

simple security policy editor

Johannes Hubertz

hubertz-it-consulting GmbH

26. Oktober 2005

GUUG - Regionaltreffen West

- Vorstellung, Übersicht
- Firewall
- VPN
- Betrieb
- Ausblick

- 1954 in Köln-Lindenthal geboren
- 1973 Abitur in Köln-Mülheim
- bis 1980 E-technik RWTH und FH Aachen
- ab 1980 bei europ. IT-Hersteller
- ab 2002 bei europ. IT-Dienstleister
- verheiratet, 2 Kinder
- seit 1973 Bundesanstalt THW
- seit 2001 Segeln auf Salzwasser
- seit August 2005 hubertz-it-consulting GmbH

- 1994 Erstkontakt mit IP
- 1996 root@www.bundestag.de, . . .
- 1997 SSLeay, ipfwadm mit shell-scripts
- 1998 Ins Allerheiligste, iX, Heise Verlag
- 1998 ipfwadm mit LPFC
- 1999 IT-Security Mgr. D-A-CH
- 2001 Gibraltar, FreeSwan, iptables . . .
- 2001 Erste Gedanken zu sspe, Reinraum
- 2002 April: Online mit 2 Standorten

- Bellovin and Cheswick: Firewalls and Internet Security
- Fazit: keep it simple!

- zentrale Administration mit **minimalem** Aufwand
- mehrere Standorte am Internet mit internen privaten Netzen
- verteilte Firewall für beliebig viele Server und User-PC
- voll vermaschtes IPSec-VPN mit FreeSwan, X.509 oder PreSharedKeys

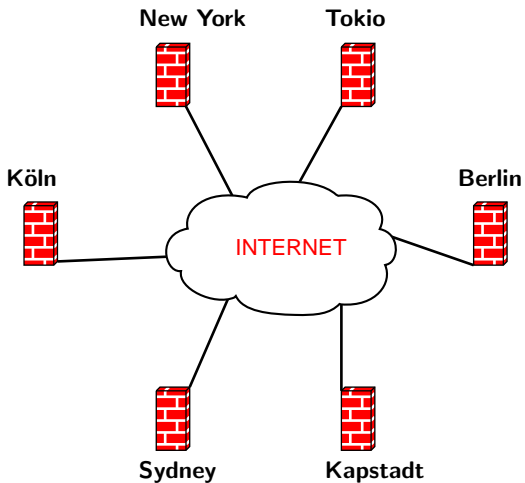
Übersicht: Randbedingungen

- Bash und Perl sichern einfache Nachvollziehbarkeit
- Verschlüsselung: IPSec und ssh, anerkannte kryptographische Sicherheit
- Freie Software: Quellen mit überprüfbarer Sicherheit
- Freie Software: dauerhafte und zuverlässige KnowHow-Quelle

