

[IPsec|SSL]-VPNs |  $\{s \in \mathbb{N}, s \ll \infty\}$   
(SAGE-)/GUUG-Treffen in Hamburg

Johannes Hubertz

hubertz-it-consulting GmbH

Hamburg, 14. Feb. 2008

Vorstellung

VPN – kein reales öffentliches Vergnügen

OpenVPN – ssleay, libssl, X.509

Skalierung – Userland, RAM, MHz, Bits/sec

IPsec – RFC sei Dank

FreeSwan – Cisco, Nortel, et.al.

StrongSwan – Andreas Steffen, Hochschule für Technik, Rapperswil

Volle Vermaschung (mit Ausnahmen)

Kompatibilität zahlt sich aus, Checkpoint, Cisco, Nortel, ...

Konfigurationsschnipsel

Hochverfügbarkeit fürs VPN

Randnotizen – Zusammenfassung – Diskussion

# Vorstellung: Johannes Hubertz

- 1973 Studium der Elektrotechnik in Aachen
- 1980 Honeywell Bull, Ersatzteilreparatur
- 1984 Entwicklung Sonderprodukte, Assembler, PLM
- 1994 Erstkontakt mit IP
- 1996 Xlink, Erstkontakt mit dem Netz, root@www.bundestag.de, ...
- 1998 „Ins Allerheiligste“, iX 1/1998, Heise Verlag
- 1999 IT-Security Mgr. D-A-CH der Bull AG
- 2002 Entwicklung und Betrieb sspe für Steria GmbH und deren Kunden
- 2005 Gründung der hubertz-it-consulting GmbH
- ... Weiterentwicklung und Betrieb von sspe
- seit 1973 Bundesanstalt Technisches Hilfswerk in Köln-Porz
- seit 2001 Segeln, am liebsten auf Salzwasser



## Erkenntnisse aus dem Berufsleben

Bellovin and Cheswick: Firewalls and Internet Security, 1994

Fazit: Keep it simple!

Oder mit Einstein: So einfach wie möglich, aber nicht einfacher!

## Etwas Erfahrung war Voraussetzung

Gründung am 8. August 2005, Sitz in Köln

Geschäftsinhalt: Dienstleistungen im Umfeld der IT-Sicherheit

Schwerpunkte: Netzwerk, Switches, Router, VPNs, Firewalls, Hochverfügbarkeit, X.509, Betrieb, Schulung, freie Software . . .

Logo: Johannes Hubertz Certificate Authority als ASCII-7Bitmuster

Diese paar Bits befinden sich in einigen  $10^4$  X.509 Zertifikaten in der Seriennummer

Mitgliedschaft bei ECO e.V., GUUG e.V. und Kooperation mit der Bull GmbH

Wir sind käuflich ;-)

**Virtual** – virtuelles Netz, also kein reales, physikalisches Netzwerk

**Private** – privates, verschlüsseltes Netz, im Gegensatz zum Öffentlichen

**VPN** – zusätzliches Netzwerk auf bestehendem, zumeist dem Internet

**VPN** – Hersteller-Konsortium schuf FreeSwan – Referenz für IPsec auf IPv4

**FreeSwan** – kompatibel zu (fast) allen kommerziellen VPN-Lösungen

**FreeSwan** – beendet, StrongSwan ist Nachfolge-Projekt aus der Schweiz

**StrongSwan** – starke Authentisierung mit z.B. 2048 bit RSA-Keys (X.509)

**OpenVPN** – kleine Entwicklergruppe, basiert auf OpenSSL, X.509, tcp / udp

**VPN** – derzeit verfügbare Technologien: IPsec-, SSL-, Obscure-VPNs

# VPN – obscure version

## A.Kerkhoff (1883)

Die Geheimhaltung der Algorithmen soll **nichts**, die Geheimhaltung der Schlüssel jedoch alles zur Sicherheit beitragen

## Ingenieurskunst?

Kommerzielle Geräte (closed source) enthalten Hintertüren, (un)absichtlich eingebaut durch Hersteller und/oder andere –

## Dem Inschinör ist **nix** zu schwör

Reverse Engineering rekonstruiert einen Quelltext, um zielgerichtet Hintertüren zu finden. Werkzeuge (Softwaredebugger, Logikanalysatoren etc.) sind am Markt, die Nutzung manchmal nicht legal ... (Ortsabhängige Legalität beachten!)

