BEUTH HOCHSCHULE FÜR TECHNIK BERLIN
University of Applied Sciences

Virtualisierte IPv6 Testumgebung mit Xen

Prof. Dipl. Inform. Thomas Scheffler

Frankfurt/Main, 28.05.2009

Übersicht

- Motivation für ein virtualisiertes IPv6-Testsystem
- Anforderungen
- Realisierung
- Ausblick

Motivation

Motivation

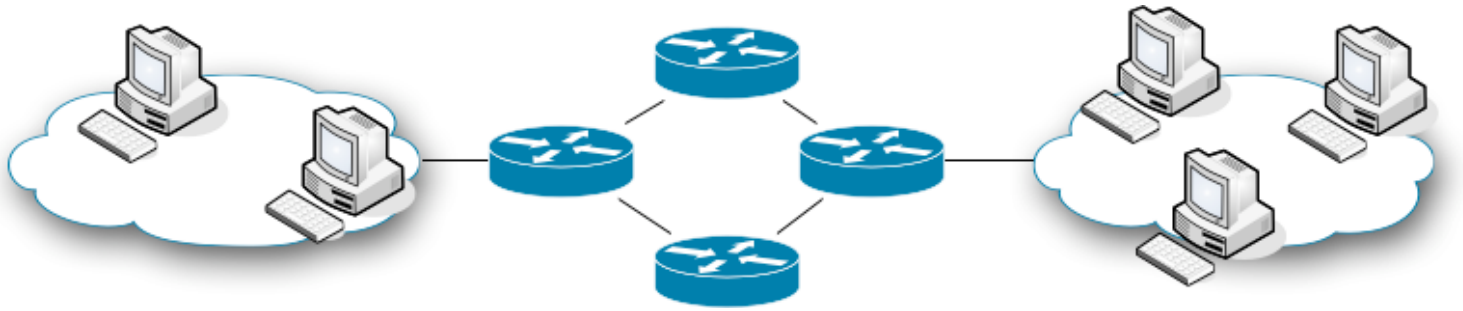
Idee: Test- und Entwicklungsumgebung für IPv6-fähige Systeme und Anwendungen

- Geringer Einfluss auf bestehende Systeme
  - Platzbedarf, Geräuschentwicklung, Energiebedarf, ...
- Kein ‚stand-alone'-Netzwerk, aber hinreichend separiert vom Rest des Labors
- Unterstützung verschiedener Betriebsysteme
- Bedienbar durch Studenten
  - Root-Rechte auf den Maschinen
  - Gespeicherte Standardkonfigurationen
- Flexibel einsetzbar für Lehre und Forschung
- Low-Cost

Virtualisierte Zielarchitektur

- Mehrere Virtuelle Maschinen mit unterschiedlichen Betriebssystemen (Windows, Linux)

- Unterstützung mehrerer physikalischer Netzwerkinterfaces
  - Exklusiver Zugriff der virtualisierten Gast-VM‘s auf Netzwerkinterfaces
  - Externes und internes Routing und Switching der beteiligten Netzwerke ohne VLAN Konfiguration möglich
  - Host hat Zugriff auf den gesamten Netzwerkverkehr (transparent für Gast-VM)



Warum Xen?

- Weit verbreitet
- Ausgereifter Entwicklungsstand
- Gut dokumentiert
- Kostenlos
- Unterstützt flexible Netzwerkkonfigurationen

- Bietet IPv6 Support

Plattform

Hardware

- **PC AMD Athlon 64X2 2,8 GHz**
  - 6 GByte RAM (ca. 512 MByte pro DomU)
  - Netzwerk-Adapter
    - 4 Port D-Link DFE-580TX 100 Mbit/s Ethernet
    - 3Com 10 Mbit/s Ethernet (Internetanbindung)
    - **Gesamtkosten:** ca. 500 Euro
- **Router**
  - Cisco 2610 (2x Ethernet, IOS 12.3)

