INSTITUT FÜR INFORMATIK

DER LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Fortgeschrittenenpraktikum

Aufbau einer Testumgebung für die sichere TCP/IP-basierte Kommunikation

vorgelegt von

Igor Radisic

Aufgabensteller:	Prof. Dr. H.-G. Hegering, LMU München
Betreuer:	Stephen Heilbronner, LMU München
	Dr. Bernhard Neumair, LMU München
	Norbert Jungfleisch, BMW AG
Abgabetermin:	19. November 1997

Inhaltsverzeichnis

Kapitel 1. Einführung

1.1 Motivation

Die Internet-Welt beschäftigt sich schon seit längerem mit der Lösung eines sehr wichtigen, aber oft vernachlässigten Problems, nämlich der Sicherheit der TCP/IP-basierten Kommunikation. In bezug auf die Kommunikationssicherheit sollen v.a. die folgenden Anforderungen erfüllt werden:

- **Vertraulichkeit der übertragenen Daten:**
  Es soll keinem unberechtigten Dritten durch Abhören von Datenleitungen möglich sein, sicherheitssensible Daten und Informationen zu erlangen. Eine derartig geschützte Kommunikation über ein Netz kann durch kryptographische Verfahren gewährleistet werden.

- **Anonymität der Kommunikationsbeziehung:**
  Indem verschleiert wird, wer mit wem wie oft kommuniziert, soll das Anlegen von Benutzerprofilen erschwert werden. Auch diese Anforderung kann durch Einsatz kryptographischer Verfahren gewährleistet werden.

- **die Verbindlichkeit von Benutzeraktionen und -daten:**
  Es soll eine eindeutige Zuordnung zwischen durchgeführten Aktionen/Daten und einem Benutzer bestehen, um z.B. nachvollziehen zu können, ob eine Nachricht tatsächlich vom angegebenen Absender kommt. Dies wird durch Erstellung von digitalen Signaturen für jeden Benutzer gewährleistet, die einer Unterschrift auf Papierdokumenten entspricht.

Es liegt auf der Hand, daß v.a. Unternehmen die das Internet kommerziell nutzen wollen bzw. TCP/IP zur unternehmensinternen Kommunikation (Intranet/Internet) einsetzen, sich vor aktiven und passiven Angriffen[1] Dritter schützen wollen.
Da das Protokollpaar TCP/IP derzeit noch keine Möglichkeit bietet solch eine Kommunikationssicherheit zu gewährleisten, müssen die Protokolle, Dienste und Anwendungen, die auf TCP/IP aufsetzen, dies durch geeignete Verfahren kompensieren. In der Internet-Kommunikation existieren bereits für die folgenden Protokolle, Dienste und Anwendungen geeigneter Ersatz bzw. geeignete Erweiterungen:

- HTTP: S-HTTP, HTTPS (setzt auf das von Netscape entwickelte Secure Socket Layer [SSL] auf)

- e-mail: Verschlüsselung der Nachricht durch z.B. PGP oder Verwendung von S/MIME [S/MIME][RFC 1847] anstatt von MIME [RFC 1521][RFC1522]

[1] **aktiver Angriff** (engl. *tampering*): unautorisierter Zugriff auf Daten, Maskieren oder Entfernen von Datenpaketen aus einem Strom von übertragenen Daten, Denial-of-service-Attacken, d.h. Angriffe auf die Verfügbarkeit von System-Ressourcen, z.B. durch das Überfluten des Netzes mit Nachrichten
passiver Angriff (engl. *wiretapping*): Abhören von Datenleitungen in vernetzten Systemen
siehe auch [Eckert1][Eckert2].

- rlogin, rsh, rcp: Ersatz durch SSH

- FTP: Nutzung der in SSH integrierten Weiterleitung von Ports um die Steuerverbindung von FTP kryptographisch zu sichern (siehe auch Kapitel 3.7 dieser Ausarbeitung)

Desweiteren existieren zahlreiche kommerzielle Produkte verschiedener Firmen, die die Authentifizierung von Benutzern durch Ausstellung von Zertifikaten unterstützen.

Die BMW AG wird nun zunehmend ihre Datenkommunikation auf TCP/IP-basierte Netze abstützen. Auch hier werden in wachsendem Umfang sensible Daten über die leicht angreifbaren Netzstrukturen übertragen. In einer parallel laufenden Diplomarbeit [Reis97] wird ein Konzept entwickelt, mit dem die Kommunikation, die u.a. mit den Protokollen HTTP, FTP und SMTP abgewickelt wird, abgesichert werden kann. Dabei werden u.a. die oben vorgestellten Erweiterungen untersucht.

1.2 Aufgabenstellung

Das Ziel dieses Fortgeschrittenenpraktikums war es, eine geeignete Testumgebung bei der BMW AG für die sichere TCP/IP-basierte Kommunikation aufzubauen, um damit die parallel laufende Diplomarbeit [Reis97] am Institut zu unterstützen. Dies sollte vor allem durch die Installation von sich bereits auf dem Markt befindenden Produkten geschehen. Die Hauptaufgabe bestand dabei in der Suche von geeigneter Software, dem Protokollieren der einzelnen Installationsschritte dieser Software auf verschiedenen Plattformen, sowie des Testens der Software in heterogener Umgebung, um damit eine mögliche Empfehlung für den tatsächlichen unternehmensweiten Einsatz geben zu können. Ebenso sollte bei jedem Produkt auch der mögliche Einsatz am Institut geprüft werden.
An dieser Stelle sei darauf hingewiesen, daß z.T. Beta-Versionen getestet wurden, und die bei der betreffenden Software aufgetretenen Probleme im Endprodukt durchaus nicht mehr vorkommen können.

1.3 Gliederung und Aufbau der Ausarbeitung

In Kapitel 2 werden jeweils die Netzwerkumgebungen am Institut und der BMW AG vorgestellt. Es werden jeweils die zur Verfügung gestellten Endgeräte, deren Betriebssysteme und ihre Verbindung untereinander anhand einer Graphik erklärt.

In Kapitel 3 werden die installierten und getesteten Softwareprodukte folgendermaßen vorgestellt:

- im ersten Abschnitt wird in knapper Form erklärt, was das Produkt leisten soll
- im zweiten Abschnitt werden die grundlegenden Voraussetzungen (wie z.B. bereits vorhandene Software) für eine erfolgreiche Installation genannt
- im dritten Abschnitt werden die Installationsschritte und möglich auftretende Probleme sowie deren Beseitigung beschrieben
- im vierten Abschnitt werden die Vor- und Nachteile der Software gegenübergestellt

- im fünften Abschnitt werden wichtige Daten, wie z.B. Installationszeit, Mindestplattenspeicher, etc. in Form einer Tabelle notiert.

Im Anhang befindet sich schließlich eine genaue Beschreibung der Prozedur der Zertifikatsbeantragung und -ausstellung mit dem Netscape Certificate Server V1.01. Weiterhin ist im Anhang ein Verzeichnis zu den verwendeten Abkürzungen angegeben.