SSPE unterliegt der
GNU General Public License

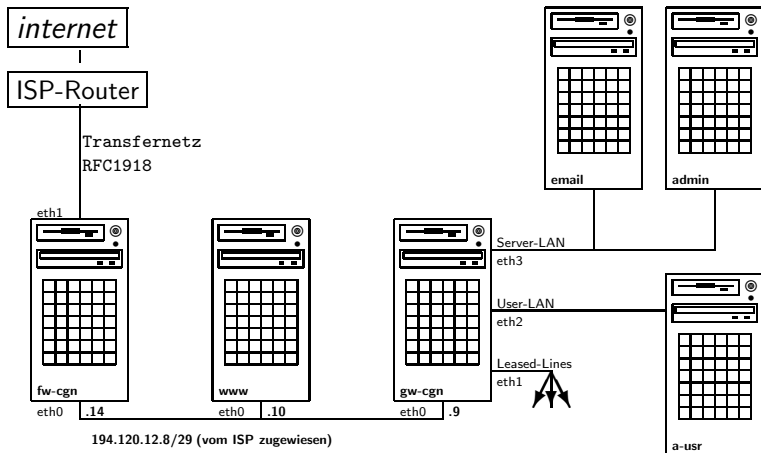


Übersicht: das Firmennetzwerk



6 Standorte an beliebigen Internet-Providern

Übersicht: ein typischer Firmenstandort



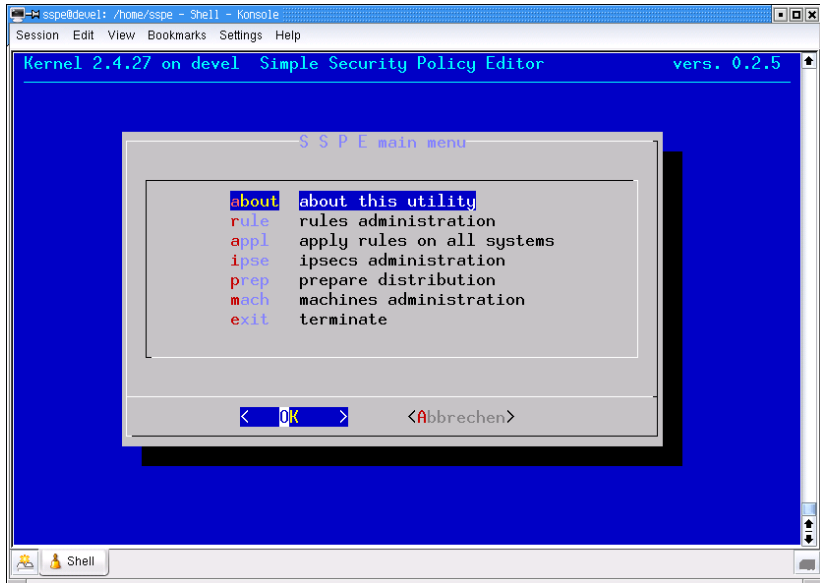
Der Standort des Admin-PC spielt keine Rolle.

- Vorstellung, Übersicht
- Firewall
- VPN
- Betrieb
- Ausblick

- Minimalsystem aus debian/stable und debfoster
- monolithischer Kernel
- root, sonst keine Benutzer
- keine unnötigen Services, nur ssh

- Admins Traum: sowenig Arbeit wie möglich \iff **Faulheit** stärkt die Glieder
- **zentrale** Administration \Rightarrow **Konsistenz**
- Fehler führen nicht zum Abbruch \Rightarrow **Verfügbarkeit**
- Top-Down Softwareentwurf
- Inselumgebung für die ersten Versuche
- LinuxTM und CiscoTM als erste Plattformen
- Dialog als Rahmen

Firewall: dialog



Hauptmenü

Firewall: Verzeichnisstruktur

```
/home/sspe/adm/  
/home/sspe/adm/bin           # Programme  
/home/sspe/adm/etc          # gobale Konfig  
/home/sspe/adm/desc         # Maschinen  
/home/sspe/adm/desc/fw-ber  # Firewall Berlin  
/home/sspe/adm/desc/fw-cgn  # Firewall Cologne  
/home/sspe/adm/desc/fw-nyc  # Firewall New York  
/home/sspe/adm/desc/gw-ber  # Gateway Berlin  
/home/sspe/adm/desc/gw-cgn  # Gateway Cologne  
/home/sspe/adm/desc/gw-nyc  # Gateway New York  
/home/sspe/adm/desc/www-cgn # DNS/Webserver Cologne  
/home/sspe/adm/hardware     # Hardware-versionen  
/home/sspe/adm/software     # Software-, Kernel-versionen
```