Quelloffenheit  $\iff$  überprüfbare Sicherheit  $\iff$  Vertrauen

**Roadwarrior** – Homeoffice mit Internetanschluss zur Zentrale

**Standortvernetzung** – Netze mehrerer Firmenstandorte verbinden

**b2b** – Netze von Geschäftspartnern miteinander verbinden

**b2b oder privat** – vertraulich telefonieren?

normale IP-Pakete werden verschlüsselt

vorangestellter zusätzlicher Header erlaubt normalen Versand

Empfang: zusätzlicher Header nach Plausibilitätsprüfung entfernt

Das IP-Paket wird entschlüsselt und an seine Ziel-IP zugestellt

Transparent für den Endanwender, Sicherheit hängt an vielen Faktoren

RFC4303 IP Encapsulation Payload (ESP)

RFC4305 Cryptographic Algorithm Implementation Requirements for ESP



Schlüssel müssen auf vertraulichem Wege ausgetauscht werden

Schlüssel müssen regelmässig gewechselt werden

RFC4306 Internet Key Exchange (IKEv2) Protocol

RFC4307 Cryptographic Algorithms for Use in the IKEv2



<a href="http://www.strongswan.org">http://www.strongswan.org</a>	<a href="http://www.openvpn.net">http://www.openvpn.net</a>
	
IPsec	SSL-VPN
RFC-konform	RFC-konform
Hersteller-kompatibel	nicht Hersteller-kompatibel
UDP, ESP, NAT	UDP, TCP, NAT
Passworte, X.509	X.509
Verschlüsselung im Kernel	Verschlüsselung im Userland

# neue Produkte, neue Kryptographie, mehr Sicherheit?

Bruce Schneier, amerikanischer Krypto-Experte, in „Secrets and Lies“:

Jeder, der eine eigene kryptographische Grundfunktion erstellt, ist entweder ein Genie oder ein Narr. Angesichts des Genie/Narr-Verhältnisses stehen die Chancen nicht gut.

ISBN 3-89864-302-6, S.110

## Kein Bedarf für Neues

- OpenVPN und StrongSwan skalieren ausgezeichnet.
- Vertraulichkeit skaliert deutlich schlechter. ;-)
- Moderne Hardware, freie Software und KnowHow schaffen vertrauliche Umgebungen

## Mit ssleay zu OpenSSL und OpenVPN

1995: Netscape stellt ssl und https vor

Eric A. Young und Tim Hudson bauen ssleay, eine freie Implementierung  
Steven Henson, Ralf Engelschall et.al. machen daraus OpenSSL mit libssl  
freie Software zum Umgang mit X.509, TLS, und Verschlüsselung  
Beständige Pflege begründet einiges Vertrauen

OpenSSL mit Shellscripts kann X.509-Zertifikate erzeugen

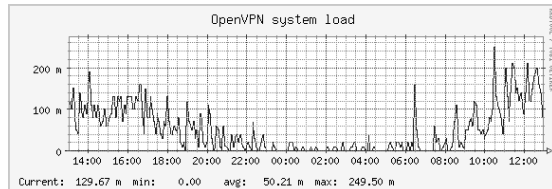
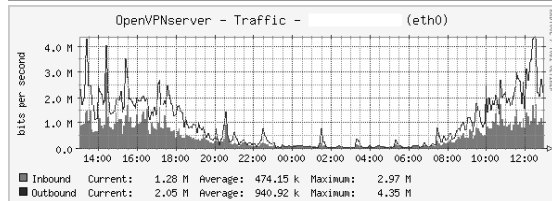
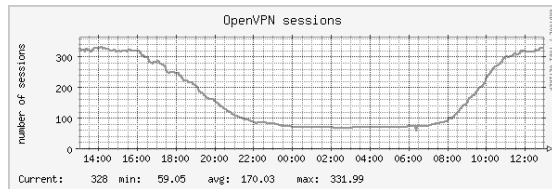
OpenVPN nutzt X.509 mit libssl, nutzt wahlweise TCP oder UDP

