SDN & OpenStack

Eine Einführung

Martin Gerhard Loschwitz

hastexo!

h!

Wer?


WIEN


LINUX
MAGAZIN

h!

h!

h!


LINUX
MAGAZIN


ADMIN
www.admin-magazin.de


SLAC
Secure Linux
Administration
Conference 2014
13.-14. Mai 2014 | Berlin


Pacemaker

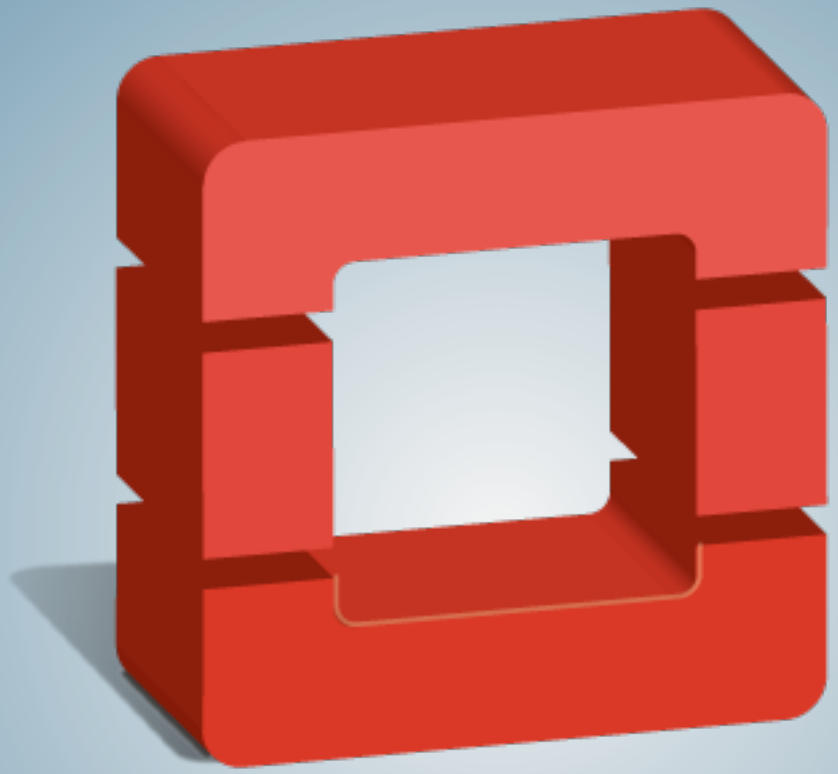
h!


XTREME BOWL

hastexo!

ceph

h!



openstack™

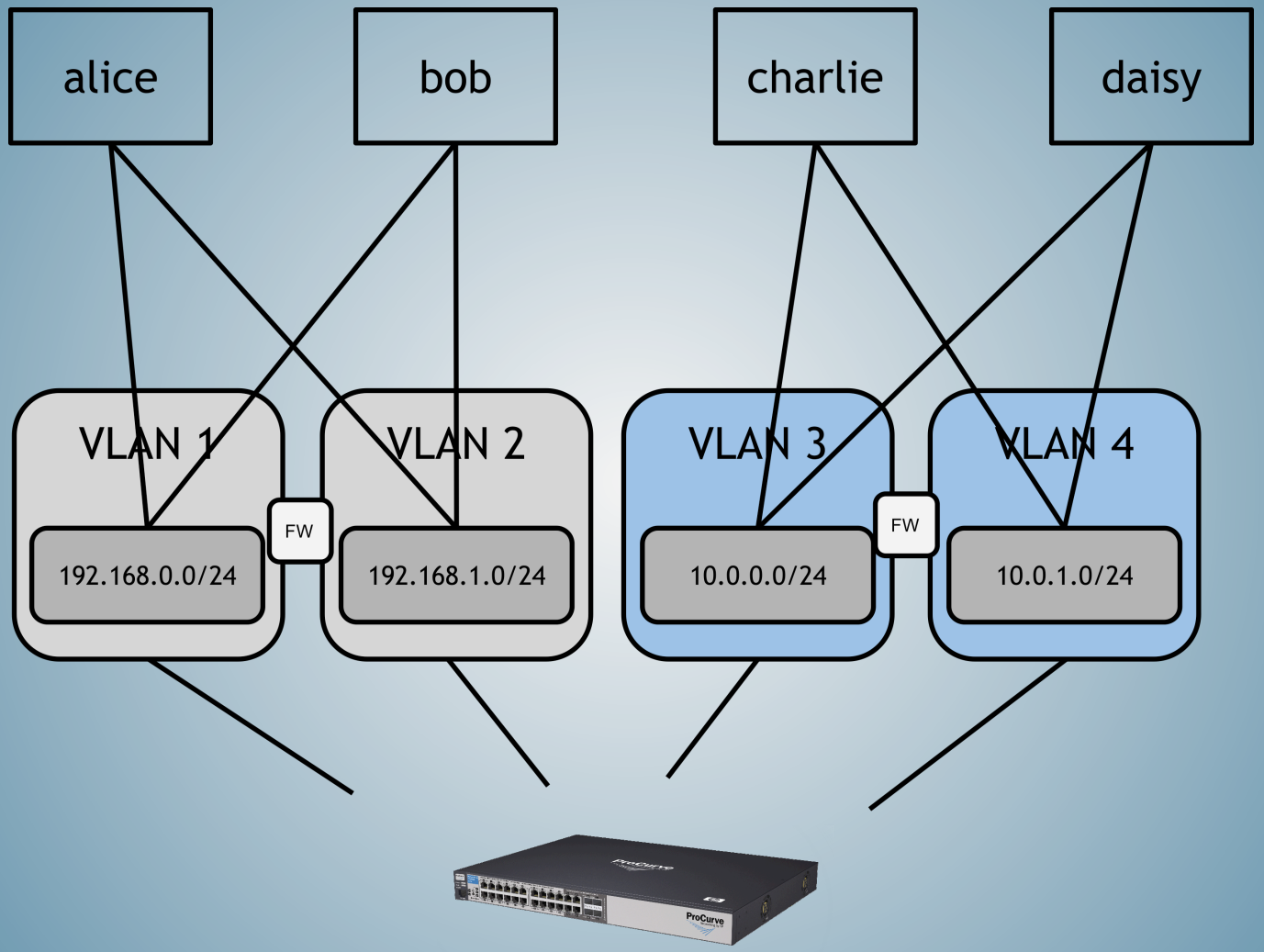
h!

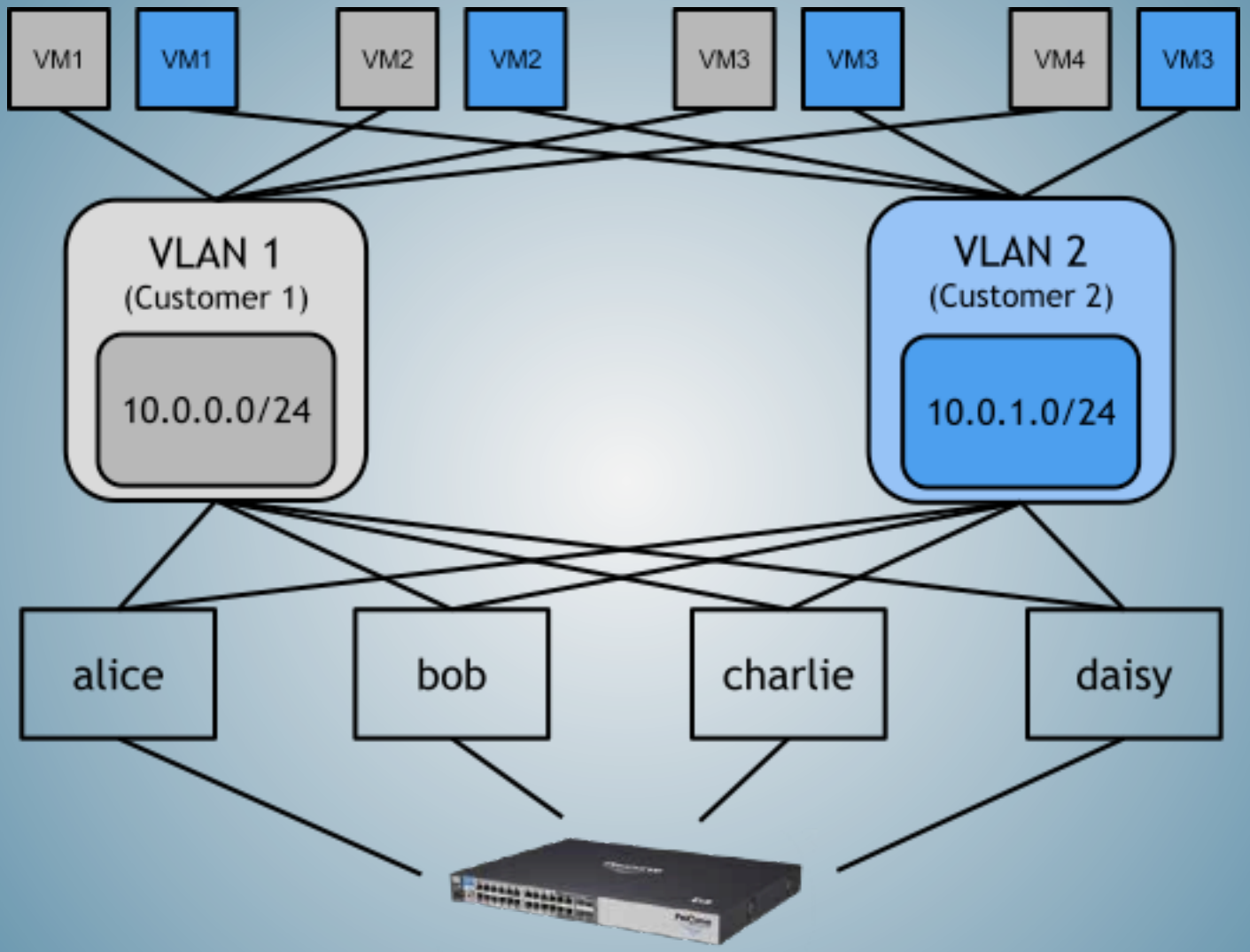
Cloud Computing

Konventionelle Netze basieren auf einem **statischen** Design.