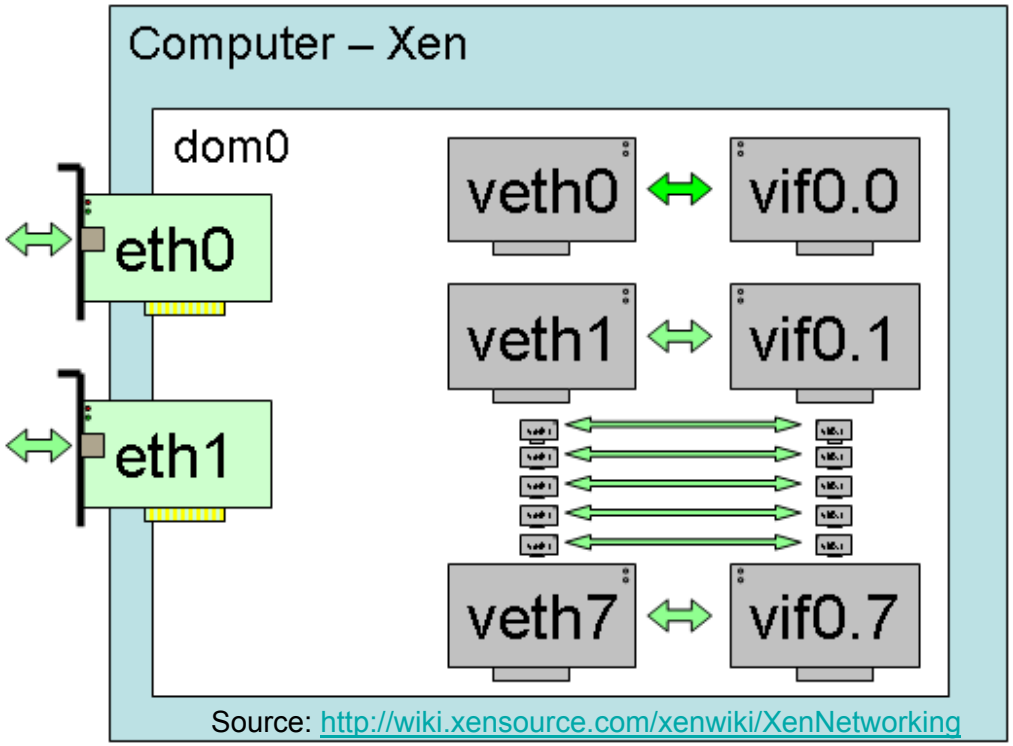
Betriebssysteme

- Dom0: CentOS 5.3
- DomU: CentoOS, Fedora, WindowsXP

Xen Networking

Xen Networking

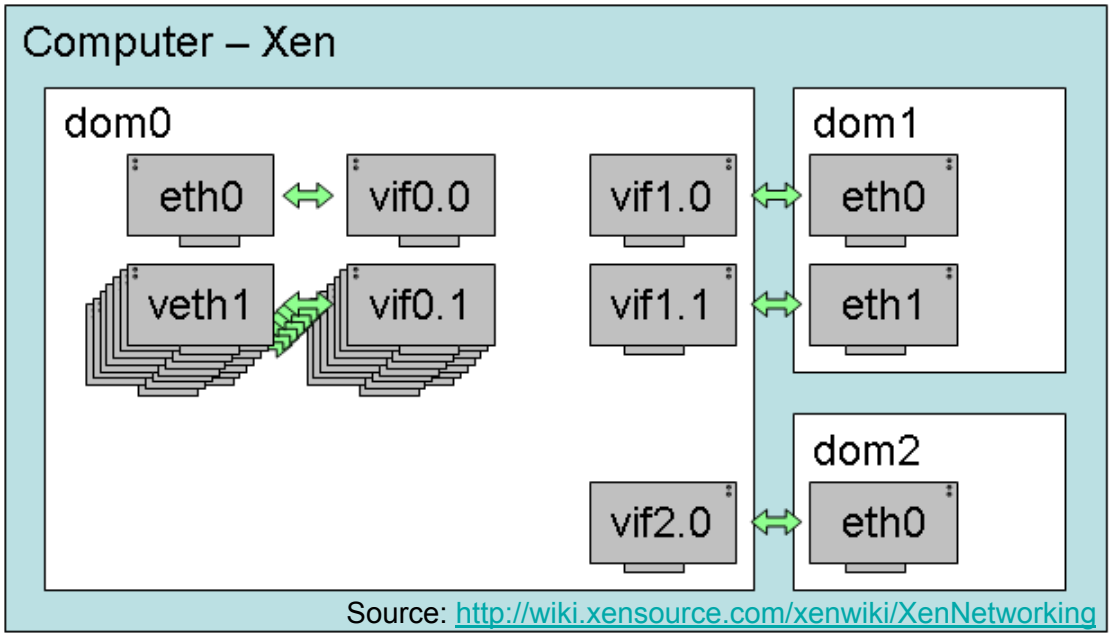
- **Xen Dom0 Networking**
  - Dokumentation http://wiki.xensource.com/xenwiki/XenNetworking
  - Xen erstellt per default 7 virtuelle Netzwerkkarten und Interfaces