Verzeichnis einer Maschine: (gw-cgn)

```
-rw-r--r-- 1 sspe sspe 34 Apr 13 2004 apply-options
-rw-r--r-- 1 sspe sspe 2042169 Apr 18 16:52 commented-rules
-rw-r--r-- 1 sspe sspe 1971780 Apr 1 18:55 commented-rules.old
-rw-r--r-- 1 sspe sspe 27 Jun 20 2003 desc
lrwxrwxrwx 1 sspe sspe 17 Jun 20 2003 hostnet -> ../../etc/hostnet
-rw-r--r-- 1 sspe sspe 7 Jun 18 2003 hw
-rw-r--r-- 1 sspe sspe 12 Jun 18 2003 ip
-rw-r--r-- 1 sspe sspe 878973 Apr 18 16:52 iptable-rules
-rw-r--r-- 1 sspe sspe 844513 Apr 1 18:55 iptable-rules.old
-rw-r--r-- 1 sspe sspe 535 Apr 1 10:35 mangle-end
-rw-r--r-- 1 sspe sspe 135 Apr 1 10:35 mangle-start
-rw-r--r-- 1 sspe sspe 16926 Apr 18 16:51 parameter
-rw-r--r-- 1 sspe sspe 3315 Apr 18 16:51 routes
lrwxrwxrwx 1 sspe sspe 18 Jun 20 2003 nathosts -> ../../etc/nathosts
lrwxrwxrwx 1 sspe sspe 18 Jun 20 2003 privates -> ../../etc/privates
lrwxrwxrwx 1 sspe sspe 21 Jun 20 2003 rules.admin -> ../../etc/rules.admin
lrwxrwxrwx 1 sspe sspe 21 Jun 20 2003 rules.ipsec -> ../../etc/rules.ipsec
-rw-r--r-- 1 sspe sspe 115 Feb 18 2004 rules.local
-rw-r--r-- 1 sspe sspe 216 Jun 25 2003 rules.tail
lrwxrwxrwx 1 sspe sspe 21 Jun 20 2003 rules.users -> ../../etc/rules.users
```

apply-options:

```
sleep-before=3
wait-before=fw-cgn
```


Definitionen in CIDR-Notation:

```
# File: hostnet
# Name      Address          # Comment
#
any         0.0.0.0/0       # the whole      internet
many       0.0.0.0/1       # lower half     internet
many       128.0.0.0/1     # upper half     internet
#
a-usr      192.168.1.126/32 # Alice          user-LAN
a-usr      192.168.1.125/32 # Bob            user-LAN
admin      192.168.1.193/32 # sspe-home     server-LAN
gw-cgn     192.168.1.222/32 # gateway cologne server-LAN
gw-cgn-e   194.120.12.9/32  # gateway cologne external
cgn-e      194.120.12.8/29  # cologne net   external
fw-cgn     194.120.12.14/32 # firewall cologne external
user-cgn   192.168.1.0/25   # users         user-LAN
cgn-net    192.168.1.0/24   # cgn completely internal
```

Gruppierung erfolgt durch Namensgleichheit

Firewall: rules

```
# File: rules.admin
# Src      Dst        Dir Prot  Port  Action Options
#
a-usr      admin      1    tcp   ssh   accept INSEC
many       admin      1    tcp   ssh   deny
admin      gw-cgn     1    tcp   ssh   accept
#

Dir      = [ 1 | 2 ]
Prot     = [ ip | icmp | tcp | udp | esp | 0 ... 255 ]
Port     = [ name | num = 0 ... 65535 | :num | num: | num1:num2 ]
Action   = [ accept | reject | deny ]
```

Zeitliche Abhängigkeiten während der Generierung

apply-options (sleep, wait)

Inhaltliche Abhängigkeiten der generierten Kommandos

- Host-, Netzdefinitionen
- Firewall Regelsatz
- Interfaces, Routingtabelle
- nathosts, privates
- Paketmangling-Dateien