# OpenVPN – Server Konfiguration

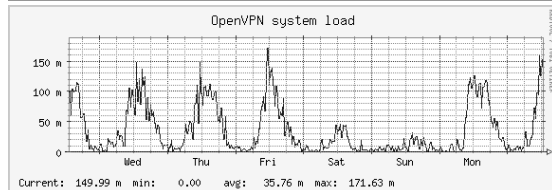
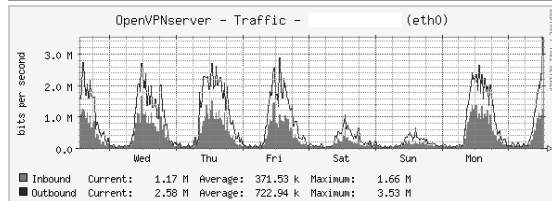
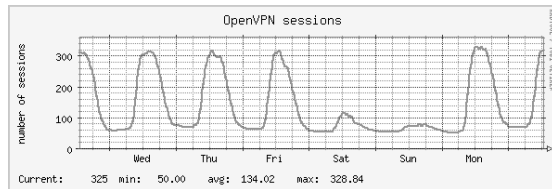
```
local u.x.y.z
proto tcp
ca /etc/openvpn/cacert.pem
key /etc/openvpn/vpn-srv-key.pem
dh /etc/openvpn/dh2048.pem
ifconfig-pool-persist /etc/openvpn/ipp1.txt
push "redirect-gateway"
push "dhcp-option DNS 172.16.0.132"
comp-lzo
user nobody
persist-key
status /var/log/ovpn1-status.log
verb 4
reneg-sec 18000

port 10000
dev tun
cert /etc/openvpn/vpn-srv.pem
crl-verify /etc/openvpn/crl.pem
server 172.24.0.0 255.255.0.0
push "route-delay 6 20"
push "dhcp-option DNS 172.16.0.131"
keepalive 10 120
max-clients 10000
group nogroup
persist-tun
log-append /var/log/openvpn1.log
mute 10
tls-exit
```

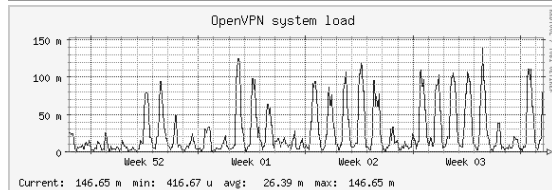
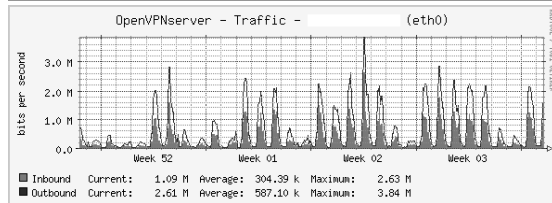
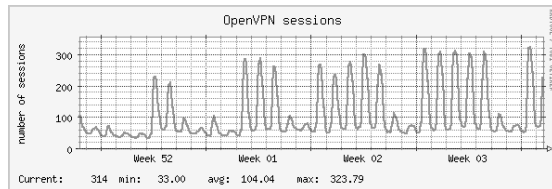
# OpenVPN Korrelation Traffic – Sessions