Source: http://wiki.xensource.com/xenwiki/XenNetworking

Xen Networking

- **Xen DomU Networking**
  - Wenn eine DomU gestartet wird vergibt Xen eine neue Domain ID für die laufende Instanz
  - Xen erstellt für jede neue DomU ein virtuelles Interface `vif<id#>.0` welches mit der laufenden Instanz verbunden wird.

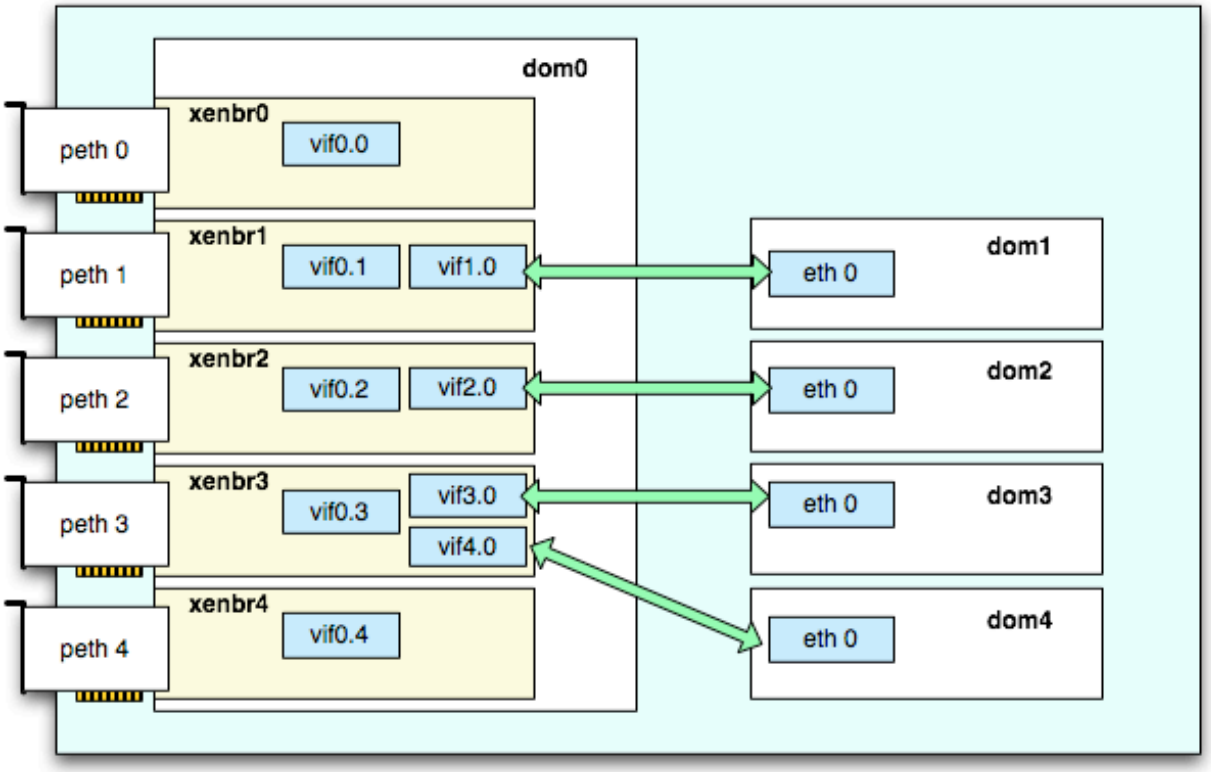


Source: http://wiki.xensource.com/xenwiki/XenNetworking

- **Xen-Bridging**
  - When xend starts up, it runs the network-bridge script, which:
    1. creates a new bridge named `xenbr0`
    2. "real" ethernet interface `eth0` is brought down
    3. the IP and MAC addresses of `eth0` are copied to virtual network interface `veth0`
    4. real interface `eth0` is renamed `peth0`
    5. virtual interface `veth0` is renamed `eth0`
    6. `peth0` and `vif0.0` are attached to bridge `xenbr0`
    7. the bridge, `peth0`, `eth0` and `vif0.0` are brought up

