Linux Networking, Ninja-Style

c¼h von sECuRE

NoName e.V., 2012-03-29

powered by LaTeX

Situation (Netzzugang)

- Verbindung via Ethernet (evtl. mal Wireless)
- Adressvergabe via DHCP
- Authentifizierung via (Cisco-)VPN
- Vertraulichkeit via OpenVPN
- Alles instabil, alles modifiziert die routing table

Typisches Setup

```
# apt-get install dhclient vpnc openvpn
# echo 'iface eth0 inet dhcp' >> /etc/network/interfaces
# ifup eth0
# /etc/init.d/vpnc start
# /etc/init.d/openvpn start
```

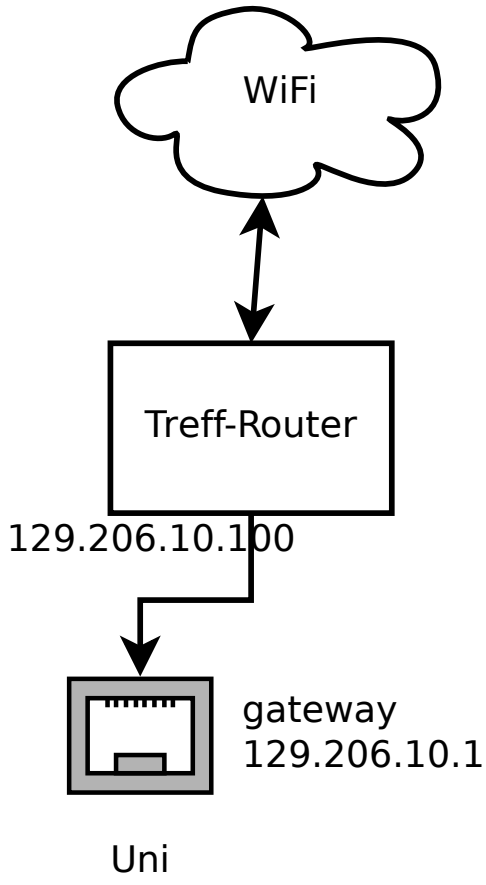
Das Problem

- Alle Komponenten modifizieren die routing table
- Keine weiß von irgendeiner anderen
- Bei Änderungen „in der Mitte“ kracht es außen
- Internetzugang instabil, -EFUN

Workaround (Holzhammer-Methode)

- `ping` in einer Schleife laufen lassen
- Keine Antwort?
  $\rightarrow$ `killall vpnc openvpn dhclient`, neu aufbauen
  $\rightarrow$ Verbindungen oftmals kaputt :-(