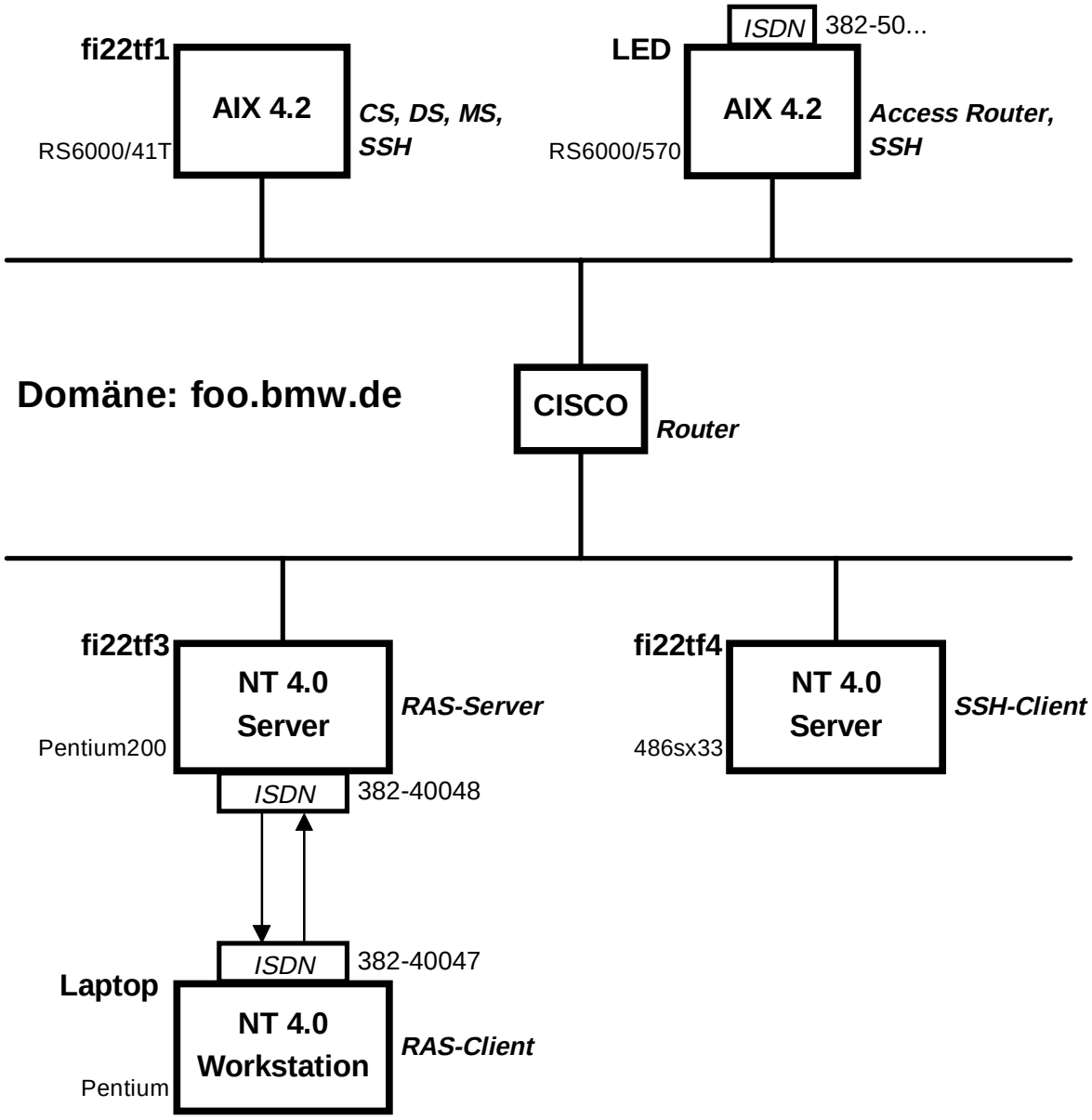
Im Literaturverzeichnis sind neben der verwendeten Literatur, den RFCs und den Internet Drafts, auch die URLs derjenigen WWW-Seiten notiert, die weitergehende und detailliertere Informationen zu dem Themenbereich bieten.

Kapitel 2. Die Netzwerkumgebungen

2.1 Die Testumgebung bei der BMW AG München

Im folgenden wird das von der BMW AG zur Verfügung gestellte Testfeld anhand einer Skizze und in Form von tabellarischen Beschreibungen der Endgeräte, vorgestellt.

2.1.1 Übersichtskizze des Testfeldes



Skizzenerläuterung

- Ein Rechteck repräsentiert jeweils ein Endgerät
- das jeweilige **Betriebssystem** eines Endgeräts ist innerhalb des Rechtecks notiert
- der **Name** eines Endgeräts steht fettgedruckt links über dem Rechteck
- die jeweils auf einem Endgerät **installierte Software** ist kursiv rechts neben dem Rechteck notiert und folgendermaßen abgekürzt worden:

CS	*Netscape Certificate Server*
DS	*Netscape Directory Server*
MS	*Netscape Mailserver*
RAS	*Remote Access Service*
SSH	*Secure Shell (sowohl daemon als auch client)*
SSH-Client	*SSH-Client für Windows*

2.1.2 Beschreibung der Endgeräte

Im folgenden werden die genauen Daten der zur Verfügung gestellten Endgeräte vorgestellt:

1. Laptop:

Marke, Typ	Compaq LTE 5280
Prozessor	Intel Pentium
Arbeitsspeicher	16 MB RAM
Betriebsystem	Windows NT 4.0 Workstation
Erweiterungskarten	PCMCIA Ethernetkarte PCMCIA ISDN-Karte: Eicon.Diehl Diva
ISDN-Telefonnummer	382-40047

2. LED:

Marke, Typ	IBM RS6000/570
Plattenspeicher	2 * 2 GB
Betriebsystem	IBM AIX 4.2
ISDN-Telefonnummer	382-50xx

3. fi22tf1:

Marke, Typ	IBM RS6000/41T
Plattenspeicher	2 * 2 GB
Betriebsystem	IBM AIX 4.2

4. fi22tf3:

Marke, Typ	Compaq Deskpro 6000
Prozessor	Intel Pentium Pro 200Mhz
Arbeitsspeicher	32 MB RAM
Betriebsystem	Windows NT 4.0 Server
Erweiterungskarten	Ethernetkarte: Integrated NetFlex 10T UTP PCI ISDN-Karte: Eicon.Diehl Diva Pro
ISDN-Telefonnummer	382-40048

5. fi22tf4:

Marke, Typ	Compaq Deskpro 4000
Prozessor	Intel 486sx 33Mhz
Arbeitsspeicher	24 MB RAM
Betriebsystem	Windows NT 4.0 Server
Erweiterungskarten	Ethernetkarte: Intel Ether Express 16 LAN-Adapter

2.2 Die Netzwerkumgebung am Institut für Informatik der LMU München

Die Netzwerkumgebung am Institut für Informatik der LMU München wird in mehrere Domänen eingeteilt. Der für alle Studenten der Informatik zugängliche **CIP-Pool** wird dabei durch die Domäne *.cip.informatik.uni-muenchen.de repräsentiert. Zusätzlich besitzt jeder Lehrstuhl des Instituts seinen eigenen Rechnerpool, der wiederum durch eine eigene Domäne repräsentiert wird. Der **Rechnerpool des Lehrstuhls** für Kommunikationssysteme und Systemprogrammierung, in dem ein Teil der Installationen durchgeführt worden ist, hat hierbei die Domäne *.nm.informatik.uni-muenchen.de.
Im folgenden werden die zur Installation eingesetzten Rechner am Institut für Informatik der LMU München in Form von Tabellen vorgestellt.

2.2.1 Die Endgeräte des CIP-Pools

Im CIP-Pool sind ca. 40 gleichartige Workstations miteinander verbunden. Die für die durchgeführten Installationen relevanten Workstations lauten:

- juno : 129.187.214.172
- mars : 129.187.214.173
- merkur : 129.187.214.174
- minerva : 129.187.214.175
- neptun : 129.187.214.176

Die genauen Daten der Endgeräte lauten:

Marke, Typ	HP-workstation 9000/710
Arbeitsspeicher	32 MB RAM
Plattenspeicher	402 MB
Betriebsystem	HP-UX 10.20
Domäne	*.cip.informatik.uni-muenchen.de

2.2.2 Die Endgeräte des Rechnerpools des Lehrstuhls

Im Rechnerpool des Lehrstuhls für Kommunikationssysteme und Systemprogrammierung sind ca. 15 Endgeräte miteinander verbunden. Dabei handelt es sich um HP-workstations 9000/710-715, SUN sparc-workstations und IBM RS-workstations. Für die Installation der Software wurde dabei nur die folgende HP-workstation genutzt:

Hostname	hpheger9
IP-Adresse	129.187.214.29
Marke, Typ	HP-workstation 9000/715
Arbeitsspeicher	64 MB RAM
Plattenspeicher	402 MB
Betriebsystem	HP-UX 10.20
Domäne	*.nm.informatik.uni-muenchen.de

Kapitel 3. Die installierten Softwareprodukte

In den folgenden Unterkapiteln sind jeweils die einzelnen Softwareprodukte sowie die zugehörigen Installationsprotokolle notiert.
Folgende Abkürzungen werden verwendet:
[TEMP] : Pfad zum temporären Verzeichnis, in das das gepackte File als erstes verschoben und in dem es schließlich entpackt worden ist
[INST_PATH] : Pfad zum Verzeichnis, in dem die Software letztendlich installiert wurde (wird während der Installation meist interaktiv abgefragt)

3.1 Pretty Good Privacy (PGP) Version 2.6.3i

Bei PGP handelt es sich um ein Freeware-Produkt von Philip Zimmermann zur Verschlüsselung von Dateien nach einem RSA-IDEA-Hybridverfahren. Jede zu verschlüsselnde Nachricht wird mit einem neugenerierten symmetrischen Session-Key (Länge: 128 Bit) verschlüsselt. Dieser Session-Key wird anschließende mit dem Public-Key des Empfängers verschlüsselt und der verschlüsselten Nachricht angehängt. PGP kann aufgrund der Verwendung des RSA-Public-Key-Verfahrens [Eckert2] sowohl zur **Verschlüsselung** als auch zum **Signieren von Nachrichten** bzw. Dateien eingesetzt werden. Damit kann einerseits die Vertraulichkeit der übertragenen Nachrichten als auch deren Verbindlichkeit zum angegebenen Absender gewährleistet werden. Mit Hilfe von zusätzlicher Software kann PGP direkt in bereits vorhandenen Mail-Clients integriert werden (siehe auch Kapitel 3.2 dieser Ausarbeitung). Installiert und getestet wurde es auf den folgenden Plattformen: HP-UX 9.X und 10.X, Solaris 2.51 und IBM AIX 4.25.