Xen Networking

- Unterstützung mehrerer Netzwerkports (Bridge-basiertes Netz)



- Dokumentation: http://www.debian-administration.org/articles/470

Xen Netzwerkkonfiguration – Dom0

Dedizierter Netzwerkport pro DomU

- Anpassen des Scripts `/etc/xen/xend-config.sxp`

```
...
  #(network-script network-bridge)
  (network-script network-bridge)
  ...
to
  ...
  #(network-script network-bridge)
  (network-script network-xen-multi-bridge)
```

- Hinzufügen eines auf die Hardwarekonfiguration angepassten Scripts `/etc/xen/scripts/network-xen-multi-bridge`

Xen Netzwerkkonfiguration – Dom0

- /etc/xen/scripts/network-xen-multi-bridge

```
#!/bin/sh
# network-xen-multi-bridge
# Exit if anything goes wrong.
set -e
# First arg is the operation.
OP=$1
shift
script=/etc/xen/scripts/network-bridge
case ${OP} in
start)
        $script start vifnum=0 bridge=xenbr0 netdev=eth0
        $script start vifnum=1 bridge=xenbr1 netdev=eth1
        $script start vifnum=2 bridge=xenbr2 netdev=eth2
        $script start vifnum=3 bridge=xenbr3 netdev=eth3
        $script start vifnum=4 bridge=xenbr4 netdev=eth4
        ;;
stop)
        $script stop vifnum=0 bridge=xenbr0 netdev=eth0
...
```

Xen Netzwerkkonfiguration – DomU

- In der DomU Konfigurationsdatei wird angegeben welche mit welcher Xen-Bridge die Virtuelle Maschine verbunden werden soll
  - `/etc/xen/vm.cfg`

```
vif = [ "mac=00:16:3e:0c:19:28,bridge=xenbr2]
```

- Damit lassen sich flexible Zuordnungen der DomU‘s zu den Xen-Bridges und den verbundenen physikalischen Netzwerken realisieren

Xen Netzwerkkonfiguration – Dom0

```
# /usr/sbin/xm list
Name                                        ID Mem(MiB) VCPUs State   Time(s)
Domain-0                                     0     4871     2 r-----   4091.3
WinXP                                        5      359     1 -b----     28.9
centOS_5.3_noX                               6      263     1 -b----    385.9
Attacker_noX                                 8      263     1 -b----     60.0

# /usr/sbin/brctl show
bridge name     bridge id               STP enabled     interfaces
...
xenbr0          8000.feffffffffff       no              peth0
                                                        vif0.0
xenbr1          8000.4e0a3aafd465       no              vif6.0
                                                        peth1
                                                        vif0.1
xenbr2          8000.5ef4c658ea35       no              vif8.0
                                                        vif5.0
                                                        peth2
                                                        vif0.2
xenbr3          8000.feffffffffff       no              peth3
                                                        vif0.3
```

Cisco IPv6 Konfiguration

Cisco IPv6 Konfiguration

router(config)#

```
ipv6 unicast-routing
```

- Startet IPv6 Routing auf dem Cisco-Router

router(config-if)#

```
ipv6 enable
```

- Weisst dem Interface eine Link-Lokale Adresse zu und started das IPv6-Forwarding

router(config-if)#

```
ipv6 address ipv6-prefix/prefix-length [eui-64]
```

- Spezifiziert eine IPv6 Adresse auf dem Interface und started das IPv6-Forwarding (durch Angabe des Schlüsselwortes **eui-64** werden die unteren 64 Bit der Adresse als Interface Identifier (ID) vergeben

Cisco IPv6 Konfiguration

router(config-if)#

```
ipv6 nd prefix ipv6-prefix/prefix-length
```

- Spezifiziert den IPv6-Prefix welcher der Router in IPv6 Router Advertisements den angeschlossenen Systemen bekannt gibt