# OpenVPN Korrelation Traffic – Sessions



# OpenVPN Korrelation Traffic – Sessions



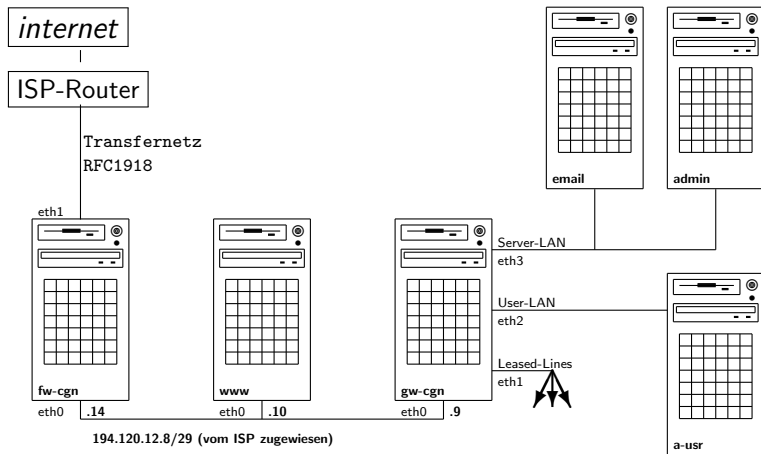
# IPsec – Herstellerunabhängig und kompatibel

- Ursprünge im IPv6-Umfeld für die Zukunft des Netzes
- konkreter Bedarf schon im Jetzt
- Herstellerkonsortium (Nortel, Cisco, u.v.a.m.)
- FreeSwan entsteht auf Linux und BSD, ab ca. 1996 nutzbar
- FreeSwan ist IPsec-Referenz auf IPv4
- FreeSwan ist 2003 fertiggestellt, Projekt eingestellt
- OpenSwan und StrongSwan treten das Erbe an



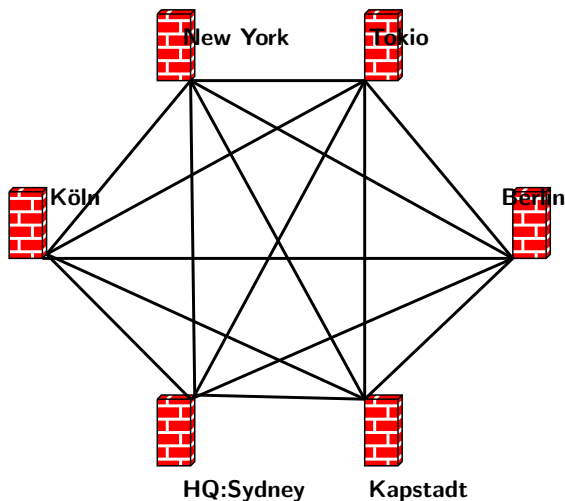
- Prof. Andreas Steffen und sein Team betreuen StrongSwan (Eidgenössische Technische Hochschule Rapperswil, Schweiz)
- X.509-Patch schon für FreeSwan hergestellt
- StrongSwan Version 2 beherrscht NAT-Traversal
- StrongSwan Version 4 benutzt IKEv2

# etwas Hardware am typischen Firmenstandort



Der Standort des Admin-PC spielt keine Rolle.

# das Firmennetzwerk



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

# 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
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

# erster Denkansatz: gleiche *ipsec.conf*

- 1. Voraussetzung für Verteilung: **alle** sind erreichbar per ssh/scp
- 2. Voraussetzung für Verteilung: Zeitsynchronisation per ntp funktioniert
- Script generiert Konfiguration und PreSharedKeys aus *ipsecs*
- per scp auf jedes Gateway: */etc/ipsec.conf* und */etc/ipsec.secrets.new* und
- cron: supervisor-script prüft und aktiviert Konfiguration jede Minute
- Resultat: voll vermaschtes Netz
- Overhead für Änderungen ist erträglich, **30** Sekunden downtime durch */etc/init.d/ipsec restart*
- singuläre Standort-Anbindung zusätzlich möglich

voll vermaschter Standort:

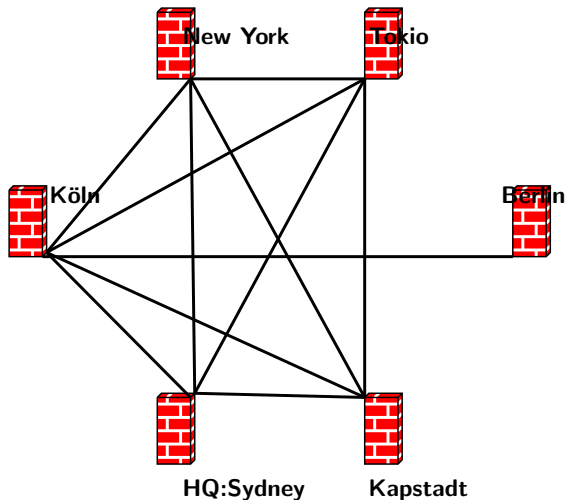
```
#!/bin/bash
if [ -f /etc/ipsec.secrets.new ] ; then
    if [ -f /etc/ipsec.conf.const ] ; then
        /bin/cat /etc/ipsec.conf.const >> /etc/ipsec.conf
    fi
    /bin/mv /etc/ipsec.secrets.new /etc/ipsec.secrets
    /etc/init.d/ipsec restart
fi
```

*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
```

# das Firmennetzwerk vor der Vollendung



5 Standorte an beliebigen Internet-Providern, einer an ISDN

```
01001010
01001000
0
01000011
0
01000001
01000001
```

**h**

hubertz-it-consulting GmbH

Standort mit singulärer Anbindung (z.B. ISDN):

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

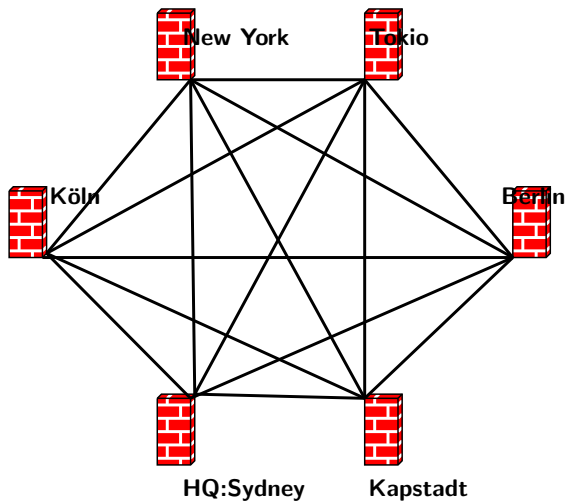


# ipsecs Konfigurationsdatei

# loc.	gateway	next-Hop	subnet
bln	172.22.0.25	172.22.0.30	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

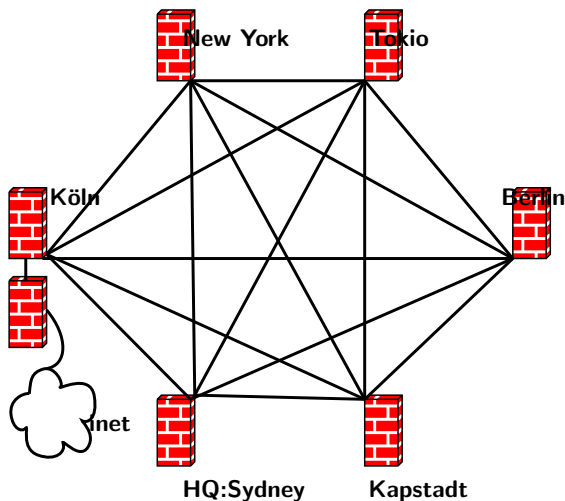
Hieraus werden alle *ipsec.conf* und *ipsec.secrets* generiert

# das Firmennetzwerk vor dem Umbau



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

# 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: 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
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
# inet4all via cgn
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, nun mit perl
- Overhead für Änderungen bleibt erträglich, 36 Sekunden downtime
- Erkenntnis 1: Routen des Internet ist auch per IPsec möglich
- Erkenntnis 2: Routinglücke für ssh zur Administration sinnvoll

- Handlungsreisende (roadwarrior) mit X.509-Authentisierung
- Sperrliste für einzelne Clients: CRL der PKI
- vpngdialer.sf.net für IPsec und L2TP vom beliebigen M\$-PC (freie Software von Thomas Kriener, läuft auf w2k und xp)
- L2TP durch IPsec zur Änderung des Routings im PC (durch vpngdialer initiiert) ermöglicht zentrale Adressvergabe

- Für drei Meinungen braucht man maximal zwei Rechtsanwälte
- Mehrere Geschäftspartner nutzen VPN-Geräte verschiedener Hersteller
- IPsec ist nicht gleich IPsec
- Wollen Sie für jeden Geschäftspartner ein anderes Device aufstellen?
- StrongSwan (FreeSwan) ist kompatibel zu (fast) allen IPsec-Devices