3.1.1 Voraussetzungen für eine erfolgreiche Installation

Für die Installation auf UNIX-Plattformen müssen folgende Voraussetzungen erfüllt sein:

- C-Compiler muß vorhanden sein (z.B. gcc oder cc)
- die Funktion ‘memmove’ muß vorhanden sein (nicht notwendigerweise in jeder Standard-C-Library enthalten)

3.1.2 Installationsschritte

1. Download von der „**International PGP Homepage**“
   URL: http://www.ifi.nio.no/pgp/

2. File-Name: **pgp263is.tar.gz** (Größe 593 KB)

3. Dieses File in ein temporäres Verzeichnis kopieren

4. Mit **gunzip** das File extrahieren → pgp263is.tar bleibt übrig

5. Mit ‘**tar -xvf pgp263is.tar**’ das File entpacken

→ es werden neue Verzeichnisse angelegt, darunter auch das **Folder 'src'**

6. In das Verzeichnis '***[TEMP]*/src**' wechseln

7. Mit '**make hpux -pa -gcc**' wird der **Source-Code für HP-UX10.X** mit dem gcc-Compiler kompiliert
   mit der **Eingabe von 'make'** alleine werden **alle möglichen make-Optionen erklärt** (u.a. wird die Liste der unterstützten UNIX-Plattformen und C-Compiler ausgegeben)
   → es entsteht das ausführbare File 'pgp'

8. Bei **lokaler** Installation folgende Schritte durchführen:
   - Im lokalen Hauptverzeichnis (z.B. /users/stud/radisic) neues Verzeichnis anlegen, z.B. mit dem Namen 'mybin'
   - 'pgp' in dieses Verzeichnis kopieren (hier also in das Verzeichnis /users/stud/radisic/mybin)
   - 'setenv PATH [Pfad des lokalen bin-Verzeichnisses] :$PATH' eingeben (hier also: 'setenv PATH /users/stud/radisic/mybin:$PATH')
     → damit wird der Suchpfad beim Aufruf von Befehlen erweitert
   - Im home-Verzeichnis ein Unterverzeichnis mit dem Namen '.pgp' kreieren
   - In dieses Unterverzeichnis unbedingt die Dokumentationsfiles 'pgpdoc1.txt' und 'pgpdoc2.txt' verschieben (das Programm sucht beim Aufruf nach diesen Files)
   - Weiterhin sollten die folgenden Files in dieses Unterverzeichnis verschoben werden: 'language.txt', 'config.txt' und '.hlp' (u.a. kann in 'config.txt' die Sprache und der verwendete Zeichensatz eingestellt werden → weitere Informationen zu den verschiedenen Einstellungsmöglichkeiten sind in 'pgpdoc1.txt' und 'pgpdoc2.txt' zu finden)

9. Bei **systemweiter** Installation folgende Schritte durchführen:
   - 'pgp' in ein globales '/bin'-Verzeichnis kopieren (z.B. in '/users/bin')
   - Die man-pages 'pgp.l' an die entsprechende Stelle verschieben
   - Ein Verzeichnis mit dem folgenden Pfad kreieren: 'usr/local/lib/pgp'
   - Die Files 'language.txt.', 'config.txt', und '.hlp' in dieses Verzeichnis kopieren
   - Weiterhin benötigt jeder User ein eigenes '.pgp'-Verzeichnis in seinem home-Directory (u.a. werden in diesem Unterverzeichnis die eigenen Schlüssel abgelegt)

10. Die Installation ist damit beendet

Befehlsübersicht

- '**pgp -h**' gibt eine **Befehlsübersicht** in der Shell aus

- Zunächst mit '**pgp -kg**' ein **Schlüsselpaar generieren**
  → bei der Frage nach der userid folgendes Format beachten:
  Vorname Nachname <e-mail-Adresse>

→ das generierte Schlüsselpaar wird jeweils in die eigenen public- und secret-keyring-files hinzugefügt (Endung '.asc'; befinden sich dann im '.pgp'-Verzeichnis)

- Zum **Extrahieren eines Schlüssels** aus dem public- oder secret-keyring:
  pgp -kx userid keyfile [keyring]

- Zum **Hinzufügen eines** oder **mehrerer Schlüssel** in die eigenen Keyrings:
  pgp -ka keyfile [keyring]

- Zum **Verschlüsseln** einer Datei:
  pgp -e Klartextdatei Empfänger-userid

- Zum **Signieren** einer Datei:
  pgp -s Datei

- zum **Entschlüsseln einer Datei** und **Überprüfen der Signatur**:
  pgp verschlüsselte_Datei

3.1.3 Vor- und Nachteíle von PGP

Vorteile	Nachteile
- bietet starke Verschlüsselung - Quasi-'Standard' im Internet - Freeware	- umbequeme Handhabung durch Kommandozeilenmodus

3.1.4 Übersichtstabelle der wichtigsten Daten

URL-Adresse	http://www.ifi.nio.no/pgp/
Filename	pgp263is.tar.gz
Filegröße	593 KB
Minimum-Plattenspeicher	5 MB
Installationszeit	ca. 15-20 min.
Verschlüsselungsverfahren	RSA-IDEA Hybridverfahren
Dokumentation	*[TEMP]*/doc/pgpdoc1.txt *[TEMP]*/doc/pgpdoc2.txt setup.doc
Getestete Plattformen	HP-UX 9.X, 10.X Solaris 2.51 IBM AIX 4.25

Im CIP-Pool des Instituts für Informatik wurde PGP unter **/soft/IFI/text/pgp** installiert. Der Source-Code befindet sich unter **/soft/IFI/src/INST-pgp**.

3.2 PGPPine

Wie schon in Kapitel 3.1 erwähnt, besteht der größte Nachteil von PGP darin, daß es nicht direkt von einem Mailclient aus aufgerufen werden kann, sondern daß bei jeder Verschlüsselung von Emails immer der Umweg über die Kommandozeile erfolgen muß. Um dieses Manko zu beseitigen, werden zahlreiche Programme und Perl-Skripten angeboten, die ein **Interface zwischen** einem bestehendem **Mailclient** und **PGP** bilden. '**PGPPine**' ist in diesem Fall ein Perl-Skript, das eine Schnittstelle zu dem Mailclient 'Pine' ab der Version 3.92 zur Verfügung stellt. Installiert und getestet wurde dieses Perl-Skript auf HP-UX 9.X und 10.X, Solaris 2.51, IBM AIX 4.25.

3.2.1 Voraussetzungen für eine erfolgreiche Installation

Folgende Voraussetzungen müssen für eine Installation auf UNIX-Plattformen erfüllt sein:

- Pine ab Version 3.92 bereits vorhanden
- PGP bereits installiert

3.2.2 Installationsschritte

1. Download-URL-Adresse: **http://www.rhein.de/~roland/pgppine**

2. Filename: **pgppine.tar.gz**

3. Im Homeverzeichnis entpacken und extrahieren
   → es wird ein neues Verzeichnis mit dem Namen '**/pgppine**' angelegt
   Inhalt: **pgpdecode, pgpencrypt, pgpsign** (Perl-Skripten)

4. '**pgpdecode**' mit einem Editor aufrufen:
   in der 25.Zeile fogende Änderung durchführen: 'cbreak' durch 'rem' ersetzen

5. Aufruf von **PINE** aus dem Hauptverzeichnis heraus

6. Innerhalb von PINE **Menüpunkt Setup/Config** auswählen (S, C)
   → Liste von Konfigurationsfeldern erscheint

7. Auf den Punkt '**display-filters**' (ziemlich am Ende der Liste) gehen und folgende Zeile eingeben:
   BEGINNING("-----BEGIN PGP") /.../pgppine/pgpdecode
   anstelle der drei Punkte ('...') muß unbedingt der **vollständige** Pfad angegeben werden

8. Auf den Punkt '**sending-filters**' gehen und folgende Zeilen eingeben:
   /.../pgppine/pgpsign
   /.../pgppine/pgpencrypt

auch hier muß wieder anstelle der drei Punkte (‘...’) der vollständige Pfad zu diesen Dateien angegeben werden

10. Die Konfiguration mit ‘E’ verlassen

11. Die Installation ist damit beendet

Bedienungsübersicht

- Beim Schreiben von Emails mit Pine, die verschlüsselt oder signiert werden sollen, unbedingt ‘-----BEGIN PGP’ am Anfang der mail notieren
  → daran erkennt Pine beim Erhalt dieser Nachricht, daß es sich um eine verschlüsselte/signierte Botschaft handelt und führt damit das Perl-Skript ‘pgpdecode’ aus

- Beim Senden der Nachricht (Ctrl-X) stellt Pine folgende Optionen zur Verfügung:
  not filtered — filtered through pgpsign — filtered through pgpencrypt
  (mit Ctrl-N und Ctrl-P kann zwischen den Optionen gewechselt werden)