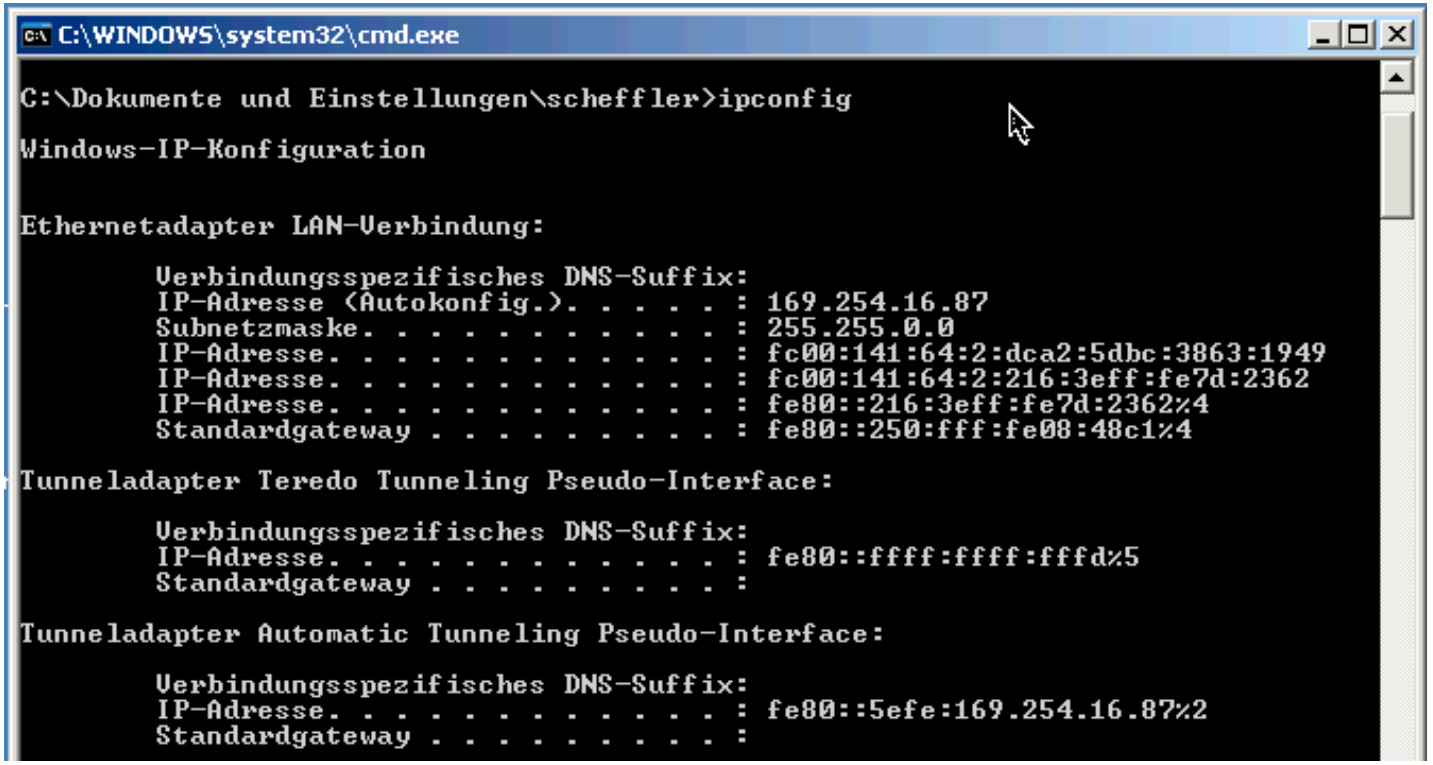
```
ipv6 unicast-routing
!
interface Ethernet0/0
 ip address 10.0.1.1 255.255.255.0
 ipv6 address FC00:141:64:1::1/64
 ipv6 enable
 ipv6 nd prefix FC00:141:64:1::/64
!
```

