SSPE

Simple Security Policy Editor

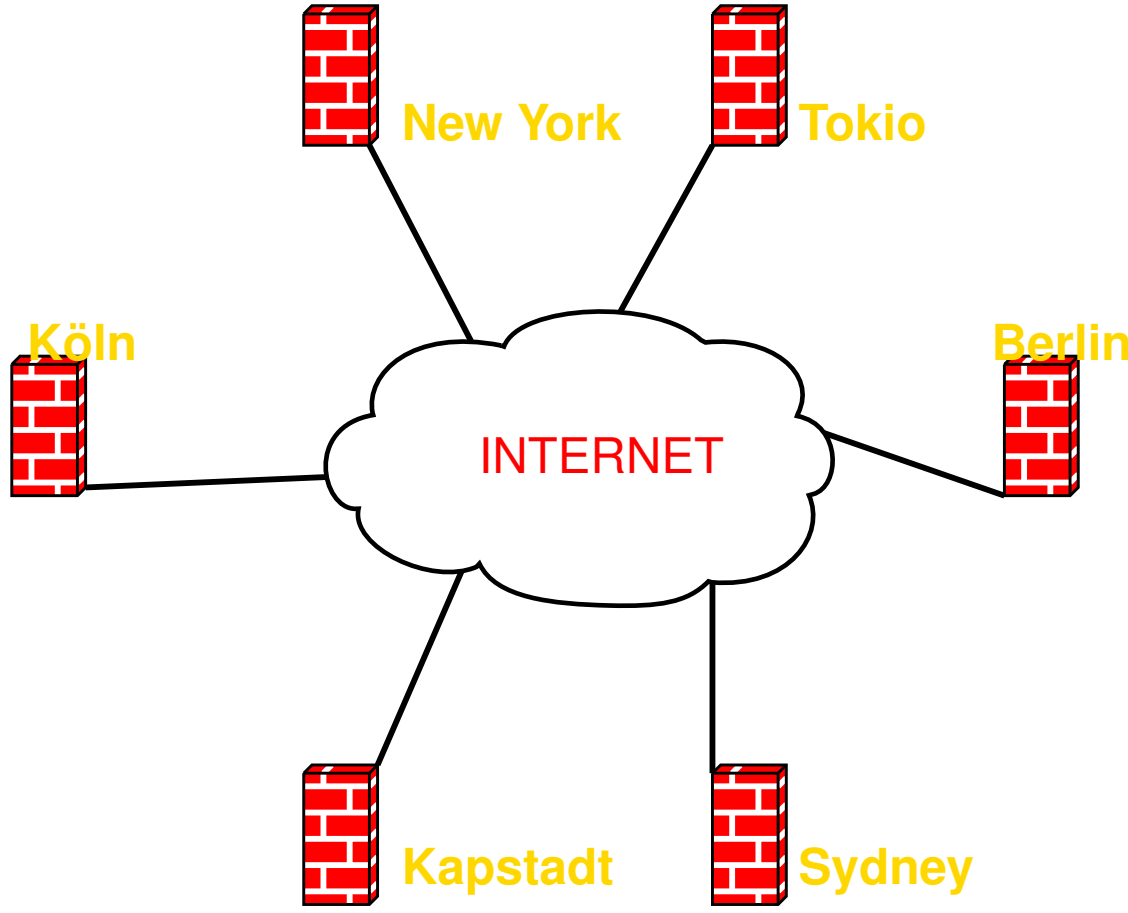
Johannes Hubertz

SSPE: Gliederung

1. Übersicht

2. Firewall
3. VPN
4. Betrieb
5. Ausblick

SSPE: Ein Firmennetzwerk



6 Standorte an beliebigen Internet-Providern

SSPE: Übersicht I

- zentrale Administration mit minimalem Aufwand

SSPE: Übersicht I

- zentrale Administration mit minimalem Aufwand
- mehrere Standorte am Internet mit internen privaten Netzen

SSPE: Übersicht I

- zentrale Administration mit minimalem Aufwand
- mehrere Standorte am Internet mit internen privaten Netzen
- verteilte Firewall für beliebig viele Server und User-PC

SSPE: Übersicht I

- 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

SSPE: Übersicht II

- Bash und Perl sichern einfache Nachvollziehbarkeit

SSPE: Übersicht II

- Bash und Perl sichern einfache Nachvollziehbarkeit
- Verschlüsselung: IPSec und ssh, anerkannte kryptographische Sicherheit