Erste Prämisse:
Das Netzwerk ist
zentral verwaltet.

Zweite Prämisse:
Server **„gehören"**
spezifischen Kunden


alice
bob
charlie
daisy
VLAN 1
VLAN 2
VLAN 3
VLAN 4
FW
FW
192.168.0.0/24
192.168.1.0/24
10.0.0.0/24
10.0.1.0/24
ProCurve


VM1
VM1
VM2
VM2
VM3
VM3
VM4
VM3
VLAN 1
(Customer 1)
10.0.0.0/24
VLAN 2
(Customer 2)
10.0.1.0/24
alice
bob
charlie
daisy

Dritte Prämisse:
Wenig bis kein Bedarf
für Scale-Out

h!

Cloud Computing?
Extrem skalierte Setups?
Reality Check

h!

Erste Prämisse:
Das Netzwerk ist
zentral verwaltet.

Netzwerk-Management
obliegt dem **Kunden**

Zweite Prämisse:
Server **„gehören"**
spezifischen Kunden

Server hosten VMs. Die VMs jedes Kunden müssen auf **jedem Server** laufen.

h!

Dritte Prämisse:
Wenig bis kein Bedarf für Scale-Out

In Clouds ist Scale-Out **notwendig**, und es muss **einfach** erreichbar sein.

Klassische Netzwerkdesigns bieten all diese Funktionen **nicht** oder nur auf Umwegen.

Erste Konsequenz:

Bei Netzwerk-Equipment sind Hardware und Software voneinander zu trennen (“**decouple**”)

Zweite Konsequenz:

Die Cloud muss sich **selbst** um ihr Netzwerk **kümmern**.

Dritte Konsequenz:

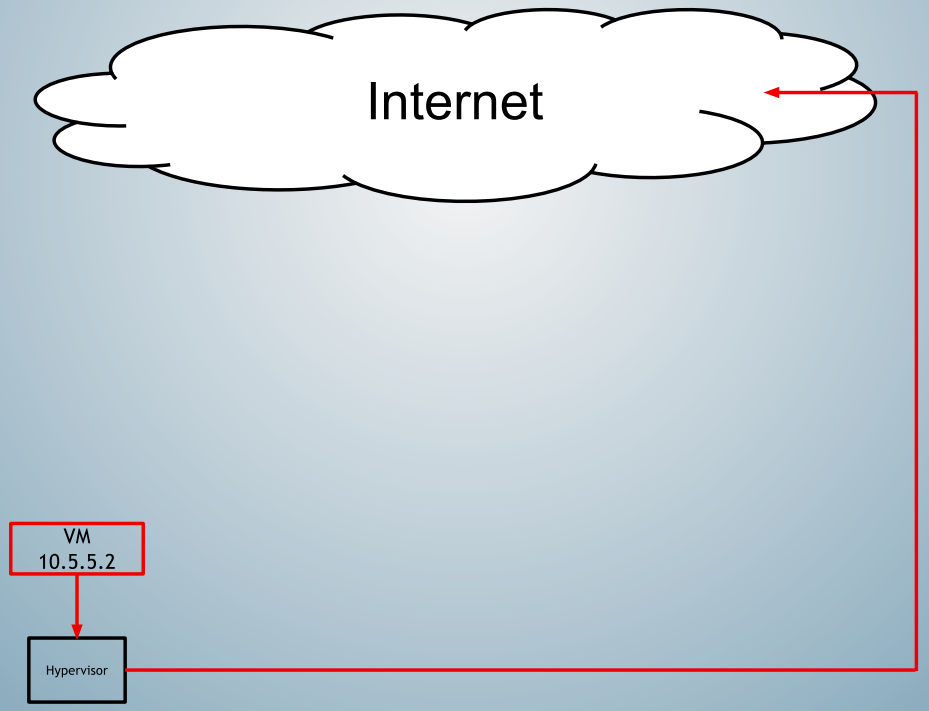
Die Cloud-Software muss **wissen**, wie sie das erreicht.

h!

In Clouds begegnen uns
zwei Arten von Traffic!

Traffic zwischen VMs des gleichen Kunden:

Traffic zwischen VMs und anderen Netzen:



Virtueller **L2**-Traffic

Open vSwitch (+OpenFlow),
VMware NSX, Midonet,
OpenContrail, PLUMGrid ...

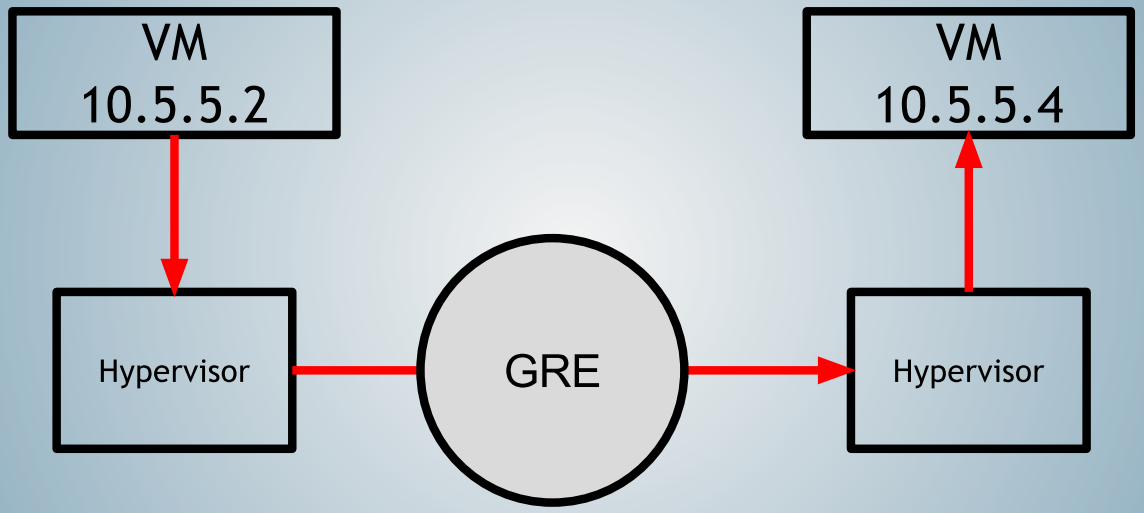
Open vSwitch (+OpenFlow),
VMware NSX, Midonet,
OpenContrail, PLUMGrid ...

OpenFlow: Virtuelle
Forwarding Plane auf
jedem beteiligten Host

Open vSwitch: Baut SDNs auf Basis von Open Flow, ist Front-End zur **Konfiguration** der Forwarding Plane

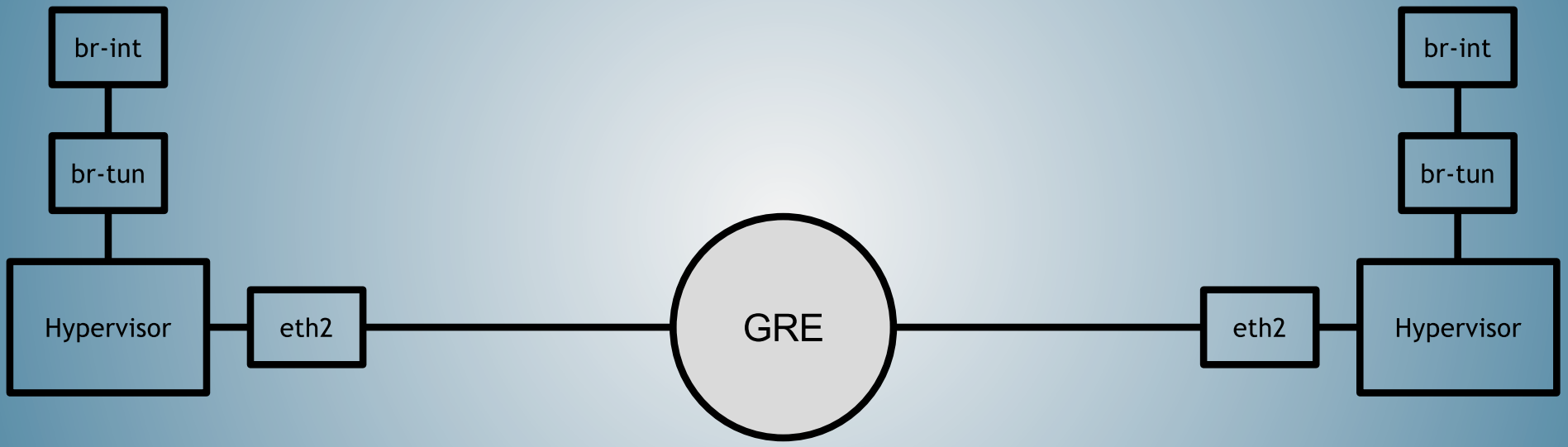
Open vSwitch verwirklicht
auch die **Trennung** von
SDN- und Nicht-SDN-Traffic