IPv6 Konfiguration der Virtuellen Maschinen

IPv6 Konfiguration in den DomU‘s

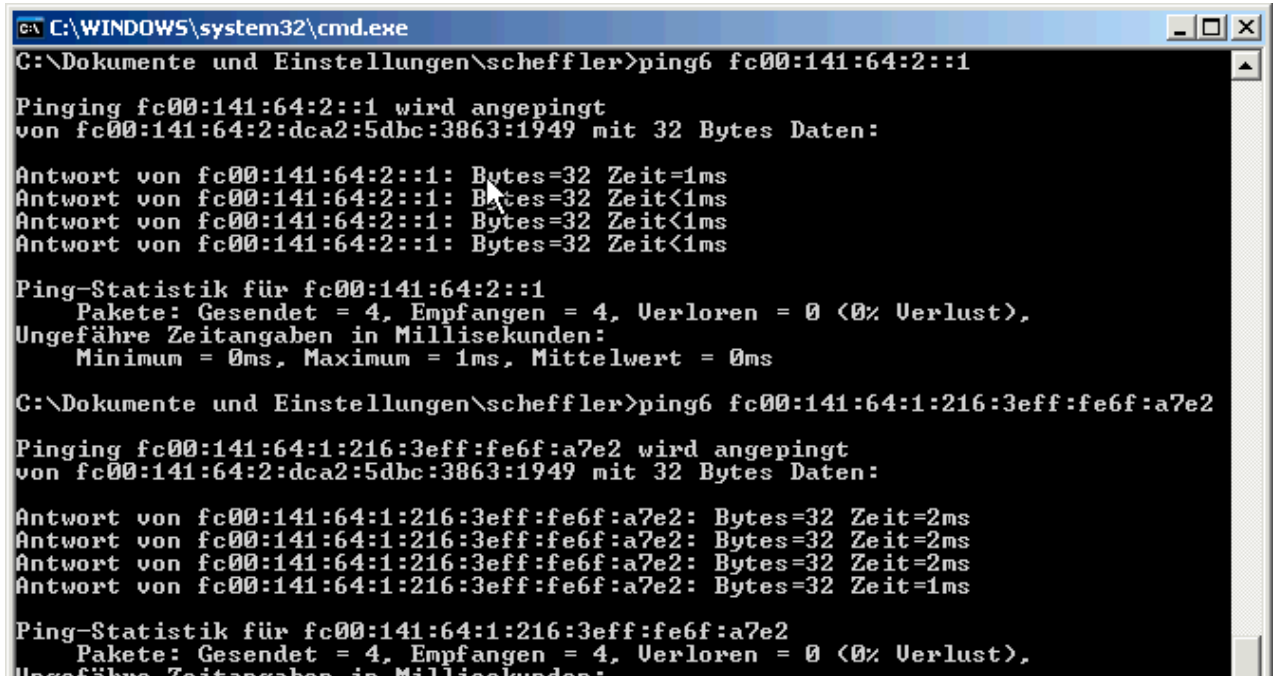
- **WindowsXP**

```
C:\ netsh interface ipv6 install
OK.
```



IPv6 Konfiguration in den DomU‘s

- **WindowsXP** (continued)
  - IPv6 Privacy Extension (RFC 3041) ist per default aktiviert und wird genutzt.



  - RFC3041 kann explizit deaktiviert werden

```
C:\ netsh interface ipv6 set privacy disabled
```

IPv6 Konfiguration in den DomU‘s

- **Linux** (CentOS)
  - IPv6 per default enabled

/etc/sysconfig/network-scripts/ifcfg-eth0

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
NETWORK=10.0.1.0
GATEWAY=10.0.1.1
IPADDR=10.0.1.11
NETMASK=255.255.255.0
IPV6INIT=yes
```

