



Netzwerkbaukasten OpenVSwitch

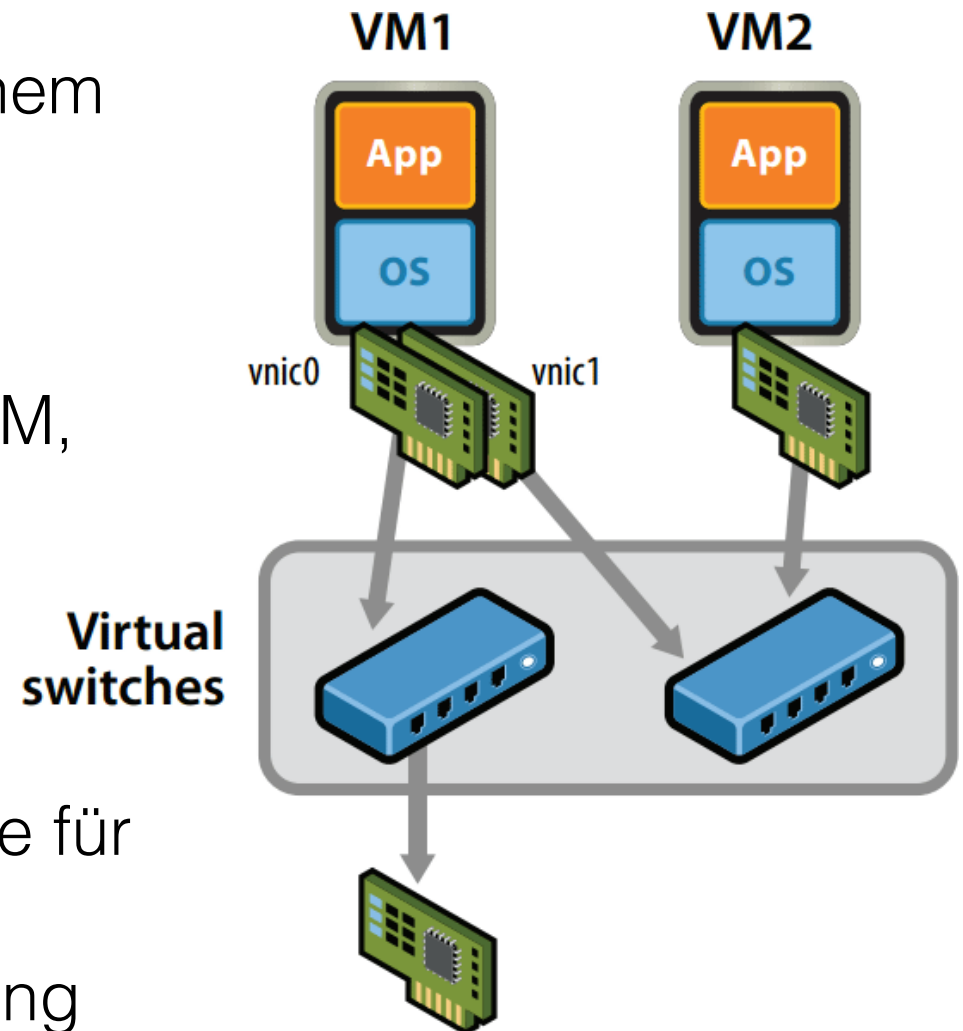
Sebastian Klingberg
Tübix 2016, 11.06.2016

Agenda

- Was ist ein virtueller Switch?
- OpenVSwitch - Komponenten und Beispiel
- Software-Defined Networking (SDN) ?!?
- Was bringt mir das jetzt ?

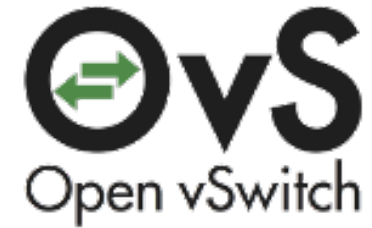
Was ist ein virtueller Switch?

- L2/3 Switch als Dienst auf einem Hostsystem
--> Hardware Virtualisierung
- Hypervisor VMWare, Xen, KVM, Virtualbox, libvirt,
- OpenStack Neutron
- Verteilte, komplexe Netzwerke für VMs und Container
--> Distributed Virtual Switching