(GRE, VLAN, VXLAN)


VM
10.5.5.2
VM
10.5.5.4
Hypervisor
GRE
Hypervisor

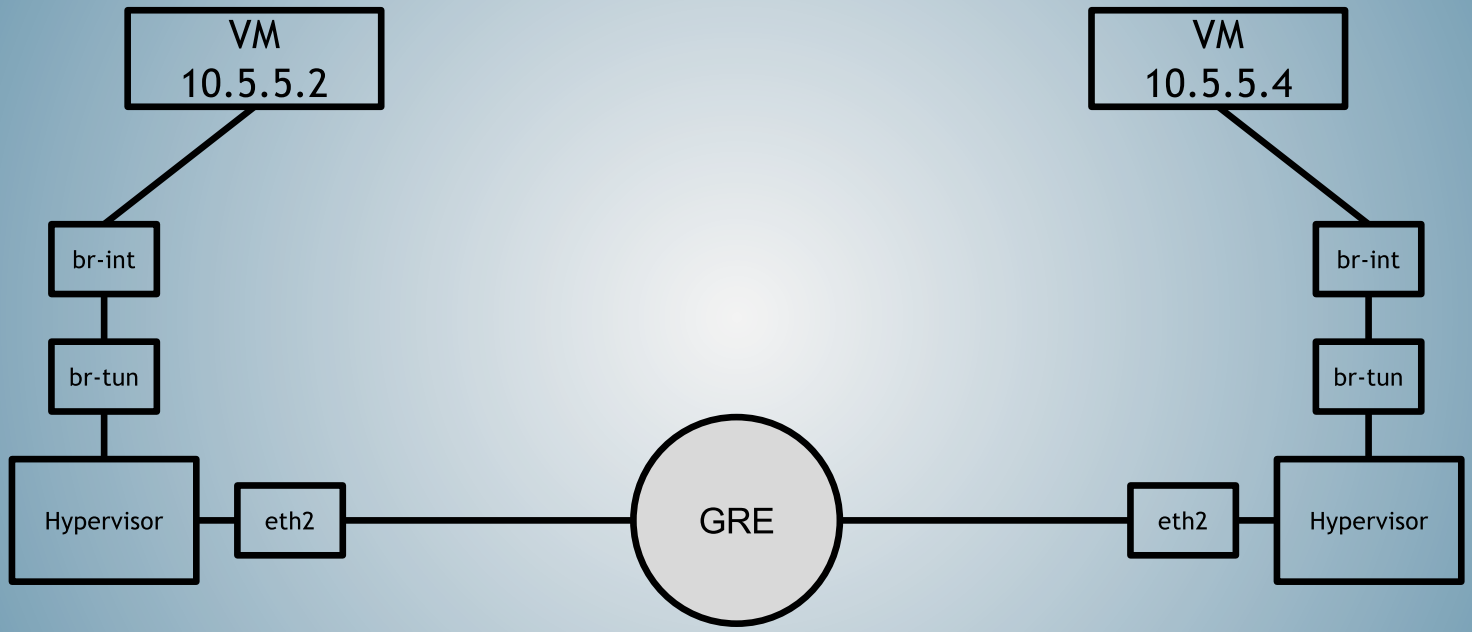
Jede VM bekommt eine
virtuelle NIC (TAP device)

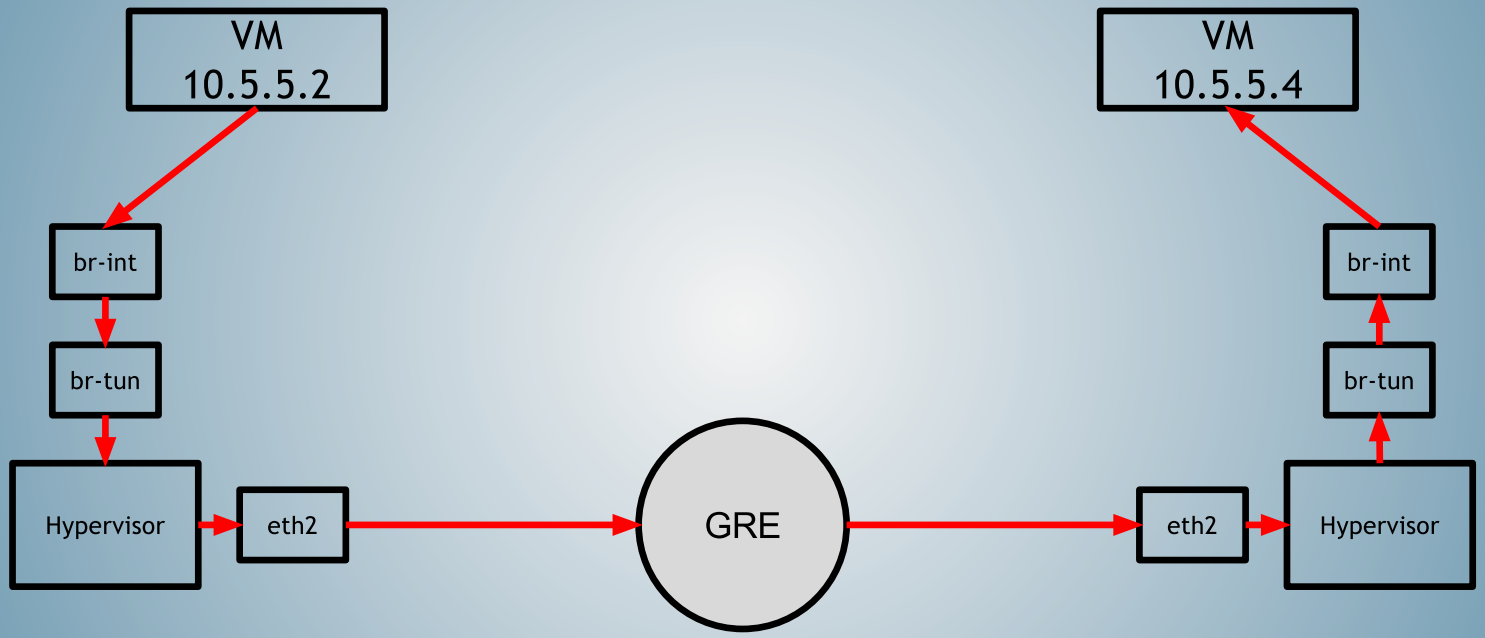
Open vSwitch verwaltet
virtuelle Switches (Bridges),
die den Weg von Paketen
im LAN bestimmen.

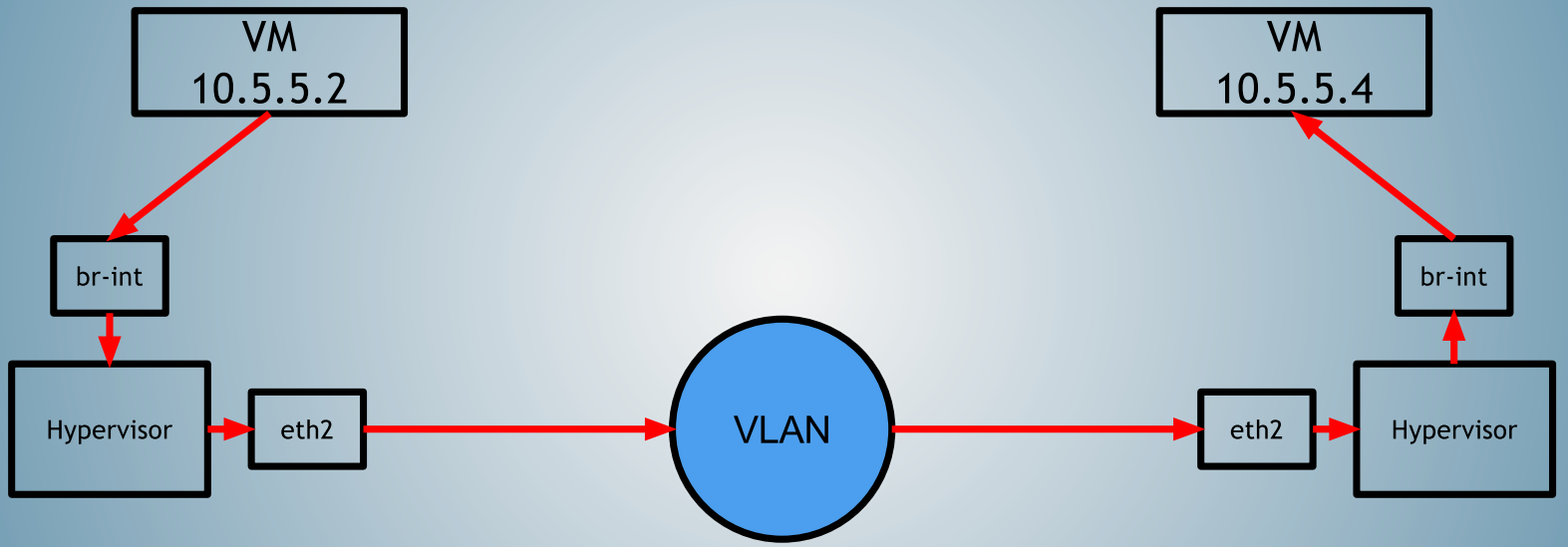

br-int
br-tun
Hypervisor
eth2
GRE
eth2
br-int
br-tun
Hypervisor

Die TAP-Devices der VMs sind **direkt** mit den virtuellen Switches verbunden.