SSPE: Übersicht II

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

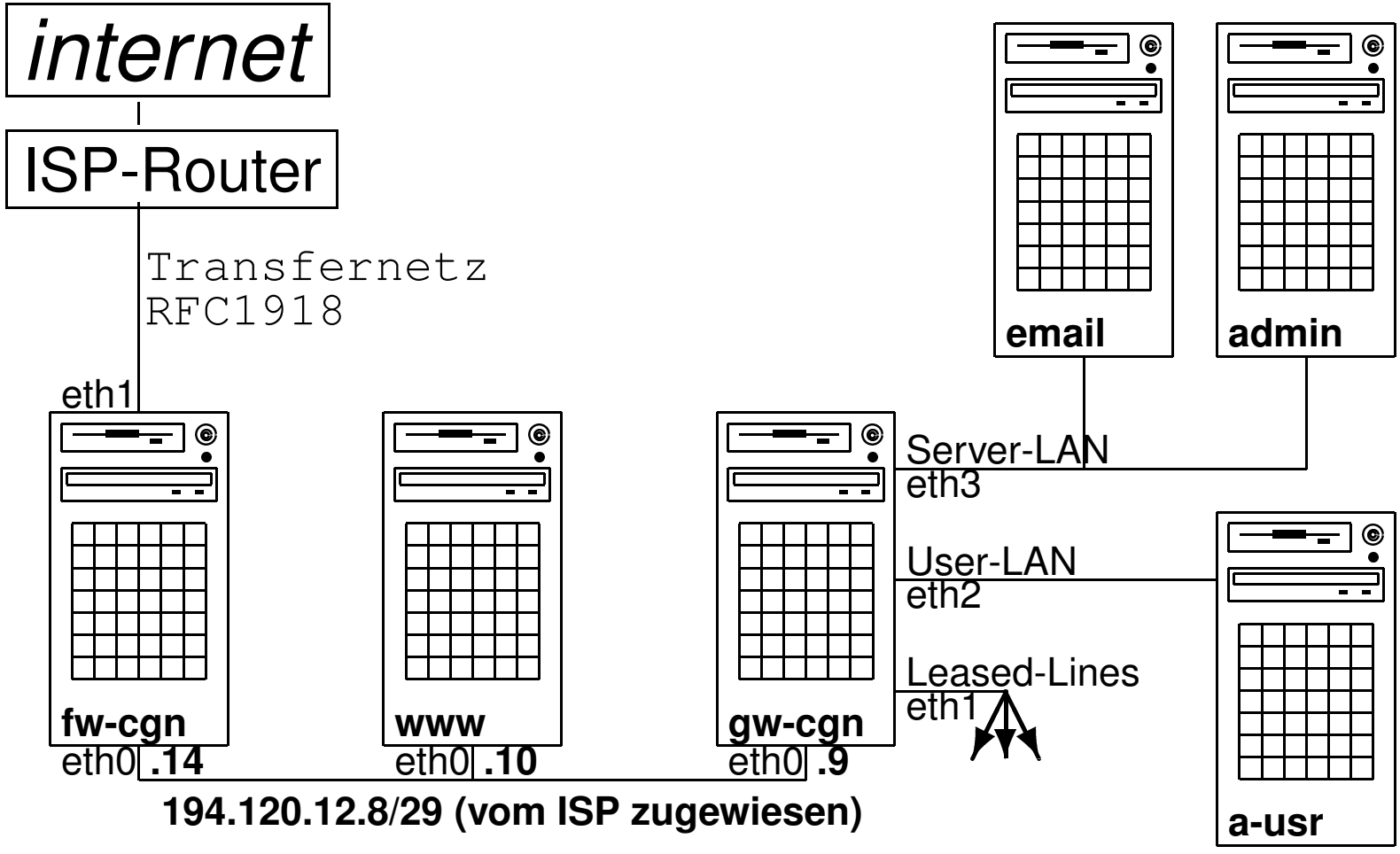
SSPE: Übersicht II

- 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

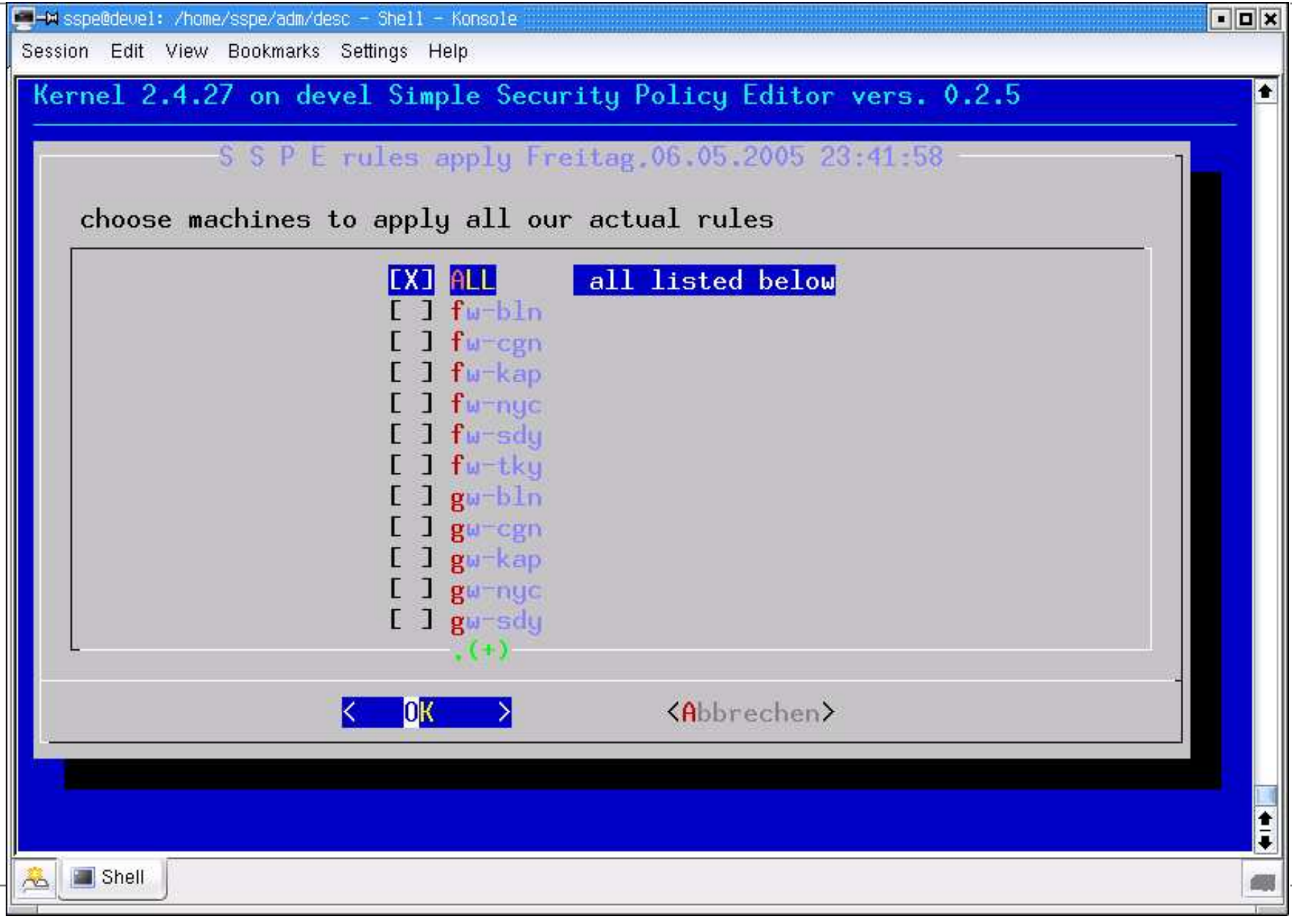
SSPE: Lizenz

SSPE unterliegt der
GNU General Public License

SSPE: Typischer Firmenstandort



SSPE: Anwendung



SSPE: Gliederung

1. Übersicht
2. Firewall
3. VPN
4. Betrieb
5. Ausblick

SSPE: Firewall Voraussetzungen

- Minimalsystem aus debian/stable und debfoster

SSPE: Firewall Voraussetzungen

- Minimalsystem aus debian/stable und debfoster
- monolithischer Kernel

SSPE: Firewall Voraussetzungen

- Minimalsystem aus debian/stable und debfoster
- monolithischer Kernel
- root, sonst keine Benutzer

SSPE: Firewall Voraussetzungen

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

SSPE: Firewall Voraussetzungen

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

SSPE: Firewall I

- eine Abstraktionsstufe oberhalb von iptables

SSPE: Firewall I

- eine Abstraktionsstufe oberhalb von iptables
- einfache Text-Dateien, symbolic-links

SSPE: Firewall I

- eine Abstraktionsstufe oberhalb von iptables
- einfache Text-Dateien, symbolic-links
