

Zusammenfassung

Simple **S**ecurity **P**olicy **E**ditor heißt im folgenden SSPE und ist eine zentrale, einfach zu administrierende Firewall- und Sicherheitslösung für beliebig viele Maschinen an mehreren Standorten mit jeweils eigenem Internet-Zugang. Die privaten Netze werden mit IPSec verbunden. Außendienstmitarbeiter sind ans VPN angeschlossen. Unter anderem stellt die Authentifizierung durch X.509 Zertifikate sicher, dass ungebetener Besuch im LAN ausbleibt ...

Für Anregungen, Kommentare und Hinweise auf Fehler bedanke ich mich vorab.

Gewidmet ist dieses Werk meinen beiden Kindern Clara und Niklas.

Inhaltsverzeichnis

Im Text verwendete Schriftarten

Schriftart	Beispiel	Kommentar
Normal	Normal	Normaler Text
italic	*Wichtig*	Hervorgehobener Text
`Typewriter`	`read_routes`	Funktionen aus Shell- oder Perlprogramm
Small Caps	Drop	Feststehender Ausdruck
Fett	**/etc/hosts**	Dateinamen, evtl. mit Pfadangabe

Diese Dokumentation wurde von Mai bis Oktober 2003 in LaTeX geschrieben, einzelne Teile erst 2004 fertig, manche sind immer noch Entwurf.

Kapitel 1

Keep it simple!

Wer seinen Bellovin [BC94] gelesen hat, wird um den Grund der wiederholten Forderung nach Einfachheit in der IT-Sicherheit wissen. Komplexe Zusammenhänge sind weit verbreitet, damit verbundene Ungewissheit leider ebenso. Diese wird mancherorts auf fatale Weise durch Unwissenheit begleitet ...
IT-Sicherheit wird allgemein verstanden als Kombination dreier Dinge:

1. Confidentiality – Vertraulichkeit
2. Integrity – Datenintegrität
3. Availability – Datenverfügbarkeit

Die gesunde, auf ein Unternehmen zugeschnittene Mixtur der drei sich gegenseitig ausschließenden Forderungen sollte in einer Sicherheitsvorschrift niedergelegt sein. Es ist Aufgabe des Administrators, die Forderungen einer solchen Security-Policy umzusetzen. Nicht nur im Netzwerkbereich bietet sich Open-Source an, die IT-Landschaft kostengünstig zu bereichern. Die allgemeine Verfügbarkeit der Quelltexte macht ein Nachvollziehen der benötigten Funktionen nicht nur dem Programmierer möglich; jeder kann sich jederzeit davon überzeugen, wie etwas funktioniert. Fehler werden so aufgrund der Vielzahl interessierter Zeitgenossen effektiv und zeitnah bereinigt. Im Bereich kommerzieller Produkte ist dies nur wenigen Insidern vorbehalten, möglicherweise sehen weniger Augen weniger Fehler? Zumindest scheint der Glaube unbegründet, kommerzielle Produkte seien weniger fehlerbehaftet als solche mit offenen Quelltexten.
Inwieweit das Gegenteil der Fall ist, bleibt Spekulation.

Linux ™ [1] verfügt seit ca. 1996 über Mechanismen, Netzwerk-Verkehr zu regulieren. [Tho96] Dies meint sowohl gezieltes Zulassen oder Verhindern als auch Logging von Verkehr auf der Grundlage von Netzwerkadressen und Ports. Linux-Kernel ab Version 2.0 sind in der Lage, mittels geeigneter Konfigurationsdirektiven IP-Pakete abhängig von Quelladresse, Zieladresse und Protokoll passieren zu lassen, zu verwerfen oder auch zusätzlich zu loggen. [KD95] Der Kommandozeilen-Befehl `ipfwadm` setzt die gewünschten Parameter anhand der übergebenen Parameter. Der erste Wurf eines Lösungsansatzes in Form eines Shell-Scripts sieht vielleicht so aus:

```
ipfwadm -I bla bla ACCEPT
ipfwadm -I blw bla DENY
ipfwadm -O bla bla
```

Diese Kommandos dienen nur dem Schutz des Gerätes, auf dem sie ablaufen. Wenn es mehrere Schnittstellen hat und als IP-Router fungiert, kommt noch hinzu:

```
ipfwadm -F bla bla
```

Damit wird der durchgehende Verkehr reguliert. Natürlich muß es dafür auch je eine Input- und Outputregel geben. Das sieht das Filterkonzept in Kernel 2.0.x und auch in der Nachfolgeversion 2.2.x vor. Vorlagen für solche Scripts existieren viele, manche sind brauchbar. Mit solchen kann eine Maschine als Firewall bezeichnet werden, wenngleich dazu noch einiges mehr gehören sollte als nur IP-Pakete zu filtern. Als Prototyp eines Application-Level-Gateway, also eines Filters auf Anwendungsebene, kann das TIS-Firewall-Toolkit[2] bezeichnet werden. Aber auch so einfache und nützliche Programme wie `rinetd` von Boutell [Bou02] sind in ihrer kombinierten Wirkung mit IP-Filtern, Proxies und Authentifizierungsmechanismen als Tor zur Welt wirkungsvoll zu gebrauchen. Ein durch solche Maßnahmen geschütztes internes Netz verspricht dem Verantwortlichen eher einen ruhigen Schlaf als auf jeden Schutz aus Kostengründen zu verzichten.
Darüber hinausgehende Hilfsmittel existieren auch, z.B. fwbuilder. Leider wird oft auf die zum Prinzip erhobene Einfachheit wenig Rücksicht genommen, sogar graphische Oberflächen kommen zum Einsatz. Nicht, daß ich solche verteufeln will, aber braucht man X11, welches Schwachstellen haben könnte? Müssen Megabytes Software, die kaum zu durchschauen sind, auf einer Maschine laufen, die Sicherheit produzieren soll? Aus dem Prinzip Einfachheit lässt sich ein vernünftiger Minimalismus ableiten, der sehr zur Übersichtlichkeit beiträgt. Natürlich hat eine solche puritanische Arbeitsweise an

[1] Linux wurde ab 1992 von Linus Thorvalds und anderen aus Spass an der Sache entwickelt und ist zum möglicherweise grössten Softwareprojekt der Geschichte gewachsen

[2] Trusted Information Systems, USA, leider keine wirklich freie Software

gewissen Stellen gegenüber der "Point-and-Click"-Version Mängel. Aber sie bietet den unschätzbaren Vorteil, ohne jahrelanges intensives Studium von Megabytes Quellcode recht schnell verstehen und nachvollziehen zu können, was im Gerät vorgeht. Der Kernel ist schon kompliziert genug; wer sich dessen Quelltexte ansieht, wird trotz der hervorragenden Qualität nicht unmittelbar sehen können, was mit seinen IP-Paketen geschieht. Und daher auch nicht, was mit unerwünschtem Traffic geschieht.
Je umfangreicher das interne Netz ist, je vielfältiger die Anforderungen der Nutzer sind, umso komplexer wird die Administration einer solchen Firewall. Schnell wird durch Unachtsamkeit oder Überforderung ein 'Loch' in die Firewall gebohrt, welches ein inakzeptables Sicherheitsrisiko darstellt. Die Überwachung auf Konformität zu einer vorgegebnen 'Sicherheits-Richtline' übersteigt schon bald die Leistungsfähigkeit auch des besten Administrators. Daraus ist unmittelbar abzuleiten, daß Vereinfachungen nötig sind, die der Nachvollziehbarkeit der Regulierung dienen. Selbstverständlich darf die Sicherheit nicht davon in Mitleidenschaft gezogen werden. Der vielleicht wichtigste Grundsatz "Keep it simple" dient als Maßstab.
Eine erste Idee ist die Ersetzung der IP-Adressen durch symbolische Namen. Die Datei `/etc/hosts` liefert ein Muster, allerdings ist die Syntax nicht zur Kennzeichnung von Netzwerken gedacht. Ergo muß jeder Eintrag noch um die Netzmaske erweitert werden.
Hostdefinitionen: Symbolischer Name IP-Adresse Netzmaske

```
def-gw 10.0.0.1 255.255.255.255
adm-pc 10.0.0.2 255.255.255.255
```

Die Namen finden sich in Regeln wieder:

```
Quelle   Ziel     Richtung Protokoll Portnummer Aktion Optionen
adm-pc   def-gw   Oneway   TCP       ssh        accept LOG
def-gw   adm-pc   Twoway   TCP       all        deny
```

Ein von 'dialog' (Abbildung 1.1) gesteuertes Shellscript erstellt aus den Regeln passende ipfwadm-Kommandos in einer ausführbaren Datei, welche anschließend und beim System-Boot ausgeführt wird. Da die erzeugte Datei jederzeit durch den Systemadminstrator einsehbar ist, kann eine Entstörung relativ einfach durch Ansehen erfolgen. Der Mechanismus ist komplett als Shell-Script implementiert und funktioniert seit Jahren an einigen Stellen problemlos.
Aufgrund der eingebauten Rückwärtskompatibilität ist die Konstruktion auch noch mit dem zur Zeit neuesten stabilen Linux-Kernel 2.4.20 problemlos anwendbar. Allerdings bieten neuere Kernel weit bessere Möglichkeiten. Zusätzlich stellt sich bald ein weiterer Wunsch ein: Mehr und mehr Maschinen an

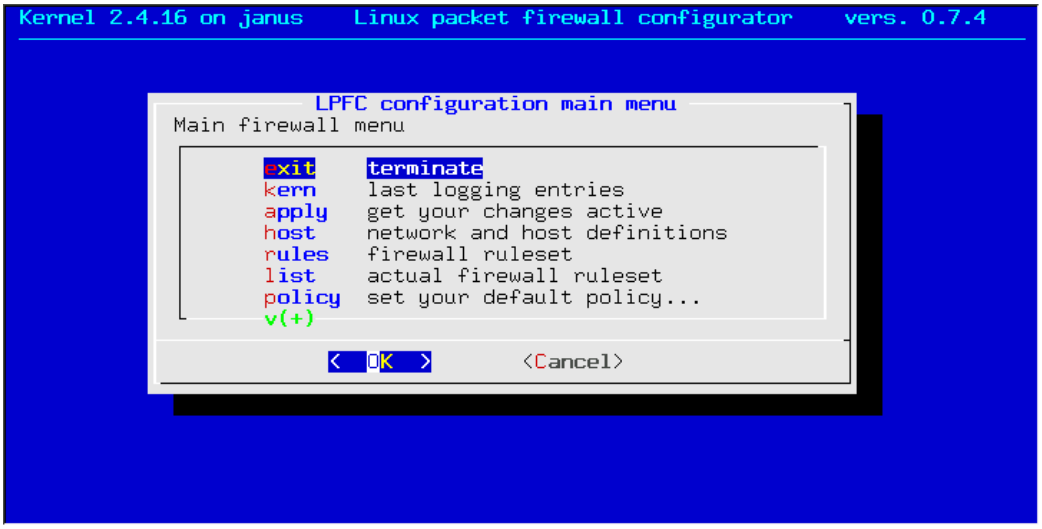

Abbildung 1.1: Linux Packet Firewall Configurator

verschiedenen Standorten wollen per Firewall-Regeln geschützt werden, der administrative Aufwand nimmt mit steigender Anzahl Geräte überproportional zu, da die Verkehrs-Beziehungen nicht im linearen Verhältnis zur Anzahl stehen. Zusätzliche Komplexität ergibt sich durch die Verwendung von NAT an den Grenzen zum Internet und aus der Notwendigkeit, die privaten Netze der Standorte mittels IPSec zu verbinden.

Kapitel 2

Viele Puzzleteile ergeben ein Bild

Eine einfache Lösung zur Administration mehrerer Linux-Firewalls soll entstehen. Aus Kosten- und Sicherheitsgründen kommt *nur* OpenSource in Frage. Für eine zentrale Administration spricht, die Definitionen einheitlich und damit konsistent zu pflegen, was die Fehlermöglichkeiten des Administrators stark einschränkt. Geringfügige Modifikationen an der Definition der Hosts in der **hostnet**-Datei führt zu einer wesentlichen Vereinfachung: Gruppierung von Hosts durch Eintragung unter gleichem Namen reduziert die Anzahl Regeln. Ein Zugang zu allen Firewall-Systemen vom Adminstratorplatz sollte per SSH ohne Passwort [JS01] möglich sein, RSA oder DSA-Authentifizierung erscheint auch sicherer als 8-buchstabige Passwörter. Der hierdurch gewonnene Automatismus lässt sich sowohl zur Verteilung als auch zur Ausführung der Scripts benutzen.

2.1 Architektur

Eine zentrale Maschine soll dazu dienen, alle Firewalls und sonstigen sicherheitsrelevanten Systeme mit Iptables-Kommandos zu beschicken. Hier bietet sich Debian GNU/Linux an, erstens weil es als OpenSource kostengünstig ist, zweitens weil es bis ins letzte Bit nachvollziehbar funktioniert und damit möglicherweise sicher ist, und drittens, weil es sich mit den besten, einfachsten, und zuverlässigsten Mitteln administrieren und insbesondere jederzeit auf neuestem Stand halten lässt, *ohne* daß in irgend einer ausländischen Firma Buch darüber geführt würde, welche Pakete wann eingespielt oder auf den neuesten Stand gebracht werden. Im kommerziellen Umfeld ist dies nicht immer gegeben, die Sicherheit eines jeden Unternehmens verbietet aber

dem gesunden Menschenverstand nach, anderen Institutionen, insbesondere Software-Lieferanten, exakte Abbilder der eigenen Maschinen anzuvertrauen. Mit DEBIAN GNU/LINUX und seinen vielfach gespiegelten Archiven im Internet ist a priori ausgeschlossen, daß ein solches Abbild herstellbar ist. Der Paketverwaltung wird nachgesagt, allen anderen mindestens in Puncto Paket-Abhängigkeiten weit überlegen zu sein. Eine DEBIAN-Minimal-Installation, per Internet auf einen aktuellen Stand gebracht, anschließend gehärtet, und danach sauber mit sicherheitsrelevanten Updates gepflegt, sollte auch gehobenen Ansprüchen genügen. Selbstverständlich kann die Funktionalität auch durch andere, möglicherweise sichere Systeme erbracht werden, wie z.B. BSD-Varianten.

2.2 Linux wird gehärtet

Bei der Installation der Linuxsysteme ist darauf zu achten, daß die Dateisysteme ausreichend dimensioniert und entsprechend zugeordnet werden. Log-Dateien sollten in einem eigenen Dateisystem untergebracht werden, so daß auch im Falle von `DoS-` oder `DDoS-`Angriffen nicht mit Systemstillstand zu rechnen ist. DEBIAN bietet mit dem Paket 'aide' ein leicht zu handhabendes, flexibles und automatisierbares hostbasiertes Intrusion-Detection-System an, welches mit wenig Aufwand sicherstellt, dass Änderungen an beliebigen Dateien nicht unbemerkt bleiben. Tripwire® funktioniert natürlich ebenso.

Nach der Installation sollten zunächst alle Netzwerk-Einstellungen überprüft werden, am einfachsten per `netstat -an` oder per `lsof` solange nach lauschenden Prozessen forschen, bis diese alle deaktiviert sind. Üblicherweise lauscht z.B. `portmap` auf UDP/111 und TCP/111. Einmal beendet und in `/etc/init.d/portmap` als zweite Zeile "exit 0" eingefügt, wird er nie wieder gestartet. Evtl. sind auch andere Prozesse bei `portmap` registriert, wer es wissen will schaut (solange portmap noch läuft) mit `rpcinfo -p localhost` einfach nach. Die dabei gezeigten Prozesse sollten analog behandelt werden. Hardliner mögen auch gerne die entsprechenden Dateien löschen. Dann sind sie auch nicht mehr zu reaktivieren.

Nicht benötigte User-Namen sollten aus der `/etc/passwd` gelöscht werden. Falls gewünscht, ebenfalls aus der `/etc/group` und `/etc/shadow`. Die Administration der Systeme erfolgt selbstverständlich als ROOT, die Administration der anderen Systeme erfordert keine root-Rechte auf dem Adminsitrations-PC. Daher ist das Anlegen eines Benutzers zu diesem Zweck, z.B. "adm", auf diesem sinnvoll.

Der einzige Zugang ins System sollte die Konsole sein, als Alternative kann ssh mit deaktivierter Passwort-Authentifizierung dienen. Daß jemand RSA-

```
# Name   CIDR-Adresse
any      0.0.0.0/0        # the whole internet
internal 10.0.0.0/8       # company-wide
adm-pc   10.0.0.2/32      # administrators workbench
def-gw   10.0.0.1/32      # internet-firewall internal Interface
gw-all   10.0.0.1/32      # internet-firewall internal Cologne
gw-all   194.45.252.1/32  # internet-firewall external Frankfurt
gw-all   194.45.253.1/32  # internet-firewall external Berlin
```

Abbildung 2.1: `hostnet` Beispiel mit Gruppierung

oder DSA-Schlüssel errät oder nachmacht, sei hier nicht bestritten, wird aber in der Praxis bisher nur sehr selten beobachtet. Selbstverständlich müssen per Filter die IP-Adressen eingeschränkt werden, von denen aus eine ssh-Session erlaubt wird.

Für die Firewalls und sonstigen zu administrierenden Systeme gilt entsprechendes, die Sicherheit einer Firma ist sicher nicht besser als die des schwächsten Systems. Weitergehende Hinweise zur Härtung von Linux finden sich im Kapitel 3 von [Bau02].

2.3 Host- und Netzdefinitionen – hostnet

Die Datei **hostnet** hat eine einfache Syntax mit Folgen. Als Beispiel zeigt Abbildung 2.1 u.a. eine Gruppe: `gw-all`. Diese Gruppe kann im Regelwerk ebenso benutzt werden wie jeder Einzel-Eintrag. Allerdings sind gewisse Voraussetzungen nötig, um Ipfwadm-, Ipchains- oder Iptables-Kommandos sinnvoll zu generieren: Sowohl die Interface-Adressen als auch die Routingtabellen der betroffenen Systeme sollten dem generierenden Prozess zugrunde liegen. Die Netfilterarchitektur sieht unterschiedliche Kommandos für ein-, aus- und durchgehende IP-Pakete vor. Daher muss im Script entschieden werden, ob eine Regel für die Maschine selbst (Input/Output) oder für durchgehenden Traffic (Forward) das Iptables-Kommando erzeugt. Dabei liegt die Überlegung zugrunde, dass nur dann Traffic durch die Maschine fließt, wenn unterschiedliche Zeilen der Routingtabelle auf Quelle und Ziel verweisen. Insbesondere liegt der Schluss nahe, daß die Maschine genau dann von der Regel nicht betroffen ist, wenn Quelle und Ziel nur durch die default-route erreicht werden. Im Ergebnis mindert dies die Anzahl Iptables-Kommandos erheblich.

```
#CIDR           # Comment
10.0.0.0/8      # company-wide
192.168.1.0/24  # Cologne DMZ
```

Abbildung 2.2: Alle benutzten privaten Adressen

```
#Source-CIDR NAT-Adresse  # Comment
10.0.0.0/24  194.120.12.9 # Cologne
10.0.1.0/24  194.45.253.1 # Berlin
```

Abbildung 2.3: NAT an verschiedenen Stellen

2.4 RFC1918 – Private Adressen mit Folgen

Die Nutzung RFC1918-konformer Adressen innerhalb eines Unternehmens ist sinnvoll, hat aber im Verkehr mit der Aussenwelt einen Haken: Entweder werden Proxies zwischengeschaltet oder ein Rechner macht NAT[1]. Proxies sind nicht Bestandteil von SSPE. Innerhalb von SSPE ist ausschließlich SNAT[2] im Gebrauch , d.h. ausgehende Pakete werden mit der Quelladresse des Gateways abgschickt, in einer Kernel-Tabelle werden Quell- und Ziel-IP sowie die beiden beteiligten Port-Nummern festgehalten. Damit kann ein empfangenes Antwort-Paket wieder eindeutig einer TCP-Session auf einer internen Ziel-Adresse zugeordnet werden. Dabei werden die IP-Header beider Pakete mit neuen Checksummen versehen. Die zweite Alternative an den Gateways macht daher eine weitere, knifflige Angelegenheit nötig: Ein Paket aus Standort A muss nur dann per NAT ausgesendet werden, wenn sein Ziel nicht auch im privaten Netz liegt. Da ausschließlich SNAT zum Einsatz kommt, reduziert sich diese Angelegenheit auf eine einzige XOR-Operation von (Quelle ist privat) mit (Ziel ist privat). Hierfür ist eine Datei **privates** vorgesehen, ein Beispiel findet sich in Abbildung 2.2

2.5 NAT - Network Address Translation

Da innerhalb von SSPE an verschiedenen Stellen im Netz NAT sein muß, ist die Sache nicht ganz trivial. Es wird eine Tabelle `nathosts` benötigt, das Beispiel in Abbildung 2.3 auf Seite 12 sagt sicherlich mehr als lange theoretische Erklärungen. Grundsätzlich erfolgt innerhalb von SSPE-Maschinen nur am ersten Ethernet-Interface eth0 SNAT. Dies dient der Vereinfachung und

[1] Network Address Translation
[2] Source-NAT

```
192.168.88.0    172.16.10.2     255.255.255.248 UG 0  0  0 eth1
194.120.12.0    0.0.0.0         255.255.255.240 U  0  0  0 eth2
192.168.102.0   0.0.0.0         255.255.255.0   U  0  0  0 eth0
10.0.4.0        172.16.10.2     255.255.252.0   UG 0  0  0 eth1
172.16.0.0      0.0.0.0         255.255.0.0     U  0  0  0 eth1
0.0.0.0         192.168.102.10  0.0.0.0         UG 0  0  0 eth0
```

Abbildung 2.4: Beispiel für eine Routingtablle

hat zur Generierung der Iptables-Kommandos eine weitere Konsequenz: innerhalb der Datei `hostnet` müssen o.a. NAT-Adressen mit einem Namen gw-XX-e, also z.B. 'gw-k-e' erscheinen. Damit kann das 'Postrouting'-Kommando einer Regel eineindeutig zugeordnet werden.