Über **OpenFlow-Regeln** ist festgelegt, **wohin** Pakete in den virtuellen Switches gehen.


VM
10.5.5.2
VM
10.5.5.4
br-int
br-tun
br-int
br-tun
Hypervisor
eth2
GRE
eth2
Hypervisor


VM
10.5.5.2
VM
10.5.5.4
br-int
br-tun
Hypervisor
eth2
GRE
eth2
Hypervisor
br-int
br-tun


VM
10.5.5.2
VM
10.5.5.4
br-int
br-int
Hypervisor
eth2
VLAN
eth2
Hypervisor

Bob:

```
# ovs-vsctl show
    Bridge br-int
      Port patch-tun
            Interface patch-tun
            options: {peer=patch-int}
      Port "tape4dbc657-c4"
            tag: 1
            Interface "tape4dbc657-c4"
      Port br-int
            Interface br-int
```

Bob (cont.):

```
# ovs-vsctl show
[…]
    Bridge br-tun
      Port br-tun
        Interface br-tun
      Port patch-int
        Interface patch-int
        options: {peer=patch-tun}
      Port "gre-1"
        Interface "gre-1"
        options: {in_key=flow,
          local_ip="192.168.133.112", \
          out_key=flow,
          remote_ip="192.168.133.114"}
```

Daisy:

```
# ovs-ofctl dump-flows br-tun
[…]
 cookie=0x0, duration=27547.332s, table=10, \
     n_packets=1537, n_bytes=359538, \
     idle_age=25, priority=1 \
     actions=[…],output:1
```

Daisy (cont.):

```
# ovs-ofctl show br-tun
1(patch-int): addr:76:7c:cb:84:54:f3
[...]
2(gre-2): addr:16:14:65:eb:26:8c
[...]
LOCAL(br-tun): addr:1e:5e:83:65:02:4e
[...]
```

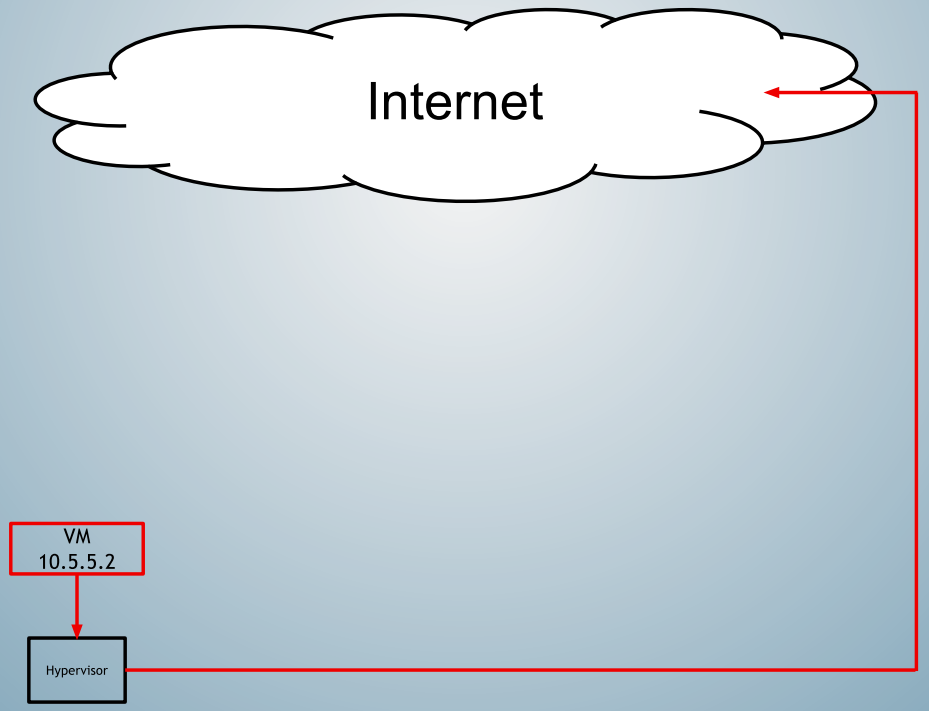
Daisy (cont.):

```
# ovs-vsctl show
Bridge br-tun
[...]
  Port patch-int
  Interface patch-int
  options: {peer=patch-tun}
[...]
```

Daisy (cont.):

```
# ovs-vsctl show
[...]
Bridge br-int
  Port "tap6dcc748c-17"
  tag: 1
[...]
  Port patch-tun
  options: {peer=patch-int}
```

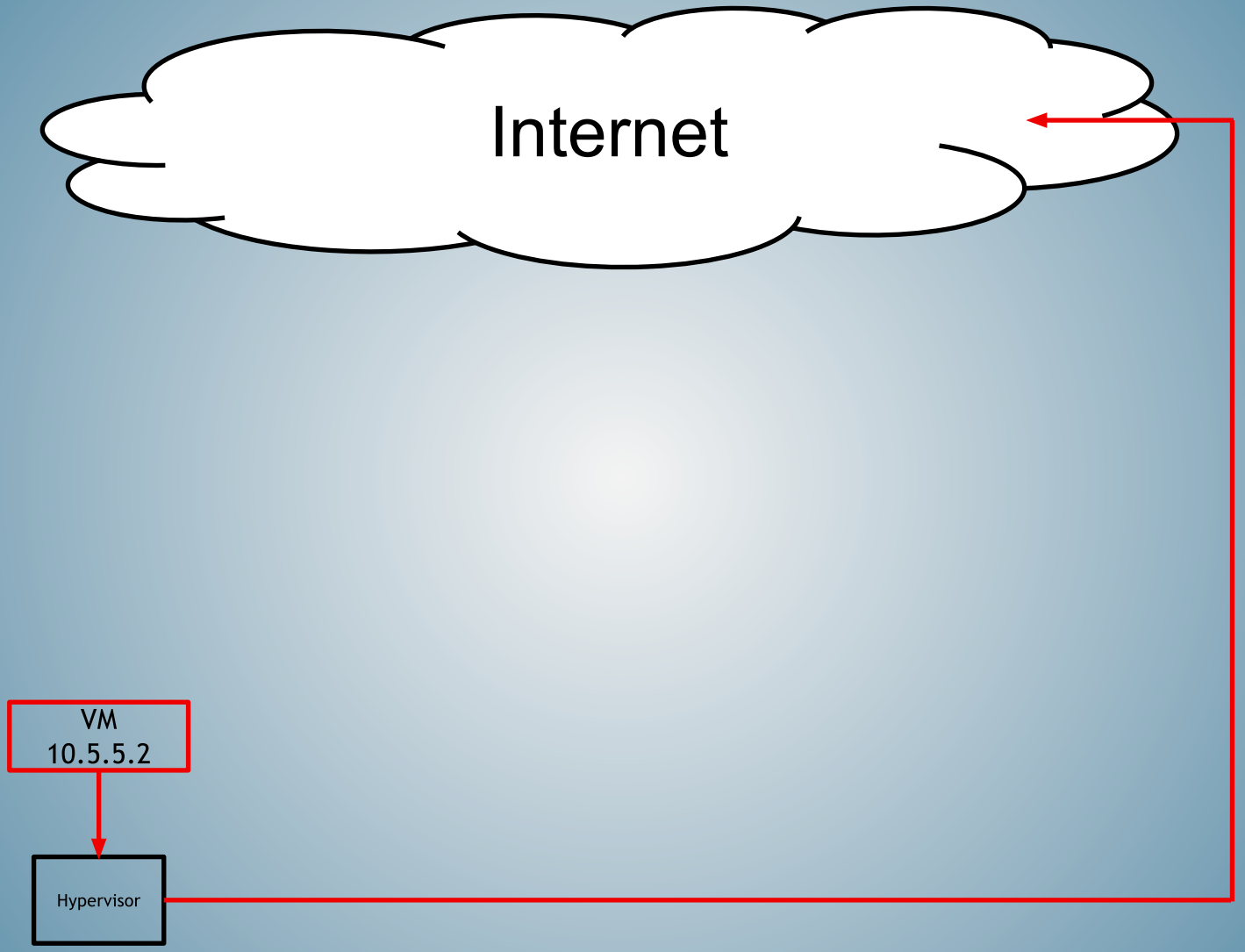
Traffic zwischen VMs und anderen Netzen:

