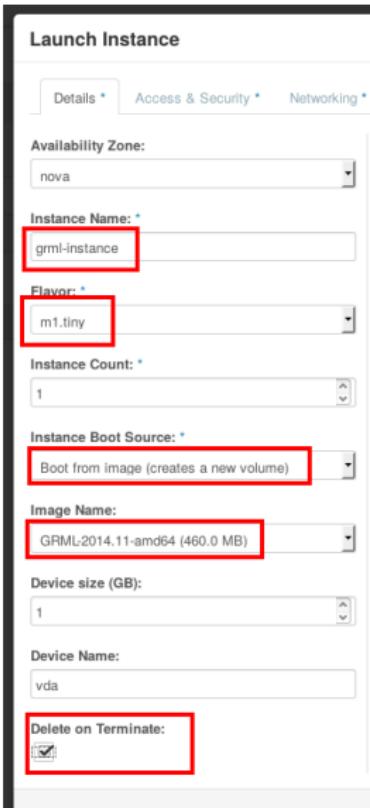
Instance Boot Source: Boot from image (creates a new volume)

Image Name: GRML-2014.11-amd64 (460.0 MB)

Device size (GB): 1

Device Name: vda

Delete on Terminate:



- Name für Instanz festlegen
- Instanztyp **m1.tiny**
 - Kein Swap/Ephemeral-Volume
- Automatisch Volume erzeugen
 - Ansonsten keine Zuweisung von Volumes
 - Volume beim Beenden löschen
- Kommandozeile:

```
$ nova image-list      # --> grml id  
$ nova network-list   # --> internal net id  
$ nova boot --flavor m1.tiny \  
    --nic net-id=<internal net id> \  
    --block-device id=<grml id>,\  
    source=image,dest=volume,size=1,\  
    shutdown=remove,bootindex=0 \  
    grml-instance
```

Achtung: Jeweils kein Leerzeichen nach dem Komma vor „source=im...“ und „shutdown=re...“ setzen.



Volume erzeugen/einhängen

Volumes & Snapshots

The screenshot shows the 'Volumes' section of the OpenStack dashboard. At the top, there are two tabs: 'Volumes' (selected) and 'Volume Snapshots'. Below the tabs is a search bar with a 'Filter' button and a red box labeled '1' highlighting the '+ Create Volume' button. To the right of the create button is a 'Delete Volumes' button. The main area displays a table of volumes:

	Name	Description	Size	Status	Type	Attached To	Availability Zone	Actions
<input type="checkbox"/>	26929a23-5e7e-4b90-a1e6-acf02d634501		1GB	In-Use	-	Attached to grml on vda	nova	<button>Edit Volume</button> <button>More</button>
<input type="checkbox"/>	retest		2GB	Available	-		nova	<button>Edit Volume</button> <button>More</button>
<input type="checkbox"/>	testv2		2GB	Available	-		nova	<button>Edit Volume</button> <button>More</button>
<input type="checkbox"/>	testv		2GB	Available	-		nova	<button>Edit Volume</button> <button>More</button>

A context menu is open for the 'testv' volume, with a red box labeled '2' highlighting the 'Edit Attachments' option. Other options in the menu include 'Extend Volume', 'Create Snapshot', and 'Delete Volume'.

- (1) Leeres Volume anlegen, benötigt Name und Größe
- (2) Volume der laufenden Instanz zuweisen
- Kommandozeile (Volume-Größe: 2 GB):

```
$ nova volume-create --display-name my-vol-name 2 # --> vol ID  
$ nova volume-attach <vol id> grml-instance
```



Eigenes Abbild als VM starten

Launch Instance

Details * Access & Security * Networking *

Availability Zone:
nova

Instance Name: *
testrun

Flavor: *
i4.tiny

Instance Count: *
1

Instance Boot Source: *
Boot from image

Image Name:
image-ce-test (2.0 GB)

- Instanztyp i4.tiny
→ Erzeugt Swap- u. Ephemeral-Disk
- Abbild direkt starten
- Kommandozeile:

```
$ nova boot --flavor i1.tiny \
    --nic net-id=<internal net id> \
    --image <image name> \
    testrun
```



Öffentliche IP zuweisen

The screenshot shows the 'Access & Security' interface with the 'Compute' project selected. Under the 'Compute' section, 'Compute' is also selected. The 'Network' section is expanded, showing 'Access & Security' selected. In the 'Access & Security' section, the 'Floating IPs' tab is active. The 'Floating IPs' table displays one item: IP Address 131.188.42.114, Instance -, Floating IP Pool i4labnet. The 'Actions' column for this item contains two buttons: 'Associate' and 'More'. Both the 'Allocate IP To Project' button at the top and the 'Associate' button in the table are highlighted with red boxes and numbered '1' and '2' respectively.