2.6 Routingtabellen

Jede Maschine hat ihre eigene Routing-Tabelle entsprechend der Stelle, wo sie im Netzwerk untergebracht ist und welche anderen mit ihr kommunizieren sollen. Beispielhaft zeigt Abbildung 2.4 eine solche von einer Maschine mit drei Schnittstellen in unterschiedlichen IP-Netzen. Linux zeigt mit dem Befehl `route -n` die Routingtabelle einfacherweise so, wie abgearbeitet wird, um einen Routing-Cache Eintrag daraus abzuleiten. Der Routing-Cache ist mit `route -C` einzusehen, die Sortierung ist jedoch nicht lesefreundlich. Erzeugt und zugleich per ssh zum Administrator-PC transferiert werden die Dateien `/boot/routes` mit den Routing-Tabellen auf jedem Ziel-System mit einem einfachen Script in Abbildung 2.5. Der Linux-Kommandozeilen-Befehl `/sbin/route` wird nur um Kommentare erleichtert und bietet die Tabelle in der Sortierfolge an, wie der Kernel sie für jedes zu sendende IP-Paket durchsucht. Die Reihenfolge wird festgelegt anhand der Netzmasken, deren CIDR-Kennzahl ist absteigend sortiert. Der restliche Inhalt der einzelnen Einträge ist für die Sortierung irrelevant. Jede Änderung des Routings hat zur Folge, daß die Routingtabelle erneut zum zentralen Administrator-PC übertragen werden muß. Anschließend sollte dann erneut eine Generierung der Iptables-Kommandos erfolgen, da sich das Routing darin widerspiegelt. Insbesondere ist dies bei Maschinen wichtig, die keine 'default-route' haben, nicht aktuelle Tabellen bei der Generierung führen zu unerwarteten Ergebnissen, da die hier gefundenen Interfaces in die Iptables-Kommandos einfließen.
Die Routing-Tabelle wird zur Generierung der Iptables-Kommandos für jede Regel wie folgt ausgewertet: Falls Quell- und Zieladresse der Regel mit der gleichen Zeile der Routingtabelle erreicht werden (z.B. die letzte, also per

```
#!/bin/bash
#
# Date:     30.12.2001
# Version:  0.1
#
# VARIABLES SECTION
RFILE=/boot/routes
ROUTE=/sbin/route
GREP=/bin/grep
TEE=/usr/bin/tee
#
# EXECUTABLE SECTION
# if we already have one keep it
[ -r $RFILE ] && mv ${RFILE} ${RFILE}.old
# build new routingtablefile
${ROUTE} -n | ${GREP} -v I | ${TEE} $RFILE
```

Abbildung 2.5: `/root/bin/rn`: Routingtabelle lesen

default-route) und es sich nicht um eine eigene IP-Adresse handelt, ist das System, für das die Regel gerade abgearbeitet wird, vom beschriebenen Traffic nicht betroffen. Ergo wird dann auch kein Iptables-Kommando erzeugt.

2.7 Regeln

Der Linux-Kernel arbeitet bei jedem IP-Paket die eingestellten Regeln in der Reihenfolge ab, wie diese festgelegt wurden. Daher ist auch die Reihenfolge der Regeln in den Dateien wichtig für die Funktion der Firewall. Eine Regel mit der Aktion DROP vor einer mit gleichen Quellen und Zielen und der Aktion ACCEPT führt zum Ergebnis, das die zweite Regel nicht zum Zuge kommt.

Hier ein kurzer Gedanke zur automatischen Sortierung der Regeln: Zunächst wird als Sortierkriterium sicherlich das Produkt der Quell- und Ziel-maske (CIDR-Kennzahl) gelten, da Regeln für Einzel-Maschinen wahrscheinlich wichtiger sind als Regeln für ganze Netze. Anhand des Ergebnisses könnten die Regeln in absteigender Reihenfolge sortiert werden. Als weiterer Faktor müßte die Aktion mit gewichtet werden, aber wer sagt, ob DROP wichtiger ist als ACCEPT? Selbst wenn hierzu etwas festgeschrieben wird, sind die Folgen nur schwer abschätzbar. Einer Entstörung müßte immer zunächst das unsortierte *und* das sortierte Regelwerk zugrundeliegen. Das ist bei einigen

```
#Quelle Ziel   Richt. Proto  Port   Aktion  Optionen
adm-pc  def-gw Oneway  TCP   ssh    accept  LOG
def-gw  adm-pc Oneway  UDP   syslog accept  INSEC LOG
def-gw  adm-pc Twoway  TCP   all    deny
def-gw  adm-pc Twoway  UDP   all    deny
```

Abbildung 2.6: Beispiel für einen Regelsatz

```
LOG    Der durch die Regel beschriebene Traffic wird geloggt
DNS    Source-Port fest auf 53
FTP    Source-Port fest auf 20
IPSEC  Source-Port fest auf 500
INSEC  Source-Port kleiner 1024 erlaubt
```

Abbildung 2.7: Optionen zu einer Regel

hundert Regeln nicht mehr mit 'Keep it simple' zu vereinbaren, zumal der Administrator dann keine Möglichkeit mehr hat, selbst Einfluß auf die Reihenfolge zu nehmen.

Eine sinnvolle Wichtung der Aktionen im Zusammenhang mit Netzmasken der Quellen und Ziele scheint also z.Zt. nicht möglich. Daher wird von einer automatischen Sortierung durch SSPE bewußt abgesehen. Die manuelle Wichtung des Administrators bleibt somit die einzige, möglicherweise sicher erscheinende Variante. Die Regeln haben stets das Format wie in Abbildung 2.6. Eine Regel kann mehrere Optionen haben, den Sinngehalt bestimmt der Administrator. Da es z.B. wenig Sinn macht, Netbios-Broadcasts[3] zu loggen, kann dies mit der Aktion DROP gezielt verhindert werden. Solche Regeln haben ausschliesslich den Sinn, Logging der implizit fest eingebauten Default-Regel[4] zu unterbinden. [5]

Die letzten beiden Regeln haben zur Konsequenz, daß keinerlei Logging bzgl. des Admin-PC durch die Default-Regel für UDP- und TCP-Pakete erfolgt. Die möglichen Optionen sind in Abbildung 2.7 gezeigt. Des weiteren können die Regeln in verschiedenen Dateien abgelegt werden, die Reihenfolge der Abarbeitung liegt jedoch fest. Dazu später mehr.

[3] any any Oneway udp 137:139 drop

[4] any any Oneway any any drop LOG

[5] Das Verhindern des Logging von IP-Broadcasts ist zur Zeit noch problematisch, da `rules.pl` diese als nicht zur Maschine gehörig klassifiziert und infolgedessen kein iptables-Kommando erzeugt wird.

`$HOME`	wo ist nicht wichtig
`$HOME/bin`	Shell und Perlprogramme von SSPE
`$HOME/etc`	globale Einstellungen, hostnet, ipsecs, nathosts, privates
`$HOME/desc`	Je ein Unterverzeichnis pro Zielmaschine hierin
`$HOME/desc/def-gw`	Unterverzeichnis der Maschine def-gw: Regeln (evtl. als sym-links), Routingtabelle, evtl. hostnet, privates
`$HOME/software`	zu verteilende Programme und Konfigurationen
`$HOME/software.gws`	auf IPSec-Gateways zu verteilende Scripts
`$HOME/hardware`	Kernel etc

Abbildung 2.8: SSPE Verzeichnisstruktur

```
#!/bin/sh
[ -r /root/rules ] && /bin/sh /root/rules && exit 0
exit 0
```

Abbildung 2.9: init-script `/etc/init.d/iptables`

2.8 Verzeichnisstruktur

SSPE findet in genau einem Verzeichnis statt. Die Unterverzeichnisse sind in Abbildung 2.8 gelistet. Die verwendete user- und group-id spielen keine Rolle. Es muß per ssh ohne Passworteingabe möglich sein, auf allen beteiligten Maschinen als ROOT aktiv zu werden.

2.9 Init-scripts

Auf alle beteiligten Maschinen wird die Datei `root/rules` verteilt und ausgeführt. Um bei erneutem Booten ebenfalls wirksam zu werden, muß ein entsprechendes init-script in den Boot-Vorgang eingebunden werden. Es kann wesentlich einfacher aussehen, als die normalen Scripte, da unabhängig vom Parameter $start \| stop$ [6] die Iptables-Kommandos nur ausgeführt werden müssen. Die vielleicht einfachste Variante zeigt Abbildung 2.9.

[6] Eine besondere Behandlung beim Kommando stop scheint nicht notwendig

Kapitel 3

Und es funktioniert doch

3.1 ASCII-Grafik

3.1.1 Das Hauptmenü

Als Hauptmenü erscheint Abbildung 3.1. 'dialog' findet auch hier Verwendung. So kommt etwas Farbe ins Spiel. Der Hinweis auf die Lizensierung darf angesichts aktueller rechtlicher Auseinandersetzungen nicht fehlen, `GNU General Public License` passt bestens zur verwendeten `Debian GNU/Linux` Distribution. Der vollständige Text liegt dem Programm bei und ist per 'Entertaste' einsehbar.

Der Eintrag[1] `rules administration` ist verbesserungs- bzw. erweiterungsbedürftig, lediglich Anschauen der Regeln und ein Sprung in den Standard-Editor zum Ändern der `rules.users` reicht aber zur bequemen Arbeit aus. Anspruchsvollere Zeitgenossen sind aufgefordert, die Funktionalität nach Belieben zu erweitern.

Mit `apply rules on all machines` wird zuerst eine Auswahlbox wie in Abbildung 3.3 angeboten. Der erste Eintrag lautet 'ALL' und ist erstmal ausgewählt. Darunter werden alle aktiven Maschinen in je einer Zeile zur Auswahl angeboten für den Fall, dass nicht 'all' gewünscht ist. Ein weiterer Tastendruck auf 'Enter' erledigt die Auswertung und Anwendung der Regeln für alle zuvor ausgewählten Maschinen im Hintergrund durch `dist_rules`. Im Vordergrund werden im drei-Sekunden Rhythmus die noch nicht fertigen Maschinen angezeigt. Wenn alle fertig sind, erscheint automatisch das Hauptmenü dann wieder, wenn keine Fehler aufgetreten sind. Sollte ein Fehler aufgetreten sein, wird er angezeigt und muß mit 'Enter' quittiert werden.

[1] Alle Menüeinträge haben Modellcharakter und können deutlich administratorfreundlicher gestaltet werden

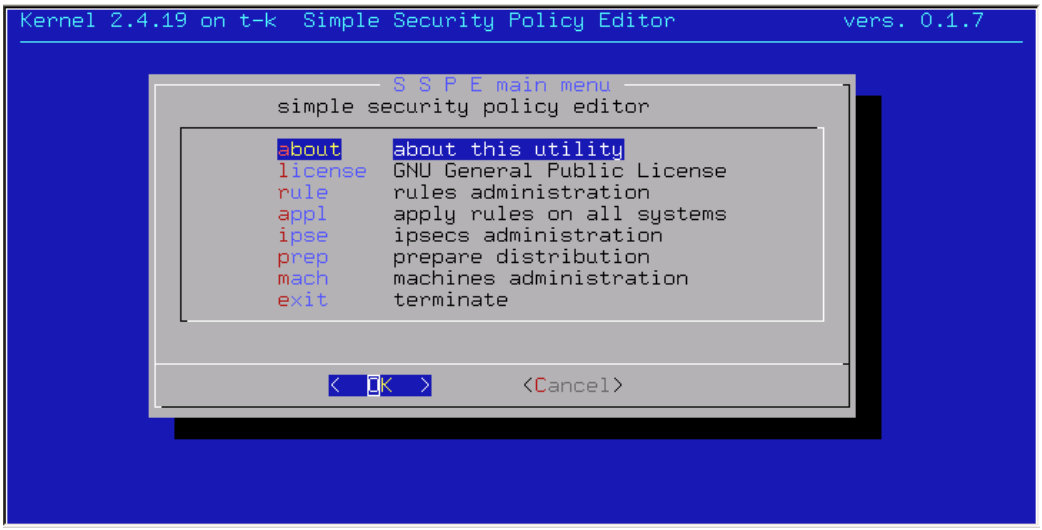

Abbildung 3.1: SSPE Hauptmenu

Hinter dem Menüeintrag `ipsec administration` verbirgt sich eine einfache Sache: Zuerst erscheint die Datei `ipsecs` ohne Kommentare, hier kann bei Nichtgefallen durch CTRL-C abgebrochen werden. Nochmals 'Enter' bewirkt Generierung und Verteilung der IPSec Konfiguration und 'Preshared Keys'. Das zugrunde liegende Programm ist `mach.ipsec`. Es ruft u.a. `ipsec.generator` auf.

3.1.2 Maschinen-Verwaltung

Nach verschiedenen Kriterien sortiert können hier die bereits eingebundenen Maschinen aufgelistet werden. Auch Eingliedern von neuen Maschinen in die zentrale Administration ist hier angedacht. Nachdem unter 'Add' zunächst der Kurzname, die IP-Adresse und eine Kurzbeschreibung eingegeben wurden, ist die Arbeit mit dem Menü beendet. Damit wurde in `$HOME/desc/` ein Unterverzeichnis mit dem Kurznamen und darin die Dateien `ip` und `desc` angelegt. Außerdem werden symbolische Links nach `etc` für `rules.admin`, `rules.ipsec` und `rules.users` angelegt. Eine Datei namnes `hw` muß manuell mit dem Hardwarenamen angelegt werden. Angedacht ist diese für eine praktikable Verwaltung unterschiedlicher Hardware-Systeme mit unterschiedlichen Kernel-Releases. Daran anschließend muß noch die Routingtabelle mit `get_routing_table` geholt werden. Selbstverständlich sollte zuvor die Kurzbezeichnung in `/etc/hosts` eingetragen sein und ein Root-Zugang per ssh mittels RSA- oder DSA-Authentifizierung erfolgreich getestet sein.
Da nicht immer alle Maschinen online verfügbar sind, kann mit 'deac' vor-

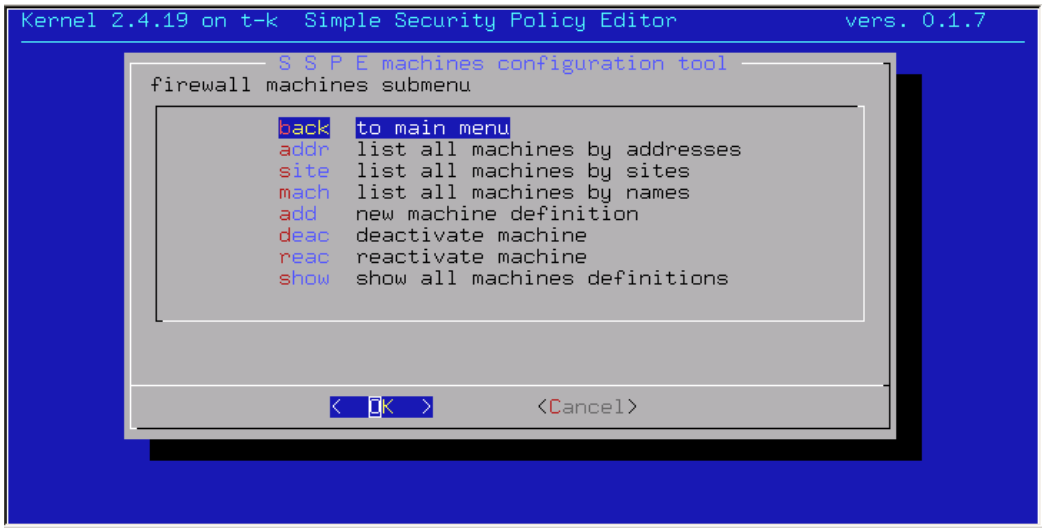

Abbildung 3.2: Beteiligte Maschinen

übergehend eine Maschine ausgegliedert werden. Sie wird dann von keinem Script mehr angesprochen, da einfach der Name des Unterverzeichnisses mit einem Punkt am Anfang erweitert wird. Die 'Reaktivierung' nimmt den Punkt wieder weg.[2]

3.1.3 Apply

Um editierte Regelsätze in den Maschinen wirken zu lassen, müssen diese erst in entsprechende Iptables-Kommandos übersetzt werden. Anschließend erfolgt die Verteilung mit unmittelbarer Anwendung. Innerhalb von `bin/rules.pl` wird derzeit die Default-Policy an zwei Stellen gesetzt:
Beim Start des `/root/rules` und unmittelbar vor Ende diese Scripts. Der Grund wird im Kapitel mit den Betriebserfahrungen nachgereicht.

3.1.4 Regeln

In dem Menü zu den Regeln (Abbildung 3.4, S. 21) ist einiges noch nicht fertig. Ansehen kann man sich die `etc/hostnet`, sortiert nach Adressen oder nach Namen. Eine Verzweigung auf maschinenspezifische Konfigurationsdateien, z.B. `desc/xxx/hostnet`, wäre wünschenswert. Leider können z.Zt. auch nur die Regeln in `etc/rules.*` angesehen werden, evtl. maschinenspezifische Dinge werden nicht mit eingebunden. In der Konsequenz kann

[2] /bin/ls zeigt Dateien mit '.' am Anfang des Namens nicht

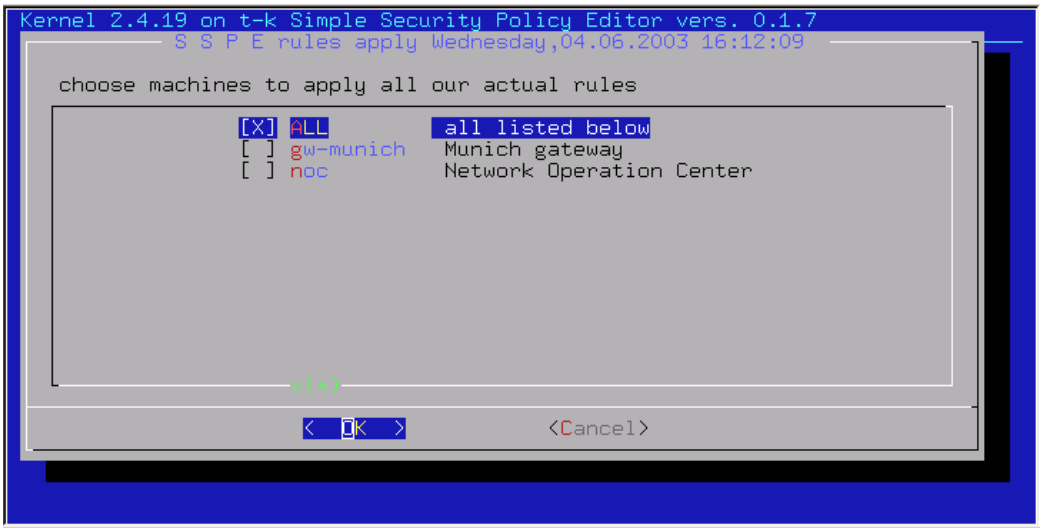

Abbildung 3.3: Apply

auch die Numerierung der Regeln falsch sein. Mit 'vi' editiert werden kann `etc/rules.users`, dies sollte im Normalfall ausreichen. Natürlich kann kein Administrator davon abgehalten werden, die Menüstrutur zu verlassen und selbst Hand an die gewünschten Dateien zu legen.

3.2 Kernprogramm

Der Kern des Programms in Abbildung 3.5 wird normalerweise innerhalb des SSPE durch den Menüpunkt 'Apply' aufgerufen.
Als Parameter wird der Name des Zielsystems übergeben, ein korrekter Eintrag in **/etc/hosts** für dieses ist selbstverständlich. Auf keinen Fall sollte sich ein sicherheitsbewusster Administrator auf die korrekte Funktion seines DNS-Server[3] verlassen, seine **/etc/hosts** sollte nur durch root schreibbar sein. Ebenso sollte ein Login via ssh durch RSA-Authentication ohne Angabe eines Passwortes unmittelbar möglich sein. Dieser Name wird in allen Unterprogrammen verwendet, um die zugehörigen Dateien im richtigen Unterverzeichnis von **$HOME/desc** zu finden. Jegliche Ausgabe erfolgt auf **stdout** und wird im aufrufenden Shellscript umgeleitet.
Zunächst wird mit `site_header` ein vordefinierter Kopf ausgegeben, der zum einen **/etc/motd** des Zielsystems auf aktuellen Stand bringt und des weiteren alle vorhandenen Iptables-Einträge löscht und die Tabellen neu anlegt.