3.2.3 Übersichtstabelle der wichtigsten Daten

URL-Adresse	http://www.rhein.de/~roland/pgppine
Filename	pgppine.tar.gz
Filegröße	4 KB
Minimum-Plattenspeicher	11 KB
Installationszeit	ca. 10 min.
Dokumentation	keine, außer der WWW-Seite
Getestete Plattformen	HP-UX 9.X, 10.X Solaris 2.51 IBM AIX 4.25

3.3 Netscape Communicator PR 4 Professional Edition (Beta-Version)

Bei dem **Netscape Communicator PR 4 Professional Edition** handelt es sich um die Beta-Version des neuesten Produkts der Firma Netscape, das den WWW-Browser Netscape Navigator ersetzen soll. Der Communicator ist nicht mehr nur ein Browser, sondern ein Zusammenschluß von mehreren Internet-Dienstprogrammen:

- Navigator 4.0 (WWW-Browser)
- Mailbox Messanger (Mail-Client)
- Page Composer (HTML-Editor)
- Conference (Aufbau einer „Konferenzschaltung“)
- Collabra Discussion Group
- IBM Host-On-Demand

Wie man an dieser Liste ersehen kann, ist das Ziel das Netscape mit dem Communicator verfolgt, alle Aspekte der Internet-Kommunikation abzudecken.
Der **Messanger** bietet zu den üblichen Funktionen eines **Mail-Clients** zusätzlich die Möglichkeit der **Verschlüsselung und Signierung von Nachrichten** durch die Unterstützung von **S/MIME** [S/MIME][RFC 1847]. Damit kann einerseits die Vertraulichkeit der übertragenen Nachrichten als auch deren Verbindlichkeit zum angegebenen Absender gewährleistet werden. Leider ist wegen des USA-Exportverbots von starken Verschlüsselungsverfahren die Schlüssellänge auf 40 Bit beschränkt. In der nordamerikanischen Version des Netscape Communicators ist eine Schlüssellänge bis zu 128 Bit möglich. Eine Verschlüsselung ist aber erst dann möglich, wenn ein **Zertifikat von einer Certification Authority** (CA) ausgestellt wurde. Bei der Beantragung dieses Zertifikats wird das Schlüsselpaar des Users generiert, das aber erst dann benutzt werden kann, wenn das beantragte Zertifikat von der CA positiv bestätigt wurde. Momentan gibt es zahlreiche kommerzielle CAs, die für ca. 10 US-$ (z.B. VeriSign) ein einzelnes Zertifikat ausstellen. Für einen unternehmensweiten Einsatz von Zertifikaten wird der **Netscape Certificate Server V1.01** angeboten (siehe auch 3.4 dieser Ausarbeitung).
Der **Navigator 4.0** unterstützt das von Netscape entwickelte Protokoll **SSL V3.0** und **SSL V2.0** [SSL] zur sicheren Internet-Kommunikation zwischen WWW-Client und WWW-Server. Natürlich ist eine sichere Übertragung nur dann möglich, wenn der WWW-Server ebenfalls SSL unterstützt.
Zum Zeitpunkt der Installation lag der Netscape Communicator nur in der Version für MS Windows 3.X, 95 und NT vor. Im BMW-Testfeld wurde der Netscape Communicator auf dem Laptop, fi22tf3 und fi22tf4 installiert.

3.3.1 Voraussetzungen für eine erfolgreiche Installation

Für die Installation auf einer Windows-Plattform sind keine besonderen Voraussetzungen bekannt, außer daß genügend Plattenspeicher vorhanden sein muß (siehe Tabelle unter 3.3.4).

3.3.2 Installationsschritte

1. Download von der '**International Netscape Page**':
   URL: http//home.netscape.com/download/

2. Aus der Liste die entsprechende Software, Sprache, System, etc. wählen
   → zum Schluß den Download-Server auswählen
   hier: Universität Augsburg, URL:http//ftp.Uni-Augsburg.DE/pub/all_OS/netscape

3. File-Name: '**p32e40b4.exe**'

4. File in ein temporäres Verzeichnis legen

5. Im 'Start'-Menü von Windows 95 oder NT 4.0 auf '**Ausführen...**' gehen
   → Fenster mit einer Eingabezeile wird geöffnet

6. Den kompletten Pfad zum File angeben (z.B. 'C:\TEMP\cb32e401.exe') und mit 'OK' bestätigen
   → Extrahierung wird gestartet und nach Abschluß dieser Aktion startet selbständig das Installationsprogramm

7. U.a. wird nach der **Art der Installation** ('Custom'), dem **Zielverzeichnis**, der zu installierenden Komponenten, usw. gefragt
   → am Schluß mit '**Install**' die Angaben **bestätigen**

8. Am Ende der Installation ist ein **Neustart des Rechners nötig**

9. Zusätzlich bei NT 4.0 Server/Workstation:
   - wird der Communicator für alle User eingesetzt, so wir beim ersten Aufruf des Communicators der '**Profile Setup Wizard**' gestartet
   - an dieser Stelle wird dann der volle Name des Users, die e-mail-Adresse, die Adresse des Mail-Servers, etc. Schritt für Schritt eingetragen

3.3.3 Vor- und Nachteile

Vorteile	Nachteile
- deckt alle Bereiche der Internet-Kommunikation ab - relativ komfortable Bedienung - unterstützt Zertifikatsausstellung	- bietet nur schwache Verschlüsselung bei Nachrichten an (40 Bit) → sehr unsicher - Verschlüsselung nur mit Zertifikat möglich - z.T. sehr schwache bis überhaupt keine Dokumentation

3.3.4 Übersichtstabelle der wichtigsten Daten

URL-Adresse	http://home.netscape.com/
Filename	p32e40b4.exe
Filegröße	15 MB
Minimum-Plattenspeicher	35 MB (Professional Edition with all plug-ins)
Installationszeit	ca. 15 min.
Sicherheitsstandards- und protokolle	S-MIME (40 Bit) bei Email SSL V3.0, V2.0 bei http
Dokumentation	• WWW-Seite von Netscape • Online-Dokumentation
Getestete Plattformen	Windows 95, NT 4.0 Server, NT 4.0 Workstation

3.4 Netscape Certificate Server V1.01

Wie schon unter 3.3 beschrieben, ist ein Verschlüsseln und Signieren von Nachrichten mit dem Netscape Messanger erst mit dem positivem Erhalt eines Zertifikats möglich. Der **Netscape Certificate Server V1.01** dient der **unternehmensweiten, selbständigen Ausstellung und Verwaltung** von **Zertifikaten**. Mit der Zertifikatsausstellung an die Mitarbeiter eines Unternehmens kann intern die Verbindlichkeit von Nachrichten und die Vertraulichkeit der übertragenen Emails gewärleistet werden. Es wird ausdrücklich ein baumstrukturierter Aufbau von mehreren CS unterstützt, das v.a. in Hinblick auf die verteilte Struktur von Unternehmen sehr dienlich ist. Statt daß alle Zertifikate von einem zentralen CS ausgestellt und verwaltet werden, autorisiert ein zentraler Root-CS mehrere verteilte, ‘lokale’ CS, die wiederum, falls nötig, weitere CS autorisieren können. Dadurch kann v.a. die Performance hinsichtlich der Ausstellung und Verwaltung von Zertifikaten gesteigert werden. Netscape hat bei Performance-Tests festgestellt, daß ein alleinstehender CS bei der Verwaltung von bis zu 1000 Zertifikaten sehr gute Ergebnisse liefert. Netscape rät jedoch ab einer Anzahl von 2000 Zertifikaten zur Installation des **Netscape Directory Servers** (siehe Kapitel 3.5), der eine bessere Datenbank-Verwaltung der Zertifikate unterstützt.
Der Netscape CS V1.01 ist für die folgenden Plattformen erhältlich:

- Digital Unix 4.0b
- HPUX 10.10
- IBM AIX 4.1 und 4.2
- IRIX 5.3, 6.2
- Solaris 2.4 und 2.5
- Windows NT ab 3.51

Installiert und getestet wurde der Netscape Certificate Server V1.01 auf einer HPUX10.20 Plattform. Im Rechnerpool des Lehrstuhls fürKommunikationssysteme und Systemprogrammierung wurde der NCS V1.01 auf der hpheger9 installiert.

Desweiteren wurde die Vorgängerversion NCS V1.0 auf einer IBM AIX 4.2 Plattform getestet und installiert. Diese Version des CS wurde im BMW-Testfeld auf der fi22tf1 installiert.