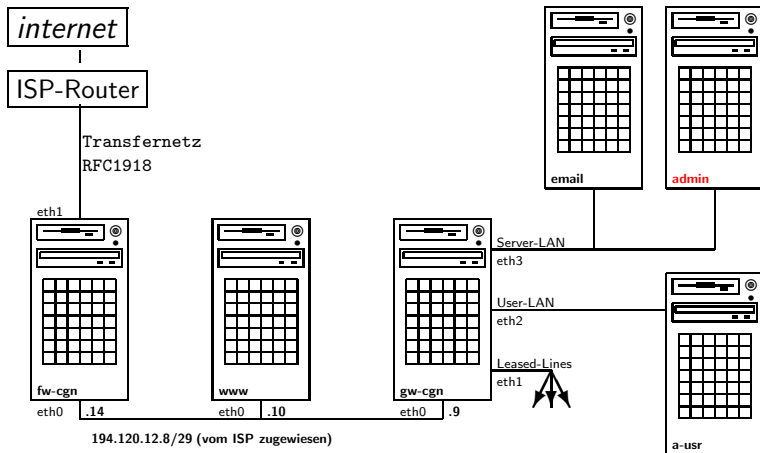
Firewall: Prolog

```
/sbin/iptables -P INPUT ACCEPT
/sbin/iptables -P OUTPUT ACCEPT
/sbin/iptables -P FORWARD ACCEPT
/sbin/iptables -F >/dev/null 2>/dev/null
/sbin/iptables -t nat -F >/dev/null 2>/dev/null
/sbin/iptables -F tcp__tab >/dev/null 2>/dev/null
/sbin/iptables -X tcp__tab >/dev/null 2>/dev/null
/sbin/iptables -F udp__tab >/dev/null 2>/dev/null
/sbin/iptables -X udp__tab >/dev/null 2>/dev/null
/sbin/iptables -F icmp_tab >/dev/null 2>/dev/null
/sbin/iptables -X icmp_tab >/dev/null 2>/dev/null
/sbin/iptables -F IPSEC >/dev/null 2>/dev/null
/sbin/iptables -X IPSEC >/dev/null 2>/dev/null
/sbin/iptables -F logdrop >/dev/null 2>/dev/null
/sbin/iptables -X logdrop >/dev/null 2>/dev/null
/sbin/iptables -N logdrop
/sbin/iptables -A INPUT -i lo -j ACCEPT
/sbin/iptables -A OUTPUT -o lo -j ACCEPT
/sbin/iptables -A INPUT -s 127.0.0.1/8 -j logdrop
/sbin/iptables -A FORWARD -s 127.0.0.1/8 -j logdrop
/sbin/iptables -N IPSEC
/sbin/iptables -A FORWARD -p esp -j IPSEC
/sbin/iptables -A FORWARD -p ah -j IPSEC
/sbin/iptables -A FORWARD -p ipencap -j IPSEC
/sbin/iptables -N tcp__tab
/sbin/iptables -A FORWARD -p tcp -j tcp__tab
/sbin/iptables -N udp__tab
/sbin/iptables -A FORWARD -p udp -j udp__tab
/sbin/iptables -N icmp_tab
/sbin/iptables -A FORWARD -p icmp -j icmp_tab
```

Firewall: Epilog

```
/sbin/iptables -A INPUT -j logdrop
/sbin/iptables -A OUTPUT -j logdrop
/sbin/iptables -A FORWARD -j logdrop
/sbin/iptables -A logdrop -j LOG --log-tcp-options --log-ip-options
                --log-level 7 --log-prefix "gw-cgn-dropped: "
                -m limit --limit 3/second --limit-burst 6
/sbin/iptables -P INPUT DROP
/sbin/iptables -P OUTPUT DROP
/sbin/iptables -P FORWARD DROP
```

Firewall: Generierung für Admin



firewall: iptables für admin

File: iptables-rules

```
# File: iptables-rules for admin
/sbin/iptables -A INPUT -i eth0 \
  -s 192.168.1.126/32 -d 192.168.1.193/32 \
  -p tcp --sport 0: --dport ssh \
  -m state --state NEW,ESTABLISHED,RELATED \
  -j ACCEPT

/sbin/iptables -A OUTPUT -o eth0 \
  -s 192.168.1.193/32 -d 192.168.1.126/32 \
  -p tcp --sport ssh --dport 0: \
  -m state --state ESTABLISHED,RELATED \
  -j ACCEPT
```