[3]DNS ist wichtig; zuverlässig im Sinne von Sicherheit aber aufgrund seiner Mächtigkeit nicht. `/etc/hosts` ist einfach und, ganz wichtig, nicht von anderen Maschinen abhängig.

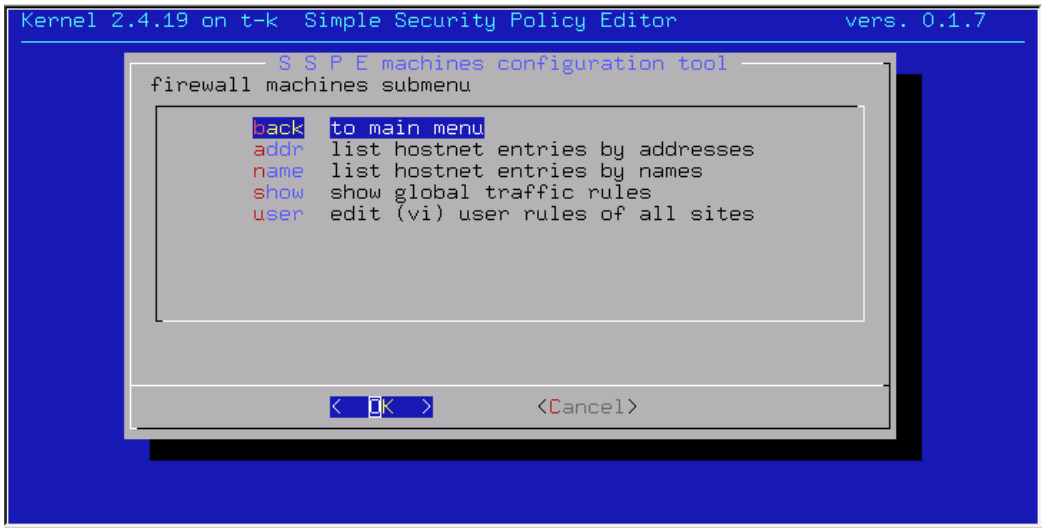

Abbildung 3.4: Verwaltung des Regelwerks

Je nach Gusto wird als letztes in diesem Kopf die default-Policy [4] eingestellt, dies kann bei vermindertem Grad der Paranoia des Administrators für die Dauer des Scriptlaufs auch ACCEPT sein. Am Ende des Scripts wird in jedem Falle die default-Policy auf DROP umgestellt. Die variable Einstellung ist einer zukünftigen Version vorbehalten.
Die nächste Funktion `hostnet` liest die Hostdefinionen zunächst im Unterverzeichnis der Zielmaschine. Sollte dort keine Datei **hostnet** vorhanden sein, wird im Verzeichnis **$HOME/etc** eine solche erwartet. Falls auch diese nicht lesbar sein sollte, bricht das Programm ab. Ein an vielen Stellen bewährtes Prinzip, Fehlermeldungen zu vermeiden, um den laufenden Betrieb nicht zu stören oder gar zu unterbrechen, ist auch hier implementiert; erst am Ende aller Aufrufe erscheint u.U. eine Fehlermeldung auf der Konsole. [5]
Dann liest `nat_hosts` die Netze und IP-Adressen, für die und an denen NAT passieren soll. Daran anschließend werden die privaten Adressen mit `read_privates` eingelesen, um definitiv NAT zu verhindern, falls Quelle und Ziel in den privaten Adressen enthalten sind. Auch diese beiden Dateien werden zunächst im Unterverzeichnis der Zielmaschine gesucht und anschließend im **$HOME/etc** Verzeichnis.
Nun folgt das Einlesen der Routingtabelle **routes** des Zielsystems mit der Funktion `routes_read`. Ebenso wird die Interface-Konfiguration aus **parameter** gelesen. Beides dient der Entscheidung, ob das Ziel-System überhaupt die im Anschluss zu lesenden Regeln abarbeiten muß. Diese Dateien werden

[4] hart kodiert in rules.pl, ist eine Einstellung per Menü wirklich wünschenswert?
[5] Bei Auftreten eines Fehlers sollten alle anderen Maschinen fertig bearbeitet werden

```
$site=$ARGV[0];
$site="" unless $site;
site_header ($site);
print "# reading hosts & network definitions\n";
hostnet ($site);
nat_hosts ($site);
print "# reading private network definitions\n";
read_privates($site);
print "# reading routing informations\n";
routes_read ($site);
print "# reading access rules.admin\n";
# read admin rules
rules_read ($site, "admin");
# define the partial ruleset filenames
@rulesets = ("head", "ipsec","local", "users", "tail");
# now read all the rest
foreach $act (@rulesets) {
        print "# reading access rules.$act\n";
        rules_read ($site, $act);
}
print "# process all access rules\n";
rules_treatment ($site);
site_trailer ($site);
exit 0;
```

Abbildung 3.5: `bin/mach` – Perl Hauptprogramm

ausschließlich im Unterverzeichnis der Zielmaschine gesucht. Damit sind die Vorbereitungen abgeschlossen, es fehlen noch die Regeln, die mittels obiger Einstellungen in Iptables-Kommandos umzusetzen sind.

Das Regelwerk ist bewußt auf mehrere Dateien verteilt. Ein administrativer Teil wird in **rules.admin** separat behandelt, diese Regeln werden sicherlich als erste eingetragen und lassen z.B. ssh ins Zielsystem zu. Der Grund für die besondere Behandlung ist folgender:

An diesen wird sicherlich am wenigsten geändert und evtl. versehentliches Löschen hat fatale Folgen: — **Aussperrung!** — Administrators Rettung ist dann nur noch eine nette Kollegin oder ein netter Kollege vor Ort oder eine mehr oder weniger unangenehme Dienstreise. Das ist nach Möglichkeit zu vermeiden. Weiter werden nun die Regeln in folgender Reihenfolge gelesen: **rules.head**, **rules.ipsec**, **rules.local**, **rules.users**, **rules.tail**. Jede dieser Dateien wird zunächst im Unterverzeichnis der Zielmaschine und anschließend in **$HOME/etc** gesucht. Falls eine Datei nicht gefunden wird, erfolgt keine Fehlermeldung. Nachdem alle Regeln eingelesen wurden, erfolgt deren Abarbeitung. Abschließend wird ein Trailer ausgegeben, in dem die Default-Regel und Default-Policy gesetzt werden. Bei ca. 600 Zeilen Regelwerk dauert ein Abarbeiten incl. Transfer und Ablaufen auf dem Zielsystem[6] bis zu 6 Minuten. Um diese lange Wartezeit für den Admin, verbunden mit Herzklopfen, Stirnschweiß, ..., zu überbrücken, zeigt die Konsole den Verlauf an.

3.3 Regelumsetzung

Das Unterprogramm `rules_treatment` gibt als Kommentar auf `stdout` zunächst Quelle, Ziel, etc. aus, um dann innerhalb von `solve_groups` Gruppierungen aus der Datei `hostnet` aufzulösen. Das bedeutet, für jeden gefundenen Eintrag in `hostnet` zu Quelle wird einmal die Regel in entsprechende Iptables-Kommandos übersetzt. Ebenso wird mit dem Ziel in der Regel verfahren. Falls Quelle oder Ziel nicht in `hostnet` gefunden werden, wird einfach mit der nächsten Regel fortgefahren. Eine Gruppierung von Services wie z.B. 'domain' erscheint aufgrund mangelnder Transparenz nicht sinnvoll.

Die in jeder Regel mögliche Aktion ACCEPT, REJECT oder DROP geben dem Administrator genügend Flexibilität im Umgang mit erwünschtem oder unerwünschtem IP-Verkehr. Zu Entstörungszwecken oder aus anderen Gründen kann eine beliebige Regel mit der zusätzlichen Option LOG versehen werden, **/var/log/kern.log** zeigt dann jedes betroffene IP-Paket sehr deutlich an.

[6] Pentium IV^{TM}, 1000MHz, 256Mb

3.4 Verschlüsseln – IPSec

Die Kommunikation zwischen den Standorten im öffentlichen Netz wird sinnvollerweise verschlüsselt. Es gibt viele verschiedene Arten, Datenverkehr zu verschlüsseln, IPSec ist aber wohl die erste Wahl, wenn man nicht zu 'Security by Obscurity' neigt. Offengelegte Verschlüsselungstechnik wird von vielen Personen durchgesehen und bietet so möglicherweise weniger Angriffsfläche als proprietäre Systeme. Wenn nur der Hersteller und der dazugehörige nationale Geheimdienst um die Methodik wissen, mag das zwar technisch einwandfrei sein, sicherheitstechnisch aber höchst bedenklich. [Bau97] [Sch00] Verschiedene Hardware-Hersteller trugen zur Normierung von IPSec bei, als Referenz-Implementierung ist FreeS/WAN bekannt. [Fre02] Es ist zu fast allen Hardware-IPSec Implementierungen ebenso kompatibel wie zu den Software-Lösungen verschiedenster Firewall-Hersteller. Verfügbar ist die Implementierung als Kernel-Patch. Er kann durch einen (unabhängigen) X.509-Patch um die wichtige Funktionalität erweitert werden, um mit X.509-Zertifikaten gegenseitig zu authentifizieren. [Cha02] [Pol01] Dies ist im Zusammenhang mit Außendienstmitarbeitern interessant.

3.4.1 IPSec Konfiguration – Kernel-Patch

Der Patch muß zum Standardkernel von kernel.org passen. Die Dokumentation auf der Web-Seite [Fre02] ist hervorragend und wenn die Pakete vorliegen, kann sofort mit der Kernel-Konfiguration begonnen werden. Da der Kernel auf Systemen eingesetzt werden soll, die zur Sicherheit beitragen, ist dringend von der ansonsten sehr beliebten Modularisierung des Kernels abzuraten. Folgende Direktiven beziehen sich auf FreeS/WAN:

```
CONFIG_IPSEC=y
CONFIG_IPSEC_IPIP=y
CONFIG_IPSEC_AH=y
CONFIG_IPSEC_AUTH_HMAC_MD5=y
CONFIG_IPSEC_AUTH_HMAC_SHA1=y
CONFIG_IPSEC_ESP=y
CONFIG_IPSEC_ENC_3DES=y
CONFIG_IPSEC_IPCOMP=y
CONFIG_IPSEC_DEBUG=y
```

Dann reicht ein Aufruf von make:

```
make dep clean bzimage
```

und nur noch die Verteilung steht dem Booten der Systeme bevor. Ach ja, FreeS/WAN braucht noch etwas Konfiguration, damit Tunnel gebaut werden können...

```
gw-cgn          194.120.12.9/32      # Cologne
gw-bln          194.45.253.1/32      # Berlin
gw-ffm          194.45.252.1/32      # Frankfurt
ipsecs          194.120.12.9/32      # Cologne
ipsecs          194.45.253.1/32      # Berlin
ipsecs          194.45.252.1/32      # Frankfurt
```

Abbildung 3.6: Ausschnitt aus **hostnet**

```
ipsecs    ipsecs  1     udp    500 accept IPSEC
ipsecs    ipsecs  1     esp    all accept
```

Abbildung 3.7: IPSec-Regeln **rules.ipsec**

3.4.2 IPSec Konfiguration – hostnet

Um IPSec weitestgehend automatisch zu handhaben, werden die Adressen der IPSec-Gateways an mehreren Stellen eingetragen; eine einheitliche Namenskonvention hilft dabei, nichts zu übersehen. Die IPSec-Gateways werden alle mit 'gw-XX' (XX ist das Kürzel des Standortes) benannt, außerdem werden sie zusätzlich mit der gleichen IP in **hostnet** als 'ipsecs' eingetragen, um so als Gruppe referenziert werden zu können. Das reduziert den Aufwand im Regelwerk deutlich, Abbildungen 3.6 und 3.7 zeigen den Zusammenhang.

3.4.3 IPSec Konfiguration – ipsecs

IPSec macht eine weitere, eigentlich unschöne Sache notwendig. Die Konfiguration von FreeS/WAN [Fre02] [KH02] gibt nicht nur die eigenen IP-Adressen, sondern auch die der jeweiligen nächsten Hops an. Ergo ist hierfür eine Datei **ipsecs** (Abbildung 3.8) nötig. Anhand der Tabelle werden die IPSec Konfiguration und die nötigen 'Preshared Keys' erzeugt, so daß am Ende nur wenig Handarbeit übrig bleibt. Später mehr dazu.
Damit wird der Einfachheit halber von einem voll vermaschten Netz ausgegangen, um die **/etc/ipsec.conf** für die IPSec-Gateways zu erzeugen.

```
# loc.  gateway         next-Hop         subnet
cgn     194.120.12.9    194.120.12.10    10.0.0.0/8
bln     194.45.253.1    194.45.253.2     10.0.1.0/24
ffm     194.45.252.1    194.45.252.2     10.0.2.0/24
```

Abbildung 3.8: `ipsecs` Konfigurationstabelle

Die Ausgangsdaten liegen in **ipsecs**, ein konstanter Vorspann ist fest in `mach.ipsec` verankert.
Die Gesamt-Anzahl der IPSec-Tunnel N steigt in Abhängigkeit von der Anzahl Standorte S gemäß $N = S * (S - 1)$ an. Für jedes einzelne Gateway folgt entsprechend $N = S - 1$ als Anzahl aktiver Tunnel. Bei mehr als drei Standorten will sicher niemand die Tunnelkonfiguration per Hand vornehmen, ein Script `mach.ipsec` erledigt das schneller und vor allem vollständig. Im dort aufgerufenen `ipsec.generator` werden neben der Konfiguration auch alle 'Preshared Keys' erzeugt.
Die zentral generierte Konfiguration wird auf alle IPSec-Gateways verteilt, FreeS/WAN sucht beim Start anhand der IP-Adressen selbständig, welche Tunnel auf dem System herzustellen sind. Dieser Automatismus hat den Nachteil, jeweils alle Tunneldefinitionen auf allen Gateways gleichzeitig in Kraft setzen zu müssen. Um Ausfallzeiten gering zu halten, ist daher eine Synchronisation per `ntp`[7] sinnvoll. Damit kann dann der durch Neukonfiguration verursachte Ausfall auf wenige Sekunden begrenzt werden. Dies setzt aber voraus, dass alle IPSec-Gateways zum Zeitpunkt der Verteilung erreichbar sind. Da dies am Internet nicht gewährleistet werden kann, ist im Konzept ein temporäres Deaktivieren vorgesehen. Der Administrator muß sich unmittelbar vor der IPSec-Konfiguration von der vollständigen Erreichbarkeit seiner Gateways überzeugen.

3.4.4 IPSec Konfiguration – Preshared Keys

Um es einfach zu machen, werden die Preshared Keys automatisch erzeugt. Dabei gehen verschiedene Werte in die Generierung ein: aktuelles Datum, Uhrzeit und eine Zufallszahl. Letztere genügt sicherlich keinen strengen kryptographischen Anforderungen, aber die erzeugte MD5 Checksumme sollte zufällig genug sein, um nicht erraten zu werden.[8] Mit der Erzeugung einer neuen IPSec-Konfiguration, die auf allen Gateways gleich ist, werden ebenfalls die neuen IPSec-Schlüssel erzeugt und verteilt. Die Konfiguration wird sofort nach `/etc/ipsec.conf` geschrieben, die Schlüssel in `/etc/ipsec.secrets.new` abgelegt. Alle Systemuhren werden per NTP synchron gehalten. So kann ein cronjob `ipsec-supervisor` (Abbildlung 3.9) jede Minute darüber wachen, ob neue ipsec-secrets angekommen sind. Falls neue verteilt wurden, werden diese in `/etc/ipsec.secrets` umbenannt und dann mittels `/etc/init.d/ipsec restart` *gleichzeitig alle* IPSec-Tunnel neu gestartet.

[7] Network Time Protocol

[8] Wenn es auch IPSec-Gateways ohne Vermaschung gibt, sind die 'Preshared Keys' dieser Geräte in `ipsec.generator` 'hart kodiert'.

```
#!/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
```

Abbildung 3.9: `ipsec-supervisor` auf voll vermaschten Systemen

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

Abbildung 3.10: `ipsec-supervisor` auf nicht vermaschten Systemen

Der durch den Restart verursachte Ausfall ist, wenn alle Systeme erreichbar sind, innerhalb von wenigen Sekunden abgeschlossen; die Retry-Timer gängiger TCP-Implementierungen reichen im Allgemeinen aus, dies unbemerkt tun zu können. Dabei darf natürlich kein ICMP DEST-UNREACHABLE in die Quere kommen; genau hier ist das Geschick des Administrators gefragt, seine Firewall-Regeln zielorientiert und korrekt zu formulieren.
Wenn bei der Neukonfiguration durch den `ipsec-supervisor` eine Datei `/etc/ipsec.conf.const` existiert, wird diese zuvor an die `/etc/ipsec.conf` angehangen. Zu den darin konfigurierten IPSec-Tunneln muß es natürlich eine Gegenseite geben. Diese kann (im Falle einer ISDN-Wählverbindung zu einem Standort, also keiner vollen Vermaschung!) IPSec-Tunnel nur zu genau einer Stelle enthalten. Dort sieht `ipsec-supervisor` dann etwa wie in Abbildung 3.10 aus. Hier wird `/etc/ipsec.conf` einfach durch `/etc/ipsec.conf.const` überschrieben. Dieser Trick funktioniert, auch wenn es lange gedauert hat, ihn herauszufinden...

Kapitel 4

Betriebserfahrungen

4.1 Ein Standort

Eine einheitliche Konzeption der Standorte beginnt damit, jeden mit der gleichen Anzahl IP-Adressen auszustatten. Durch Verwendung von RFC-1918 Adressen ist die Vergabe von je einem /20 oder mehr möglich; auch wenn nur 10 Personen dort arbeiten, hindert es nicht. Bei den offiziellen Adressen, die durch den jeweiligen Provider vergeben werden, sieht die Sache anders aus: Der Mangel beherrscht die Möglichkeiten. Der Anschluß sollte dennoch überall gleich gestaltet werden, wie z.B. in Abbildung 4.1 dargestellt. Der Router des Providers routet die offiziellen Adressen über ein Transfernetz auf unsere Firewall, diese hat das zugewiesene Netz auf dem inneren Interface. Hier sind evtl. öffentliche Standort-Server angeschlossen und das örtliche IPSec-Gateway. Dieses ist mit seinen weiteren Schnittstellen intern als Default-Gateway eingetragen, eine Unterteilung in 'User-'[1], 'Server-'[2] , 'RFU-'[3] und sog. 'Leased-LAN'[4] erscheint gleichermaßen sinnvoll wie einprägsam. So können die einzelnen Bereiche mit Firewall-Regeln gegenseitig geschützt werden. – Nach glaubwürdigen Angaben von Analysten erfolgen 80 Prozent aller Angriffe auf IT-Systeme von innen. – Im User-LAN wird zusätzlich ein weiterer Server aufgestellt: DHCP und DNS. Als nützliches 'Add-On' gibts ein NIDS[5] mit snort. Das ist lehrreich und macht z.B. Wurmbefall nicht nur durch abnorm starken Traffic deutlich.