3.4.1 Voraussetzungen für eine erfolgreiche Installation

Folgende Punkte müssen unbedingt erfüllt sein, damit der NCS V1.01 erfolgreich auf einer HPUX-Plattform installiert werden kann:

- mind. 64 MB RAM
- mind. 90 MB Plattenspeicher
- Installation muß als Superuser durchgeführt werden
- in der Domäne muß der DNS-Server laufen
- alle anderen Netscape Server (falls vorhanden) müssen vor der Installation angehalten werden

- der CS muß innerhalb einer Domäne mit einer Länderkennung am Ende des Domänen-Namenstrings (hier: nm.informatik.uni-muenchen**.de**)installiert werden; innerhalb einer Domäne die keine Länderkennung am Schluß des Namenstrings hat (z.B. .muc) kann zwar der CS installiert, aber nicht korrekt konfiguriert werden
- Netscape Navigator ab Version 3.01 bereits vorhanden

3.4.2 Installationschritte

Die folgenden Schritte beschreiben die Installation des NCS V1.01 auf der **hpheger9** im Rechnerpool des Lehrstuhls.

Die Installation läßt sich in **drei Abschnitte** gliedern:

a) Installation der Informix-Datenbank
b) Installation des Certificate Servers (Netscape Administration Server wird mitinstalliert, falls dieser noch nicht vorhanden ist)
c) Konfiguration des Certificate Servers

1. Download von der '**International Netscape Page**'
   URL: http://home.netscape.com/download/

2. Filename: '**cm101hpu.tar.gz**' (Größe: ca. 25 MB)

3. File in ein **temporäres Verzeichnis** (hier: '**/tmp/certif/**') verschieben und dort zunächst mit '**gunzip**' entpacken → '**cm101hpu.tar**' bleibt übrig

4. Mit '**tar -xvf cm101hpu.tar**' das File extrahieren
   → das Verzeichnis '*[TEMP]*/certificate-101-export-us.hppa1.1-hp-hpux10.10' wird neu angelegt

   in diesem Verzeichnis befinden sich die folgenden Files:
   - license.txt : Hinweise zu den Linzenzbestimmungen
   - INSTALL.TXT : Installationsanleitung auf englisch
   - CERTSVC.TAR : Installations-package des NCS V1.01
   - INFORMIX.SH : Installations-package des Informix Datenbank-Servers

5. Für die weiteren Installationsschritte ist eine Superuser-Kennung notwendig: hier: **su -root2**
   → das '-' darf nicht weggelassen werden, da die $HOME Umgebungsvariable des Superusers benötigt wird

a) Installation des Informix-Datenbank-Servers

1. Zunächst muß eine **Gruppe 'informix'** und ein **User 'informix'** (mit Passwort!) eingerichtet werden (mit Homeverzeichnis = /usr/informix)
   hier: Verwendung des HPUX-Administrationstools '**sam**'

2. Das File '**/etc/hosts.equiv**' des Server-hosts (hier:hpheger9) folgendermaßen editieren (falls nicht vorhanden, muß das File angelegt werden):

   <vollständiger Host-Domain-Name>
   <Abkürzung des Hostnamen>

   hier also:
   hpheger9.nm.informatik.uni-muenchen.de
   hpheger9

3. Falls NIS eingesetzt wird (im Rechnerpool der Fall), muß das file '**/etc/services**' um folgende Zeile erweitert werden:
   ifmx_srvc 2055/tcp

4. In das Verzeichnis '***[TEMP]*/certificate-101-export-us.hppa1.1-hp-hpUX10.10**' **wechseln**
   → das File 'INFORMIX.SH' befindet sich in diesem Verzeichnis

5. Die Installation mit folgendem Befehl anstoßen: '**sh INFORMIX.SH**'

6. Folgende Installationseinstellungen verwenden:
   - **Installationsort**: (hier das Homeverzeichnis des in Schritt 1. eingerichteten Users 'informix' angeben) hier: '/usr/informix'
   - **DB-Size**: 20 MB
   - **Installationsart**: Express-Installation

7. Nach ca. 15-20 min. sollte die Installation erfolgreich abschließen
   → den Bildschirm genau beobachten, da bei nicht erfolgreichen Installation Fehlermeldungen angezeigt werden

8. Der **Informix Server-Prozeß** wird nach **erfolgreicher Installation automatisch gestartet**

9. Befehlsübersicht:
   - **anhalten** des Serverprozesses: **sh /usr/informix/stop.sh**
   - **starten** des Serverprozesses: **sh /usr/informix/start.sh**

Ursachen für eine nicht erfolgreiche Installation:
- DNS-Server wurde in der Domäne nicht gestartet
- falls Re-Installation: Informix-Serverprozeß vorher anhalten
  (s.a. 'Troubleshooting' am Ende dieses Kapitels)

b) Installation des Certificate Servers

1. **Neuen User 'cmsdbusr'** (mit Passwort) einrichten
   → mit dieser Kennung wird der CS die Informix-Datenbank ansprechen
   → dieser User entspricht dem Datenbank-Administrator für die Zertifikate
   → das **Passwort** muß bei der **Konfiguration des CS** (Abschnitt c)) in Schritt 3. angegeben werden

2. In das Verzeichnis '***[TEMP]/* certificate-101-export-us.hppa1.1-hp-hpUX10.10**' **wechseln**
   → an dieser Stelle befindet sich das CS-Installationspackage 'CERTSVC.TAR'

3. Mit folgendem Befehl das Package entpacken:
   '**tar -xvf CERTSVC.TAR**'

   folgende Files werden dabei in das aktuelle Verzeichnis extrahiert:
   - ns-cms.tar : das CS-package
   - ns-setup : das Installationsprogramm, das u.a. 'ns-cms.tar' entpackt

4. './**ns-setup**' aufrufen
   folgende Angabe ist dabei zu machen:
   - **Server Root**: (hier Verzeichnis angeben, in dem der Server installiert werden soll) **hier: /usr/ns-home**

   daraufhin wird ns-cms.tar entpackt

   → falls der **Netscape Administration Server nicht** schon durch ein anderes Produkt von Netscape **installiert wurde** (wie z.B. Netscape Mailserver), wird **dieser mitinstalliert**

5. Nach ca. 10-15 min. erfolgen **Konfigurationsangaben** für den **Administration Server** (nur für den Fall, daß dieser nicht schon installiert wurde):

   Fullname: (an dieser Stelle den Namen komplett mit der Domäne des Hosts angeben, auf dem der Server gerade installiert wurde und der Serverprozeß in Zukunft laufen wird) hier: **hpheger9.nm.informatik.uni-muenchen.de**
   Admin Port: (es wird ein zufälliger Port für den AS errechnet; unbedingt merken, da man diesen später für die Konfiguration benötigt) hier: **24678**
   Admin Server: (Kennung unter der der Serverprozeß laufen soll) hier: **root2**
   Access username: (Username, den man zur Anbindung an den AS benützen will) hier: **admin**
   Password: (an dieser Stelle ein Passwort für den obigen User angeben)
   Hosts: (an dieser Stelle diejenigen Hosts angeben, die von dem AS verwaltet werden dürfen und die als einzige den AS ansprechen dürfen)
   hier: ***.nm.informatik.uni-muenchen.de,**
   ***.cip.informatik.uni-muenchen.de**

IP-Address: (dasselbe wie bei ‘Hosts’ nur über die IP-Adressen)
hier: **129.187.214.29**

location des Konfigurationsfiles: *[INST_PATH]*/ns-home/admserv/ns-admin.conf

6. Nach diesen Angaben wird gefragt, ob ein WWW-Browser zur Konfiguration des Certificate Servers gestartet werden soll:
   → dies **verneinen,** falls auf dem Server-Host kein Netscape Navigator ab Version 3.01 installiert ist
   → es wird die **URL des Administration Servers** angegeben, unter der man den **Certfication Server konfigurieren** kann; die URL setzt sich wie folgt zusammen: **http://<host>.<domain>:<Port>**

   hier: **http://hpheger9.nm.informatik.uni-muenchen.de:24678**

Ursachen für eine nicht erfolgreiche Installation:
- der Informix-Datenbank-Server wurde nicht erfolgreich installiert
- der Informix-Datenbank-Serverprozeß wurde angehalten bzw. nicht gestartet: mit ‘sh usr/informix/start.sh’ den Prozeß starten
- nicht genügend Plattenspeicher vorhanden

c) Konfiguration des Certificate Servers

1. Einen WWW-Browser, falls nötig, von einem anderem Host aufrufen (am besten mit Netscape Navigator ab Version 3.01)

2. **URL des Administration Servers** eingeben:
   hier: http://hpheger9.nm.informatik.uni-muenchen.de:24678

3. Auf der neuaufgebauten Seite ‘**Install a new Certificate Server**’ auswählen
   → **Installation Wizard** für die Konfiguration des Certificate Servers **wird gestartet**
   → die nachfolgenden Schritte werden im Browser jeweils genau beschrieben