- (1) Öffentliche IP aus Pool allokinieren, **nur einmalig nötig**
- (2) IP-Adresse an laufende Instanz zuweisen
- Kommandozeile:

```
$ nova floating-ip-create i4labnet  
$ nova floating-ip-associate my-vm-instance 131.188.42.115
```

- Abfrage innerhalb laufender VM:

```
$ wget -qO - http://169.254.169.254/latest/meta-data/public-ipv4
```



Zugriffsregeln für Netzwerkverbindungen

The screenshot shows the 'Access & Security' section of the OpenStack interface. On the left, a sidebar lists 'Project' (Compute selected), 'Overview', 'Instances', 'Volumes', 'Images', 'Access & Security' (selected and highlighted with a red border), and 'Network'. The main area is titled 'Access & Security' and contains tabs for 'Security Groups' (selected), 'Key Pairs', 'Floating IPs', and 'API Access'. Under 'Security Groups', there is a table with one item: 'default' (Name: default, Description: default). A red box highlights the 'Manage Rules' button next to the 'default' entry. Below the table, it says 'Displaying 1 item'.

- TCP-Ports müssen für öffentlichen Zugriff freigegeben werden
- Kommandozeile, z. B. für TCP-Port 22 (SSH):

```
$ nova secgroup-add-rule default tcp 22 22 0.0.0.0/0
```



Firewall-Zugriffsregeln

Add Rule

Rule: *

Direction

Open Port *

Port

Remote: *

CIDR

Description:

Rules define which traffic is allowed to instances assigned to the security group. A security group rule consists of three main parts:

Rule: You can specify the desired rule template or use custom rules, the options are Custom TCP Rule, Custom UDP Rule, or Custom ICMP Rule.

Open Port/Port Range: For TCP and UDP rules you may choose to open either a single port or a range of ports. Selecting the "Port Range" option will provide you with space to provide both the starting and ending port numbers. For ICMP rules you instead specify an ICMP type and code in the spaces provided.

Remote: You must specify the source of the traffic to be allowed via this rule. You may do so either in the form of an IP address block (CIDR) or via a source group (Security Group). Selecting a security group as the source will allow any other instance in that security group access to any other instance via this rule.

Ingress = Eingehende Verbindungen, Egress = Ausgehende Verbindungen



- Passwortloser Zugriff mit SSH

```
$ ssh-keygen -R <instanz_ip>      # Alten Host-Key entfernen  
$ ssh -i <gruppen_name> cloud@<instanz_ip>
```

→ Schlüssel aus letzter Tafelübung, Instanz-IP aus vorheriger Zuweisung

→ Anderes VM-Image unter selber IP erfordert evtl. Zurücksetzen von Host-Key

- Instanzen beenden: „Terminate“ auf der Weboberfläche, oder

```
$ nova list    # id heraussuchen  
$ nova delete <instanz id>
```

- Alte Abbilder/Volumes löschen: Weboberfläche, oder

```
$ nova volume-delete <volume id>  
$ nova image-delete <image id>
```



- Modifikationen des VM-Abbilds über Grml-Instanz
 - Installation weiterer Software-Pakete
 - Anpassung der Startskripte
 - Systemkonfiguration
- Limitationen der Cloud-Umgebung des Lehrstuhls
 - Ressourcen der drei Node-Controller sind **beschränkt**
 - Beenden von nicht (mehr) benötigten Instanzen
 - Jederzeit auf faire Verwendung achten
- Infrastruktur
 - Bitte sendet bei Problemen oder Ungereimtheiten schnellstmöglichst eine E-Mail an mw@i4.informatik.uni-erlangen.de