```
#  
config setup  
    interfaces="ipsec0=eth0 "  
    klipsdebug=none  
    plutodebug=none  
    plutoload=%search  
    plutostart=%search  
    uniqueids=yes  
#
```



```
#  
version 2  
config setup  
    interfaces="%defaultroute"  
    klipsdebug=none  
    plutodebug=none  
    uniqueids=yes  
    nat_traversal=yes  
#
```

# ipsec.conf Konfigurationsdateischnipsel I

```
.  
. #  
conn to-nortel  
    auto=start  
    type=tunnel  
    authby=secret  
    compress=no  
    esp=3des-md5  
    ike=3des-md5-modp1024  
    auth=esp  
    pfs=no  
    left=ip-of-his-gateway  
    leftsubnet=10.137.61.0/27  
    right=ip-of-my-gateway  
    rightnexthop=ip-of-my-router  
    rightsubnet=172.18.210.114/32  
#
```

# ipsec.conf Konfigurationsdateischnipsel II

```
.  
.  
#  
conn firewall-one-old-release  
    auto=start  
    type=tunnel  
    authby=secret  
    auth=esp  
    pfs=no  
    keyexchange=ike  
    keyingtries=0  
    keylife=45m  
    ikelifetime=1h  
    disablearrivalcheck=no  
    left=ip-of-his-gateway  
    leftsubnet=172.24.253.0/24  
    right=ip-of-my-gateway  
    rightnexthop=ip-of-my-router  
    rightsubnet=172.18.210.114/32  
  
#
```

# ipsec.conf Konfigurationsdateischnipsel III

```
.  
.  
#  
conn to-firewall-one  
    auto=start  
    type=tunnel  
    authby=secret  
    auth=esp  
    esp=aes256-sha1  
    pfs=yes  
    keyexchange=ike  
    keyingtries=0  
    keylife=60m  
    ikelifetime=24h  
    disablearrivalcheck=no  
    left=ip-of-his-gateway  
    leftsubnet=192.168.20.0/24  
    right=ip-of-my-gateway  
    rightnexthop=ip-of-my-router  
    rightsubnet=172.18.210.114/32  
#
```

# ipsec.conf Konfigurationsdateischnipsel IV

```
.  
.  
#  
conn to-checkpoint  
    auto=start  
    type=tunnel  
    authby=secret  
    auth=esp  
    pfs=no  
    keyexchange=ike  
    keyingtries=0  
    keylife=120m  
    ikelifetime=1h  
    disablearrivalcheck=no  
    left=ip-of-his-gateway  
    leftsubnet=10.100.111.205/32  
    right=ip-of-my-gateway  
    rightnexthop=ip-of-my-router  
    rightsubnet=172.18.210.114/32  
#
```

# ipsec.conf Konfigurationsdateischnipsel V