- Host- und Netzwerk Definitionen in CIDR-Notation

SSPE: Firewall I

- eine Abstraktionsstufe oberhalb von iptables
- einfache Text-Dateien, symbolic-links
- Host- und Netzwerk Definitionen in CIDR-Notation
- Regeln mit Definitionen, Protokollen und Aktionen

SSPE: Firewall hostnet

- 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 durch Namensgleichheit

SSPE: 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]

SSPE: Firewall Abhängigkeiten

Der generierte Filter für jede Maschine hängt ab:

- inhaltlich: Host-, Netzdefinitionen

SSPE: Firewall Abhängigkeiten

Der generierte Filter für jede Maschine hängt ab:

- inhaltlich: Host-, Netzdefinitionen
- inhaltlich: Firewall Regelsatz

SSPE: Firewall Abhängigkeiten

Der generierte Filter für jede Maschine hängt ab:

- inhaltlich: Host-, Netzdefinitionen
- inhaltlich: Firewall Regelsatz
- inhaltlich: Interfaces, Routingtabelle

SSPE: Firewall Abhängigkeiten

Der generierte Filter für jede Maschine hängt ab:

- inhaltlich: Host-, Netzdefinitionen
- inhaltlich: Firewall Regelsatz
- inhaltlich: Interfaces, Routingtabelle
- inhaltlich: nathosts, privates

SSPE: Firewall Abhängigkeiten

Der generierte Filter für jede Maschine hängt ab:

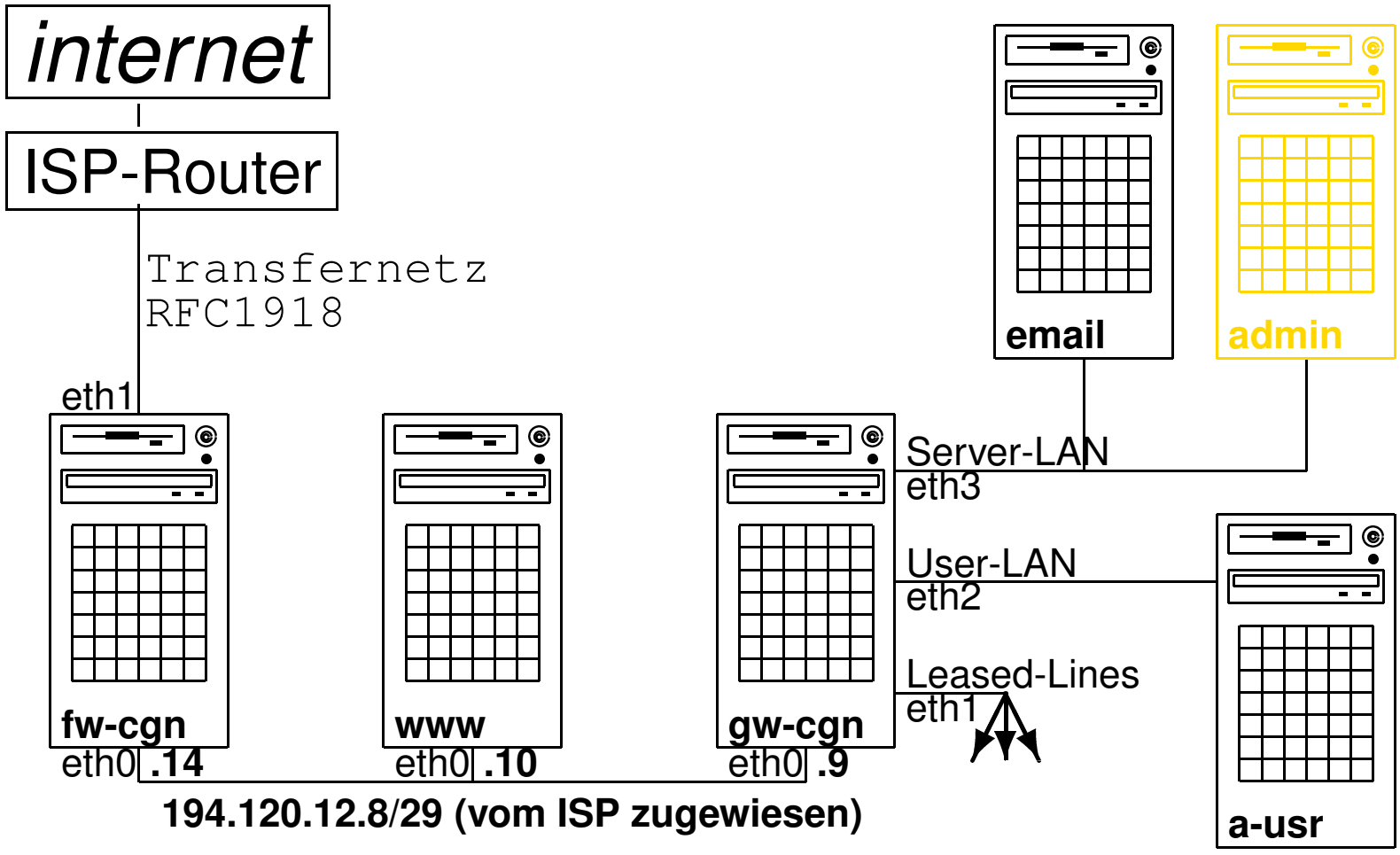
- inhaltlich: Host-, Netzdefinitionen
- inhaltlich: Firewall Regelsatz
- inhaltlich: Interfaces, Routingtabelle
- inhaltlich: nathosts, privates
- inhaltlich: Paketmangling-Dateien

SSPE: Firewall Abhängigkeiten

Der generierte Filter für jede Maschine hängt ab:

- inhaltlich: Host-, Netzdefinitionen
- inhaltlich: Firewall Regelsatz
- inhaltlich: Interfaces, Routingtabelle
- inhaltlich: nathosts, privates
- inhaltlich: Paketmangling-Dateien
- zeitlich: apply-options (sleep, wait)