Regel:

```
a-usr      admin    1  tcp  ssh  accept INSEC
```

firewall: iptables für gw-cgn

File: iptables-rules

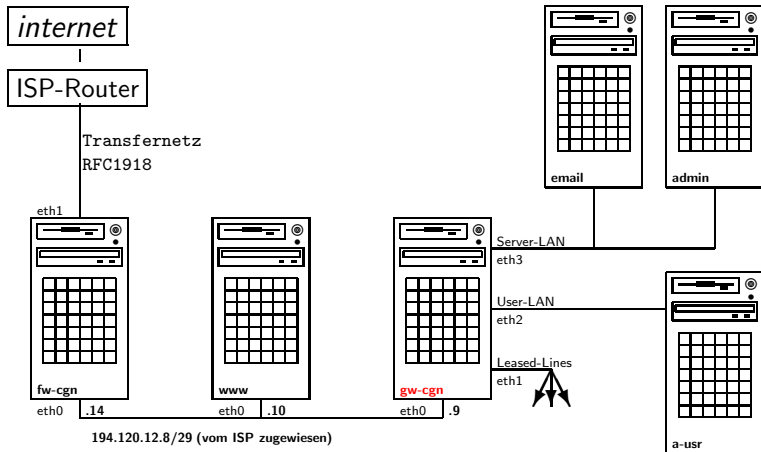
```
# File: iptables-rules for gw-cgn
/sbin/iptables -A tcp__tab \
  -s 192.168.1.126/32 -d 192.168.1.193/32 \
  -p tcp --sport 0: --dport ssh \
  -m state --state NEW,ESTABLISHED,RELATED \
  -j ACCEPT

/sbin/iptables -A tcp__tab \
  -s 192.168.1.193/32 -d 192.168.1.126/32 \
  -p tcp \
  -m state --state ESTABLISHED,RELATED \
  -j ACCEPT
```

Regel:

```
a-usr      admin    1  tcp  ssh  accept INSEC
```


Firewall: Regeln für gw-cgn



Private Netze untereinander: nie NAT!

```
# File: privates
172.16.0.0/16    # Berlin
172.17.0.0/16    # Cologne
172.18.0.0/16    # New York
172.19.0.0/16    # Kapstadt
172.20.0.0/16    # Sydney
172.21.0.0/16    # Tokio
```

Wer macht NAT für wen wo?

```
# File: nathosts
172.16.0.0/16      194.120.12.9      # Berlin
172.17.0.0/16     192.168.111.1     # Cologne
172.21.0.0/16     192.168.119.1     # Tokio
```

Firewall: packet-mangling

packet-mangling:

```
#
# experimental
#
# FILE: mangle-start for gw-cgn
#
# force icmp to minimize-delay
#
/sbin/iptables -t mangle -F
#
# 0x10 = minimize Delay!
/sbin/iptables -t mangle -A PREROUTING -p icmp -j TOS --set-tos 0x10
#
```

- Zentrale Generierung der Kommandos
- Verteilung per scp, Anwendung der Regeln durch ssh
- Reihenfolge der Zielmaschinen steuerbar (apply-options)
- Anstoss des Vorgangs aus einem Dialog-Menue
- Interface-Konfiguration und Routingtabelle sind Grundlage

- Splittung des Regelsatzes: admin, head, ipsec, local, users, tail
- Regelsuche: erst im lokalen, dann im gemeinsamen Verzeichnis
- Optionen in Regeln: LOG, NONAT, NOIF, DNS, FTP, SYSL, NTP, IPSEC, VNC, ...