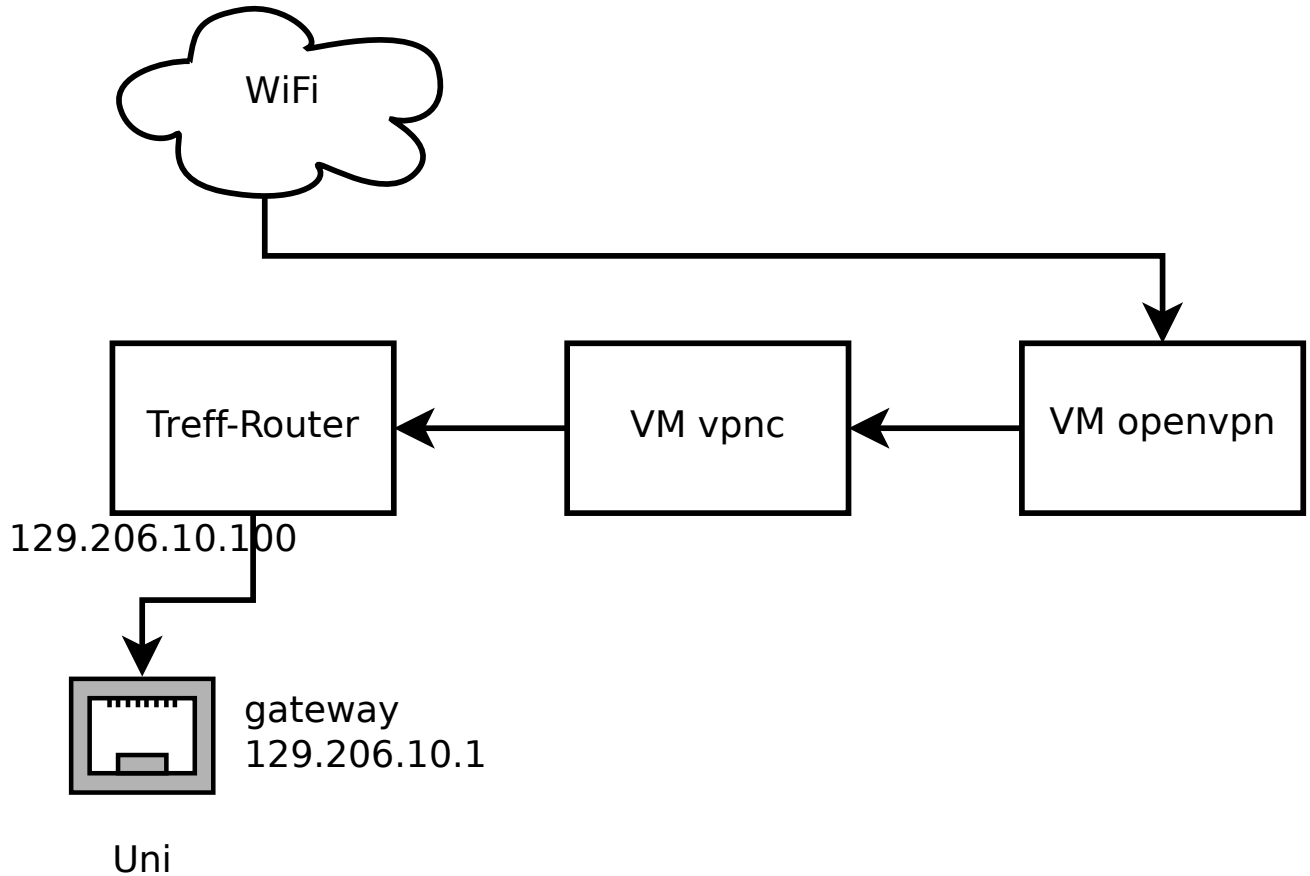
Netz-Übersicht



Die Idee

- Eine VM pro Netz-Ebene:
  eine für DHCP, eine für Cisco-VPN, eine für OpenVPN
  $\rightarrow$ Nur je eine routing table wird modifiziert
- Beinhaltet ganz viel NAT. Ausnahmsweise hilfreich!
- Aber: Router hat einen Intel Atom $\rightarrow$ keine VMs :-|

Übersicht mit VMs



Die Lösung: Namespaces

- Linux network namespaces (netns), seit 2.6.27
- Virtueller Netzwerk-Stack für einen Prozess
- `lxc-unshare -s 'NETWORK' -- /bin/zsh`
- (Host) Neues Bridge-Paar erstellen:
  `ip link add name vto_ns type veth peer name v_ns`
- (Host) Ein Ende in den Namespace stecken:
  `ip link set v_ns netns 1234`

Watchdog

- Saubere Trennung erreicht, fehlt noch Zuverlässigkeit
- Watchdog: 1 Ping/s, 5 Timeouts in Folge gelten als Fehler
- Äußerer und innerer Watchdog:
  outer-wd säubert Netzconfig und startet inner-wd
  inner-wd startet $Befehl und sendet Pings

Watchdog: Zombies

- Im Fehlerfall muss komplett aufgeräumt werden (vpnc-script, …)
- cgroups wären eine Lösung, aber nicht auf dem System
- PID-Namespaces!
- `lxc-unshare -s 'PID|MOUNT' -- /bin/sh -c 'exec inner-wd $PARAMS'`
- Stirbt der Watchdog, sterben alle Prozesse im selben PID-Namespace

Interaktiver, bitte

- Statt einer Shell starten wir `screen`
- Im screen wird das Netz eingerichtet, danach über einen FIFO nach außen kommuniziert
- Danach wird der Watchdog im screen gestartet

Befehle in screen ausführen

```
# Attach ns-$name (-x),
# Select the first window (-p0),
# Run screen command (-X),
# execute command with stdin/stdout/stderr
# connected to screen (as opposed to the
# program running in the screen window)
system(
   "screen -x ns-$name -p0 -X exec ... /bin/sh -c '$command'"
);
```

Befehle in screen injecten

```
system(
    q|screen -x ns-| .
    $name .
    q| -p1 -X eval "stuff \"| .
    $command .
    q|\015\"" >/dev/null|
);
```