SSPE: Regeln für admin



admin: INPUT / OUTPUT

a-usr admin 1 tcp ssh accept INSEC

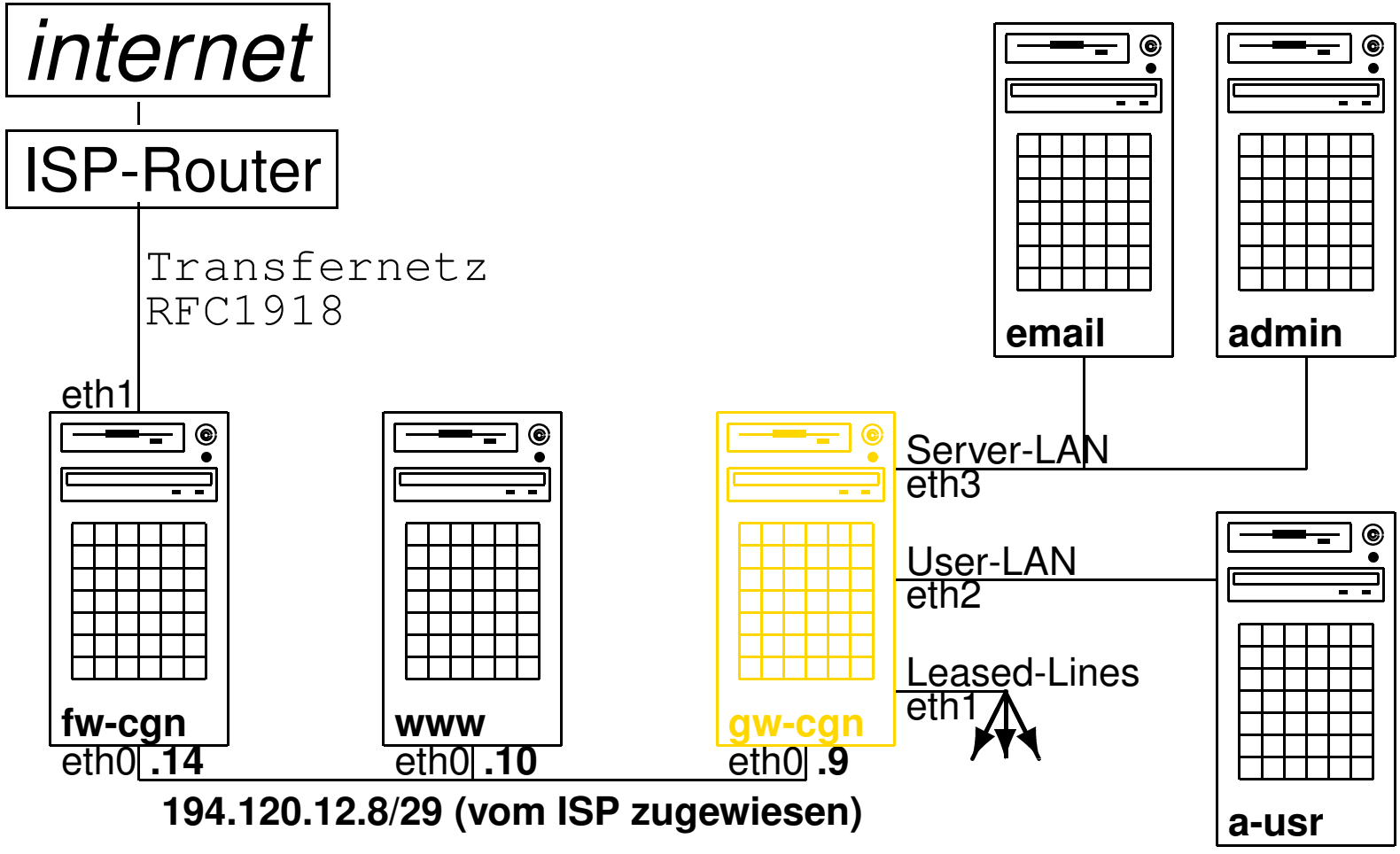
SSPE: admin's iptables

- 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
```

SSPE: Regeln für gw-cgn



gw-cgn: FORWARD

a-usr admin 1 tcp ssh accept INSEC

SSPE: gw-cgn's iptables

- 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
```

SSPE: Firewall privates

- 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
```

SSPE: Firewall nathosts

- 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
```