Firewall: Regeloptionen

```
if (/LOG/)      { $logging=1;                }
if (/INSEC/)   { $sport=$insec;              }
if (/DNS/)     { $sport=53;                  }
if (/TIMED/)   { $sport=525; $sport=525;     }
if (/WHO/)     { $sport=513; $sport=513;     }
if (/SYSL/)    { $sport=514; $sport=514; $nstate=1; }
if (/NTP/)     { $sport=123;                 }
if (/FTP/)     { $sport="ftp-data"; $dport=$unpriv; }
if (/NETBIOS/) { $sport="137:139";           }
if (/BOOTP/)   { $sport=67;                 }
if (/AUTH/)    { $sport=113;                }
if (/IPSEC/)   { $sport=500; $nstate=1; $noif=1; }
if (/NONAT/)   { $nonat=1;                  }
if (/NOSTATE/) { $nstate=1;                 }
if (/NOIF/)    { $noif=1;                   }
if (/FORCED/)  { $forced=1;                 }
if (/VNC/)     { $sport=$insec; $dport="5900";
                 $nstate=1; $noif=1; }

```

- NAT ausschliesslich an Gateways auf eth0
- IPSec ebenfalls nur an Gateways auf eth0
- temp. Deaktivierung eines Zielgerätes
- Fehler werden erst angezeigt, wenn alle Ziele fertig sind

- syslog lokal oder remote
- Drop-Regeln vermindern Logaufkommen
- Logging auf Accept-Regel hilft entstören

ssh und IPSec

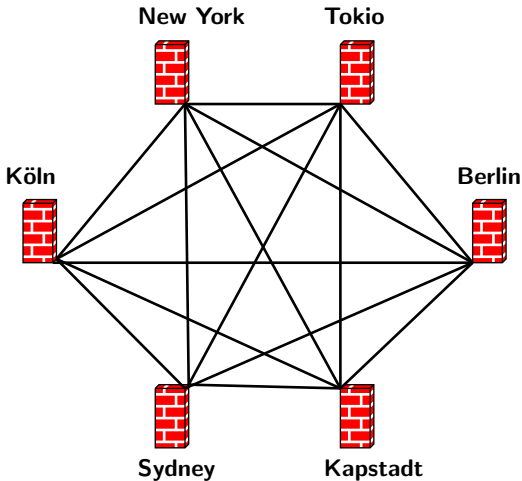
- ausschliesslich ssh zur Administration
- IPSec und ssh nicht wechselseitig abhängig
- ssh durch IPSec nur zu internen Maschinen ohne IPSec
- IPSec verändert Routing, hat also Einfluß auf Generierung!

Erkenntnis:

Paranoid zu sein bedeutet nicht, daß keiner hinter einem her wäre!

- Vorstellung, Übersicht
- Firewall
- **VPN**
- Betrieb
- Ausblick

VPN: das Firmennetzwerk



6 Standorte an beliebigen Internet-Providern
per IPSec voll vermascht mit $S * (S - 1) = 30$ Tunneln

- **Gleiche** ipsec.conf an allen Standorten
- Voraussetzung: **alle** sind erreichbar
- Zeitsteuerung per cron, ntp
- Overhead für Änderungen ist erträglich, **30** Sekunden downtime

- Konfiguration und PreSharedKeys aus `sspe-konfig: ipsecs`
- voll vermaschtes Netz, singuläre Standort-Anbindung zusätzlich möglich
- Verteilung mit `scp: ipsec.conf.new`
- supervisor-script prüft und aktiviert Konfiguration

VPN: ipsec-supervisor mit Vermaschung

Script am vernetzten Standort:

```
#!/bin/bash
if [ -f /etc/ipsec.secrets.new ] ; then
    if [ -f /etc/ipsec.conf.const ] ; then
        cat /etc/ipsec.conf.const >> /etc/ipsec.conf
    fi
    mv /etc/ipsec.secrets.new /etc/ipsec.secrets
    /etc/init.d/ipsec restart
fi
```

ipsec.conf und ipsec.secrets.new werden gemeinsam übertragen
ipsec.conf.const enthält die Konfiguration für singuläre Anbindungen
und wird manuell einmal erstellt und auf die beiden Endpunkte verteilt