[1] Alle Arbeitsplatz-PCs
[2] Mailserver,Datenbanken,...
[3] Reserved for Future Use
[4] zum Anschluß von Kunden, Home-Offices...
[5] Network Intrusion Detection System

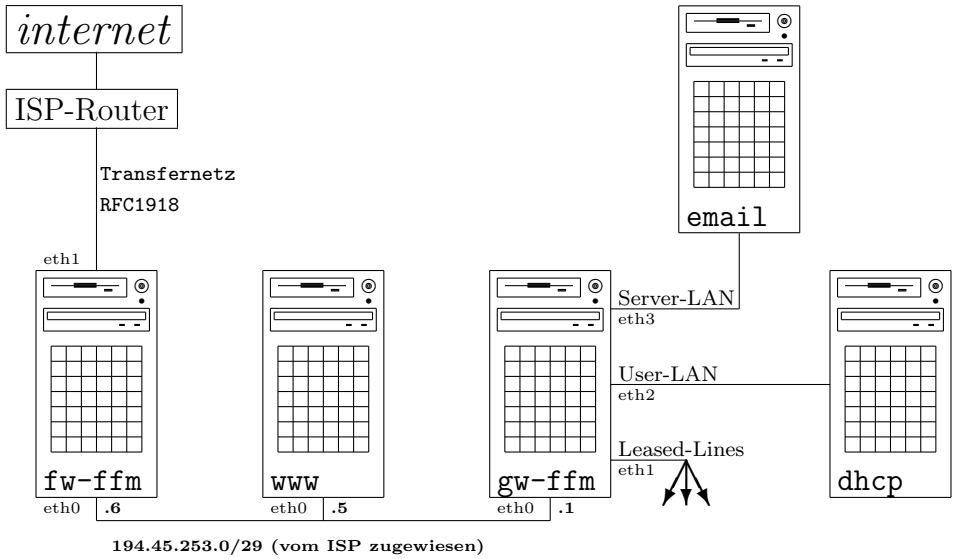

Abbildung 4.1: Typischer Firmen-Standort

4.2 Inbetriebnahme

Nach mehr als einem Vierteljahr reiner Entwicklung (natürlich neben dem Alltagsgeschäft) sind die ersten Internetanschlüsse verfügbar: zwei Standorte mit 2MBit/s, drei andere mit je 128 kBit/s. Die Entwicklungsumgebung, in welcher das Internet durch einige Router[6] simuliert wurde, wird zugunsten der realen Welt verlassen. Als erstes wird der zentrale Administrator-PC dupliziert, Debian ist die erste Wahl. Die Installation erfolgt problemlos.
Wegen angeblicher Support-Vorteile wird seitens des Managements für die Rechner am Internet eine weit-verbreitete kommerzielle Linux-Distribution gewählt. Die Duplizierung der gewählten Paket-Zusammenstellung zeigt sich einfach, wenn die Handbücher richtig gelesen werden. Als Problemfall erweist sich eine Hardware-Komponente: der IDE-RAID-Controller. Solange er mit dem Kernel der Distribution betrieben wird, funktioniert er. Die RAID-Funktion wird erfolgreich getestet und für gut befunden. Allerdings widerspricht diese Kernel-Konfiguration deutlich den Sicherheitsvorstellungen: modularisierte Kernel sollten vermieden werden, wenn die Systeme am Internet betrieben werden. Eine aktuelle Version sollte es ebenfalls sein. Da alle Systeme ausschließlich mit IDE-Platten und einer Sorte Netzwerk-Karten beschafft wurden, fällt mit dem Ziel geringen Wartungsaufwandes die Entscheidung,

[6]Mehrere über Serial-Interfaces verbundene Router lassen Bandbreiten-Simulation bis zu 4 MBit/s zu

nur einen Kernel zu backen und auf allen Systemen zu nutzen. Nur mit dem RAID-Controller will dieser Kernel nicht booten. Lilo zeigt zwar ordentlich und wie erwartet an, welche Kernel im Bootsektor eingetragen werden, jedoch wird das System anschließend mit dem alten Kernel gestartet. Da soll dann der Support, für teures Geld gekauft, helfen. Eine schnelle Mail an den Hersteller, und schon überzeugt die Antwort von der Leistungsfähigkeit des kommerziellen Software-Supports: diesen RAID-Controller unterstütze man zwar mit den eigenen Kerneln, aber mit dem Problem sollten wir uns doch an den Hersteller des FreeS/WAN-Patches wenden. Ein inhaltlicher Zusammenhang wurde nicht hergestellt. Letztlich wird auf den Einsatz der RAID-Controller verzichtet; nur *ein* Ausfall einer IDE-Platte wird seit Betriebs-Beginn von 24 Systemen beobachtet. Auch dieser erwies sich als unkritisch, das System (Debian woody) lief 'read-only' weiter, bis eine Ersatz-platte zur Stelle war. Aufgefallen war der Ausfall des DHCP-Servers, diese Funktion wurde kurzfristig durch ein anderes Ger ät übernommen. Andere Dienste der Maschine waren nicht betroffen.

4.2.1 Administrations-PC

Aus Personalgründen sollen die Firewalls von verschiedenen Standorten aus administriert werden. Daher werden als erstes die IP-Adressen festgeschrieben, von denen aus per ssh Zugang zum Administrations-PC erlaubt sein soll. Die meisten administrativen Dinge können so tatsächlich im Alltagsgeschäft erledigt werden. Nur wenn, wie die Erfahrung zeigt, Internetanschlüsse nicht verfügbar sind, funktioniert das naturgemäß nicht so performant wie gewünscht, da am zentralen Standort zusätzlich noch ein Wähleingang steht. Auch die dort verwendeten IP-Adressen der Admins sind zugelassen. Als Backup für den Normalbetrieb ist dieser zusätzliche Eingang nicht vorgesehen. Damit steht der erste Regel-Teil `rules.admin` fest.

4.2.2 Backup nötig?

Der Einfachheit halber werden Firewalls und Gateways mit gleicher Hardware ausgestattet. 5 Ethernet-Schnittstellen sollten pro Maschine reichen. Durch den von der Distribution bereitgestellten Mechanismus ist die Erstinstalltion neuer Maschinen in sehr kurzer Zeit getan. Nur etwa 15 Minuten dauert es, wenn die zweite CD-ROM sofort nach der Aufforderung eingelegt wird. Diese Eigenschaft wird zum wichtigsten Eckstein des Backup-Konzeptes: Original-Distribution installieren, Konfigurations-Dateien einspielen, fertig. Hierzu muß weder ein Band-Roboter noch ein spezielles Backup-Programm benutzt werden: Ein einfaches Shell-Script mit tar wie in Abbildung 4.2.2

```
#!/bin/bash
# Date: 25.1.2002
# Script to quick backup remote filesystems $DIRS
if [ -z "$1" ] ; then
    echo "    Usage: `basename $0` [-v] machine-name"
    exit 1
fi
verbose=""
case $1 in
    -v) verbose=v ; shift ;;
esac
DATE=`/bin/date +%Y%m%d%H%M`
DIRS="/boot /root /etc"
cd $HOME/desc
if [ ! -d "$1" ] ; then
    echo "`basename $0`: Error: machine-name $1 NOT found, abort"
    exit 1
fi
NAME=$1
STORAGE=/var/qbs
FILE="${NAME}.${DATE}.tgz"
echo "`basename $0`: start for ${NAME}, keep patience"
ssh ${NAME} "tar cz${verbose}lf - ${DIRS}" >${STORAGE}/${FILE}
exit 0
```

Abbildung 4.2: Backup-Script

reicht völlig aus. Als Parameter wird nur der Hostname mitgegeben, ssh erledigt den Rest. Mit dem Script wird von allen Maschinen ein sog. 'Quick-Backup' gezogen und zentral abgelegt. Diese nur wenige MBytes großen Dateien liegen auf dem Administrator-PC, der selbstverständlich mit Bändern regelmäßig gesichert wird. Damit im Fall der Fälle in den Standorten nicht auf den Transport einer Maschine gewartet werden muß, ist eine als Reserve vor Ort. Ein Satz Original-CDs und eine 'Rescue-CD' mit allen Quick-Backups ebenfalls.

4.2.3 Firewalls und Gateways

Als erste Systeme werden die Firewalls und Gateways an den beiden Standorten installiert, die mit 2Mbit/s angeschlossen sind. Die Performanz, mit der die Gegenseite erreichbar ist, entspricht den Erwartungen aus der Entwicklungsumgebung, wo schon die Taktrate auf 2Mbit/s begrenzt wurde. Die

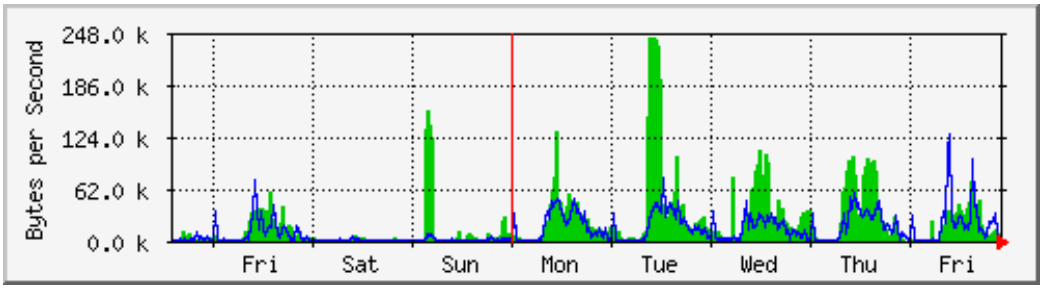

Abbildung 4.3: Bandbreiten-Nutzung

Geschwindigkeit von Datenbank-, Webserver- und anderen Zugriffen überzeugt auch die letzten Zweifler von der Richtigkeit der Vorgehensweise. Im weiteren Verlauf wird die Nutzung der verfügbaren Bandbreite mit MRTG, einem Hilfsmittel zur graphischen Aufbereitung eben solcher Daten, auf einer internen Webseite dargestellt. Engpässe treten danach nur dann auf, wenn Backups von Anwendungsservern über die IPSec-Verbindungen 'geschaufelt' werden. In Abbildung 4.3 ist ein typisches Bild einer Wochen-Übersicht eines der beiden Standorte.

In den ersten Wochen realen Betriebes kommt es immer wieder zu Änderungen der Regeln. Infolgedessen wird manchmal auch mehrfach während der normalen Arbeitszeit ein 'Apply' durchgeführt. Da zu Anfang immer alle Tabellen gelöscht werden, um neu aufgebaut werden zu können, werden dabei einige TCP-Sessions gekappt. Dies führt zu Ärger mit den Kollegen, insbesondere mit denjenigen, die sich anschließend neu anmelden müssen. Solche Pein hatte es mit den alten Scripts (LPFC, siehe Kapitel 1) nie gegeben. Iptables funktioniert eben anders. Abhilfe schafft ein sicherheitstechnisch vielleicht bedenklicher Versuch, während des 'Apply' die Default-Policy auf ACCEPT zu stellen, bevor am Ende DROP gesetzt wird. Andererseits ist nirgendwo bekannt, wann eine solche Lücke aufgemacht wird.

4.2.4 Email - aber wie?

Sendmail[7] ist sowohl für seine Flexibilität als auch für seine Sicherheitsprobleme bestens bekannt. Auch bekannt ist, dass diese Probleme nur aufgrund der hohen Komplexität des Programms beim Empfang auftreten. Es sollte also im Internet keine Emails annehmen, wenn nicht zwingende Gründe dafür sprechen. Es widerspricht jedem auch noch so halbherzig gestellten Anspruch auf Einfachheit. Andere, leicht durchschaubare Hilfsmittel lassen sich finden,

[7] http://www.sendmail.org

```
allow:*mydomain.de:ALL:ALL
noto:ALL:*we.dont.like.you.com:ALL
noto:ALL:ALL:former.employe@mydomain.de
allow:ALL:ALL:*mydomain.de
allow:Gateway-IP:*mydomain.de:ALL
noto:ALL:*mydomain.de:*mydomain.de
allow:ALL:ALL:*mydomain.de
noto:ALL:ALL:ALL:551 Sorry %H(%I), no relaying from %F to %T
```

Abbildung 4.4: smtpd-Konfiguration

so z.B. `smtpd` von Obtuse[8]. Sendmail kann dabei immer noch zum Versenden eingesetzt werden, altbekannte und lieb-gewonnene Mechanismen wie MAILERTABLE und MAILDOMAINS lassen sich so weiter nutzen. Den Empfang leistet das durch `xinetd` aufgerufene und in CHROOT-Umgebung laufende `smtpd` mit einer sehr einfachen, aber wirkungsvollen Konfiguration wie z.B. in Abbildung 4.4. Ankommende Mail wird in das ein-kompilierte Verzeichnis geschrieben, ohne weitergehende Analyse dessen, was in den Daten enthalten ist. Natürlich dürfen die Header keine Steuerzeichen enthalten, nicht zu lang sein und müssen syntaktisch einwandfrei sein. Lediglich 'From-' und 'To-Header' müssen mit der Konfiguration übereinstimmen. Falls das nicht der Fall ist, beendet der Prozess mit einer Fehlermeldung die TCP-Session und schreibt eine Meldung ins LOG. Ansonsten ist der Prozess mit dem Schreib-Vorgang beendet. Dann kann ein lokaler Virenchecker bestens die Email untersuchen und weiterleiten, alternativ kann sie direkt mit `smtpfwdd` an den internen Mailserver oder einen separaten Virenchecker weitergeleitet werden. Abgehende Mail wird vom internen Mailserver direkt per NAT versandt. Gegen DENIAL OF SERVICE in der Form, daß die Festplatte vollgemüllt wird, hilft das alles nicht. Aber wer Email haben will, muß dieses Risiko eingehen. Nur mit größerem Aufwand ließe sich das vermeiden.

4.3 Neuen Standort einbinden

Immer wieder mußten neue Standorte in die Verwaltungsmechanismen von SSPE eingebunden werden. Hier hat sich am zentralen Standort bestens bewährt, an der Firewall ein weiteres Interface mit einem weiteren Router zu verbinden, der seinerseits gegenüber der neu aufgebauten Firewall den Router des dortigen Providers mit dem dortigen Transfer-Netz und den dort zugewiesenen offiziellen Adressen simuliert. Dann reicht es, auf der bereits im

[8] http://www.obtuse.com

Betrieb befindlichen Firewall eine Route für die offiziellen neuen Adressen zu legen, und schon kann vom Administrator-PC zunächst die Firewall des neuen Standortes und dadurch die anderen Geräte anschließend erreicht werden. IPSec gegenüber dem zentralen Gateway funktioniert dann auch schon so wie später via Internet, die Vermaschung klappt natürlich nicht. Als Test reicht dies aber vollständig aus; wenn alle Funktionen erfolgreich getestet wurden, steht dem Transport nichts mehr im Wege. Manchmal kommt es vor, dass die Erfordernisse schneller wachsen, als ein Provider Internet an bestimmten Stellen liefern kann ...

4.4 ISDN Wählverbindungen und andere Besonderheiten

An fast jedem Ort ist es kurzfristig möglich, eine normale ISDN-Leitung zu erhalten. Mit einem kleinen Router kann darüber Datenverkehr laufen. Eine spätere Umstellung auf einen normalen Internet-Anschluß gestaltet sich dann einfacher, wenn dieses von Anfang an eingeplant wird. Ergo wird dann sinnvollerweise sogleich eine Firewall und ein Gateway als Minimal-Ausstattung in Betrieb genommen. IP-Filterung funktioniert wie am Internet, nur IPSec funktioniert auf einer solchen, privaten Leitung nicht, da keine offiziell gerouteten IP-Adressen genutzt werden. Um trotzdem von allen anderen Standorten ins interne Netz des neuen Standortes zu gelangen, ist ein kleiner Trick anzuwenden: von allen voll vermaschten Orten wird das Netz des neuen Standortes zu dem Standort geroutet, an dem die ISDN-Strecke terminiert. Der neue Ort routet alle internen Netze via IPSec zum zentralen. Und wenn dann alles richtig konfiguriert ist, funktioniert das sogar.
Da die IPSec-Konfiguration per Script erzeugt wird, ist der bereits beschriebene Kunstgriff mit `/etc/ipsec.conf.const` und dem modifizierten `/root/bin/ipsec-supervisor` entstanden. Auch für andere Dinge, z.B. den Traffic einer anderen Firma übergangsweise mit zu transportieren, ist er bestens geeignet. Da FreeS/WAN die 'Preshared Keys' anhand der IP-Adressen der abgehenden Interfaces festmacht, sind für zusätzliche Tunnel keine weiteren erforderlich. Eine weitere, winzige Änderung war für jene Firma ebenfalls erforderlich: Bestehende Cisco^{TM}IP-IP-Tunnel (IP-Protokoll 4) sollten weiterhin funktionieren. Also mußte `bin/rules.pl` um dieses IP-Protokoll erweitert werden. Es funktioniert wie ESP oder AH unidirektional, so daß einfach der ESP-Teil `do_esp_protocol` zusätzlich genutzt werden konnte.

Kapitel 5

VPN für den Außendienst

Die Programme der Zertifizierungsstelle sind nicht Bestandteil von SSPE. Zum Verständnis der benutzten Mechanismen erscheint eine kurze Beschreibung jedoch notwendig.

5.1 X.509-Zertifikate

Seit Netscape die SSL-Spezifikationen veröffentlichte und Eric Young sein freies SSLeay-Paket veröffentlichte, steht dem nachvollziehbaren Gebrauch von X.509-Zertifikaten nichts im Wege. Erste Ansätze gab es schon lange in Form der 'Privacy Enhanced Mail' Spezifikationen in den RFC 1420-1424 und anderen. Ein Durchbruch geschah aber erst durch Netscapes SSL-Anwendung auf das Hypertext Transport Protocol http. Das verwandte 'https' sieht verschiedene Anwendungen vor:

1. Nur Serverseitiges Zertifikat
2. Client kann und Server muß ein Zertifikat nutzen
3. Client- und Server muß je ein Zertifikat nutzen

Ein Zertifikat besteht aus mehreren Komponenten. Der DN[1] besteht aus dem Namen der Organisation, dem Land, der Provinz und dem Namen des Eigners, dies ist im Falle eines Serverzertifikates der FQDN[2] des Servers. Der Name des Ausstellers (Issuer Name), eine eindeutige Seriennummer, Beginn und Ende des Gültigkeitszeitraumes, ein Public-Key[3] sowie der zugehörige

[1] DestinguishedName

[2] fully qualified domain name

[3] der passende Private-Key ist nicht Bestandteil des Zertifikates, muß aber im Browser bzw. im Web-Server vorhanden sein

Algorithmus[4] und Checksummenalgorithmus[5] gehören ebenfalls dazu. Der Aussteller hat das Zertifikat mit seinem Private-Key signiert, daher ist dies jederzeit mit dem öffentlichen Zertifikat[6] der Zertifizierungsstelle verifizierbar. Das allererste Zertifikat ist selbstsigniert und beinhaltet als DN den Namen der Zertifizierungsstelle. Mit RSA-Schlüsseln ausgestatte Zertifikate sollten nur einige Jahre gültig bleiben, 2048 Bit Schlüssellänge dürften zur Zeit ausreichen.
Einmal ausgestellte Zertifikate sind bis zum Ende ihrer Gültigkeitsdauer verwendbar. Dies wird im Falle eines Verlustes problematisch. Daher gibt die zertifizierende Instanz in regelmäßigen Abständen eine CRL[7] heraus, die alle von dieser Zertifizierungsstelle jemals ausgestellten Zertifikate anhand ihrer Seriennummer eindeutig identifiziert, welche als nicht länger gültig bekanntgemacht wurden. Wenn nun einem Web-Server die aktuelle CRL vorliegt, kann er ankommnde Client-Zertifikate anhand der internen Zeitstempel und anhand der Seriennummern in der CRL eindeutig als gültig oder ungültig erkennen. Natürlich setzt das die Disziplin der Benutzer voraus, verlorengegangene Zertifikate der Zertifizierungsstelle zu melden. Daher muß eine solche Vorgehensweise auch mit organisatorischen Maßnahmen begleitet werden, wenn Erfolg in Form von Sicherheit das Ziel sein soll.
Die RSA-Schlüssel werden ausschließlich dazu genutzt, Sitzungs-Schlüssel zu Begin der TCP-Session auszuhandeln. Dieser wird dann als Schlüssel eines schnellen Strom-Chiffrierers wie zB. RC4[8], AES[9] oder 3DES[10] genutzt, Inhalte zu schützen. Ein Seiteneffekt ist nützlich: Der Client prüft anhand des Serverzertifikates, ob dessen FQDN mit dessen Namen im angefragten URL übereinstimmt. Wenn nicht, wird der Benutzer auf diesen Mißstand hingewiesen. Falls der Server so eingestellt ist, dass der Client sich mittels Zertifikat authentisieren muß, können aus dem DN des Clients weitere Rechte abgeleitet werden. Unbeliebte Nutzer können so sehr einfach auf der Serverseite ausgeschlossen werden. Nutzer ohne Client-Zertifikat haben keine Chance, an Inhalte des Servers zu kommen.
X.509-Zertifikate lassen sich im Zusammenhang mit IPSec zur Authentisierung nutzen. Dazu später mehr.

[4] zumeist RSA, DSA und andere möglich
[5] zumeist MD5 oder SHA1, andere möglich
[6] dies enthält den dazugehörigen Public-Key
[7] Certificate Revokation List
[8] Rivest's Code 4, 128 Bit Schlüssel
[9] Advanced Encryption Standard, 256 Bit Schlüssel
[10] Triple Data Encryption Standard, 112 Bit oder 168 Bit Schlüssel

5.2 Zertifizierungsstelle – was ist das

Die zuständige Ordnungs- oder Polizeibehörde stellt Personalausweise und Reisepässe nur für solche Bürger aus, die danach fragen, sich ausweisen können und im örtlichen Register ohne Versagungsgründe bekannt sind. Ähnlich kann eine firmeninterne Instanz dazu dienen, aufgrund eines Mitarbeiterverzeichnisses auf berechtigtes Verlangen hin Zertifikate auszustellen. Die Mitarbeiter können diese dann zur Authentisierung gegenüber internen Servern oder Diensten benutzen. Die Nutzung eines Directory/LDAP-Servers[11] zur Speicherung der Zertifikate kann den Nutzen z.B. für S/MIME[12] deutlich steigern, ist aber hier nicht von Bedeutung. Eine Verteilung und Nutzung der Zertifikate auch im Außenverhältnis einer Firma kann in Deutschland rechtliche Probleme aufgrund der vorbildhaften und äußerst fortschrittlichen Signaturgesetzgebung SigG[13] mit sich bringen. Die im weiteren beschriebene Zertifizierungsstelle genügt den strengen Vorschriften dieses Gesetzes *an keiner Stelle*.