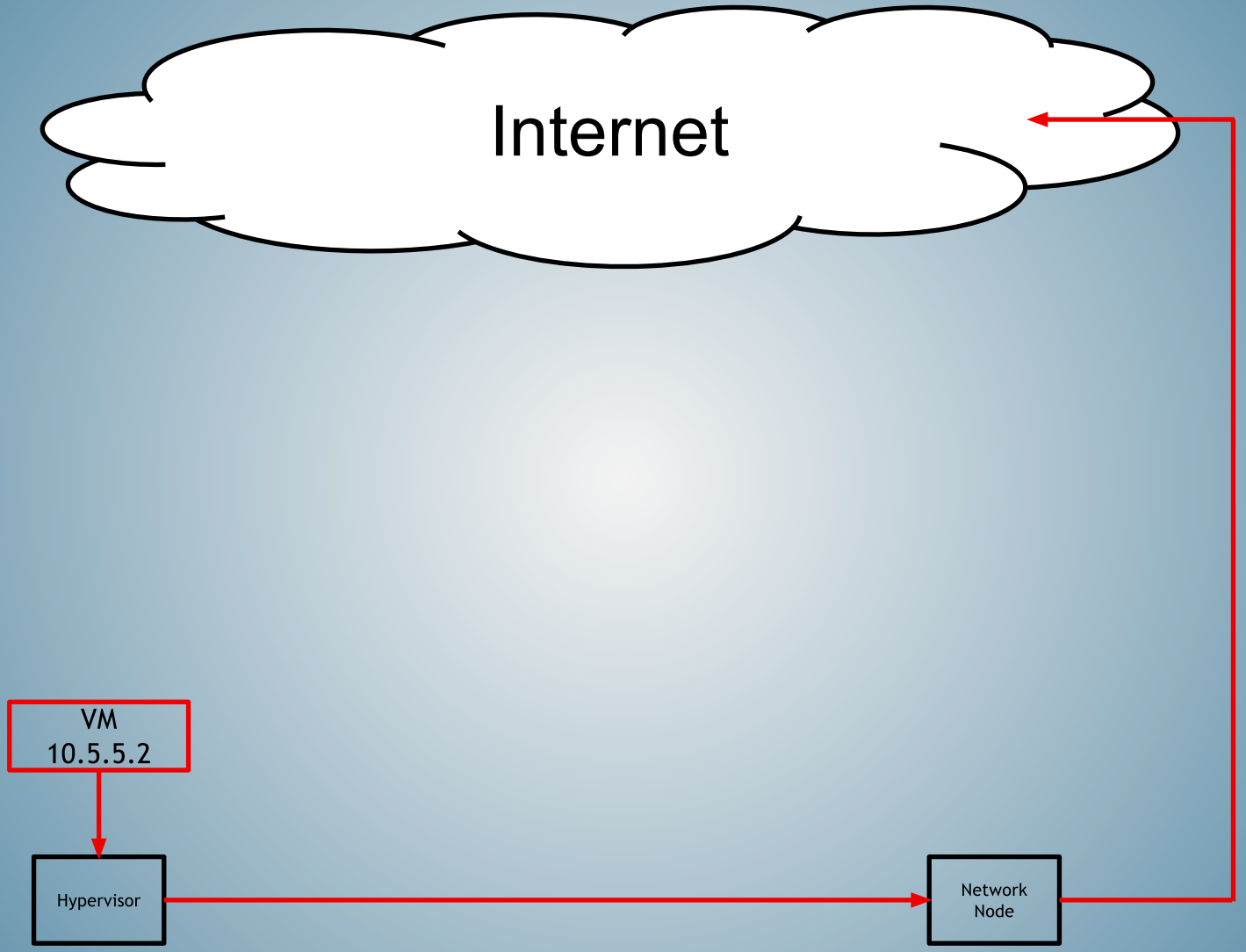


VMs sollen im Hinblick
auf Internet-Zugriff **zentral**
verwaltbar sein.

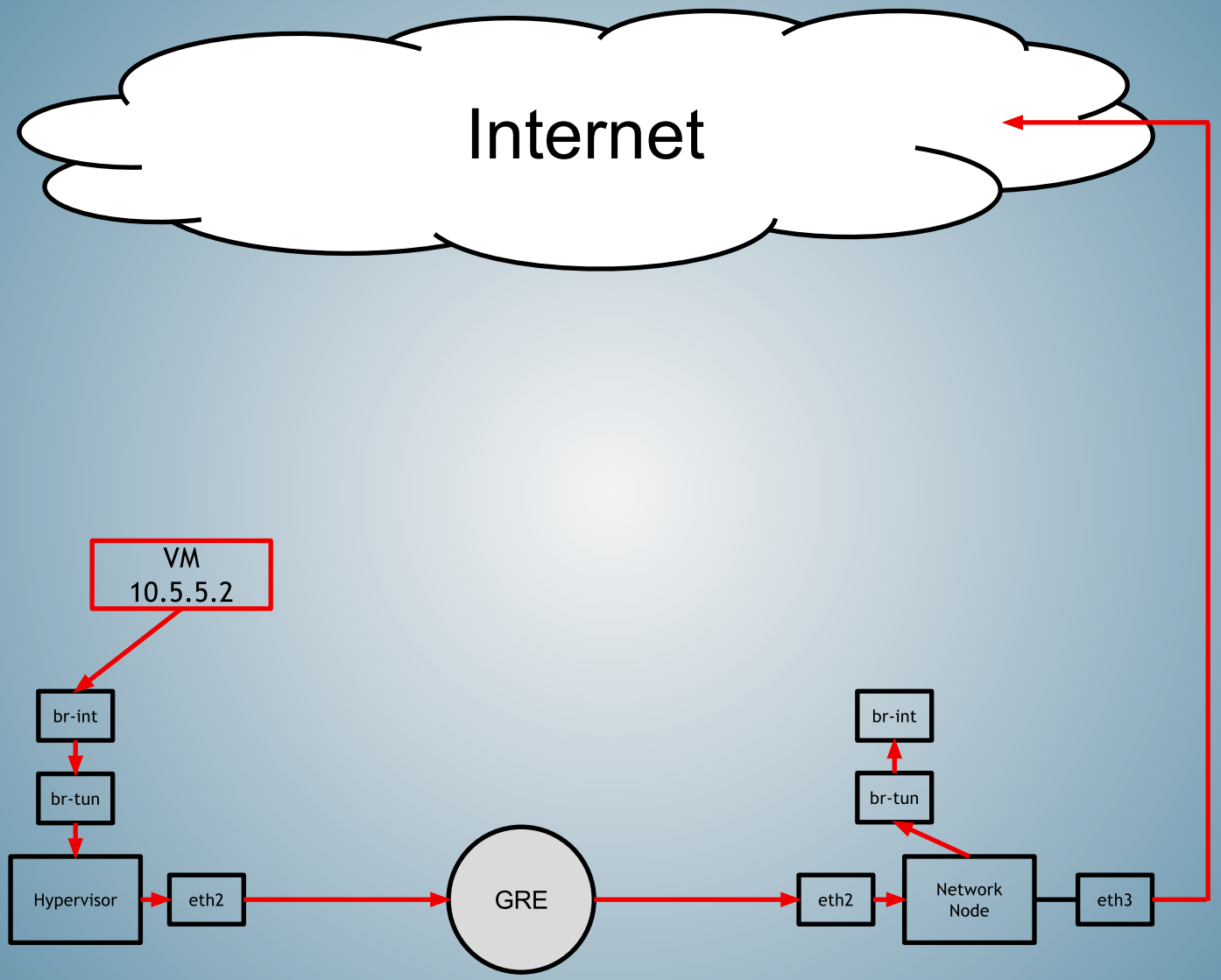
Dienste wie Firewalling,
virtuelles Routing und
LBaaS setzen das voraus.

Internet-Traffic nutzt
"**Network Nodes**" zur
Sicherstellung der
zentralen Verwaltung.