crontab:

```
* * * * * /root/bin/ipsec-supervisor >/dev/null 2>/dev/null
```

VPN: ipsec-supervisor ohne Vermaschung

Script am Standort mit singulärer Anbindung:

```
#!/bin/bash
if [ -f /etc/ipsec.secrets.new ] ; then
    if [ -f /etc/ipsec.conf.const ] ; then
        cat /etc/ipsec.conf.const > /etc/ipsec.conf
    fi
    mv /etc/ipsec.secrets.new /etc/ipsec.secrets
    /etc/init.d/ipsec restart
fi
```

ipsec.conf und ipsec.secrets.new werden gemeinsam übertragen
ipsec.conf.const enthält die Konfiguration für singuläre Anbindungen
und wird manuell einmal erstellt und auf die beiden Endpunkte verteilt

crontab:

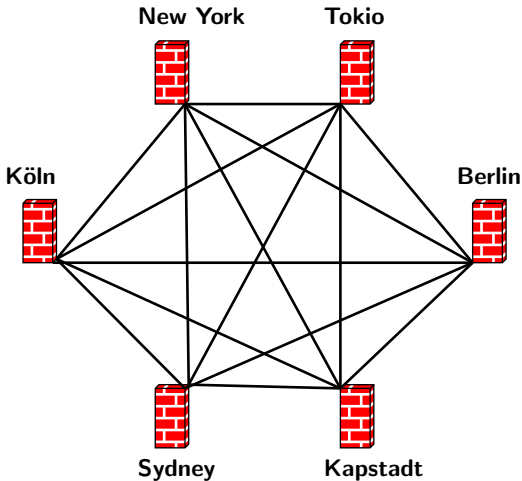
```
* * * * * /root/bin/ipsec-supervisor >/dev/null 2>/dev/null
```


VPN: ipsecs Konfigurationsdatei

# loc.	gateway	next-Hop	subnet
bln	172.22.0.41	172.22.0.46	10.11.48.0/21
cgn	172.22.0.25	172.22.0.30	10.11.40.0/21
nyc	172.22.0.65	172.22.0.70	10.11.4.0/22
sdv	172.22.0.17	172.22.0.22	10.0.0.0/8
kap	172.22.0.9	172.22.0.14	10.11.56.0/21
tok	172.22.0.1	172.22.0.6	10.11.16.0/21
to2	172.22.0.1	172.22.0.6	10.11.80.0/21

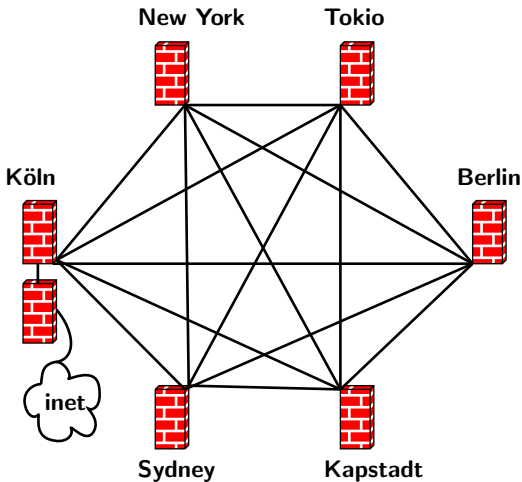
Hieraus werden alle ipsec.conf und ipsec.secrets generiert