SSPE: 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
#
```

SSPE: Firewall II

- Zentrale Generierung der Kommandos

SSPE: Firewall II

- Zentrale Generierung der Kommandos
- Verteilung per scp, Anwendung der Regeln durch ssh

SSPE: Firewall II

- Zentrale Generierung der Kommandos
- Verteilung per scp, Anwendung der Regeln durch ssh
- Reihenfolge gezielt steuerbar (apply-options)

SSPE: Firewall II

- Zentrale Generierung der Kommandos
- Verteilung per scp, Anwendung der Regeln durch ssh
- Reihenfolge gezielt steuerbar (apply-options)
- Anstoss des Vorgangs aus einem Dialog-Menue

SSPE: Firewall II

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

SSPE: Firewall III

- Splittung des Regelsatzes: admin, head, ipsec, local, users, tail

SSPE: Firewall III

- Splittung des Regelsatzes: admin, head, ipsec, local, users, tail
- Regeln im Maschinenverzeichnis: admin, ipsec, users

SSPE: Firewall III

- Splittung des Regelsatzes: admin, head, ipsec, local, users, tail
- Regeln im Maschinenverzeichnis: admin, ipsec, users
- Regelsuche: erst im lokalen, dann im gemeinsamen Verzeichnis

SSPE: Firewall III

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

SSPE: Firewall IV

- NAT an Gateways auf eth0

SSPE: Firewall IV

- NAT an Gateways auf eth0
- IPSec ebenfalls auf eth0

SSPE: Firewall IV

- NAT an Gateways auf eth0
- IPSec ebenfalls auf eth0
- Packet-mangling per Dateien im Maschinenverzeichnis

SSPE: Firewall IV

- NAT an Gateways auf eth0
- IPSec ebenfalls auf eth0
- Packet-mangling per Dateien im Maschinenverzeichnis
- temp. Deaktivierung eines Zielgerätes möglich

SSPE: Firewall IV

- NAT an Gateways auf eth0
- IPSec ebenfalls auf eth0
- Packet-mangling per Dateien im Maschinenverzeichnis
- temp. Deaktivierung eines Zielgerätes möglich
- Ein Fehler führt nicht zu Abbruch!

SSPE: Firewall V

- Logging per syslog lokal oder remote

SSPE: Firewall V

- Logging per syslog lokal oder remote
- Logging einzuschränken per DROP-Regeln