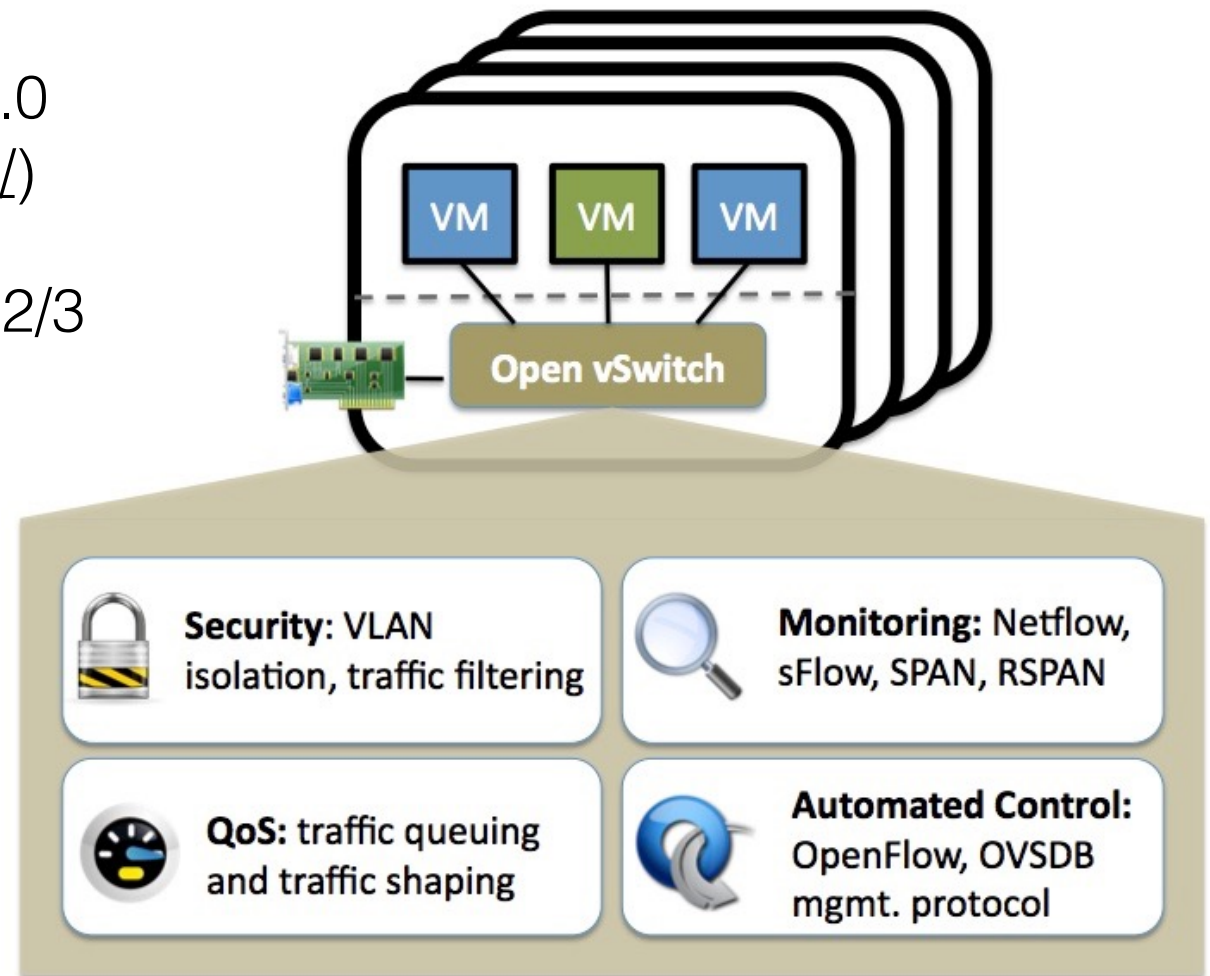


Source: <http://www.govmlab.com/what-is-virtual-switch-and-how-its-different-from-physical-switch/>

OpenVSwitch

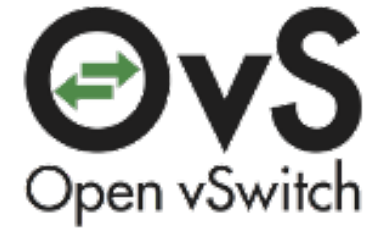


- OpenSource Apache 2.0 (<http://openvswitch.org/>)
- Switch Standards auf L2/3 + einiges mehr!
- User & Kernel space
- Datapath Modul im Linux Kernel !
- SDN Einbindung
 - OpenFLOW
 - Monitoring



Source: <http://openvswitch.org/>

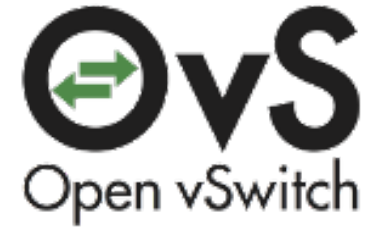
OpenVSwitch Komponenten



- <https://github.com/openvswitch/ovs>

ovs-vswitchd	OpenVSwitch Deamon
ovsdb-server	Datenbank mit OVS Konfiguration
ovs-vsctl	Konfigurationswerkzeug
ovs-appctl	vswitchd Steuerung zur Laufzeit (z.B. Debug)
ovs-ofctl	OpenFLOW Konfiguration