```
.  
.  
#  
conn to-sonicwall  
    auto=start  
    type=tunnel  
    authby=secret  
    auth=esp  
    pfs=yes  
    keyexchange=ike  
    keyingtries=0  
    keylife=8h  
    ikelifetime=1h  
    disablearrivalcheck=no  
    left=ip-of-his-gateway  
    leftsubnet=192.168.33.32/32  
    right=ip-of-my-gateway  
    rightnexthop=ip-of-my-router  
    rightsubnet=172.18.210.114/32  
#
```

# ipsec.conf Konfigurationsdateischnipsel VI

```
.  
.  
#  
conn to-cisco-vpn  
    auto=start  
    type=tunnel  
    authby=secret  
    auth=esp  
    pfs=yes  
    keyexchange=ike  
    keyingtries=0  
    keylife=8h  
    ikelifetime=8h  
    lifetime=8h  
    disablearrivalcheck=yes  
    left=ip-of-his-gateway  
    leftsubnet=192.168.30.32/32  
    right=ip-of-my-gateway  
    rightnexthop=ip-of-my-router  
    rightsubnet=172.18.210.114/32  
#
```

# ipsec.conf Konfigurationsdateischnipsel VII

```
.
#
conn to-x509-gw-xyz02
    auto=add
    authby=rsasig
    esp=3des-md5-96
    keyingtries=3
    disablearrivalcheck=no
    left=%any
    leftid="C=DE, ST=Germany, O=hubertz-it-consulting GmbH, \
        OU=IPSec-gateways, CN=gw-xyz02"
    lefttrsasigkey=%cert
    leftsubnet=172.25.2.0/24
    righttrsasigkey=%cert
    rightid="C=DE, ST=Germany, O=hubertz-it-consulting GmbH, \
        OU=IPSec-gateways, CN=gw-zentrale"
    rightcert=/etc/ipsec.d/gw-zentrale.der
    right=ip-of-my-gateway
    rightnexthop=ip-of-my-router
    rightsubnet=172.18.210.114/32
#
```



## MTBF, MTTR, 100%

MTBF – mean time between failure

MTTR – mean time to repair

100% – nie erreichbar

98,5% – üblich bei ISP-Verträgen

5,4785 Tage Downtime pro Jahr – Ist das genug, um Ihr Geschäft zu runinieren?

## Mehr Verfügbarkeit $\Leftrightarrow$ mehr Aufwand

Einfache Lösung: alle Geräte doppelt vorhalten  $\rightarrow$  halbe Ausfallzeit?

Nicht alle Geräte gehören Ihnen, Ihr ISP hat keine Lust, kein Personal ...

Sie nehmen einen zweiten, unabhängigen ISP hinzu

Alles wird gut ...

## noch mehr Aufwand nötig

noch ein paar neue Geräte – Router

noch etwas KnowHow (nicht aus dem Supermarkt!)

und etwas Zeit

# Strikte Trennung: Vertraulichkeit vs. Verfügbarkeit

## Schwerpunkt Verfügbarkeit

Wir verschlüsseln, damit beliebig oft gesniffert werden kann!

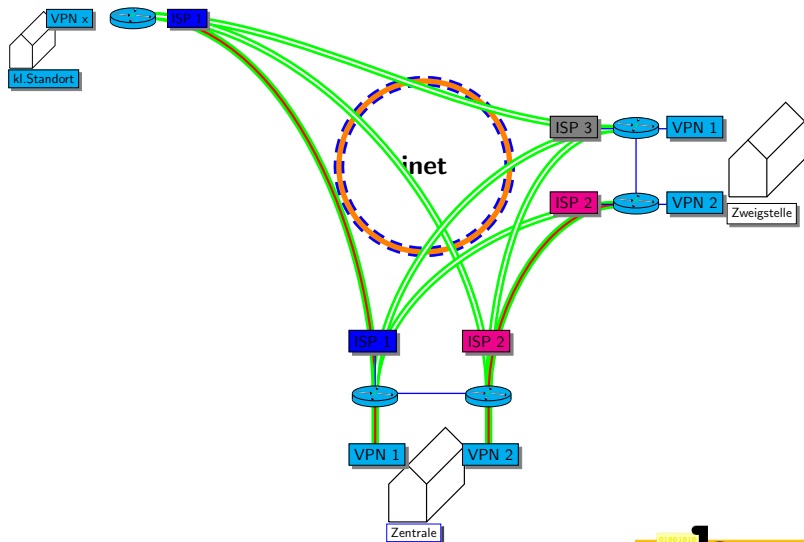
→ keine Anforderungen bzgl. Herkunft der Geräte

## Schwerpunkt Vertraulichkeit

Wir verschlüsseln, damit beliebig oft gesniffert werden kann!