4. Die **Konfiguration gliedert** sich in die **folgenden acht Schritte**:

   1. *CA-Signing Key Generation:*
      In diesem Schritt wird das **Schlüsselpaar der Certification Authority** (CA) generiert, mit denen dann die mit dem CS ausgestellten Zertifikate signiert werden.
      - Auf den **Host wechseln**, auf dem der **CS installiert** wurde (hier: hpheger9)
      - In das Verzeichnis ***[INST_PATH]*/ns-home** wechseln (hier: /usr/ns-home)
      - Zur Schlüsselgenerierung folgenden Befehl ausführen:
        bin/cms/admin/bin/gen-sgn-key -k 1024
        → die Anweisungen auf dem Bildschirm verfolgen
        → zum Schluß wird nach einem Paßwort für dieses Schlüsselpaar gefragt

location des CA-Schlüsselpaares: /usr/ns-home/cms-hpheger9/
/config/CASigningkey.db

- Wieder auf den **Host wechseln**, auf dem der **Browser mit dem Installation Wizard** läuft → am **Ende der Browserseite** das oben angegebene **Signing-Key-Paßwort eingeben** und bestätigen

2. Server SSL key Generation:
In diesem Schritt wird das **Schlüsselpaar für die SSL-Kommunikation** generiert, das in Schritt 5. benötigt wird.
- Die im Browser-Fenster angegebenen Anweisungen durchführen (ähnliche Prozedur wie in Schritt 1.)

location des SSL-Schlüssels: /usr/ns-home/cms-hpheger9/config/
/ServerKey.db

- Am **Ende der Browserseite** das Paßwort für das **generierte SSL-Schlüsselpaar** eingeben

3. Database Configuration:
In diesem Schritt wird die **Datenbank zur Verwaltung der Zertifikate angelegt**.
- Folgende Angaben sind zu machen:

Database Root Directory: (Home-directory des Informix-Datenbank-Servers)
hier: **/usr/informix**
Database Server: (Name des Datenbank-Servers, das die Daten des CS enthalten wird; default-Wert übernehmen)
hier: **ifmx_online**
Database Name: (Name der Datenbankinstanz für den CS) hier: **cmsdb**
Database User: (Kennung des Users, mit dem der CS die Datenbank anspricht; wurde bei der CS-Installation in Schritt 1. kreiert)
hier: **cmsdbusr**
password: (Passwort des Database Users)

4. Issue CA Certificate:
An dieser Stelle werden **Angaben** gemacht, die jeweils **in jedem** von dem CS **ausgestelltem Zertifikat** notiert werden und damit die dadurch repräsentierte **CA ausweisen**. An Hand dieser Angaben kann jeder User entscheiden, ob er eine dritte Person, die sich mit einem **Zertifikat dieser CA ausweist, akzeptiert oder ablehnt** (im Netscape Communicator kann ein User z.B. generell angeben von welcher CA er Zertifikate grundsätzlich akzeptiert oder ablehnt; siehe Anhang A, Punkt iv) dieser Ausarbeitung). Im Prinzip läßt sich dies mit der Ausstellung eines Zertifikats für den eigenen CS vergleichen, d.h. man autorisiert sich selbst Zertifikate ausstellen zu dürfen.
- Folgende Angaben sind zu machen:

Common-Name: (Name der CA) hier: **LMU - Informatik CA**
Organization Unit: (Name der Abteilung) hier: **MNM**

Organization: (an dieser Stelle unbedingt den Host auf dem der CS läuft und Teil des Domänenstrings **ohne** Länderkennung angeben) hier: **hpheger9.nm.informatik.uni-muenchen**
Country: (Länderkennung) hier: **DE**

5. *Issue Server SSL Certificate:*
Der CS wird über einen Browser angesprochen und administriert. Damit die **Verbindung zwischen Browser und CS gesichert**, also verschlüsselt, abläuft, wird das von Netscape entwickelte **Protokoll SSL** [SSL] eingesetzt. Ein User, der mit diesem Server kommunizieren will, prüft wieder an Hand eines Zertifikats, ob er diesem Server vertraut und damit einer SSL-Verbindung zustimmt. Bei Zustimmung werden schließlich mit dem Server vertrauliche Daten ausgetauscht. Wie in Schritt 4. stellt man sich hier ein eigenes Zertifikat aus.
- Folgende Angaben sind zu machen:

Common-Name: (hier unbedingt den vollständigen Domainnamen des hosts auf dem der CS installiert wird, angeben) hier: **hpheger9.nm.informatik.uni-muenchen.de**
Organization Unit: (Name der Abteilung) hier: **MNM**
Organization: (wie in Schritt 4.) hier: **hpheger9.nm.informatik.uni-muenchen**
Country: (Länderkennung) hier: **DE**
Length of Validity Period: hier: **2 years**

6. *Create Server Administrator:*
An dieser Stelle wird der **Superuser für die Server Administration** kreiert. D.h. es wird eine Kennung des Superusers angegeben sowie ein Schlüsselpaar mit dem sich der Superuser ausweist, generiert. Für diesen User wird daraufhin im nächsten Schritt ein Zertifikat ausgestellt, der dann in den gerade verwendeten Browser importiert wird. Nur die Person, die gerade den CS konfiguriert und installiert, wird in der Lage sein, sich nach Beendigung dieser Schritte unter dem angegebenen Namen beim CS als ‘priviledged’ anzumelden und zwar nur von dem gerade verwendeten Browser aus (das Zertifikat wird nur in diesen Browser importiert).
- Folgende Angaben sind zu machen:

Username: (Kennung des Superusers) hier: **radisic**
Common-Name: (Name des Superusers) hier: **Service Admin**
Organization Unit: hier: **MNM**
Organization: hier: **hpheger9.nm.informatik.uni-muenchen**
Country: hier: **DE**
Keysize: hier: **512**
Length of Validity Period: hier: **2 years**
- Mit der Bestätigung dieser Angaben wird das Schlüsselpaar des Superusers generiert

7. *Import Aministrator Certificate:*
An dieser Stelle wird dem **Superuser ein Zertifikat** von der eben in Schritt 4. generierten CA **ausgestellt**. Mit diesem Zertifikat, das u.a. sein in Schritt 6. generierten Public Key enthält, weist sich der Superuser aus. Desweiteren

wird **dieses Zeritifkat** in den **gerade geöffneten** und zur Installation verwendeten **Browser importiert**. D.h., daß der Superuser in Zukunft nur über diesen Browser den CS ansprechen wird können.
- Den Button '**Import Certificate**' betätigen

8. Installation Complete:
Dieser achte Schritt weist nur darauf hin, daß die Installation des CS und Administration Servers erfolgreich abgeschlossen wurde.

5. zum Schluß muß noch der Serverprozeß mit folgendem Befehl gestartet werden:
   - Auf host wechseln, auf dem der Server installiert wurde (hier: hpheger9)
   - ***[INST_PATH]*****ns-home/cms-hpheger9/start**

6. Die Installation ist damit beendet

Befehlsübersicht für den CS auf der hpheger9

- **Starten der Serverprozesse**:

  jeweils beim Hochfahren des CS die folgende Reihenfolge beachten:

  1. **Administration Server**: /usr/ns-home/start-admin
  2. **Informix-Server**: sh /usr/informix/start.sh
  3. **Certificate Server**: /usr/ns-home/cms-hpheger9/start

- **Anhalten der Serverprozesse**:

  jeweils beim Herunterfahren des CS die folgende Reihenfolge beachten:

  1. **Certificate Server**: usr/ns-home/cms-hpheger9/stop
  2. **Informix-Server**: sh usr/informix/stop.sh
  3. **Administration Server**: usr/ns-home/stop-admin

Der AS und der CS werden jeweils über ihre URLs angesprochen.

In **Anhang A** dieser Ausarbeitung ist die Prozedur der Beantragung und Ausstellung von Zertifikaten beschrieben.

In **Anhang C** dieser Ausarbeitung sind die am **Bootvorgang** des CS-hosts hpheger9 beteiligten Dateien für das automatische Starten des CS, dokumentiert.

Troubleshooting

- Informix-Server läßt sich nicht starten und folgende Fehlermeldung wird angezeigt:

```
network error: -25572: Network driver cannot bind a name to the port
```

  In diesem Fall wurde wahrscheinlich der Host heruntergefahren ohne daß der Informix-Server mit dem Befehl ‘sh /usr/informix/stop.sh’ korrekt angehalten wurde.