5.2.1 Zertifizierungsstelle – wozu?

Mit Zertifikaten können viele Dinge geregelt werden, die ohne fast unmöglich sind. Eine Anwendung für Zertifikate stellt die Authentisierung und Autorisierung des Benutzers gegenüber einem Webserver dar; aus dem präsentierten DN können fein differenziert Berechtigungen des Benutzers z.B. datenbankbasiert abgeleitet werden. Ein benutzerspezifisches Portal kann so einfach auf einem SSL-Apache aufbauen. Genauso kann für IPSec die gegenseitige Authentisierung der Tunnelenden durch Zertifikate gestaltet werden, ein spezieller X.509-Patch von Andreas Steffen bringt die Funktionalität in FreeS/WAN hinein. Eine Software-Instanz, die Zertifikate ausstellen und verwalten kann, ist mit OpenSSL[14] einfach herstellbar. In dem Maße, wie der Zugang zu sensiblen Unternehmensinformationen mit Zertifikaten gesichert wird, steigt natürlich der Wert der Zertifizierungsstelle, da die Unternehmenssicherheit nur von der Schwierigkeit abhängt, Zertifikate zu reproduzieren. Ist jemand im Besitz des zertifizierenden Schluessels, können seine (falschen) Zertifikate nicht von den Echten unterschieden werden. Daher sollte der Maschine bestmöglicher Schutz zuteil werden. Eine altbewährte Strategie ist mit ssh und Shell-Scripts einfach realisierbar.

[11] Leightweight Directory Access Protocol, Untermenge von X.500

[12] secured mime, verschlüsselte und signierte Email mit Zertifikaten

[13] Eine SigG-konforme Zertifizierungsstelle bedarf in Deutschland der Zulassung durch die Aufsichtsbehörde (Bundesamt für Sicherheit in der Informationstechnik BSI) und ist mit vielfältigen, teilweise kostenintensiven Auflagen verbunden

[14] http://www.openssl.org

5.2.2 Fort Knox zum Nulltarif

Die Benutzerzertifikate werden durch Signieren eines Requests mit dem privaten Schlüssel der Zertifizierungsstelle erstellt. Der private Schlüssel stellt gewissermaßen den gesamten Wert [15] des Unternehmens dar und muß daher durch geeignete Maßnahmen vor Mißbrauch geschützt werden.

- 3DES-Verschlüsselung zwingt zur Eingabe der 'Passphrase' vor jeder Nutzung. Die 'Passphrase' stellt den Schlüssel für 3DES des RSA-Private-Keys dar.
- Die Maschine sollte im Netzwerk nicht für alle Anwender erreichbar sein. Geeignete Firewallmechanismen, fest definierte Administratoren, so wenige wie zwingend nötige Services ...
- Basic-Auth oder besser Client-Zertifikate sollten zur Administration eingesetzt werden.
- Alle Aktionen loggen, die Zeitstempel sollten nachvollziehbar mit amtlicher Uhrzeit geschehen[16]
- Regelmäßige Generierung einer CRL sichert aktuelle Zugriffsrechte an allen durch Zertifikat geschützten Stellen. Die Verteilung sollte durch die Administratoren der Zertifizierungsstelle erfolgen.
- Backup der Maschine muß mit besonderer Sorgfalt vor unbefugten Einblicken geschützt werden.

Sinnvollerweise werden die Benutzer ihre Zertifikatsanforderungen nicht auf dieser Maschine ablegen, da jede unternehmensweit verfügbare Schnittstelle auch unternehmensweit zu mißbrauchen ist. Die Requests lassen sich einfach als Datei mit jedem beliebigen, bereits vorhandenem Webserver ablegen. Von der Zertifizierungsstelle können sie dann mit SSH und einfachen Shell-Scripts abgeholt werden. Dieses, auch als 'Fort Knox Methode'[17] bekannte Vorgehen, erfordert nur wenig Aufwand, um höchste Sicherheit zu erreichen. Die Verteilung der fertigen Zertifikate stellt keine hohe Anforderung, da ein Schieben des Zertifiaktes auf o.a. zweiten Webserver kein Risiko darstellt. Eine Benachrichtigung für den Benutzer (oder User-help-Desk) wie das Zertifikat zu

[15] Sein Mißbrauch führt im einfachsten Fall zu einem Zertifikat, dessen unautorisierte Benutzung nicht von der autorisierten Nutzung anderer Zertifikate zu unterscheiden ist, da die kryptographischen Mechanismen nicht verletzt werden.

[16] ntp.ptb.de

[17] Daten werden ausschließlich geholt oder geschoben, d.h. nie durch eigene offene Ports in die Maschine verbracht

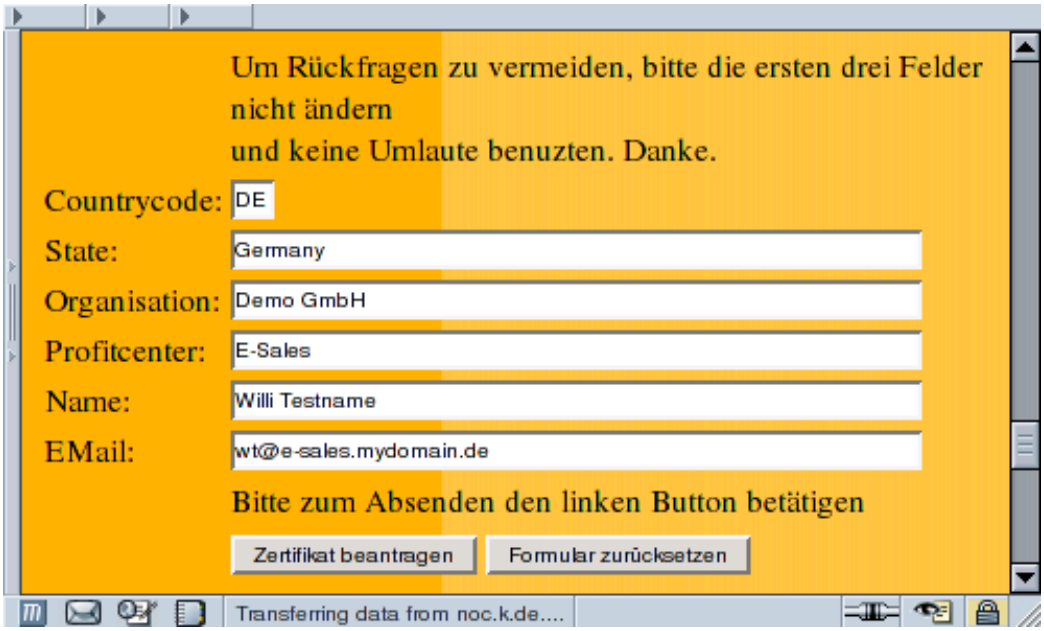

Abbildung 5.1: Benutzeransicht

erlangen ist und wie es zu installieren ist, kann problemlos per Mail erfolgen. Auch dies öffnet keine TCP-Session von außen, die evtl. zu mißbrauchen wäre.

5.2.3 Benutzeransicht

Der Benutzer sieht von alledem nur eine HTML-Form wie in Abbildung 5.1, in die er seinen Namen und seine EMail-Adresse eingeben darf, um ein Zertifikat zu beantragen. Mit seinem Klick auf die 'Submit-Taste' erhält der Administrator eine Mail, damit er weiß, etwas zu tun zu haben.

5.2.4 Administratoransicht

Der Administrator kann nach Eingabe seines Passwortes für den BasicAuth-Mechanismus des Web-Servers auf alle Seiten zur Zertifikatsverwaltung wie in Abbildung 5.2 gezeigt zugreifen. Wenn ein Antrag vorliegt, kann er mit einem Klick auf 'Ausstellen' wie in Abbildung 5.3 sichtbar gemacht werden. Darin muß nur die 'Passphrase' eingegeben werden. Ob an die im Antrag genannte Email-Adresse eine Nachricht gesandt werden soll, entscheidet der Administrator durch das kleine Kästchen. Er erhält in jedem Falle eine Email, wie das Zertifikat abzuholen ist. Der URL enthält eine MD5-Checksumme als

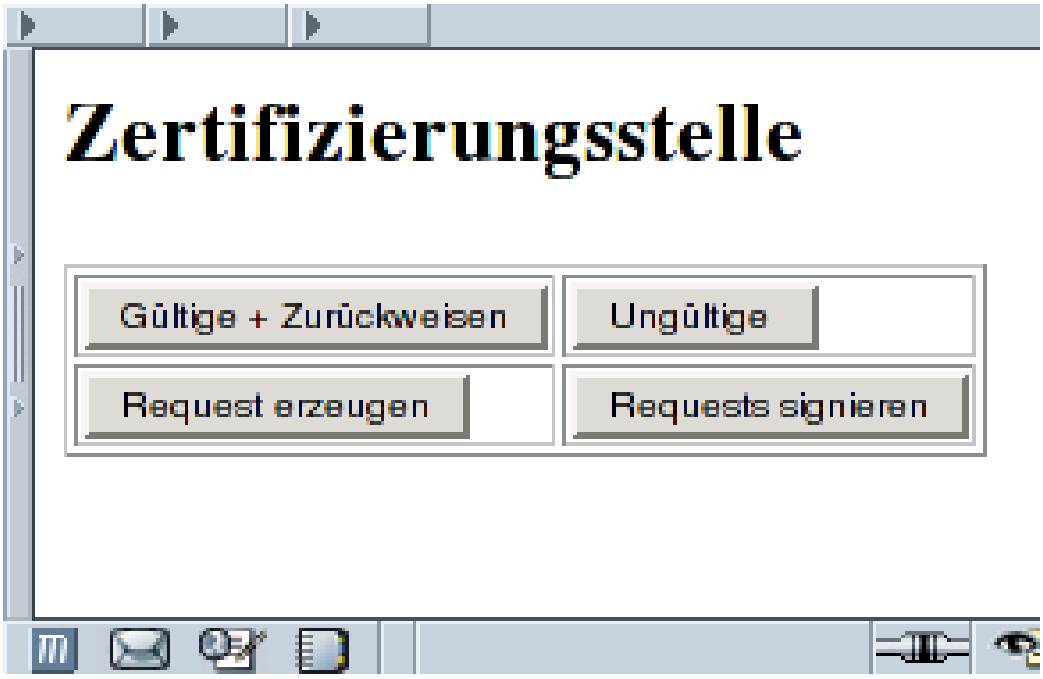

Abbildung 5.2: Administratoransicht

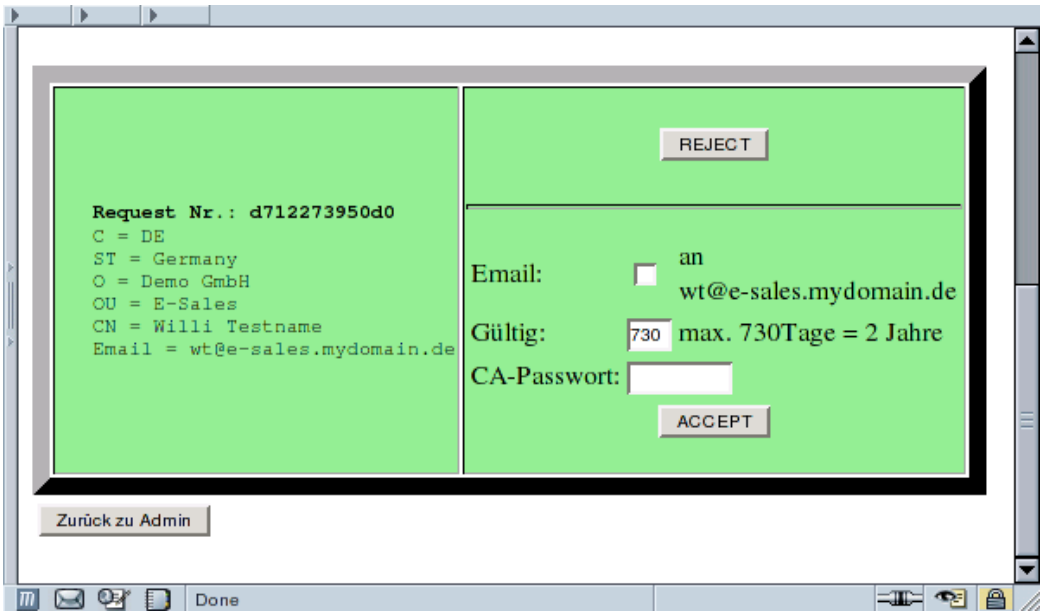

Abbildung 5.3: Zertifikatsantrag

```
[global]

[lns default]
ip range = 172.16.43.129-172.16.43.190
local ip = 172.16.47.253
require chap = yes
refuse pap = yes
require authentication = yes
name = LinuxVPNserver
ppp debug = yes
pppoptfile = /etc/ppp/options.l2tp
length bit = yes
```

Abbildung 5.4: L2TPD-Konfiguration **/etc/l2tp/l2tpd.conf**

Parameter eines CGI, ein Erraten ist so relativ unwahrscheinlich. Ist das CGI mit dem korrekten Parameter aufgerufen, wird das Zertifikat im PKCS#12-Format[18] zum Download angeboten. Danach wird es im Verzeichnisbaum des Webservers verschoben, so daß es nur genau einmal geholt werden kann ...

Das Paktet ist ca. 4,5 kBytes groß und enthält das Benutzerzertifikat, den dazugehörigen privaten Schlüssel und das Zertifikat der Zertifizierungsstelle. Damit kann der Client-PC das Zertifikat des VPN-Gateways, welches von der gleichen Zertifizierungsstelle stammt, überprüfen. Dabei zeigt sich dann auch, welche Besonderheiten die Firma aus Redmond in ihr Verständnis von Zertifikaten hineinlegt: Sie dürfen keine Sonderzeichen enthalten. Also auch kein '@'. Auch nicht in der Email-Adresse. Dies gilt aber merkwürdigerweise nur für das Zertifikat, welches überprüft wird, nicht jedoch für das eigene, welches dem Gateway präsentiert wird. Der Einfachheit halber lässt man die Email-Adresse im Zertifikat des Gateways also besser weg, insbesondere aufgrund der mangelnden Entstörmöglichkeiten.

5.3 l2tpd- und ppp-Konfiguration

Die Konfiguration des 'layer two protocol daemons'[19] ist wie in Abbildung 5.4 recht einfach. Die Angaben zum Adress-Pool sollten mit der Konfiguration des ppp-daemons wie in Abbildung 5.5 übereinstimmen. Zweckmässigerweise nimmt ein Subnetz, welches intern zu der Maschine geroutet wird, auf welcher der l2tpd läuft.

[18] Zu PKCS# siehe Dokumentation von RSA unter http://www.rsa.com

[19] l2tp ist nichts anderes als ppp über IP

```
ms-dns  172.16.44.253
ms-wins  172.16.44.81
auth
nodefaultroute
debug
lock
mtu 1400
mru 1400
```

Abbildung 5.5: **/etc/ppp/options.l2tp**

5.4 Windows 2000™kann IPSec

Inwieweit die IPSec-Implementierung durch Microsoft® kryptographischen oder sicherheitstechnischen Anforderungen genügt, mögen Experten oder solche, die sich dafür halten, entscheiden. Berechtigte Zweifel müssen aber im Verhältnis zu alternativem Aufwand eine kommerzielle, zusätzliche Lösung rechtfertigen. Fakt ist jedenfalls, dass Windows 2000 und Windows XP mit Bordmitteln IPSec unterstützen. Allerdings behauptet möglicherweise niemand, es sei benutzerfreundlich eingerichtet. Ohne eine entsprechende automatisierte Unterstützung ist es ein reines Abenteuer, muß man doch mit jeder Einwahl ins Internet die komplette Konfiguration neu einrichten, da man üblicherweise jedesmal eine andere IP-Adresse von seinen Provider erhält. Ein erster, wesentlicher Verbesserungsansatz kommt von Marcus Müller mit ipsec.exe. An die FreeS/WAN Konfiguration angelehnt, ist es leider nicht flexibel genug, alle Anforderugnen zu erfüllen. Daher wurde für den 'Roadwarrior' ein neues Einwahlprogramm nötig.

5.4.1 vpndialer

Dank an Thomas Kriener. Er hats geschrieben und es ist unter GPL erhältlich bei http://vpndialer.sourceforge.net. Sowohl vor Aufbau des IPSec-Tunnels als auch anschließend kann vpndialer eine 'RAS'-Verbindung in Gang setzen. Damit kann ein beliebiger ISP angewählt werden, zum Firmen-Gateway eine IPSec-Session installiert werden und dann durch diese eine L2TP-Verbindung geöffnet werden. L2TP ist nichts anderes als ppp-over-IP, ein geeigneter Daemon-Prozess findet sich leicht für Linux in Form von `l2tpd`. Abgewickelt wird das Protokoll über udp-Port 1701. Sinn und Zweck einer solchen zweifachen Tunnelung des Datenverkehrs ist folgender: Der IPSec-Tunnel ist zuständig für die Verschlüsselung, diese ist Garant für korrekte Authentisierung mittels X.509-Zertifikat. Er kann auf genau eine interne IP beschränkt werden, die IPSec-Konfiguration macht sogar die Beschränkung auf 1701/udp möglich.

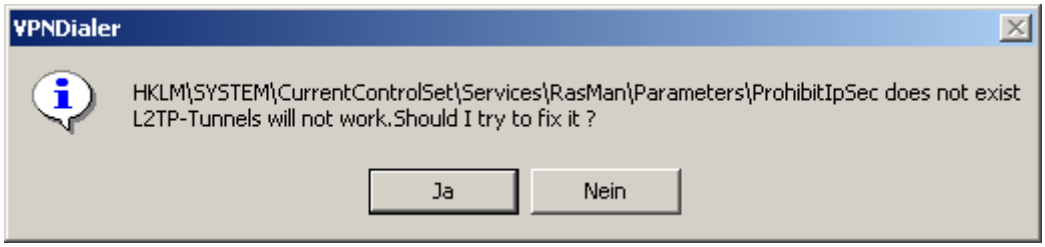

Abbildung 5.6: Der erste Start geht schief

Der L2TP-Tunnel dient vornehmlich der Konfiguration des Client-PC. Durch das PPP kann der Nameserver und anderes, Windows-spezifisches, vorgegeben werden. Eine Verschlüsselung findet hier nicht statt. Eine zusätzliche Authentisierung des Nutzers gegenüber dem ppp-daemon ist notwendig und kann zur Zuweisung einer individuellen IP genutzt werden.
Der erste Start des VPNDialer geht üblicherweise wie in Abbildung 5.6 gezeigt schief, das Programm korrigiert einen oder mehrere Registry-Einträge und meldet dies dem Benutzer. Ein Boot tut jedem Windows gut, danach ist der VPNDialer bereit, konfiguriert zu werden. Ein Klick in die linke obere Ecke des Hauptfensters und schon kann eine neue Konfiguration durch einen Klick ins weisse Bildchen wie in Abbildung 5.7 gezeigt starten. Sinnvollerweise gibt man der neuen Konfiguration zuerst einen Namen. Abspeichern der Einstellungen erfolgt mit einem Klick auf das Disketten-Bildchen ohne sichtbare Reaktion des Programms in der Windows-Registry. Eine RAS-Verbindung ins Internet sollte bereits bestehen, kann aber auch nachträglich hinzukonfiguriert werden. Falls die L2TP-Verbindung vorher konfiguriert wurde, kann sie eingebunden werden. Die restlichen Parameter ergeben sich fast von selbst.

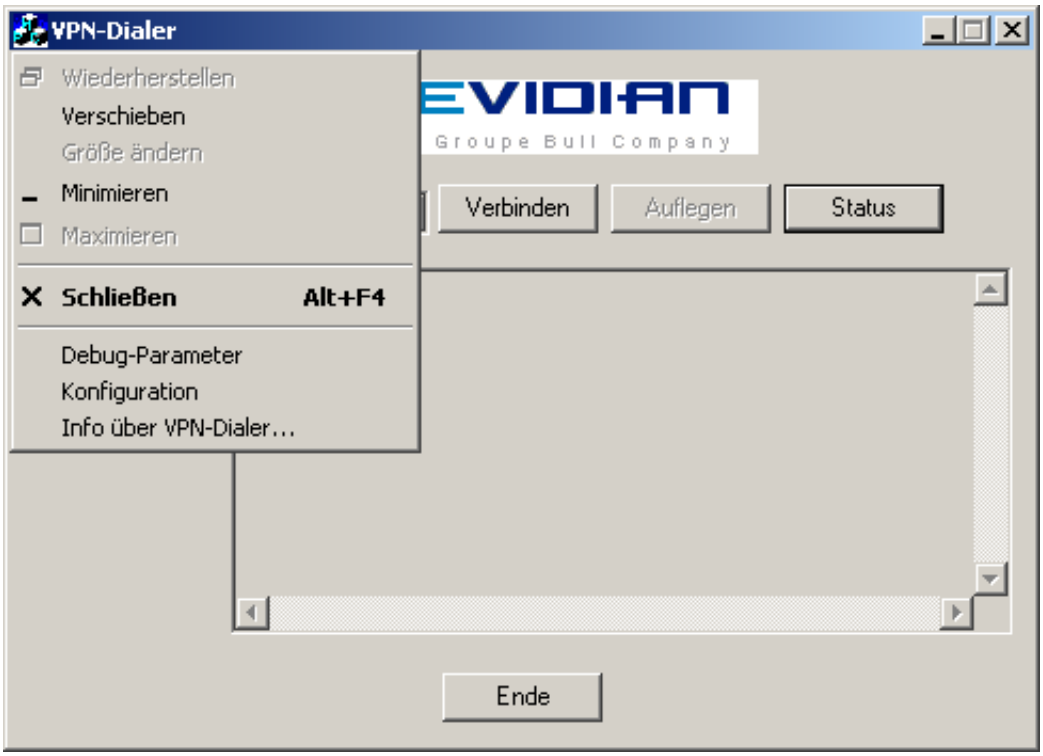

Abbildung 5.7: Boot macht gut!