Vertrauenswürdige Verschlüsselung funktioniert mit offenen Quelltexten und vielen Augen ...

# HA-VPN — mehrere Standorte



# Keine Bedenken?

- vor einigen Wochen: Zufallszahlen bei W2k im Heise-Newsticker
- B.Schneier: Auch in Linux gab es Probleme, 1996 in SSL
- CRYPTO 2007 conference, Dan Shumow, Niels Ferguson:  
Dual\_EC\_DRBG<sup>1</sup> contains a backdoor
- B.Schneier:  
We don't like to use algorithms that have even a whiff of a problem
- B.Schneier:  
Do not use Dual\_EC\_DRBG under any circumstances
- vorhersagbare Zufallszahlen machen Verschlüsselung reichlich überflüssig
- betroffen sind alle Methoden, die Zufall aus M\$-Systemen holen  
(NIST-Standard)

---

<sup>1</sup>Dual Elliptic Curve Deterministic Random Bit Generator

# Zusammenfassung

## Verfügbarkeit

Geräte mit definierten und bekannten Kommunikationen dürfen dazu beitragen  
Online-Registrierung ist kontraproduktiv, Linux-HA kostenlos

## Vertraulichkeit – ein hoher Preis

Geräte mit offenen Quelltexten können dazu beitragen  
Alle Komponenten tragen gleichermaßen dazu bei

## Sicherheiten

Mark Twain: man muß die Tatsachen kennen, bevor man sie verdrehen kann.  
Paranoid zu sein bedeutet nicht, daß keiner hinter einem her wäre ...

## Was tun?

Seid wachsam!  
Vertrauen ist gut, Kontrolle ist besser!

Have a **close** look at **your bits**.

Andreas Steffen, Sichere Internet-Telefonie? Interview in: digma, Zeitschrift für Datenrecht und Informationssicherheit, 6. Jahrgang, Heft 3, September 2006, Seite 138ff.

Bruce Schneier, Secrets and Lies, Heidelberg: dPunkt.verlag GmbH, ©2004

<http://ietf.org/rfc/rfc.4303.txt> – IP Encapsulation Payload (ESP)

<http://ietf.org/rfc/rfc.4305.txt> –

Cryptographic Algorithm Implementation Requirements for ESP

<http://ietf.org/rfc/rfc.4306.txt> – Internet Key Exchange (IKEv2) Protocol

<http://ietf.org/rfc/rfc.4307.txt> – Cryptographic algorithms for Use in the IKEv2

<http://www.openssl.org/> – OpenSSL Homepage

<http://www.openvpn.net/> – OpenVPN, James Yonan

<http://strongswan.org/> – StrongSwan, Andreas Steffen

<http://vpndialer.sourceforge.net/> – Thomas Kriener

<http://sspe.sourceforge.net/> – Johannes Hubertz

Ich bedanke mich für Ihre Aufmerksamkeit  
**Ihre Sicherheit ist uns wichtig!**

**Frohes Schaffen**

Johannes Hubertz

it-consulting\_at\_hubertz\_dot\_de



powered by **AT<sub>E</sub>X 2<sub>ε</sub>**  
and PSTricks