SSPE: Firewall V

- Logging per syslog lokal oder remote
- Logging einzuschränken per DROP-Regeln
- Best-Practice: debfoster, rescueCD

SSPE: Firewall V

- Logging per syslog lokal oder remote
- Logging einzuschränken per DROP-Regeln
- Best-Practice: debfoster, rescueCD
- Best-Practice: manuelle Kontrolle der generierten Regeln

SSPE: Firewall VI

- ssh zur Verwaltung der Firewalls und Gateways

SSPE: Firewall VI

- ssh zur Verwaltung der Firewalls und Gateways
- Firewall und VPN dürfen nicht voneinander abhängen

SSPE: Firewall VI

- ssh zur Verwaltung der Firewalls und Gateways
- Firewall und VPN dürfen nicht voneinander abhängen
- ssh und IPSec sind wechselseitig unabhängig

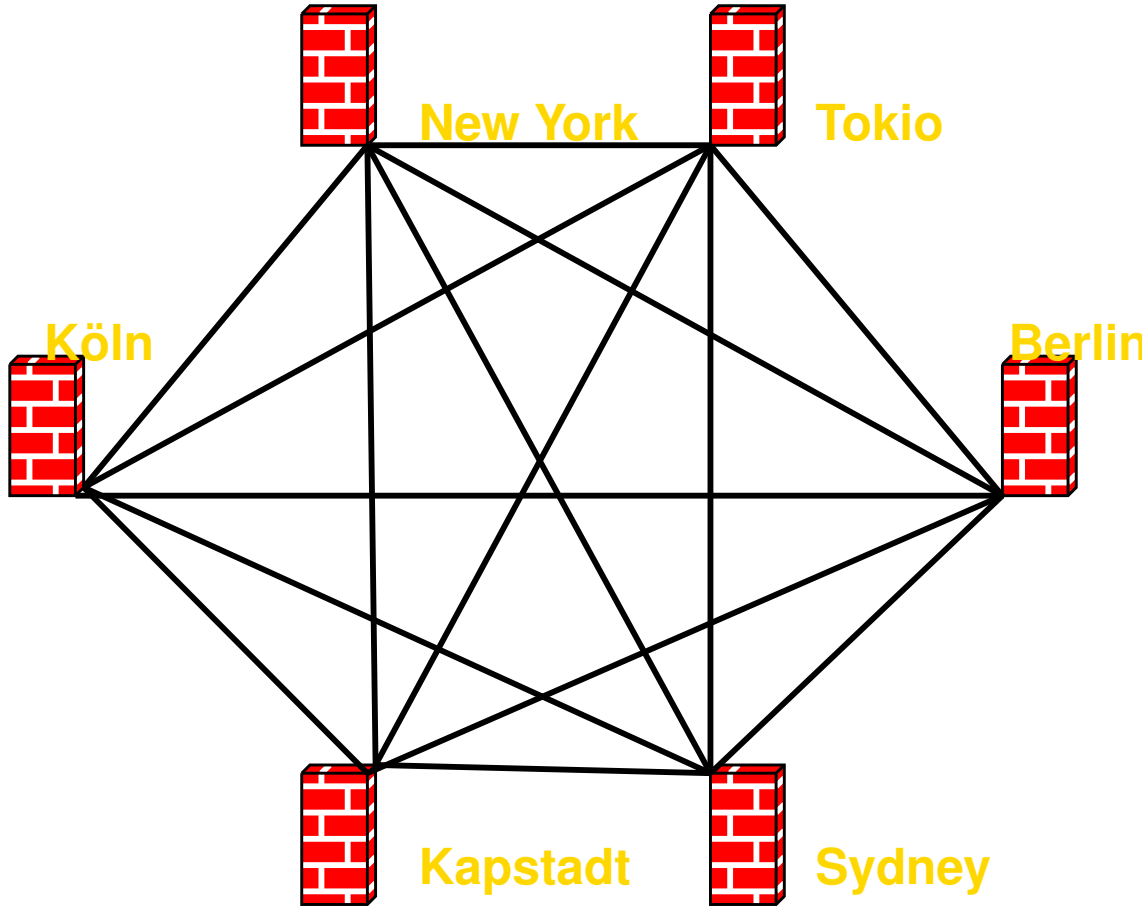
SSPE: Firewall VI

- ssh zur Verwaltung der Firewalls und Gateways
- Firewall und VPN dürfen nicht voneinander abhängen
- ssh und IPSec sind wechselseitig unabhängig
- Paranoid zu sein bedeutet nicht, daß keiner hinter einem her wäre

SSPE: Gliederung

1. Übersicht
2. Firewall
3. VPN
4. Betrieb
5. Ausblick

SSPE: VPN



Voll vermascht mit $N = S * (S - 1)$ Tunneln
bei 6 Standorten $\rightarrow$ 30 Tunnel

SSPE: VPN I

erster Ansatz

- Gleiche ipsec.conf an allen Standorten

SSPE: VPN I

erster Ansatz

- Gleiche ipsec.conf an allen Standorten
- Voraussetzung: alle sind erreichbar

SSPE: VPN I

erster Ansatz

- Gleiche ipsec.conf an allen Standorten
- Voraussetzung: alle sind erreichbar
- Zeitsteuerung per cron, ntp

SSPE: VPN I

erster Ansatz

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

SSPE: VPN II

- Konfiguration und PreSharedKeys aus sspe-konfig: ipsecs

SSPE: VPN II

- Konfiguration und PreSharedKeys aus sspe-konfig: ipsecs
- voll vermaschtes Netz, singuläre Standort-Anbindung zusätzlich möglich

SSPE: VPN II

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

SSPE: VPN II

- 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

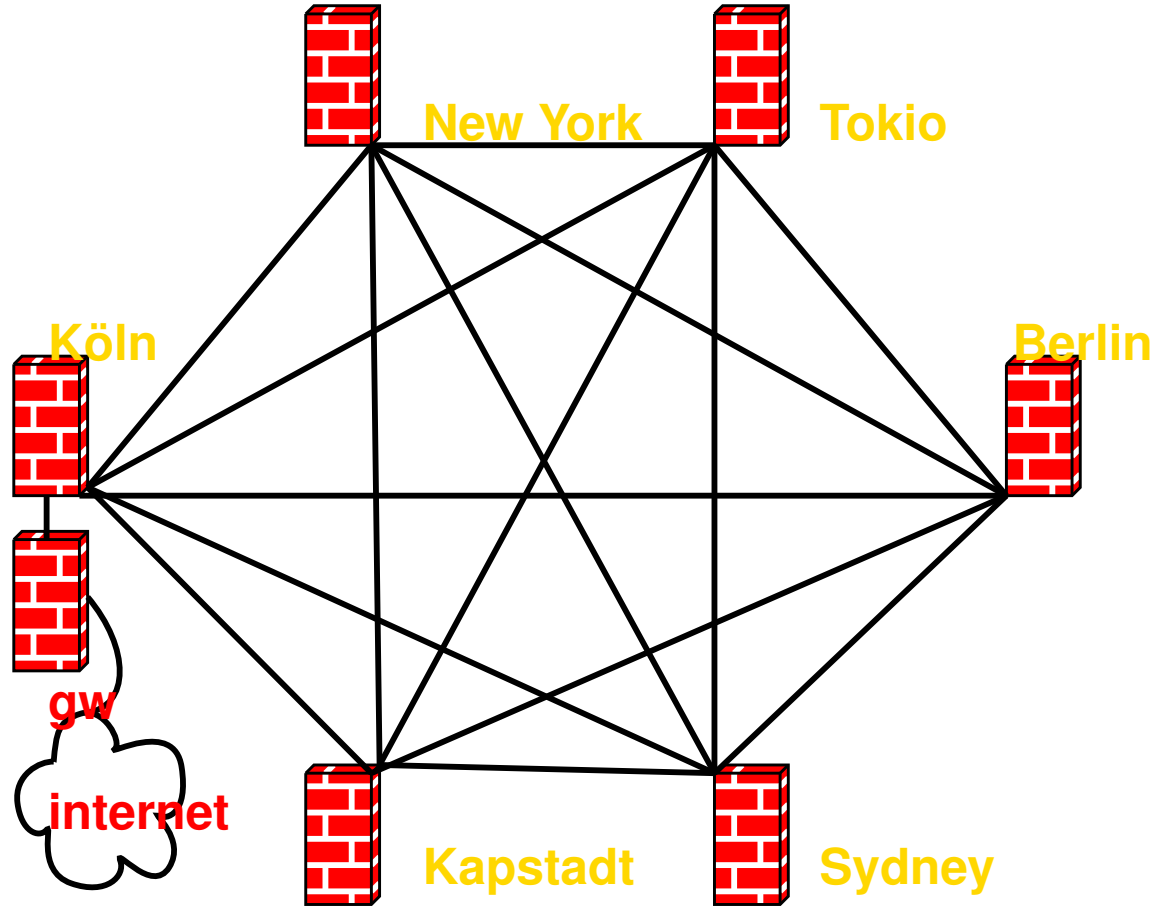
SSPE: VPN III