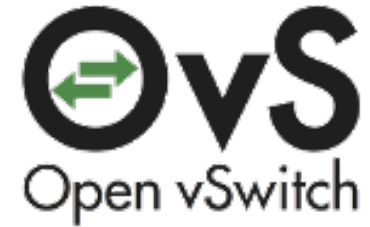
OVS Example I



- Start OpenVSwitch
--> Datapath Modul !!!

```
[root@c7vm1 ~]# systemctl start openvswitch
[root@c7vm1 ~]# journalctl -u openvswitch
-- Logs begin at Do 2016-06-09 11:18:55 EDT, end at Do 2016-06-09 11:30:25 EDT. --
Jun 09 11:19:05 c7vm1 systemd[1]: Starting LSB: Open vSwitch switch...
Jun 09 11:19:08 c7vm1 openvswitch[515]: Starting ovssdb-server [ OK ]
<<snap>>
Jun 09 11:19:09 c7vm1 openvswitch[515]: Inserting openvswitch module [ OK ]
Jun 09 11:19:09 c7vm1 openvswitch[515]: Starting ovs-vswitchd [ OK ]
Jun 09 11:19:09 c7vm1 openvswitch[515]: Enabling remote OVSDb managers [ OK ]
Jun 09 11:19:09 c7vm1 systemd[1]: Started LSB: Open vSwitch switch.
Jun 09 11:22:46 c7vm1 systemd[1]: Started LSB: Open vSwitch switch.
```

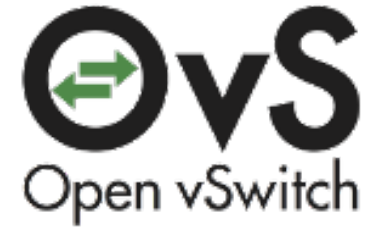
OVS Example II



- Neues Bridge-Interface erstellen:

```
[root@c7vm1 ~]# ovs-vsctl add-br ovs-br1
[root@c7vm1 ~]# ovs-vsctl show
2efc8719-0dd6-468f-9cf3-db11e0551213
    Bridge "ovs-br1"
        Port "ovs-br1"
            Interface "ovs-br1"
                type: internal
    ovs_version: "2.5.0"
```

OVS Example II



- (Docker) Container starten

```
[root@c7vm1 ~]# docker run -dit --name=con1 --net=none busybox /bin/sh
5461d2f1f6705b343cb47fb1fcf67436b4e06909b84537307b5faa6046d66550
[root@c7vm1 ~]# docker run -dit --name=con2 --net=none busybox /bin/sh
32b24c9f9f78b56c2edaa7ac8e356d49e9652e4150c4655c64bfe042bf25f573'
```


OVS Example III



- (Docker) Container verbinden

```
[root@c7vm1 ~]# ovs-docker add-port ovs-br1 eth0 con1 --ipaddress=192.168.0.1/24
[root@c7vm1 ~]# ovs-docker add-port ovs-br1 eth0 con2 --ipaddress=192.168.0.2/24
[root@c7vm1 ~]# ovs-vsctl show
2efc8719-0dd6-468f-9cf3-db11e0551213
    Bridge "ovs-br1"
        Port "0850594603354_1"
            Interface "0850594603354_1"
        Port "005a617433c04_1"
            Interface "005a617433c04_1"
        Port "ovs-br1"
            Interface "ovs-br1"
                type: internal
    ovs_version: "2.5.0"
```

OVS Example IV

- Pingtest

```
[root@c7vm1 ~]# docker ps --no-trunc --format "{{.Names}} {{.ID}}"
con2 32b24c9f9f78b56c2edaa7ac8e356d49e9652e4150c4655c64bfe042bf25f573
con1 5461d2f1f6705b343cb47fb1fcf67436b4e06909b84537307b5faa6046d66550
```

```
[root@c7vm1 ~]# docker exec con1 ping -c3 192.168.0.2
PING 192.168.0.2 (192.168.0.2): 56 data bytes
64 bytes from 192.168.0.2: seq=0 ttl=64 time=0.522 ms
64 bytes from 192.168.0.2: seq=1 ttl=64 time=0.114 ms
64 bytes from 192.168.0.2: seq=2 ttl=64 time=0.116 ms
```

```
[root@c7vm1 ~]# docker exec con2 ping -c3 192.168.0.1
PING 192.168.0.1 (192.168.0.1): 56 data bytes
64 bytes from 192.168.0.1: seq=0 ttl=64 time=0.349 ms
64 bytes from 192.168.0.1: seq=1 ttl=64 time=0.149 ms
64 bytes from 192.168.0.1: seq=2 ttl=64 time=0.115 ms
```

Software-Defined-Networking (SDN)

- Ein zentrales „Gehirn“ verwaltet die Netzwerkinfrastruktur (genannt „**Controller**“)
- Regelwerk über Protokoll verteilt --> OpenFLOW !!
- OpenSource Controller für OVS:



(<https://www.opendaylight.org/>)

Was bringt mir das jetzt?

OpenVSwitch ist ein quelloffener Netzwerkbaukasten um...

- ... in einer Welt voller VMs und Container die Vernetzung auf low-level Ebene zu verstehen
- ... in SDN einzusteigen
- ... in einer heterogenen Infrastruktur zu kommunizieren
- ... die Dynamik & Flexibilität in Cloud Umgebungen zu realisieren

Sources

- Titlepage Graphic designed by Freepik + OVS Logo
- <http://openvswitch.org/>
- <https://github.com/openvswitch/ovs>