Internet
VM
10.5.5.2
Hypervisor

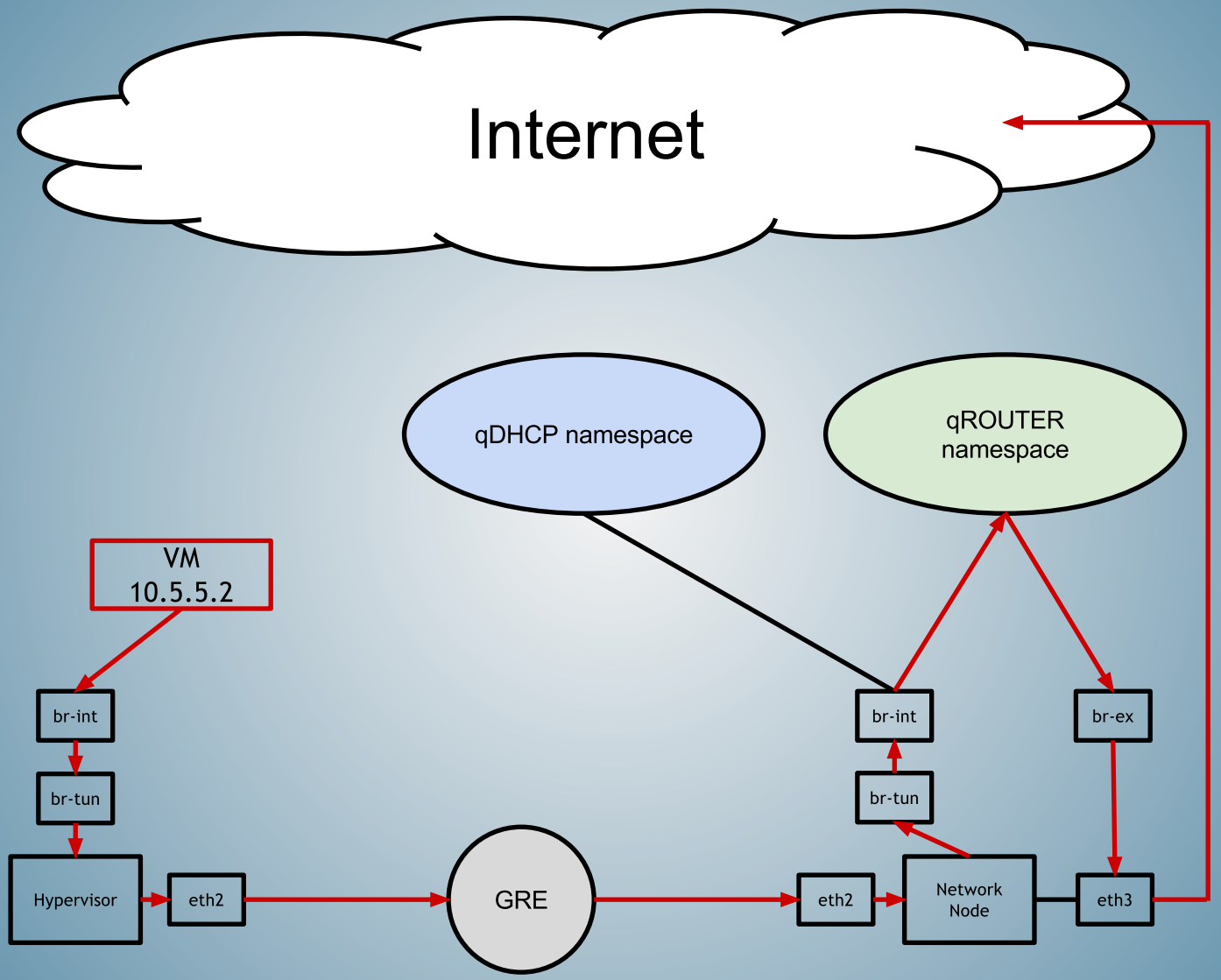

Internet
VM
10.5.5.2
Hypervisor
Network
Node

Der Packet Flow zwischen den Network Nodes und den Hypervisor-Knoten funktioniert **wie gehabt**.


Internet
VM
10.5.5.2
br-int
br-tun
Hypervisor
eth2
GRE
eth2
br-int
br-tun
Network
Node
eth3

Auf auf den Network Nodes müssen Pakete von Kunden strikt **getrennt** bleiben.

Auf den Network Nodes kommen dazu **Network Namespaces** zum Einsatz.


Internet
qDHCP namespace
qROUTER
namespace
VM
10.5.5.2
br-int
br-tun
Hypervisor
eth2
GRE
eth2
Network
Node
br-int
br-tun
br-ex
eth3

Zur Erinnerung: SDN muss **aus der Cloud-Umgebung heraus** konfigurierbar sein.

Willkommen bei OpenStack **Neutron**

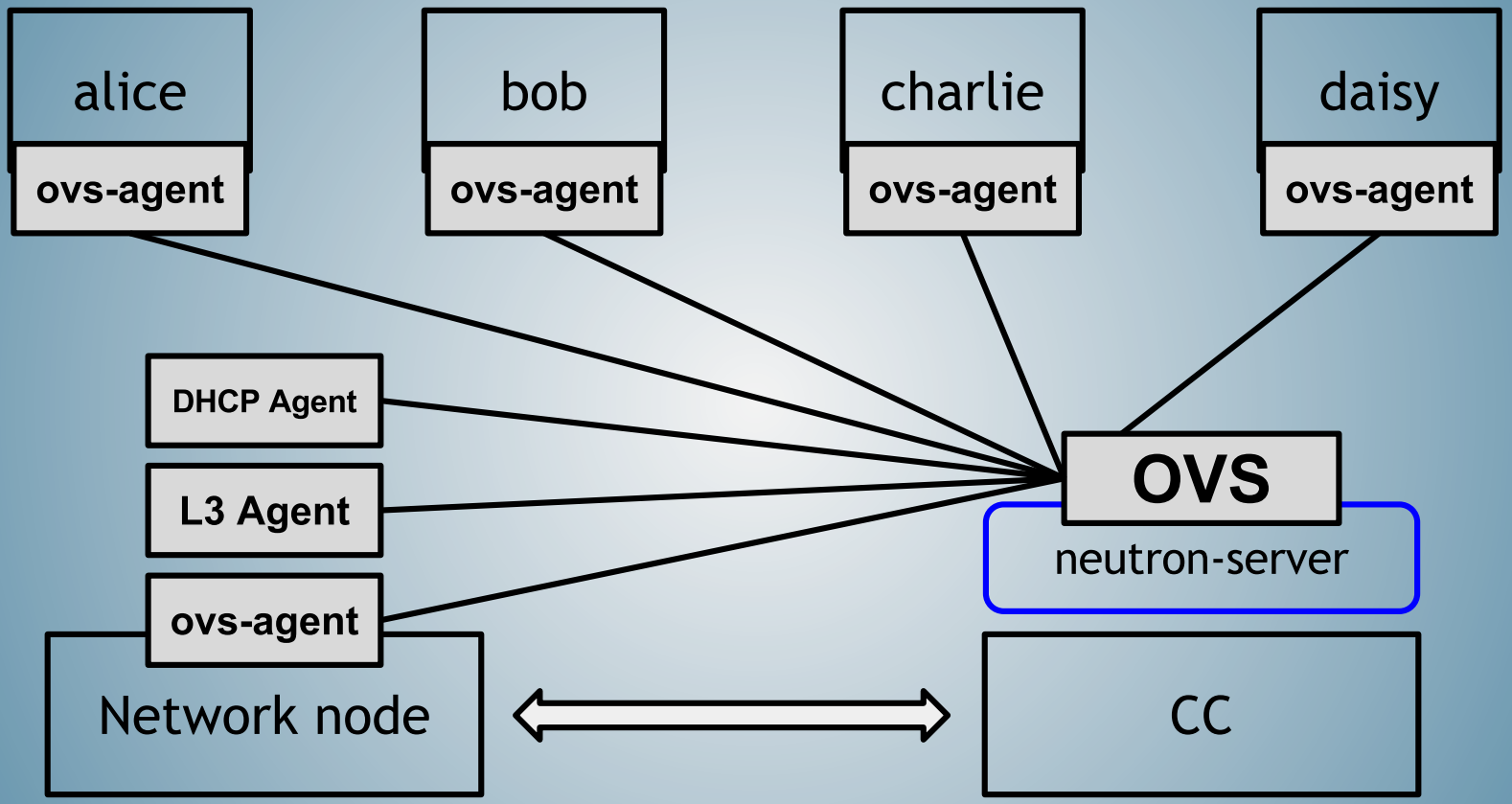
h!

Modulares Design

Neutron **Server**

Neutron **Plugins**

Neutron **Agents**


alice
ovs-agent
bob
ovs-agent
charlie
ovs-agent
daisy
ovs-agent
DHCP Agent
L3 Agent
ovs-agent
Network node
OVS
neutron-server
CC

Neutron Key Features

Virtuelle Layer2-Netzwerke
auf dem Layer 3 mit echter
Trennung von Kundenpaketen

h!

Kunden **bestimmen** ihre Netzwerktopologie **selbst**

h!

Nahtlos integrierter
Internet-Zugriff für VMs
(**Layer 3**-Networking)

Support für **diverse**
SDN-Lösungen

Open vSwitch, VMware
NSX, Ryu, Midonet

Support für Hardware-basierte Lösungen (über **Plugins**)

Cisco, Juniper,
Brocade, Mellanox

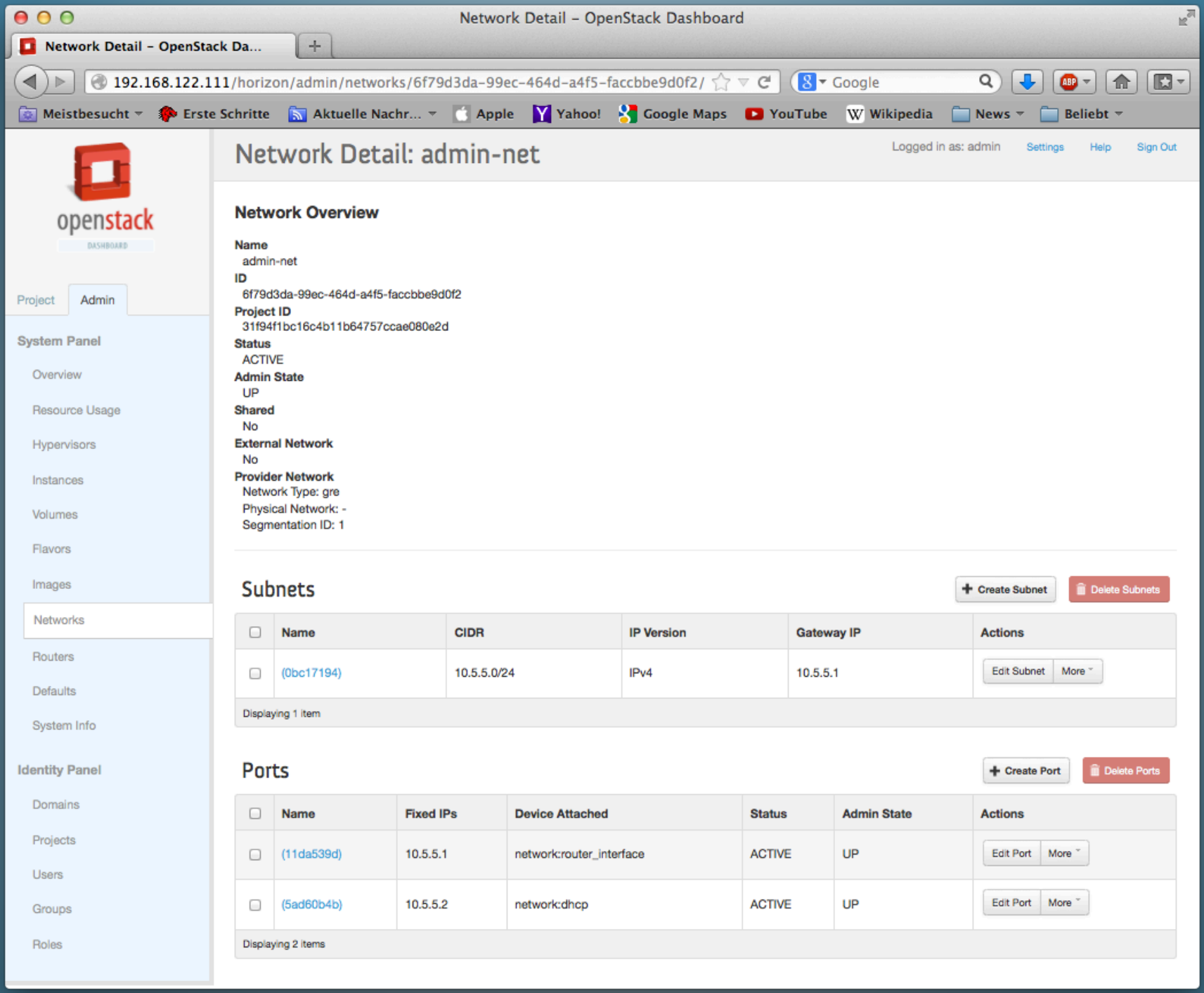
h!