  Lösung: alle Dateien im Verzeichnis ‘/INFORMIXTMP’ löschen und den Informix-Server nochmals starten

- In der Datei ‘**/usr/informix/msg/errmsg.txt**’ sind alle Fehlermeldungen und -codes des Informix-Servers beschrieben, sowie eine detailliertere Beschreibung der Fehlerursache(n) notiert

- für auftretende Fehler bei den Netscape Servern, aber auch bei Informix-Problemen, sollte der Netscape Helpserver als Hilfestellung zur Problemlösung herangezogen werden (URL:**http://help.netscape.com**)

3.4.3 Übersichtstabelle der wichtigsten Daten

URL-Adresse	http://home.netscape.com/download
Filename	cm101hpu.tar.gz
Filegröße	25 MB
Minimum-Plattenspeicher	90 MB
Minimum RAM	64 MB
Installationszeit	ca. 60 min.
URL-Zusammensetzung des AS	http://<host>.<domäne>:<port> hier: **http://hpheger9.nm.informatik.uni-muenchen.de:24678**
URL-Zusammensetzung des CS	https://<host> hier: **https://hpheger9**
Dokumentation	• WWW-Seite von Netscape • Online-Dokumentation • im Installationsverzeichnis ‘/tmp/certif/certificate-101-export-us.hppa1.1-hp-hpux10.10/’ befinden sich nützliche Textfiles
Verfügbare Plattformen	Digital Unix 4.0b, HPUX 10.10, IBM AIX 4.1 und 4.2, IRIX 5.3, 6.2, Solaris 2.4 und 2.5, Windows NT ab 3.51
Getestete Plattform	IBM AIX 4.2, HPUX 10.20

3.5 Netscape Directory Server V1.03

Der Netscape Directory Server V1.03 dient allgemein der **effizienteren Verwaltung von Daten**, die bei der Verwendung von Netscape Enterprise-Produkten (insbesondere der Server-Linie) anfallen. Eingesetzt wird der Directory Server (DS) in aller erster Linie zur effizienteren **Benutzerverwaltung** beim Netscape Mailserver (siehe Kapitel 3.6) und **Zertifikatsverwaltung** beim Netscape Certificate Server (siehe Kapitel 3.4). In der getesteten Version existiert der DS für die folgenden Plattformen:

- Solaris 2.4, 2.5 (SunOS 5.4/5.5)
- IRIX 6.2
- HPUX 10.10
- AIX 4.1, 4.2
- OSF 4.0
- Windows NT 4.0

Installiert und getestet wurde der Netscape Directory Server auf einer HPUX 10.20 Plattform. Im Rechnerpool des Lehrstuhls für Systemnahe und Technische Informatik wurde der DS auf der hpheger9 installiert.

3.5.1 Voraussetzungen für eine erfolgreiche Installation

Folgende Punkte müssen unbedingt für eine erfolgreiche Installation des DS erfüllt sein:

- mind. 32 MB RAM (64 MB empfohlen)
- für HPUX mind. 81 MB Plattenspeicher
- Netscape Navigator ab Version 3.01 bereits installiert

3.5.2 Installationsschritte

Im folgenden wird die Installation des DS auf der **hpheger9** unter der Kennung '**root2**' beschrieben. Für die Installation des DS ist allerdings eine Superuser-Kennung nicht erforderlich.

1. Download von der '**International Netscape Page**'
   URL: **http://home.netscape.com/download/**

2. Filename: '**directory-1.03-export-us.hppa1.1-hp-hpux10.10.tar.gz**'
   (Größe: 14 MB)

3. File in ein temporäres Verzeichnis verschieben und dort entpacken
   hier: **/tmp/certif/directory-server**
   → 'directory-1.03-export-us.hppa1.1-hp-hpux10.10.tar' bleibt übrig

4. Mit '**tar -xvf directory-1.03-export-us.hppa1.1-hp-hpux10.10.tar**' das File extrahieren

folgende Files werden in das aktuelle Verzeichnis gelegt:
- README : Installationsbeschreibung in englischer Sprache
- license.txt : Hinweise zu den Linzenzbestimmungen
- ns-slapd.tar : das DS-package
- ns-setup : Installationsprogramm, das u.a. das 'ns-slapd.tar'-package entpackt

6. Im temporären Verzeichnis (hier: /tmp/certif/directory-server) das Installationsprogramm folgendermaßen aufrufen:
   ./ns-setup

7. Falls der **Administration Server** (AS) **nicht schon** durch eine andere Netscape-Software wie z.B. dem Certificate Server **installiert** wurde, **wird dieser mitinstalliert** (Konfiguration des AS siehe Kapitel 3.4 Abschnitt b))

8. Nach Beendigung der Installation, den **AS über einen Browser** (Navigator ab Version 3.01 notwendig) ansprechen
   hier: URL: **http://hpheger9.nm.informatik.uni-muenchen.de:24678**

9. Auf der neuaufgebauten Seite folgenden Punkt auswählen:
   '**Install a new Netscape Directory Server**'

10. Bei der nun aufgebauten Seite müssen die folgenden Angaben gemacht werden:

    Server Name: (hier den vollständigen Host-Domainnamen angeben, auf dem der DS installiert wird)
    hier: **hpheger9.nm.informatik.uni-muenchen.de**
    Server Port: (default-Wert übernehmen, falls nicht schon belegt)
    hier: **389**
    Encryption Enabled: (an dieser Stelle angeben, ob die Kommunikation zum AS und zu den jeweils anderen Server, die den DS nutzen, verschlüsselt ablaufen soll; falls dies mit 'yes' bestätigt wird, benötigt der DS ein Zertifikat, das bei einer CA beantragt werden muß (siehe Schritt 11.))
    hier: **yes**
    Encrypted Port: (default-Wert übernehmen, falls nicht schon belegt)
    hier: **636**
    Server Identifier: ('beliebigen' ID-Namen für den DS angeben)
    hier: **hpheger9**
    Configure Directory for SuiteSpot: (falls der DS mit SuiteSpot 3.0 Servern zusammenarbeiten soll, diese Frage mit yes beantworten)
    hier: **no**
    Server User Name: (hier die Kennung angeben, unter der der DS laufen soll; es ist keine Superuser-Kennung nötig)
    hier: **root2**

Database Subtree: (dies sind Angaben für die zu generierende Datenbank:
bei ‘O =’ eine Organisation wie z.B. die Firma angeben und
bei ‘C =’ die Länderkennung angeben)
hier: **O = LMU Informatik, C = DE**
Unrestricted User: (hier einen Superuser für die Datenbank angeben, wobei
bei ‘CU =’ ein Name angegeben werden soll)
hier**: CU = DirectorySU, O = LMU Informatik, C = DE**
Password: (Passwort für Datenbank-Superuser angeben und bei ‘... again’
wiederholen)

Durch Bestätigen dieser Angaben wird der DS endgültig installiert.

11. Falls der Punkt ‘**Encryption Enabled**’ mit ‘**no**’ beantwortet wurde, kann der **Server gestartet** werden

    Falls der Punkt ‘**Encryption Enabled**’ mit ‘**yes**’ angekreuzt wurde (wie in diesem Fall geschehen), benötigt der DS ein **Server Certificate** um gestartet werden zu können

Server Certificate für den DS

Grundsätzlich müssen folgende Punkte unternommen werden, um ein Server Certificate für den DS zu erhalten:

a) Schlüssel generieren
b1) Server Certificate beantragen
b2) Server Certificate ausstellen (nur von ‘priviledged’ CS-Usern möglich)
c) Server Certificate installieren

Im folgenden wird die Prozedur beschrieben, wie man mit Hilfe eines bereits installierten CS das Server Certificate für den DS beantragt, ausstellt und installiert.

a) Schlüssel generieren

1. Den AS über einen Browser ansprechen
   hier: URL: **http://hpheger9.nm.informatik.uni-muenchen.de:24678**

2. Auf der Anfangsseite des AS den eben installierten DS auswählen
   hier: **Directory Server: hpheger9**

3. Auf der nun aufgebauten Startseite des DS, in der **obersten Menüleiste** den Menüpunkt ‘**Encryption**’ auswählen

4. Im **Seitenmenü** von ‘Encryption’ den Punkt ‘**Generate Key**’ auswählen

5. In der nun aufgebauten Seite sind die folgenden Schritte zur Schlüsselgenerierung beschrieben:
   i) Auf **Server-Host** (hier hpheger9) als '**root**' **einloggen**
   ii) In das Verzeichnis des DS wechseln
   hier: **/usr/ns-home/slapd-hpheger9**
   iii) Folgendes Programm starten:
   ./sec-key
   iv) Den Anweisungen folgen:
   → der Schlüssel wurde abgespeichert in
   '**/usr/ns-home/slapd-hpheger9/ssl/key.db**'
   v) Unbedingt (auch wenn als Option angezeigt) folgendes Programm starten:
   ./sec-dngl
   das keyfile-Passwort wurde abgespeichert in
   '**/usr/ns-home/slapd-hpheger9/ssl/Passwort.dng**'
   Nach Beendigung dieser Schritte kann ein Server Certificate beantragt werden

b1) Server Certificate beantragen

1. Dem Hyperlink '**Request a Server Certificate**' folgen

2. Das folgende Formular, das auf der neuaufgebauten Seite angezeigt wird, ausfüllen:
   Certificate authority: (an dieser Stelle wird normalerweise die e-mail-Adresse einer Kontaktperson der CA angegeben, wie z.B. netscape-cert@verisign.com; in diesem Fall aber, soll das Zertifikat von dem bereits installierten Netscape CS ausgestellt werden → deswegen wird hier die eigene, persönliche e-mail-Adresse des Antragstellers angegeben)
   hier: **radisic@nm.informatik.uni-muenchen.de**
   Common Name: (an dieser Stelle muß der volle Domain-Name des Servers angegeben werden)
   hier: **hpheger9.nm.informatik.uni-muenchen.de**
   Email address: (hier die email-Adresse der Person angeben, die die automatische Benachrichtung der CA erhalten soll; in diesem Fall ist dies nicht von Bedeutung)
   hier: **radisic@nm.informatik.uni-muenchen.de**
   Organization: (optionales Feld für die Angabe der Firma)
   hier: **LMU Informatik**
   Organizational Unit: (optional) **MNM**
   Locality: (optional) **munich**
   State: (Staat, indem die Firma registriert ist)
   hier: **germany**
   Country: (unbedingt gültige Länderkennung angeben)
   hier: **DE**
   Telephone Number: (unbedingt eine (nicht unbedingt gültige) Nummer angeben) hier: **089/123456**