```
# 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
sdy      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
```

`ipsecs` Konfigurationstabelle

Hieraus werden alle ipsec.conf und ipsec.secrets generiert

SSPE: VPN IV



Viele Tunnel bei 6 internen Standorten und einem Internet-Anschluß

SSPE: VPN V

zweiter Ansatz

- jedes VPN-GW ist anders

SSPE: VPN V

zweiter Ansatz

- jedes VPN-GW ist anders
- exakte Konfiguration erzeugen

SSPE: VPN V

zweiter Ansatz

- jedes VPN-GW ist anders
- exakte Konfiguration erzeugen
- Routen des Internet per IPSec möglich

SSPE: VPN V

zweiter Ansatz

- jedes VPN-GW ist anders
- exakte Konfiguration erzeugen
- Routen des Internet per IPSec möglich
- Routinglücke für ssh zur Administration

SSPE: VPN VI

```
# 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
sdy     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
```

SSPE: VPN VII

- Handlungsreisende (roadwarrior) mit X.509-Authentisierung

SSPE: VPN VII

- Handlungsreisende (roadwarrior) mit X.509-Authentisierung
- Sperrliste für einzelne Clients: CRL der PKI

SSPE: VPN VII

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

SSPE: VPN VII

- 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

SSPE: Gliederung

1. Übersicht
2. Firewall
3. VPN
4. Betrieb
5. Ausblick

SSPE: Betrieb I

- ca. 3 Jahre Betrieb für mehrere Kunden

SSPE: Betrieb I

- ca. 3 Jahre Betrieb für mehrere Kunden
- ca. 50 Maschinen mit iptables gesichert

SSPE: Betrieb I

- ca. 3 Jahre Betrieb für mehrere Kunden
- ca. 50 Maschinen mit iptables gesichert
- einige hundert Anwender-PC geschützt

SSPE: Betrieb I

- ca. 3 Jahre Betrieb für mehrere Kunden
- ca. 50 Maschinen mit iptables gesichert
- einige hundert Anwender-PC geschützt
- Kosten drastisch minimiert

SSPE: Betrieb II

- Debian macht security-fixes einfach

SSPE: Betrieb II

- Debian macht security-fixes einfach
- Debian stable verdient seinen Namen

SSPE: Betrieb II

- Debian macht security-fixes einfach
- Debian stable verdient seinen Namen
- RedHat funktioniert auch, SuSe vermutlich ebenso

SSPE: Betrieb II

- Debian macht security-fixes einfach
- Debian stable verdient seinen Namen
- RedHat funktioniert auch, SuSe vermutlich ebenso
- Scriptänderungen einfach machbar

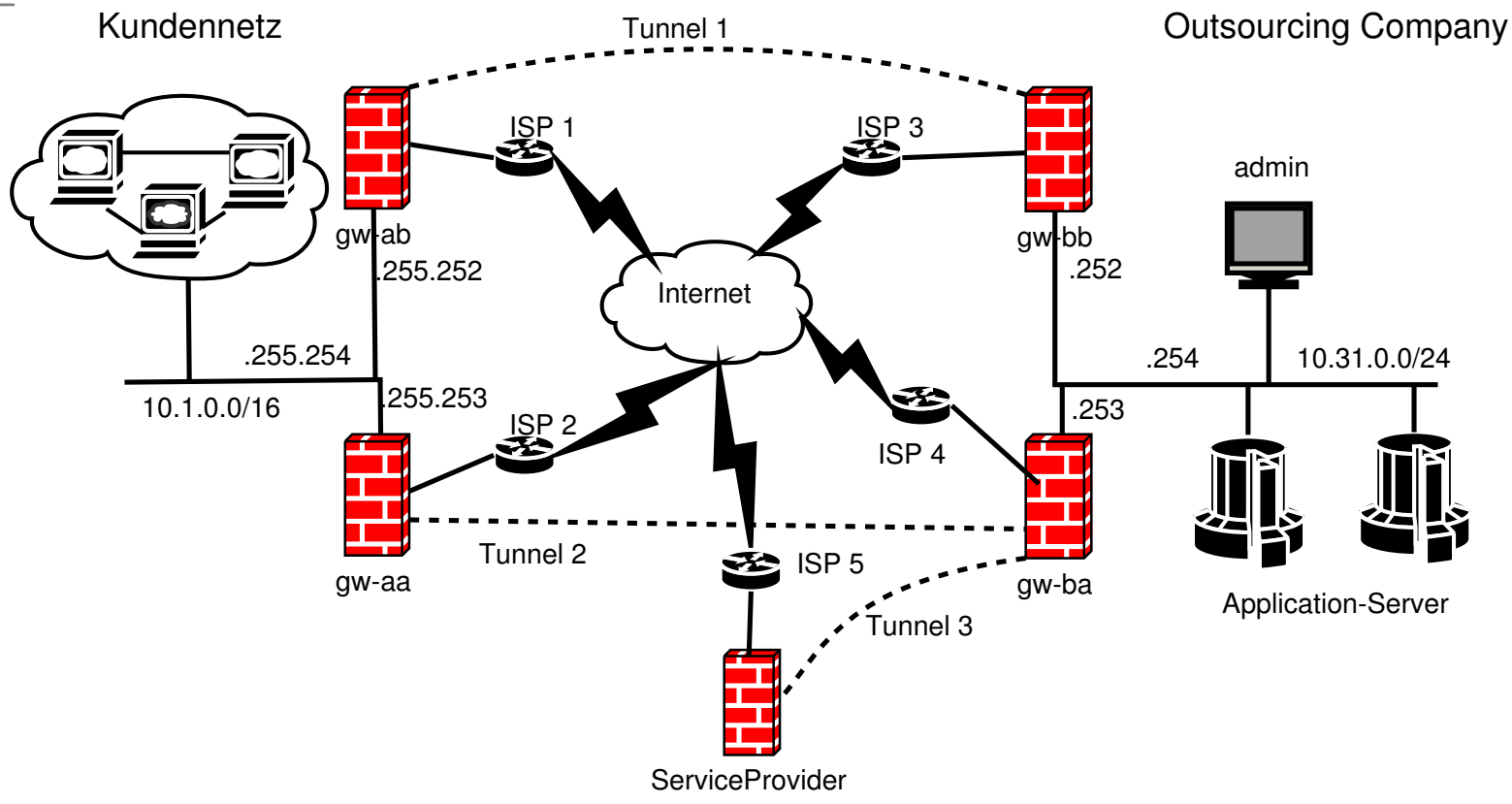
SSPE: Betrieb II

- Debian macht security-fixes einfach
- Debian stable verdient seinen Namen
- RedHat funktioniert auch, SuSe vermutlich ebenso
- Scriptänderungen einfach machbar
- Erweiterungen

SSPE: Betrieb II

- 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, . . .

SSPE: Betrieb III



Linux-Magazin 4/2005, Seite 70:
Wege sind das Ziel
Linux-Magazine MAY 2005, Page 68:
Alternate Path

SSPE: Gliederung

1. Übersicht
2. Firewall
3. VPN
4. Betrieb

5. Ausblick

SSPE: Ausblick

- MAC-Adressen im DHCP-Server und Firewalls fixieren

SSPE: Ausblick

- MAC-Adressen im DHCP-Server und Firewalls fixieren
- Cisco-Access-List Generierung fertigstellen

SSPE: Ausblick

- MAC-Adressen im DHCP-Server und Firewalls fixieren
- Cisco-Access-List Generierung fertigstellen
- Programmierung für BSD, Solaris, xyz

SSPE: Ausblick

- MAC-Adressen im DHCP-Server und Firewalls fixieren
- Cisco-Access-List Generierung fertigstellen
- Programmierung für BSD, Solaris, xyz
- Revisionsfähigkeit durch Datenbank ?

SSPE: 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

SSPE: 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!

SSPE: Öffentlichkeit

1. März 2003: http://sspe.sourceforge.net/
2. Georg Greve: Brave GNU World # 54 Oktober 2003
   http://www.linux-magazin.de/Artikel/ausgabe/2003/11/gnu/gnu.html
3. Linux-Magazin 04/2005, März 2005
   Wege sind das Ziel, HA-VPN
4. Computer Club Pascal, Köln, 24. Mai 2005
   Vortrag

Danke

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

Johannes Hubertz