In OpenStack **Icehouse**:
Verschiedene Plugins
(Layer 2) zur **gleichen** Zeit

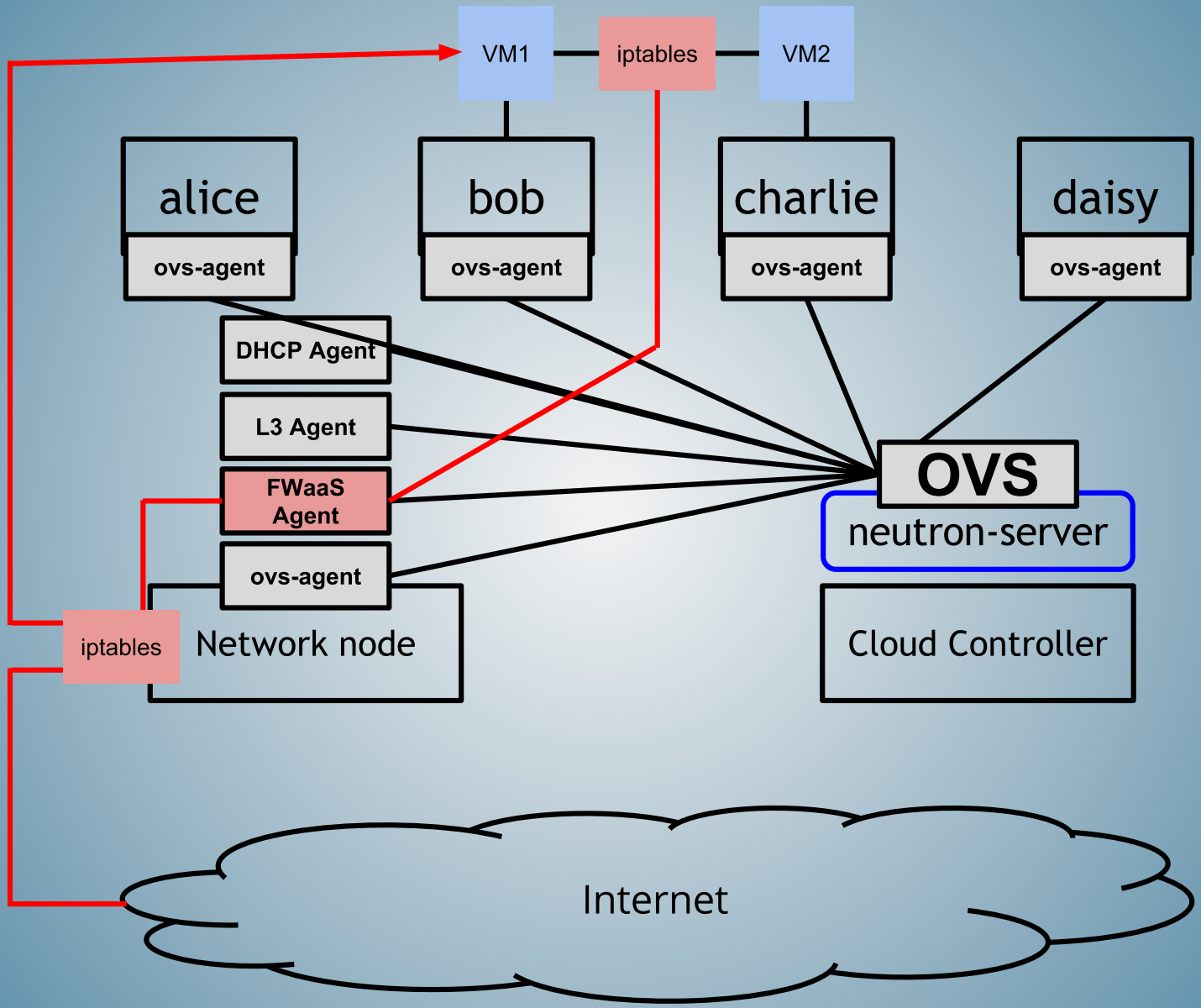
h!

Konfiguration mittels eines intuitiven **Webinterfaces** (OpenStack Dashboard)

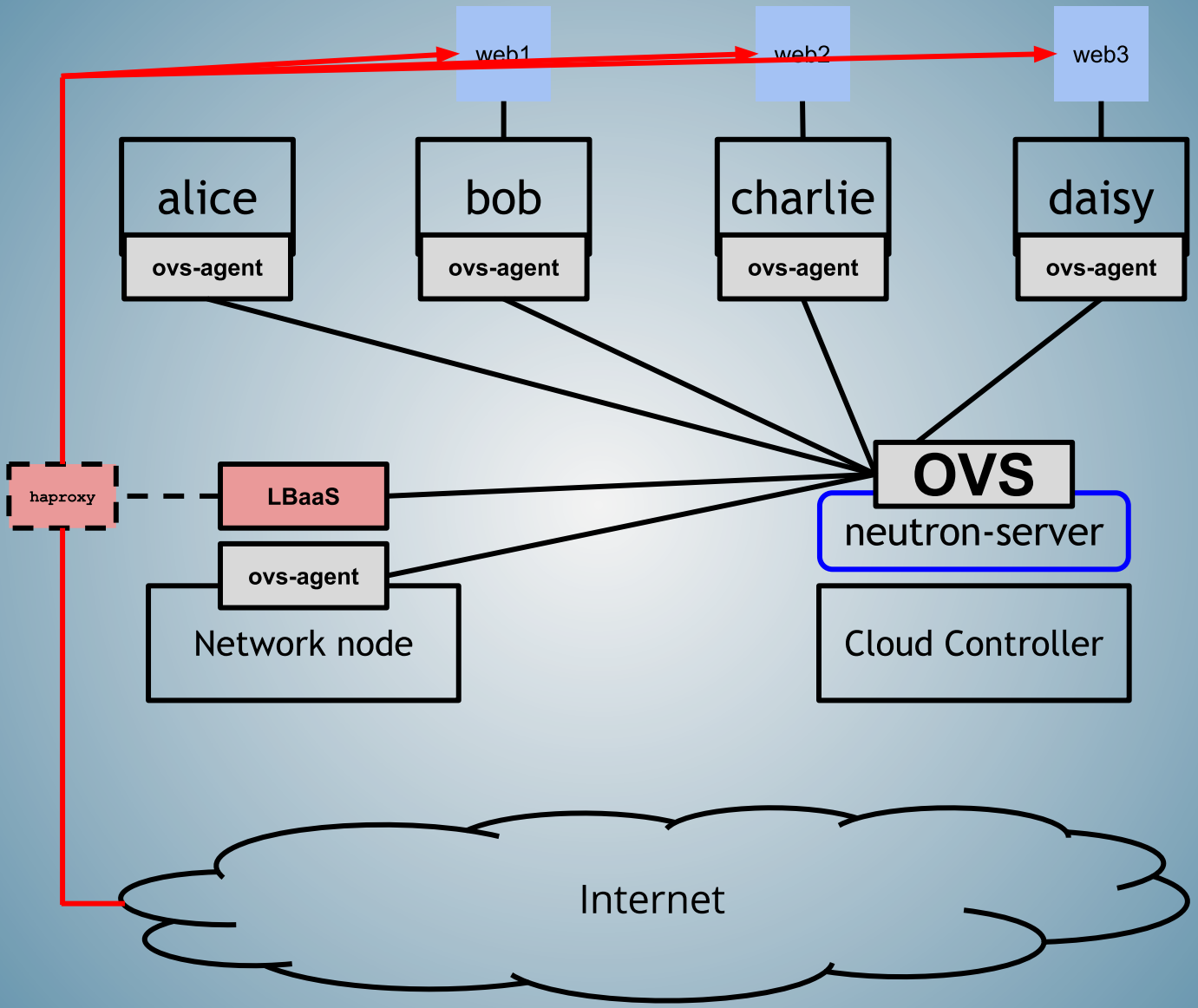
h!