```
[root@localhost ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:16:3E:6F:A7:E2
          inet addr:10.0.1.11  Bcast:10.0.1.255  Mask:255.255.255.0
          inet6 addr: fc00:141:64:1:216:3eff:fe6f:a7e2/64 Scope:Global
          inet6 addr: fe80::216:3eff:fe6f:a7e2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:14 errors:0 dropped:0 overruns:0 frame:0
          TX packets:57 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2811 (2.7 KiB)  TX bytes:9129 (8.9 KiB)
          Interrupt:177 Base address:0xc000
```

Probleme

Zugriff auf die Serielle Schnittstelle des PC

- Xen nutzt `/dev/ttyS0` für die eigene interne Console
- Über die serielle Schnittstelle soll auf den angeschlossenen Cisco-Router zugegriffen werden

- In der Grub Menü-Datei `/boot/grub/menu.lst` wurde die folgende Änderung vorgenommen:
  - `module /vmlinuz-2.6.18-92.1.18.el5xen ro root=/dev/System/root rhgb quiet xencons=tty6`

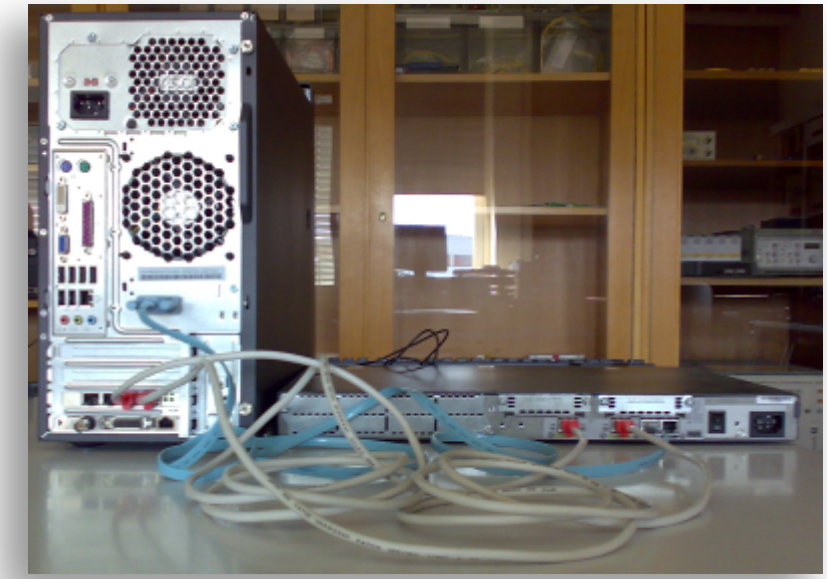
Deutsche Tastaturbelegung in den DomU‘s

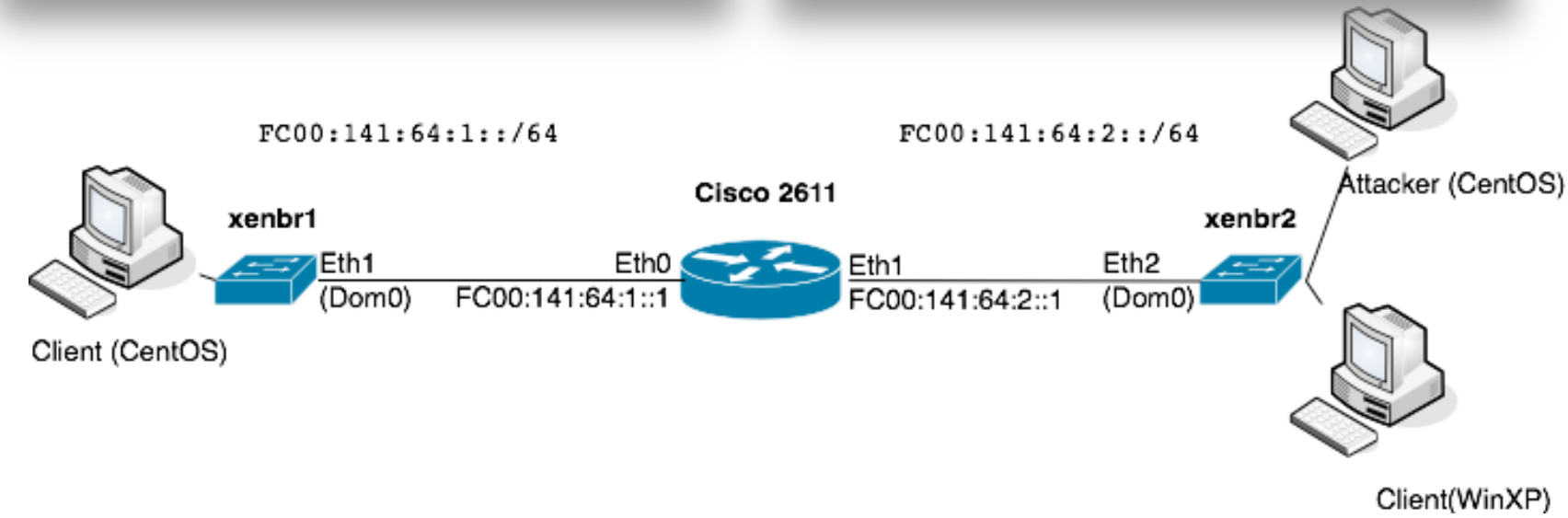
- Hinzufügen von `keymap = 'de'` in den Xen-Konfigurations-dateien