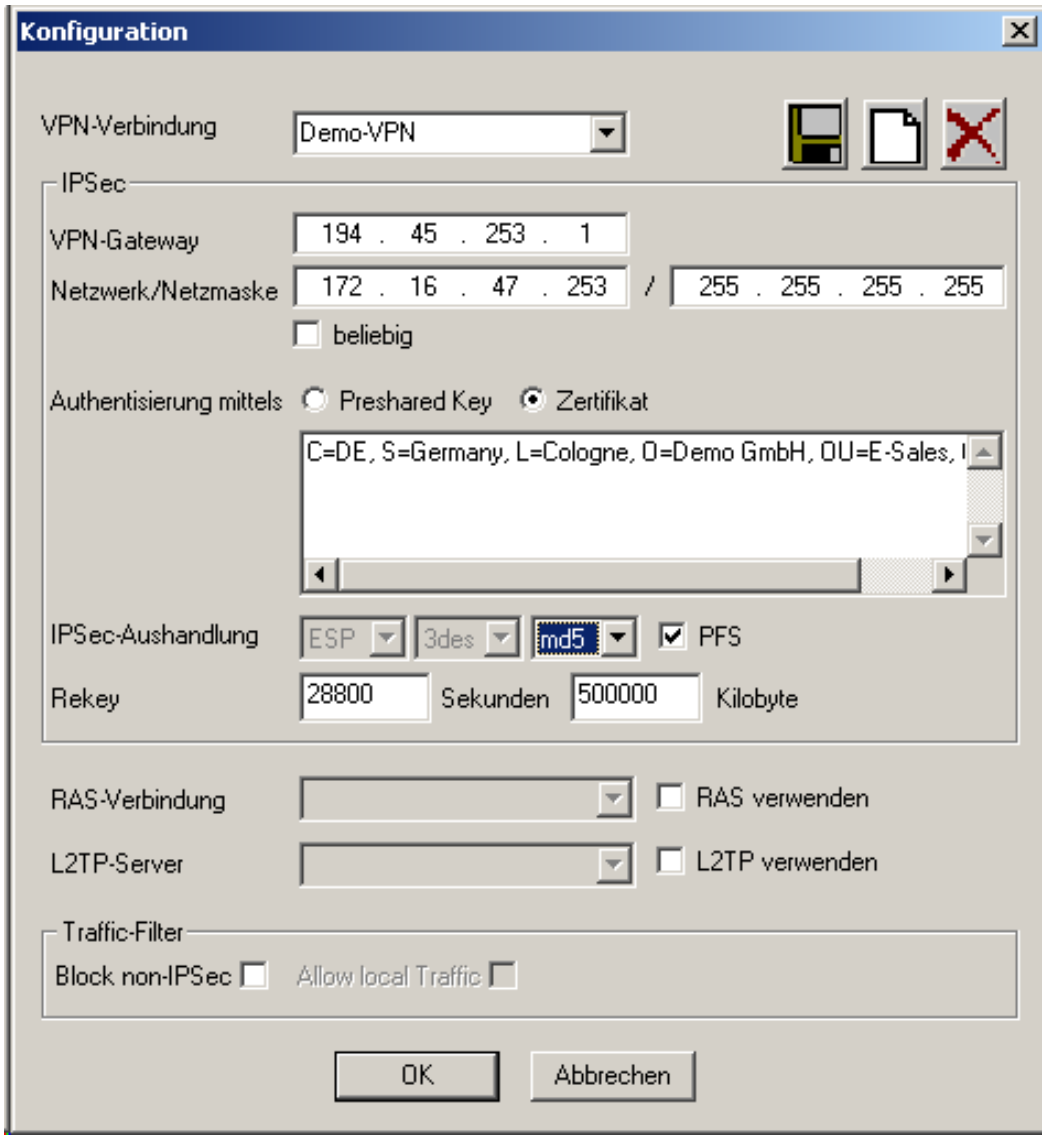

Abbildung 5.8: Hier ist einiges einzutragen

5.4.2 Management-Konsole einrichten

Um eine ausgestelltes Zertifikat zur IPSec-Authentisierung zu nutzen, muß es dem Windows-PC korrekt eingepflanzt werden. Der Weg geht über eine Management-Konsole mit eigens hierzu geschaffenem SnapIn.

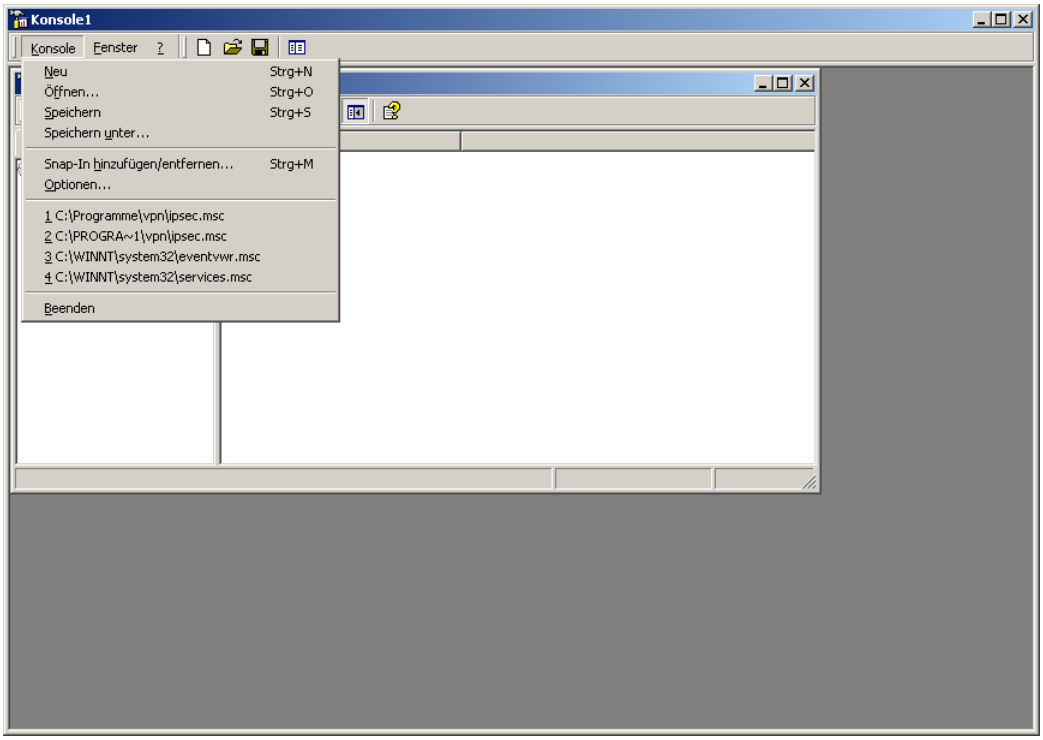

Abbildung 5.9: Start,Ausführen,mmc.exe,SnapIn Hinzufügen.

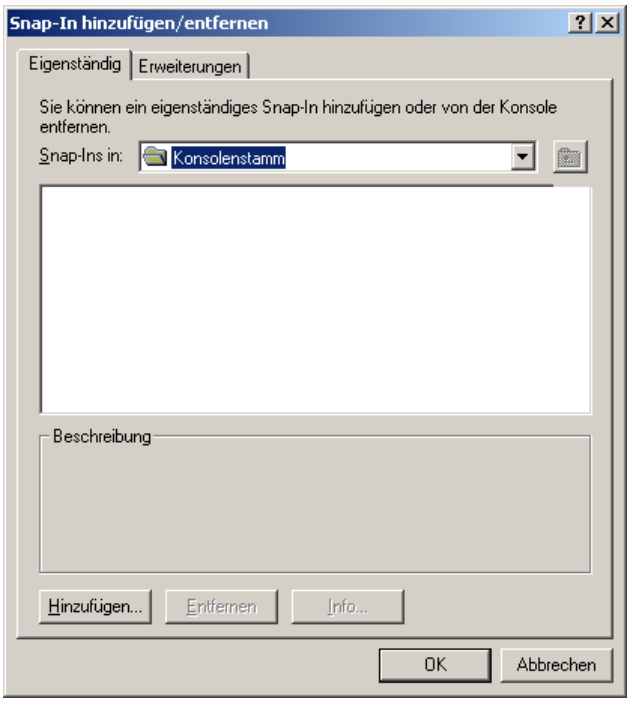

Abbildung 5.10: SnapIn Hinzufügen

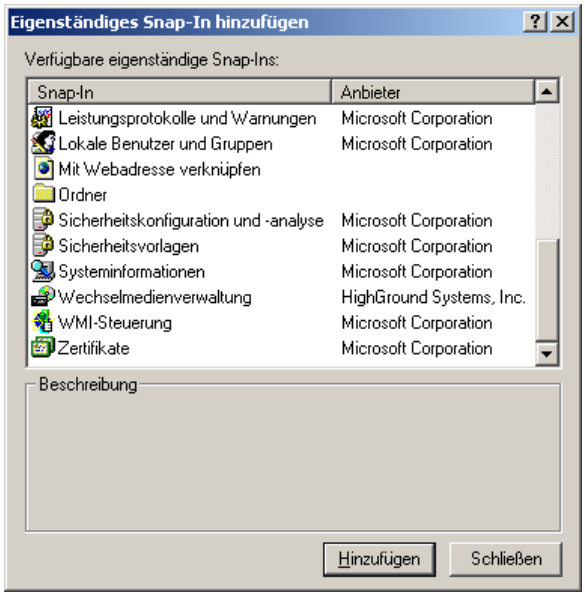

Abbildung 5.11: Zertifikate hinzufügen.

Sachdienliche Hinweise auf Dokumentation, aus welcher der Sinn unterschiedlicher Konten erkennbar wird, werden gerne entgegen genommen.

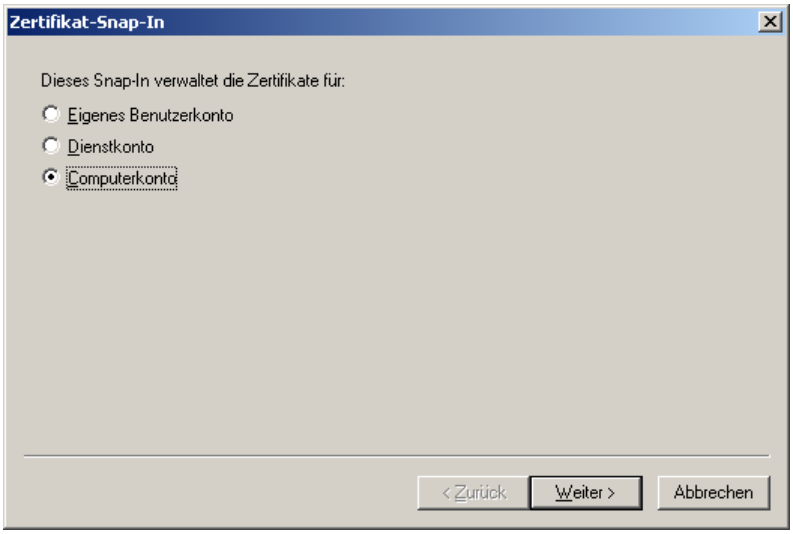

Abbildung 5.12: Computerkonto, Weiter

Nicht der lokale Computer, sondern lediglich die Management-Konsole wird hier fertiggestellt!

Abbildung 5.13: Lokalen Computer Fertig stellen

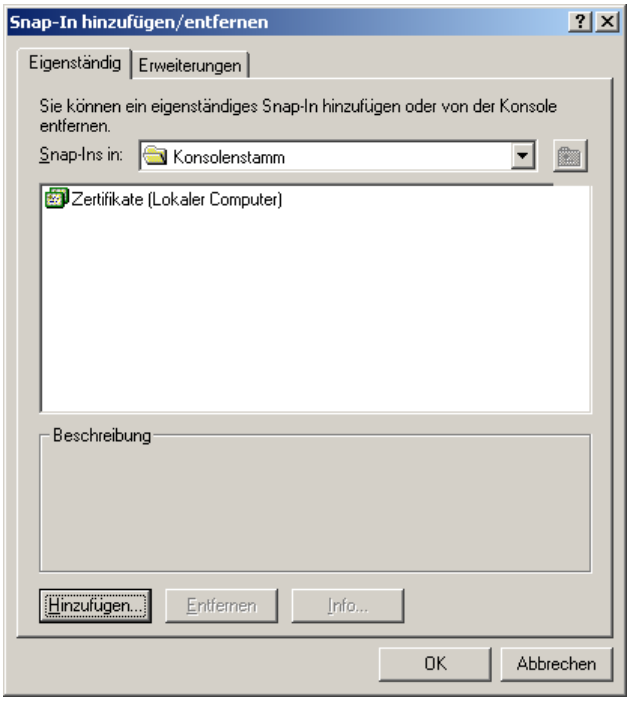

Abbildung 5.14: OK, und weiter gehts ...

Ab jetzt kann das SnapIn seiner zweckgemässen Verwendung zugeführt werden. Wer will, kann nun die Management-Konsole zur wiederholten Verwendung speichern.

5.4.3 Zertifikatsimport

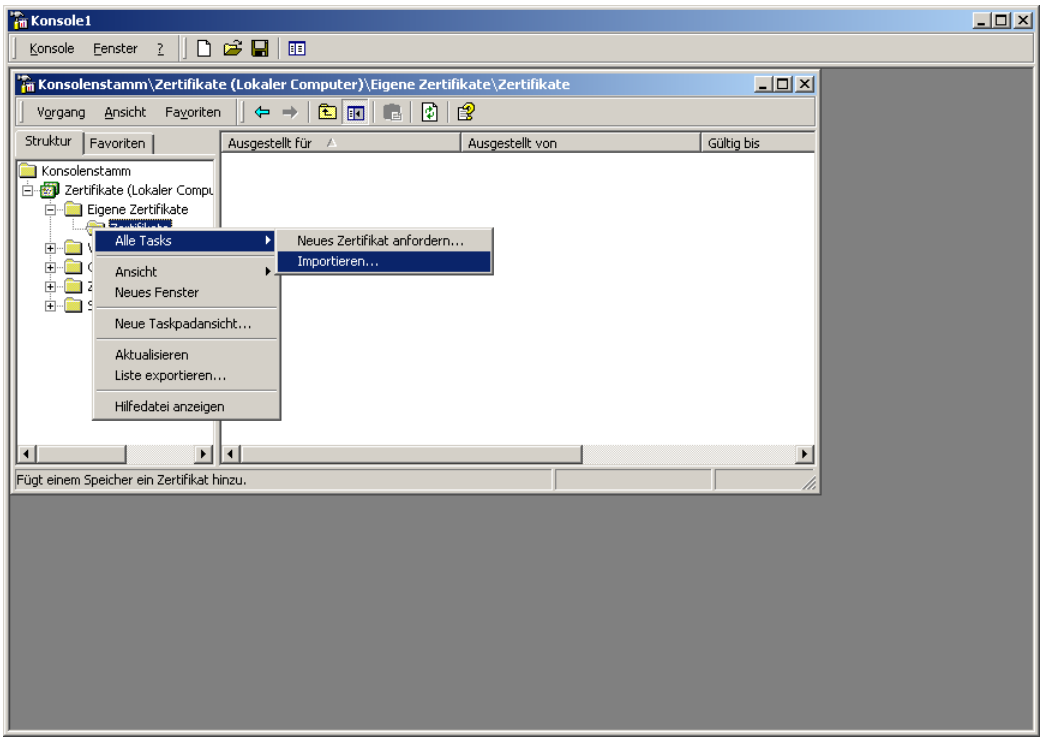

Abbildung 5.15: Eigene Zertifikate, Zertifikate, Alle Tasks, Importieren

Die Menüs und Untermenüs werden immer vielfältiger, damit der verwöhnte Anwender kein Wissen benötigt, wird wieder ein Assistent tätig ...

Abbildung 5.16: Der Assistent will weiter ...

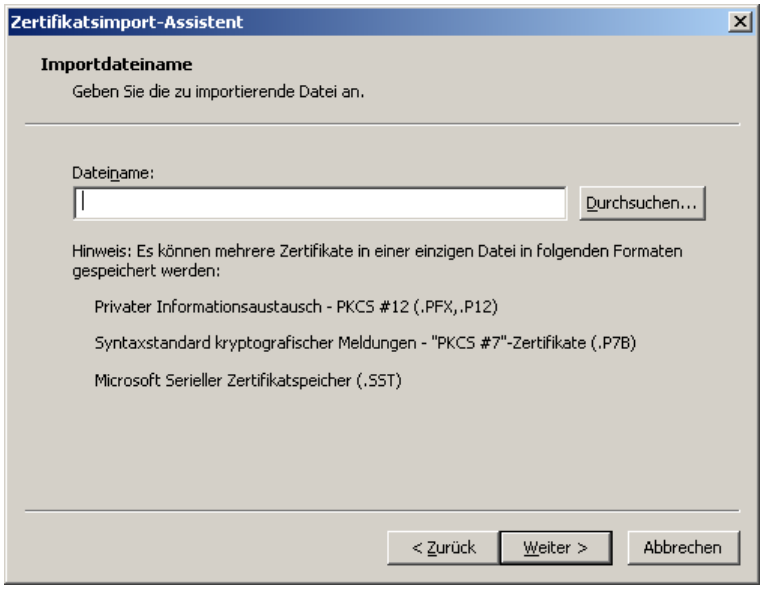

Abbildung 5.17: Wie war doch der Name? Durchsuchen hilft

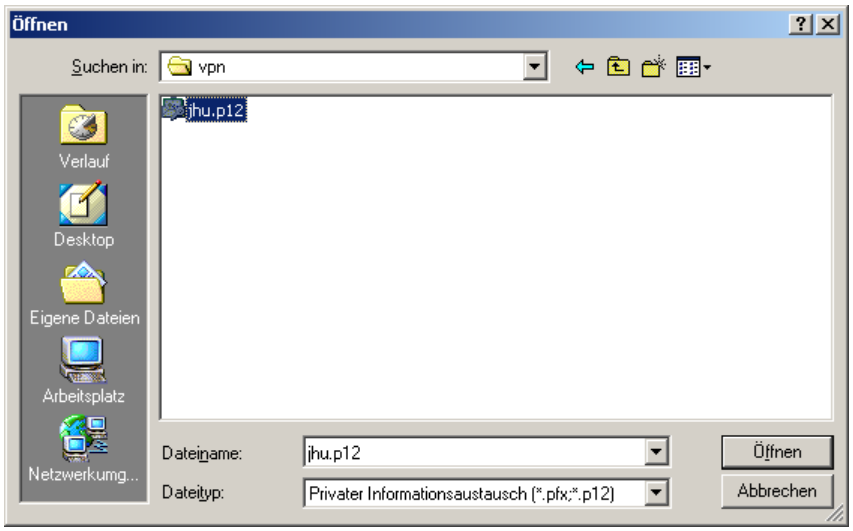

Abbildung 5.18: Da ist sie ja, öffnen!

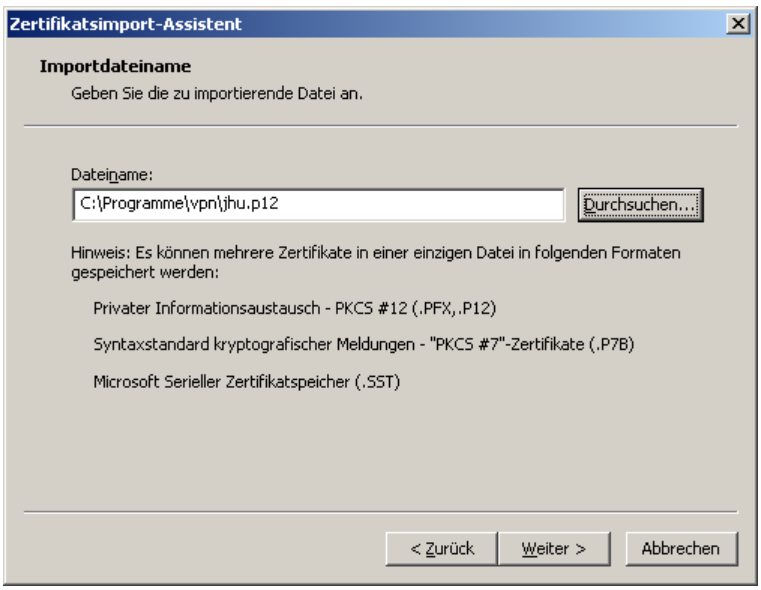

Abbildung 5.19: Weiter gehts

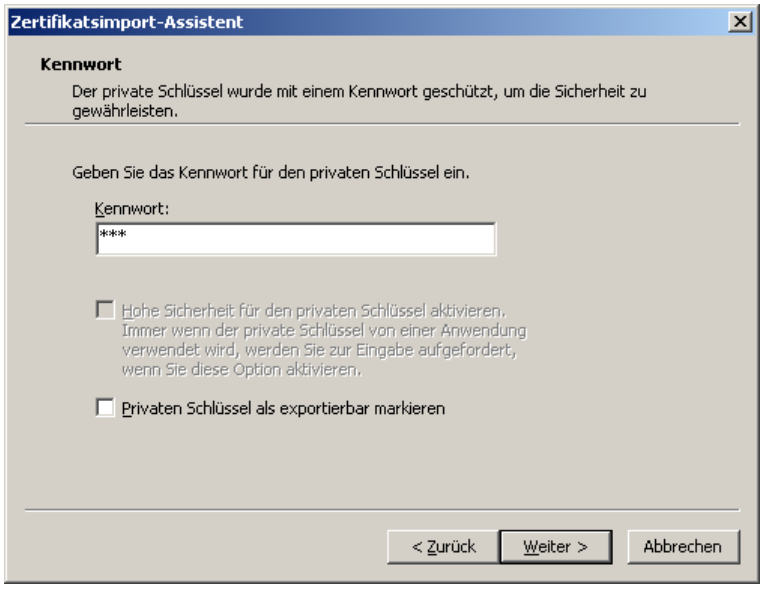

Abbildung 5.20: Wo ist nur diese v. . . Mail?