VPN: das Firmennetzwerk vor dem Umbau



6 Standorte an beliebigen Internet-Providern
per IPSec voll vermascht mit $S * (S - 1) = 30$ Tunneln

VPN: das Firmennetzwerk nach dem Umbau



1 ISP + 6 Standorte an einem ISP-MPLS-VPN,
per IPSec voll vermascht mit $S * (S - 1) + 60 = 90$ Tunneln

VPN: ipsec Konfigurationsdatei

```
# loc.      gateway          next-Hop          subnet
bln         172.22.0.41     172.22.0.46     10.11.48.0/21
cgn         172.22.0.25     172.22.0.30     10.11.40.0/21
nyc         172.22.0.65     172.22.0.70     10.11.4.0/22
sdyc        172.22.0.17     172.22.0.22     10.0.0.0/8
kap         172.22.0.9      172.22.0.14     10.11.56.0/21
tok         172.22.0.1      172.22.0.6      10.11.16.0/21
to2         172.22.0.1      172.22.0.6      10.11.80.0/21
I01         172.22.0.25     172.22.0.30     0.0.0.0/1
I02         172.22.0.25     172.22.0.30     128.0.0.0/3
I03         172.22.0.25     172.22.0.30     160.0.0.0/5
I04         172.22.0.25     172.22.0.30     168.0.0.0/6
I05         172.22.0.25     172.22.0.30     172.0.0.0/12
### !!! never open next line or gateways will be lost !!!!
### !!!Ixx 172.22.0.25 172.22.0.30 172.16.0.0/12 !!!
I06         172.22.0.25     172.22.0.30     172.32.0.0/11
I07         172.22.0.25     172.22.0.30     172.64.0.0/10
I08         172.22.0.25     172.22.0.30     172.128.0.0/9
I09         172.22.0.25     172.22.0.30     173.0.0.0/8
I10         172.22.0.25     172.22.0.30     174.0.0.0/7
I11         172.22.0.25     172.22.0.30     176.0.0.0/4
I12         172.22.0.25     172.22.0.30     192.0.0.0/3
```

Hieraus werden alle ipsec.conf und ipsec.secrets generiert

- jedes VPN-GW ist anders
- exakte Konfiguration erzeugen
- Routen des Internet per IPSec möglich
- Routinglücke für ssh zur Administration
- Overhead für Änderungen bleibt erträglich, 36 Sekunden downtime

- Handlungsreisende (roadwarrior) mit X.509-Authentisierung
- Sperrliste für einzelne Clients: CRL der PKI
- VPN-Dialer (Thomas Kriener) zur Einbindung von beliebigen Einzel-PC
- L2TP durch IPSec zur Änderung des Routings im PC

- Vorstellung, Übersicht
- Firewall
- VPN
- **Betrieb**
- Ausblick

- Betrieb seit April 2002 für mehrere Kunden
- ca. 50 Maschinen mit iptables gesichert
- einige hundert Anwender-PC geschützt
- Kosten drastisch minimiert

- Debian macht **security-fixes** einfach
- Debian stable verdient seinen Namen
- RedHat funktioniert auch, SuSe vermutlich ebenso
- Scriptänderungen einfach machbar
- Erweiterungen
- z.B. HA, dyn.Routing, . . .

- Vorstellung, Übersicht
- Firewall
- VPN
- Betrieb
- **Ausblick**

- MAC-Adressen im DHCP-Server und Firewalls fixieren
- Cisco-Access-List Generierung fertigstellen
- Programmierung für BSD, Solaris, xyz
- Revisionsfähigkeit durch Datenbank ?
- Quellen unter <http://sspe.sourceforge.net>
- Beteiligung weiterer Entwickler erwünscht!

- März 2003: <http://sspe.sourceforge.net/>
- Georg Greve: Brave GNU World # 54 Oktober 2003
<http://www.linux-magazin.de/Artikel/ausgabe/2003/11/gnu/gnu.html>
- Linux-Magazin 04/2005, März 2005
Wege sind das Ziel, HA-VPN
- Computer Club Pascal, Köln, 24. Mai 2005
Vortrag
- Linux-Magazin 07/2005, Juni 2005
Treuer Hirtenhund
- Linux-Enterprise 04/2005, August 2005
Artikel zu sspe
- GUUG-Regionaltreffen West, 26.Okt.2005, Köln

Ich bedanke mich für Ihre Aufmerksamkeit und stelle mich gerne Ihren Fragen ...