Sicheres Testbed

- Trennung des IPv6-Verkehrs vom sonstigen Laborbetrieb
  - Test und Einsatz von Sniffing und Attack-Tools auf den DomU‘s unkritisch
- Zugang über SSH
  - Public Key Authentisierung
- VNC auf Dom0 und perspektivisch auf DomU‘s
  - Tunneling über SSH

Derzeitiger Stand





Ausblick

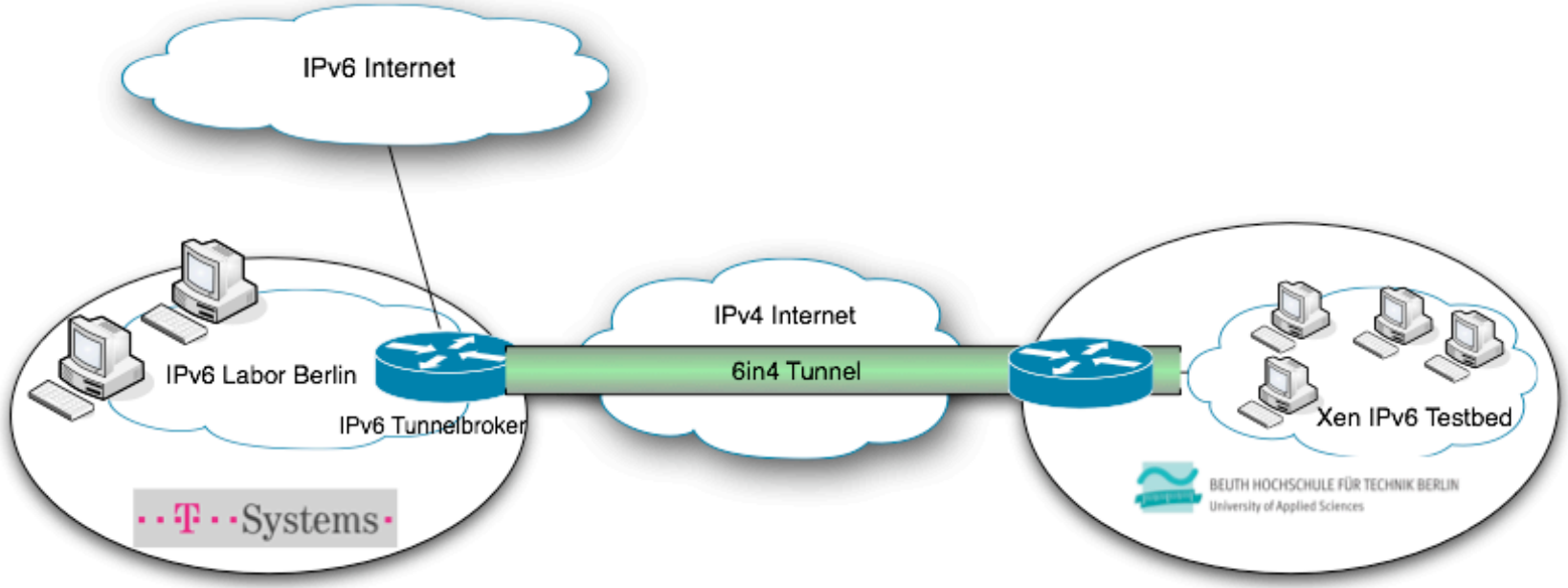
Ausblick

- Snort Installation in der Dom0
  - Verifizierung des Snort IPv6 Entwicklungsstands
  - Beobachtung des IPv6-Verkehrs in und zwischen den Bridges

- Testbed für eigene IPv6 Entwicklungen
  - Embedded IPv6 Webserver
  - ...

Ausblick

- Kooperation mit dem IPv6 Labor der T-Systems in Berlin
  - Kopplung der Labore (Nutzung und Tests der Dienste und Routernetze der Deutschen Telekom)
    - Fester Zugang über T-Systems Labor zum IPv6 Piloten der Deutschen Telekom (konfigurierter Tunnel)
    - Dynamischen Zugang für einzelne DomUs über Hexago-TunnelBroker
    - Durchführung von Basis IPv6-Tests in gemeinsamen Testbed



Fragen?

Prof. Dipl. Inform. Thomas Scheffler

Email: scheffler@beuth-hochschule.de
WWW: prof.beuth-hochschule.de/scheffler