Network Detail – OpenStack Dashboard
Network Detail – OpenStack Da...
192.168.122.111/horizon/admin/networks/6f79d3da-99ec-464d-a4f5-faccbbe9d0f2/
Google
Meistbesucht
Erste Schritte
Aktuelle Nachr...
Apple
Yahoo!
Google Maps
YouTube
Wikipedia
News
Beliebt
openstack
DASHBOARD
Project
Admin
System Panel
Overview
Resource Usage
Hypervisors
Instances
Volumes
Flavors
Images
Networks
Routers
Defaults
System Info
Identity Panel
Domains
Projects
Users
Groups
Roles
Network Detail: admin-net
Logged in as: admin
Settings
Help
Sign Out
Network Overview
Name
admin-net
ID
6f79d3da-99ec-464d-a4f5-faccbbe9d0f2
Project ID
31f94f1bc16c4b11b64757ccae080e2d
Status
ACTIVE
Admin State
UP
Shared
No
External Network
No
Provider Network
Network Type: gre
Physical Network: -
Segmentation ID: 1
Subnets
+ Create Subnet
Delete Subnets
Name
CIDR
IP Version
Gateway IP
Actions
(0bc17194)
10.5.5.0/24
IPv4
10.5.5.1
Edit Subnet
More
Displaying 1 item
Ports
+ Create Port
Delete Ports
Name
Fixed IPs
Device Attached
Status
Admin State
Actions
(11da539d)
10.5.5.1
network:router_interface
ACTIVE
UP
Edit Port
More
(5ad60b4b)
10.5.5.2
network:dhcp
ACTIVE
UP
Edit Port
More
Displaying 2 items

Firewall as a Service (**FWaaS**,
seit OpenStack **Havana**)

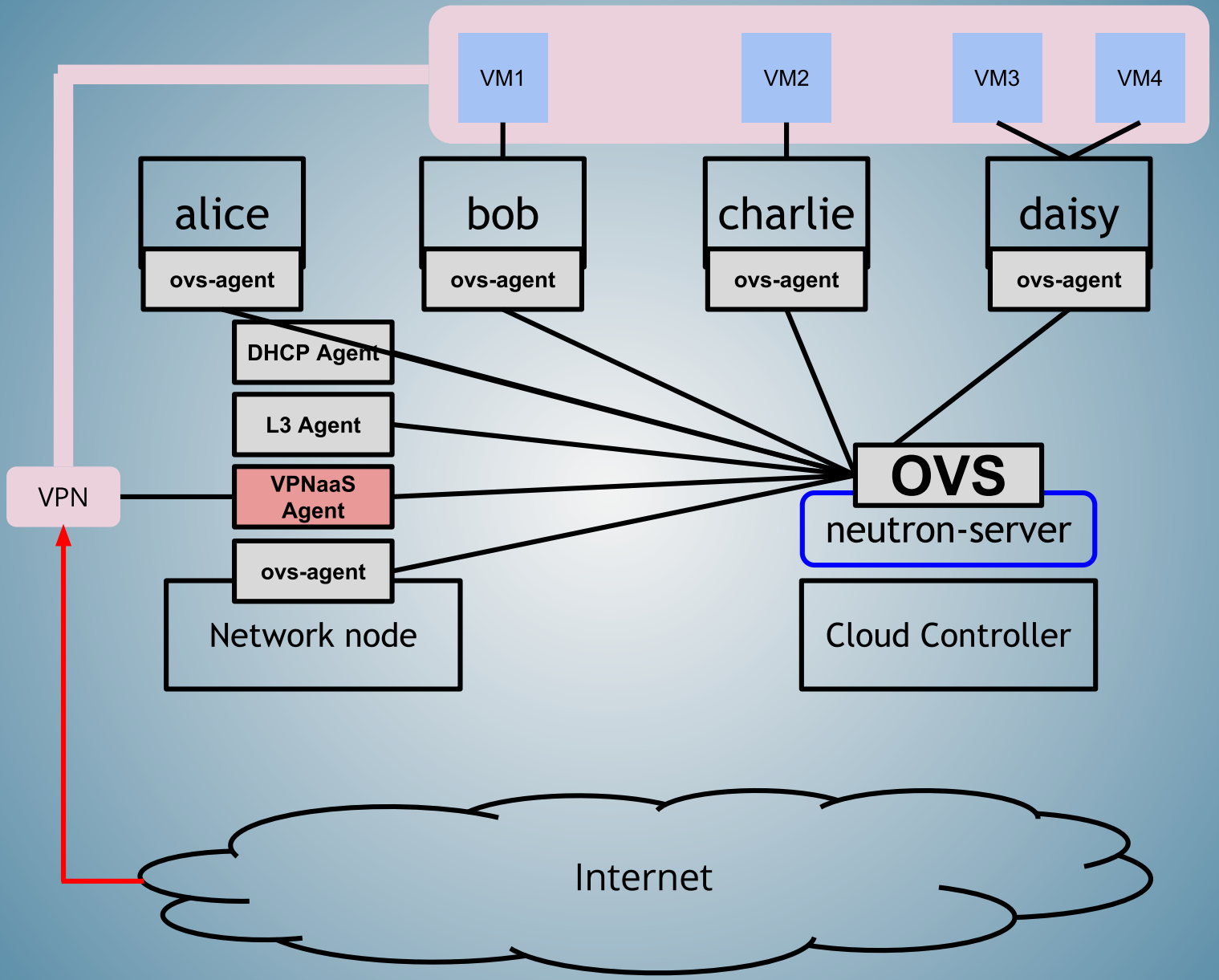

VM1
iptables
VM2
alice
ovs-agent
bob
ovs-agent
charlie
ovs-agent
daisy
ovs-agent
DHCP Agent
L3 Agent
FWaaS Agent
ovs-agent
OVS
neutron-server
iptables
Network node
Cloud Controller
Internet

Load Balancer as a Service (**LBaaS**, seit OpenStack **Havana**)


web1
web2
web3
alice
ovs-agent
bob
ovs-agent
charlie
ovs-agent
daisy
ovs-agent
haproxy
LBaaS
ovs-agent
Network node
OVS
neutron-server
Cloud Controller
Internet

VPN as a Service (**VPNaaS**,
seit OpenStack **Havana**)

h!


VM1
VM2
VM3
VM4
alice
bob
charlie
daisy
ovs-agent
ovs-agent
ovs-agent
ovs-agent
DHCP Agent
L3 Agent
VPNaaS Agent
ovs-agent
VPN
OVS
neutron-server
Network node
Cloud Controller
Internet

Neutron integriert
SDN **nahtlos** in eine
OpenStack-Cloud

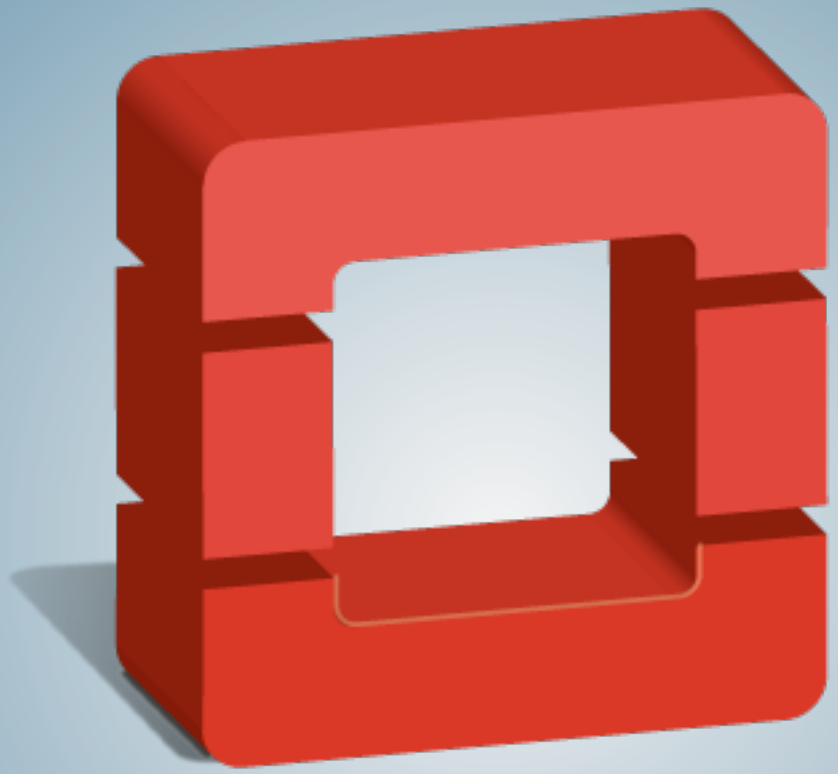
h!

Herausforderungen

Wahl der richtigen
SDN-Implementation

Hoher Lernaufwand,
steile Lernkurve

Mit dem Netz muss sich
die **interne Organisation**
erheblich verändern



openstack™

h!