Das Passwort stand doch in der Mail? Oder muss ich anrufen?

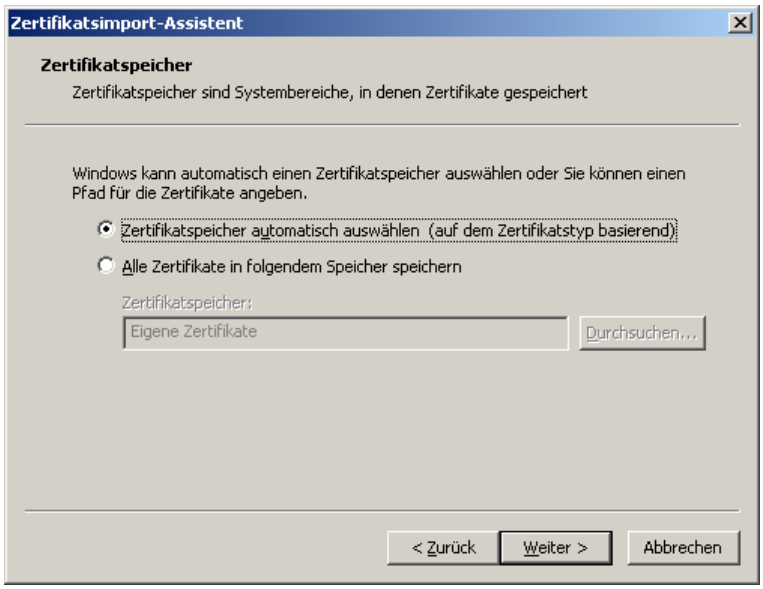

Abbildung 5.21: Natürlich automatisch, oder?

Ob es was nutzt, sehen wir späeter . . .

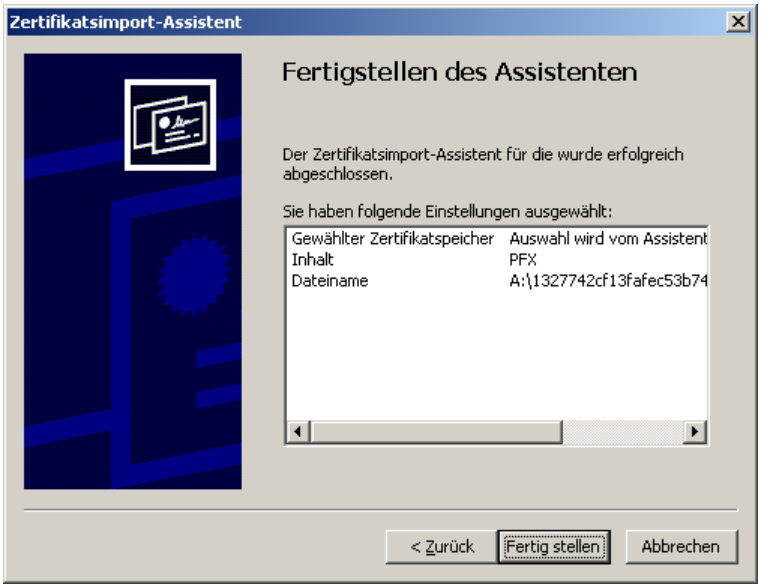

Abbildung 5.22: Fertigstellen

Abbildung 5.23: Fertig, nun ists drin .....

5.4.4 Windows L2TP-Konfiguration

Keine Hexerei ist bei der Konfiguration des L2TP-Tunnel im Spiel, lediglich einiges an Tipp- und Mausarbeit ist zu leisten. Start, Einstellungen, Netzwerk- und DFÜ-Verbindungen:

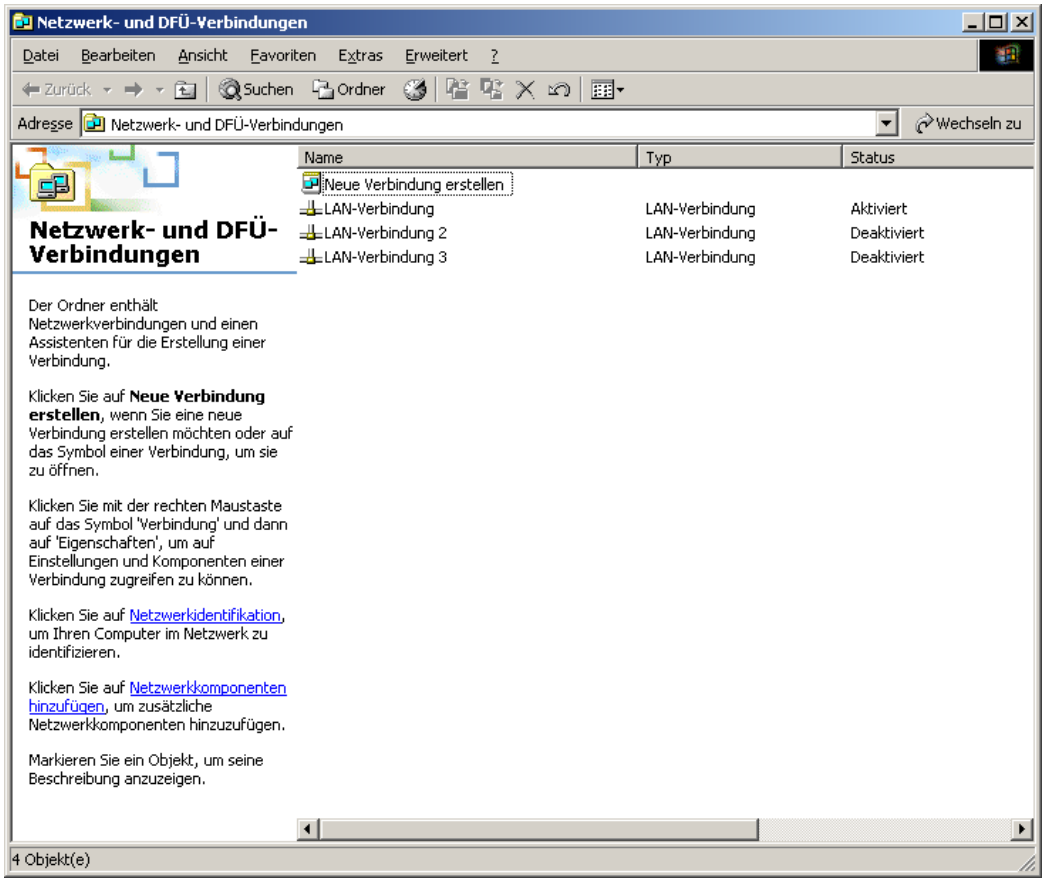

Abbildung 5.24: Netzwerkeinstellungen

Abbildung 5.25: Noch ein Assistent hilft

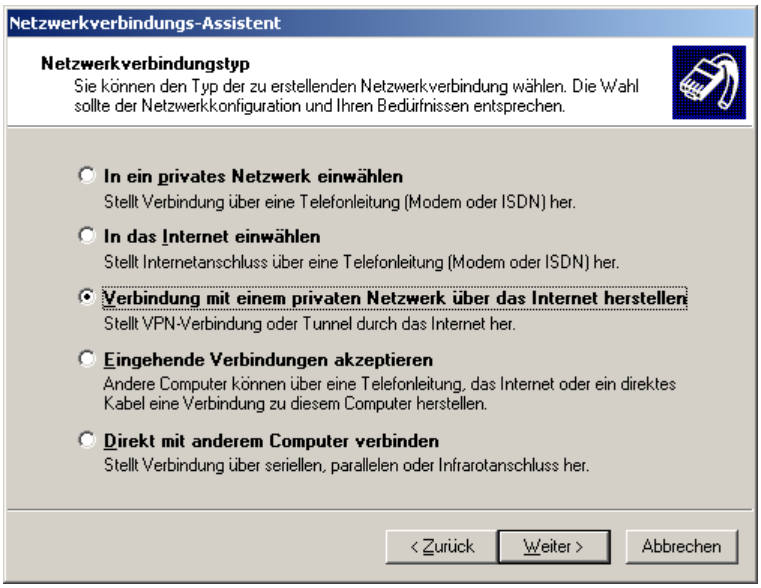

Abbildung 5.26: Die Variante entscheidet

Microsoft® kennt Tunnel durch das Internet!

VPNDialer wird diese Verbindungskonfiguration aufrufen, also ‚Keine...'

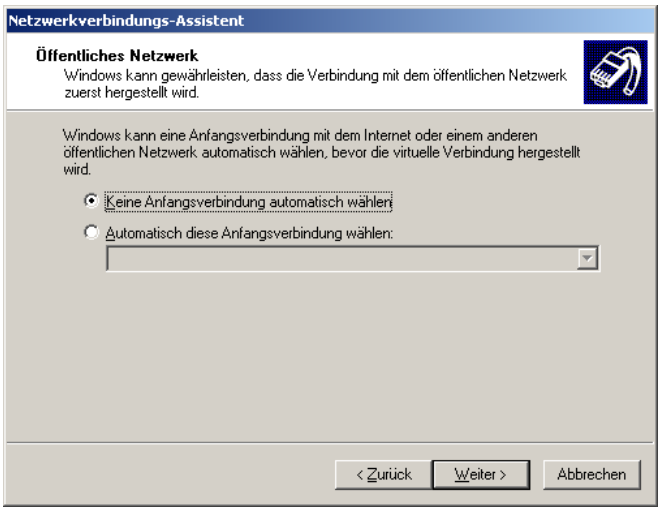

Abbildung 5.27: Erst die Henne oder erst das Ei?

Microsoft®ist hier sehr flexibel!

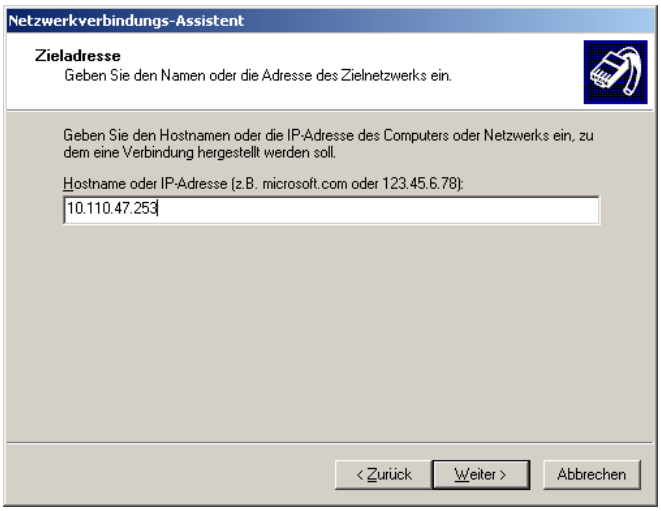

Abbildung 5.28: IP-Adresse des l2tpd-servers

Der L2TP-Server kann, muß aber keineswegs auf der Maschine laufen, die den IPSec-Tunnel terminiert.

Auf meinem Computer dürfen alle Benutzer meine Konfiguration benutzen, da nur einer real existiert:

Abbildung 5.29: Ein PC für mehrere Leute?

Abbildung 5.30: Name ist nur Schall und Rauch!

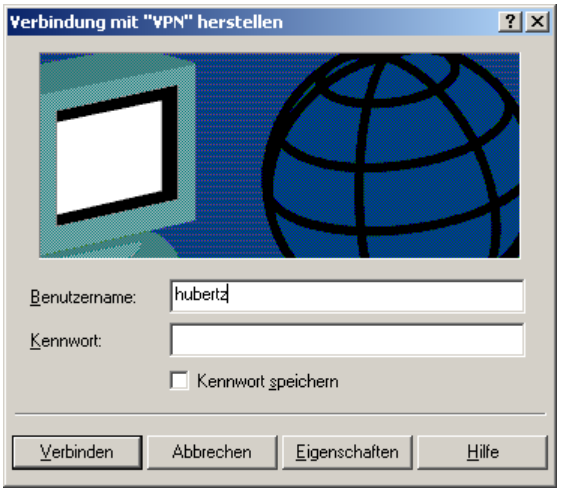

Abbildung 5.31: `/etc/ppp/chap-secrets` läßt grüßen

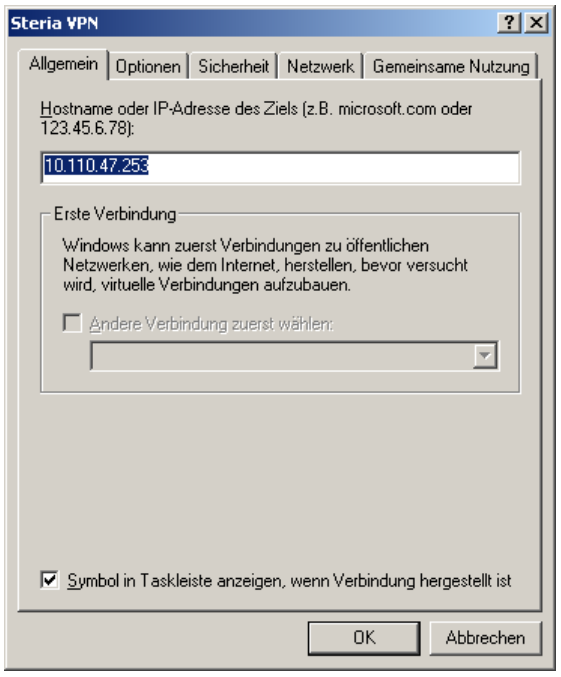

Abbildung 5.32: Nun kann der L2TP-Tunnel wirklich konfiguriert werden

Die nochmalige Eingabe der IP-Nummer hat uns der Assistent erspart, was er sonst alles noch gemacht hat, verschweigt er uns.

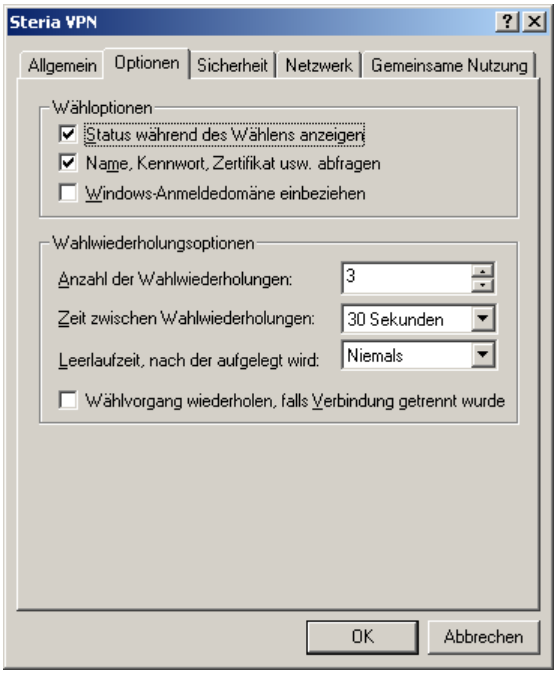

Abbildung 5.33: Optionen müssen auch sein

Den "Wählvorgang" automatisch zu wiederholen, macht wenig Sinn, da erfahrungsgemäß nur bei Zusammenbruch des IPSec-Tunnels[20] der L2TP-Tunnel nicht mehr funktioniert. Die Anzahl der Wahlwiederholungen und die Zeit dazwischen sind relativ belanglos.

[20]Der IPSec-Tunnel hängt von der RAS-Verbindung, der Laufzeit, und der übertragenen Datenmenge ab, da das 'Rekeying' nicht korrekt funktioniert

Sicherheit steht hier drauf, ist auch drin, wenn mans richtig macht:

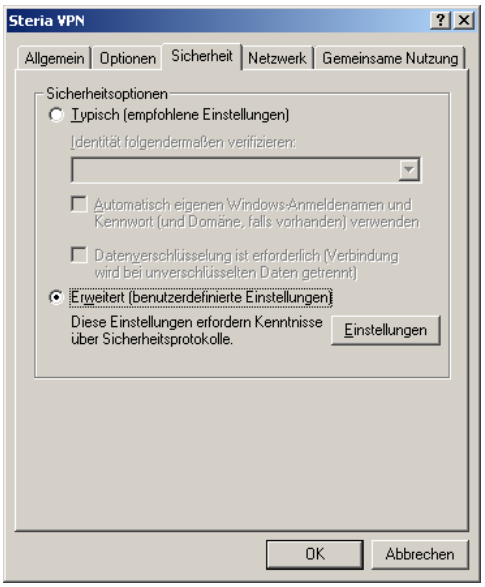

Abbildung 5.34: Hier müssen erweiterte Einstellungen ran!

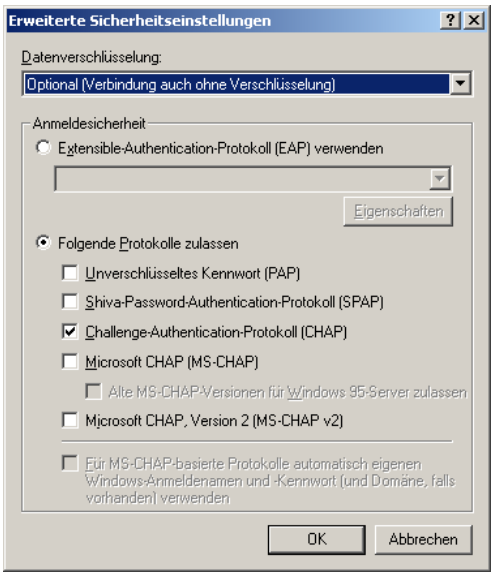

Abbildung 5.35: Einmal verschlüsseln reicht

Layer-2-Tunneling-Protokoll(L2TP) ist die Wahl, zu den Einstellungen kommen wir später auf Seite 64. Erst kommen wir zum Internet-Protokoll...

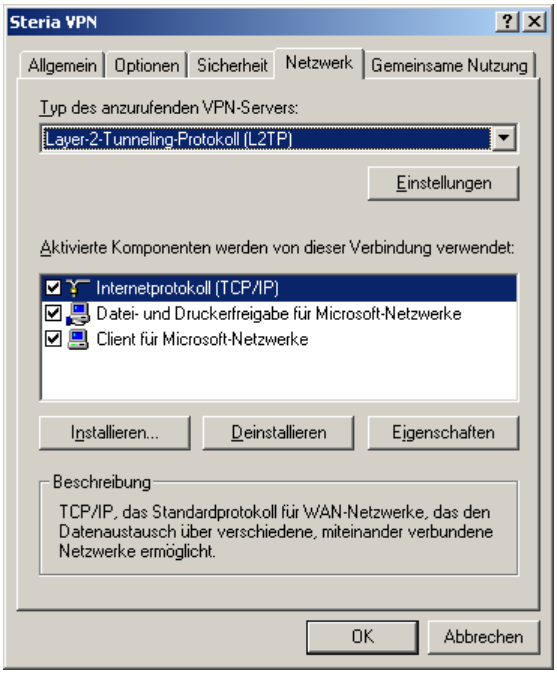

Abbildung 5.36: Internetprotokoll hat wichtige Eigenschaften!

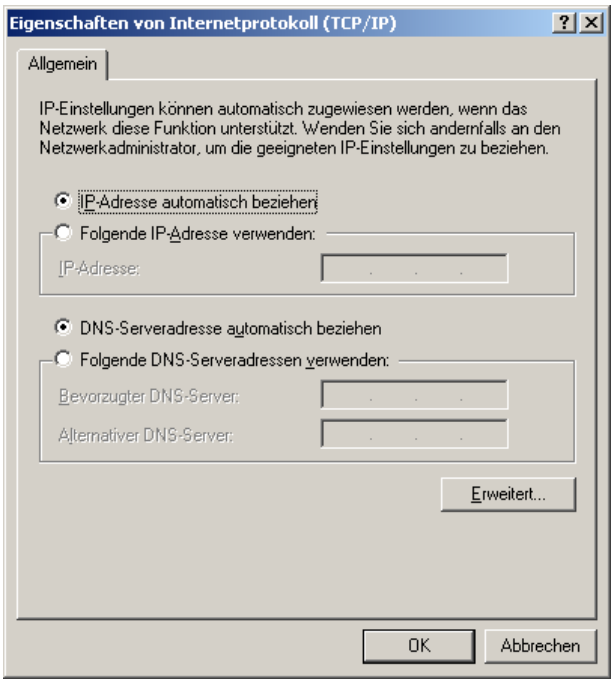

Abbildung 5.37: Hier geht was automatisch? Erweiterungen!

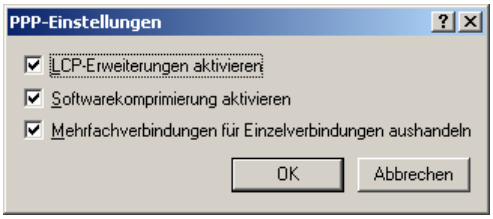

Abbildung 5.38: Erweiterte Eigenschaften: Alle Lampen an!

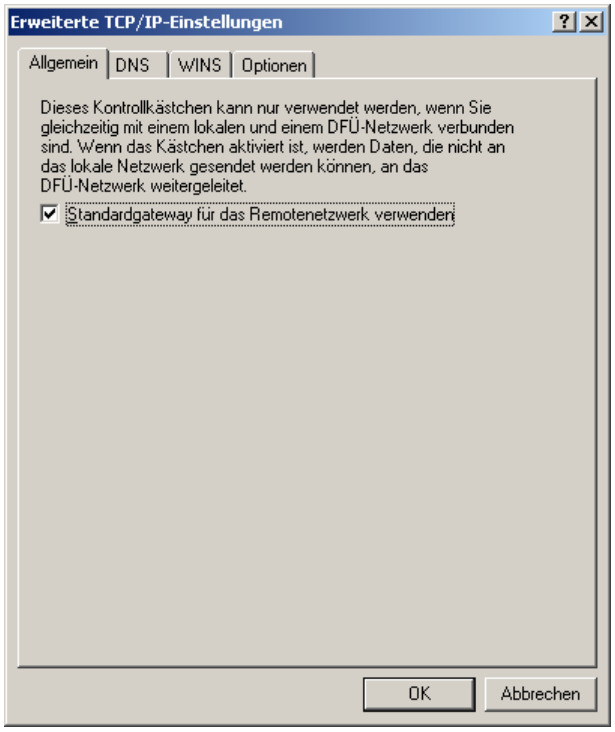

Abbildung 5.39: Dies Häkchen ist wichtig!

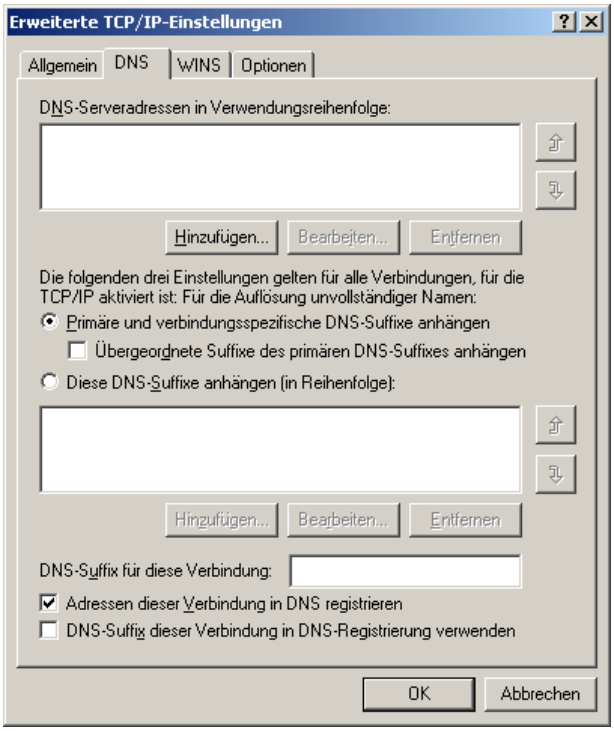

Abbildung 5.40: Die DNS-Konfiguration kommt per ppp!

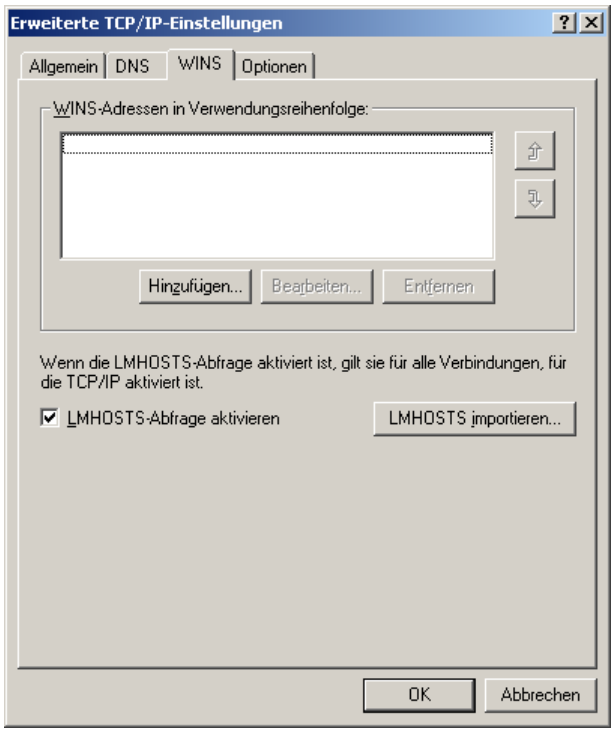

Abbildung 5.41: WINS muß wohl auch sein.

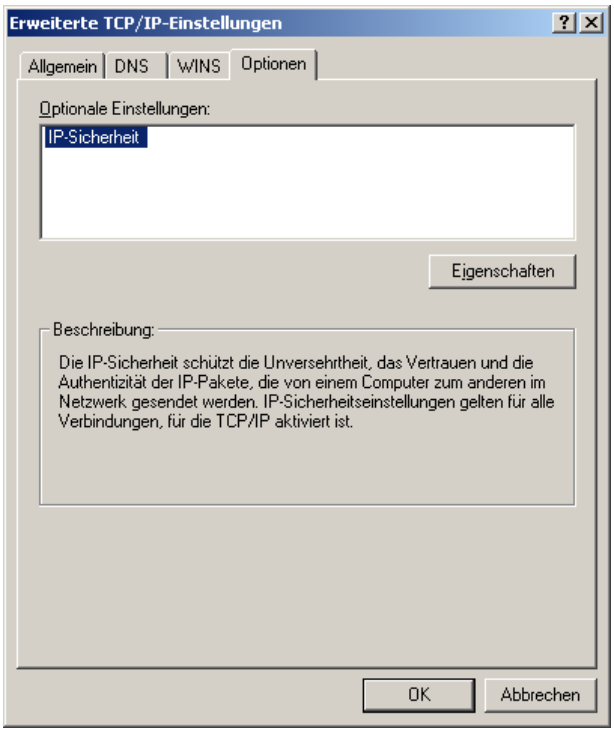

Abbildung 5.42: Das macht Hoffnung

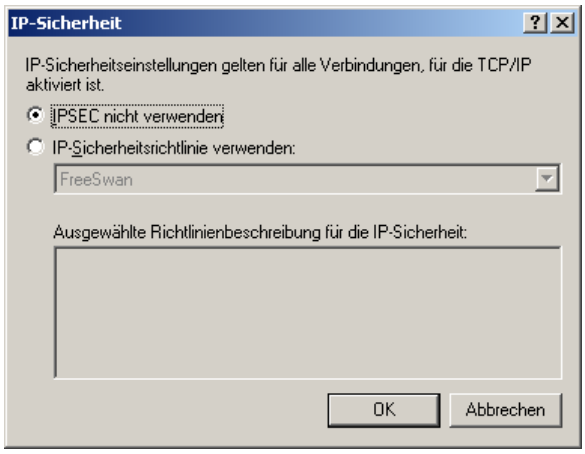

Abbildung 5.43: VPN-Dialer macht IPSec, hier nicht nochmal!

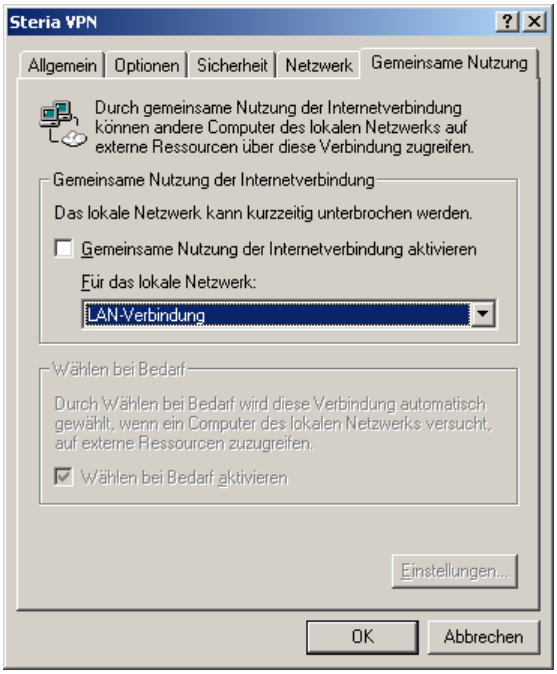

Abbildung 5.44: Firmen-PC am Heim-LAN?

Inwieweit sich Windows-PCs als Router und Verschlüsselungsmaschinen verwenden lassen, mag jeder selbst herausfinden. Aufgrund der nicht offengelegten Quellen ist jedoch davon abzuraten. `Debian GNU/Linux` zum Beispiel ist nicht nur preiswerter, sondern auch nachvollziehbar.

Kapitel 6

Zukunftswünsche

6.1 MAC-Adressen fest einstellen

An bestimmten Stellen im Netzwerk, insbesondere in nicht überwachten Außenstellen kann es sinnvoll sein, auch die am LAN angeschlossenen Geräte per MAC-Adresse fest zu konfigurieren. Dann kann niemand unbemerkt neue Geräte im LAN produktiv betreiben, ohne mit dem Firewall-Administrator Absprachen zu treffen. So ist z.B. denkbar, für unbenutzte IP-Adressen in der Firewall MAC-Adressen vorzugeben, etwa:

```
172.23.1.3    ether   52:54:05:F5:2C:2F   CM   eth1
172.23.1.4    ether   00:01:01:01:01:01   CM   eth1
```

Der erste ist ein realer Eintrag, der zweite führt dazu, daß Pakete an die IP ins 'Leere', nämlich an exakt diese gefälschte, nicht existente Mac-Adresse auf dem LAN geschickt werden. Da der ARP-Cache einen permanenten Eintrag für 172.23.1.4 hat, wird nicht mehr per ARP-Protokoll nach einem gesucht. Das verhindert natürlich nicht, daß ein Gerät mit dieser IP konfiguriert wird und im LAN Traffic macht. Lediglich die Firewall wird zu keiner Antwort imstande sein.

6.2 Kernelparameter

In `/proc/sys/net/ipv4` finden sich abhängig von der Kernelversion verschiedene Parameter, die sich zur Laufzeit einstellen lassen. Um diese nicht dem Zufall zu überlassen, sollten innerhalb des Startup-Scripts mit einem einfachen `/bin/cat VALUE > /proc/sys/net/ipv4/conf/eth0/rp_filter` alle vorhandenen Werte vorbesetzt werden. Sinnvollerweise werden die Vorgaben aus dem laufenden Kernel in eine Datei gelesen, diese wird wunschgemäß

editiert und dann die Werte bei jedem Boot mit dem Startup-Script in den Kernel gesetzt. Eine Einbindung dieser Datei in das Regelwerk unter Ausnutzung des sysctl-Befehls erscheint sinnvoll.

6.3 packet-mangling

`iptables` bietet die vielfältigen Möglichkeiten, die Kernel vor 2.4.0 schmerzlich vermisst wurden. z.B. TOS (Type of Service, Feld im IP-Header) konnte nicht manipuliert werden. Dies ist mit der Ziel-Tabelle MANGLE nun neben vielen anderen gezielten Veränderungen machbar. Eine Implementierung in Form erweiterter Regel-Optionen ist denkbar, ebenso aber auch eine "RAW-Section" z.B. zwischen `rules.head` und `rules.local`. Dies wurde mit Version 0.2.4 Realität.

6.4 Andere Plattformen

Cisco ™ Router sind mittels Access-Listen in der Lage, als einfache IP-Filter Traffic zu regulieren. Leider sind die aus SSPE erzeugten Listen sehr schnell zu groß, um im NVRAM der meisten Router Platz zu finden. Sicherlich ist es leicht möglich, BSD-Systeme ebenfalls in die Architektur einzubinden, ein Abwandeln des `rules.pl` sollte nicht allzu viel Aufwand sein. Auch andere *ix Systeme, die über geeignete Filter verfügen, sind denkbar, wenngleich nicht offengelegte Systeme ein nur schwer abschätzbares Sicherheitsrisiko darstellen.

Anhang A

Changelog

```
0.1.7    20030322    README written. At least for this release
                     its the only file beside the sources which
                     contains any useable documentaion of the
                     ideas behind the development.
                     Files ipsec, nathosts, rules.x written for
                     public release.
                     apply now has DEMO (.config) Mode, which
                     creates iptables-commands but no transfer and
                     remote execution. Neccessary for fakes of public
                     release.
                     cisco-user / password now in .config
                     no longer root account neccessary to run it. Only
                     ADMROOT env. neccessary in .profile or .bashrc
                     adm as shellalias useful.

0.1.6    20030305    hostnet, nathosts, privates now possible in
                     every machine-directory and etc
                     One must be present each!

0.1.5    20030228    privates no longer hardcoded in rules.pl
                     but as file etc/privates

0.1.4    20020508    Locking to prevent multiple simultaneous usage
                     added local ipsec.conf.const, appended to
                     ipsec.conf on the gateway by ipsec-supervisor

0.1.3    20020430    New Name:
                     Distributed Firewall Control Terminal
                     becomes
                     Simple Security Policy Editor
                     Idea to generate IOS Access-lists as well.
                     no cisco with enough memory stops realization.

0.1.2    20020322    first online try, machine de/activation
                     useful for broken internet connection and
                     simultaneous need of rules changing.
```

Anhang B

Verzeichnisse

Alle Bildschirm-Schnappschüsse wurden mit OpenSource Programmen wie 'The Gimp' und 'ImageMagick' produziert und unterliegen dem gleichen Copyright wie der Rest dieses Dokuments.

Die gezeigten Programme und Programmteile unterliegen der GNU General Public License, bekannt als GPL.

Alle im Dokument verwendeten IP-Adressen und Bezeichnungen haben mit real existierenden Geräten nichts gemeinsam. Sie dienen ausschließlich der Erklärung der Zusammenhänge.

Glossar

3DES Triple Data Encryption Standard, veraltete schnelle Stromchiffrierung, normalerweise als 3DES-EDE im CBC-Modus betrieben mit 112 oder 168 Bit Schluesseln.

AES Advanced Encryption Standard, schnelle Stromchiffrierung, auch als Rijndael bekannt, 256 Bit Schluessellaenge

AH Authenticated Header, einer der beiden IPSec-Modi

ARP Address Resolution Protocol, Bindeglied zwischen Layer 2 und 3 u.a. bei Ethernet

BSD Berkely Software Design, OpenSource Unix-Betriebssystem auf den Quellen von AT&T basierend, http://www.bsd.org

CGI Common Gateway Interface, Webserver-Funktionalitaet

CIDR Classless Inter Domain Routing, durch Adressmangel verursachte Aufgabe der vordefinierten Netzmasken in A-,B- und C-Klassen, Netze werden nun durch die Anzahl gesetzter Bits in der Netzmaske festgelegt, Netzmaskenbits immer linksbuendig auf 1 ohne Unterbrechungen. Damit werden Netze zu SuperNets und Subnetze zu Netzen zusammenfassbar. Heute sind nur noch Routen groesser /19 im Internet zu finden.

CRL Certificate Revokation List, von der CA herausgegebene Liste aller ungueltigen Zertifikate

Debian Das Debian-Projekt wurde von Ian Murdoch und seiner Frau Debra initiiert, die jeweils ersten Buchstaben der Vornamen ergeben Debian

DES Data Encryption Standard, veraltete schnelle Blockverschluesselung 56 Bit Schluessellaenge

DN Destinguished Name aus der X-500 Begriffswelt

DNAT Destination NAT, die Ziel-Adresse wird umgeschrieben

DNS Domain Name System, weltweit verteilte Datenbank zur Umsetztung von IP-Adressen auf lesbare Namen und umgekehrt

DSA Digital Signature Algorithm, asymmetrischer Crypto-Algorithmus

ESP Encapsulated Payload, einer der beiden IPSec-Modi

GNU ist eine rekursive Definition: GNU is Not Unix, mehr dazu bei der Free Software Foundation, oder auch unter http://www.gnu.org

IP Internet Protokoll, in sog. Request-for-Comments (RFC) festgelegte Art und Weise, Daten in Form von Paketen zu transportieren. IP in der noch aktuellen Version 4 legt RFC791 fest, wie die Bits der Pakete gedeutet werden.

IPSec Internet Protocol Security, zuerst mit Version 6, dann auch bei Version 4 des IP verwendet, in einigen RFC festgeschrieben und aktueller defacto-Standard der Verschluesselungstechnik

L2TP Layer two Tunneling Protocol, PPP over IP

lilo linux loader, boot-programm, um Linux von der Festplatte zu starten

NAT Network Address Translation, Quell- oder Zieladresse eines Paketes wird gezielt modifiziert. Meist bei privaten Adressen aus RFC1918 unvermeidbar

NVRAM Non volatile RAM, nicht fluechtiger Speicher

OpenSource Darunter werden Quelltexte von Programmen verstanden, die frei von Patentschutz oder anderweitigen Einschraenkungen im rechtlichen Sinne sind. Meist unter GPL oder anderen, sinnverwandten Lizenzen verbreitet. Durch das 1000-Augen-Prinzip vielleicht sicherer als nicht-offene Software.

PPP Point-to-Point-Protocol

RC4 Rivest Code 4, schnelle Stromchiffrierung

RSA asymmetrischer Crypto-Algorithmus, Rivest, Shamir, Adleman sind die Erfinder

SNAT Source NAT, die Quell-Adresse wird umgeschrieben

snort Netzwerk Einbruchserkennung, http://www.snort.org

ssh Secure Shell, cryptographischer Telnet- und Ftp-Ersatz, benutzt anerkannt sichere Verfahren, um Authentisierung und Datentransfer zu bewerkstelligen.

SSPE Simple Security Policy Editor, zu finden unter http://sspe.sourceforge.net

VPN Virtual Private Network, allgemeiner Oberbegriff in Verbindung mit Internet als Transportvehikel, Schutz der Privatheit bei gleichzeitiger Internetverbindung und Verbindung der eigenen Netze an verschiedenen Orten

X.509 Unternorm von X.500-Directories, behandelt die binaere Darstellung von Zertifikaten und derer Inhalte

Abbildungsverzeichnis

Literaturverzeichnis

[Bau97] Friedrich L. Bauer. *Entzifferte Geheimnisse.* Springer Verlag Berlin Heidelberg New York, 1997.

[Bau02] Michael D. Bauer. *Building SECURE SERVERS with LINUX.* O'Reilly, 2002.

[BC94] Bellovin and Cheswick. *Firewalls and Internet Security.* Addison Wesley, AT&T, 1994.

[Bou02] Boutell. http://www.boutell.com, 2002.

[Cha02] John Viega & Matt Messier & Pravir Chandra. *Network Security with OpenSSL.* O'Reilly & Associates, Inc., 2002.

[Erw99] Charlie Scott & Paul Wolfe & Mike Erwin. *Virtuelle Private Netzwerke.* O'Reilly Verlag, 1999.

[Fre02] FreeSWAN.org. http://www.freeswan.org, 2002.

[JS01] Daniel J.Barret and Richard E. Silverman. *SSH, the Secure Shell: The Definitve Guide.* O'Reilly & Associates, Inc., 2001.

[KD95] Olaf Kirch and Terry Dawson. *LINUX Network Administrator's Guide.* O'Reilly & Associates, Inc., 1995.

[KH02] Oleg Kolesnikow and Brian Hatch. *Building Linux^{TM} Virtual Private Networks (VPNs).* New Riders, 2002.

[Pol01] Russ Housley & Tim Polk. *Planning for PKI.* Wiley Computer Publishing, 2001.

[Sch00] Bruce Schneier. *Angewandte Kryptographie.* Addison Wesley, 2000.

[Tho96] Linus Thorvalds. http://www.kernel.org/pub/kernel/2.0/, 1996.

Anhang C

GNU Free Documentation License

Version 1.2, November 2002

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document freeïn the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.
This License is a kind of c̈opyleft;̈ which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.
We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The **"Document"**, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as **"you"**. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A **Modified Version"** of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A **SSecondary Section"** is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The **Ïnvariant Sections"** are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The **Cover Texts"** are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A **Transparent"** copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image

format is not Transparent if used for any substantial amount of text. A copy that is not Transparentïs called **Öpaque"**.
Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.
The **Title Page"** means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, Title Pagemeans the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.
A section **Ëntitled XYZ"** means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as **Äcknowledgements"**, **"Dedications"**, **Ëndorsements"**, or **"History"**.) To **Preserve the Title"** of such a section when you modify the Document means that it remains a section Ëntitled XYZäccording to this definition.
The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.
You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if

there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "Historyïn the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "Historyßection. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled Äcknowledgementsör "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled Ëndorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled Ëndorsementsör to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled Ëndorsements", provided it contains nothing but endorsements of your Modified Version by various parties–for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "Historyïn the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled Äcknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled Ëndorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an äggregateïf the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.
If a section in the Document is Entitled Äcknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.
Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License ör

any later versionäpplies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright ©YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts.line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.
If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.