Durch **Bestätigen dieser Angaben**, wird der Antrag an die **oben angegebene e-mail-Adresse der CA** geschickt → in diesem Fall erhält der Antragsteller (hier: radisic@nm.informatik.uni-muenchen.de) das Antragsformular, das in einem der folgenden Schritte wieder benötigt wird

3. Den **CS über einen Browser** ansprechen und als '**public**' anmelden
   hier: URL:**https://hpheger9**

4. Auf den Menüpunkt '**Request a Server Certificate**' des **Seitenmenüs** gehen

5. Die **erhaltene e-mail mit dem Antragsformular** in einem Fenster öffnen und den Text zwischen '**Begin Certificate Request**' und '**End Certificate Request**' (inklusive dieser Header) **markieren**

6. Den **eben markierten Text** in das Feld unterhalb der Überschrift '**Server Certificate Request**' kopieren

7. Bei '**Contact Information**' die verlangte Information angeben

8. Die Angaben durch Betätigen des Buttons '**Submit Request**' bestätigen

die folgenden Schritte unter b2) kann nur ein 'priviledged' User des CS durchführen !

b2) Server Certificate ausstellen

1. Den **CS über Browser** ansprechen und als '**priviledged**' anmelden
   hier: URL:**https://hpheger9**

2. Den Menüpunkt '**List Certificate Signing Requests**' des **Seitenmenüs** auswählen

3. Die Option '**Show waiting Requests**' auswählen und den Button '**Run Query**' betätigen

4. In der nun neu aufgebauten Seite den **entsprechenden Antrag auswählen**

5. Im Browser wird daraufhin der komplette Antrag angezeigt
   → den Hyperlink '**Assign to me**' betätigen

6. Auf der nun aufgebauten Seite, können verschiedene Optionen verändert werden:

   Length of Validity Period: hier: 2 years
   Netscape Certificate Type (Usage): hier: SSL Client, SSL Server, Secure mail
   Key Identifier: hier: Include Authority Key Identifier, Include Subject Identifier

7. Die Angaben mit ‘**Issue this Certificate**’ bestätigen und damit das Zertifikat ausstellen

8. Zum Schluß wird das ausgestellte Zertifikat angezeigt
   → wichtig ist v.a. der Text zwischen ‘**-----BEGIN CERTIFICATE-----**‘ und ‘**-----END CERTIFICATE-----**’

c) Server Certificate installieren

1. Den CS ansprechen und als ‘**public**’ anmelden
   hier: URL:**https://hpheger9**

2. ‘**List all Certificates**’ des **Seitenmenüs** wählen und ‘**Run Query**’ starten

3. Das enstsprechende (mit ‘**issued**’ bereits ausgestellte) **Zertifikat auswählen**

4. Auf der neuaufgebauten Seite den **Text** zwischen ‘**-----BEGIN CERTIFICATE-----** ‘ und ‘**-----END CERTIFICATE-----**’ (inklusive dieser Header) **markieren** und **kopieren** (im Browser-Menü ‘Edit’)

5. Den **AS ansprechen** und den **DS** (hier:hpheger9) **auswählen**
   hier: URL:**http://hpheger9.nm.informatik.uni-muenchen.de:24678**

6. Den Menüpunkt ‘**Encryption / Install a Certificate**’ auswählen

7. Auf der nun aufgebauten Seite, sind folgende Angaben zu machen:

   Certificate For: hier: This Server
   Certificate Name: hier: DS Certificate
   Install in certificate database for: hier: Directory Server

8. Jetzt den in Schritt 4 markierten Text in das Feld unterhalb von ‘**Message text (with headers):**’ kopieren

9. Mit ‘**OK**’ die Angaben bestätigen

10. Mit ‘**Add Certificate**’ das Zertifikat nochmals bestätigen

11. Jetzt kann der Server gestartet werden

Befehlsübersicht für den DS auf der hpheger9

- **Starten des DS**: /usr/ns-home/slapd-hpheger9/start

- **Anhalten des DS**: /usr/ns-home/slapd-hpheger9/stop

Administriert wird der DS über den AS: URL:http://hpheger9.nm.informatik.uni-muenchen.de:24678

3.5.3 Übersichtstabelle der wichtigsten Daten

URL-Adresse	http://home.netscape.com/
Filename	directory-1.03-export-us.hppa1.1-hp-hpux10.10.tar.gz
Filegröße	14 MB
Minimum-Plattenspeicher	81 MB (HPUX)
Minimum RAM	32 MB (64 MB empfohlen)
Installationszeit	ca. 35 min.
Dokumentation	• WWW-Seite von Netscape • Online-Dokumentation • *[TEMP]*/README
Verfügbare Plattformen	- Solaris 2.4 und 2.5 (SunOS 5.4/5.5) - IRIX 6.2, HPUX 10.10 - AIX 4.1 und 4.2 - OSF 4.0 - Windows NT 4.0
Getestete Plattform	HPUX 10.10

3.6 Netscape Mailserver V2.0

Um den Austausch von verschlüsselten e-mails mit dem Netscape Messanger testen zu können, wurde ein Mailserver benötigt. Da die Software, die bis zu diesem Zeitpunkt im BMW-Testfeld installiert wurde, ohne Ausnahme von Netscape stammte, wurde entschieden auch den Mailserver von der Firma Netscape zu beziehen. So kann ein Vergleich zu bekannten Mailservern, wie z.B. sendmail, gezogen werden. In der getesten Version existiert der Mailserver für die folgenden Plattformen:

- IBM AIX 3.25, 4.1, 4.2
- Solaris 2.4, 2.5
- HP-UX 9.05, 10.01, 10.10
- IRIX 5.3, 6.2
- Windows NT 4.0

Installiert und getestet wurde der **Netscape Mailserver V2.0** auf einer IBM AIX V4.2 Plattform. Im BMW Testfeld wurde er auf der fi22tf1 installiert.

3.6.1 Voraussetzungen für eine erfolgreiche Installation

Für die Installation auf einer IBM AIX V4.2 Plattform sind folgende Voraussetzungen zu beachten:

- mind. 32 MB RAM
- mind. 40 MB Plattenspeicher
- Netscape Navigator ab Version 3.01 bereits vorhanden

Achtung: Der Netscape Mailserver überschreibt während der Installation alle bestehenden Daten und Dateien eines fremden Mailservers, wie z.B. sendmail, falls dieser auf demselben Host installiert ist. D.h. insbesondere, daß alle **forwarding-Adressen gelöscht** werden!

3.6.2 Installationsschritte

1. Download von der '**International Netscape Homepage**'
   URL: **http://home.netscape.com/download/**

2. Filename: '**ms20aix42.tar.gz**'

3. In ein temporäres Verzeichnis verschieben und entpacken
   → '**ns-setup**' wird u.a. extrahiert

4. Mit dem Aufruf von '***[TEMP]*/ns-setup**' wird die **Installationroutine** gestartet

5. Folgende Angaben sind zu machen:

   Server root: (hier den Pfad zu dem Directory angeben, in dem der Server installiert werden soll) z.B. **/usr/ns-home**
   Configure Server: yes
   User group permission: (hier Unix-Gruppenname angeben unter der die Installation durchgeführt werden soll; default-wert ist email/email)
   Your domain name: (Name der Domäne) hier: **foo.bmw.de**
   Post Office directory: **/var/spool/postoffice** [default-wert]
   User Mailbox directory: **/var/spool/mailbox** [default-wert]
   Administration identifier: **foo.bmw.de**
   NIS module: **no**
   send greetings: (an dieser Stelle angeben, ob neue User mit einer Standard-email begrüßt werden sollen) hier: **yes**
   user for mail account: (an dieser Stelle soll angegeben werden, welche user eine e-mail-Kennung erhalten sollen; in Unix bietet sich die Möglichkeit die passwd-Datei zu durchsuchen) hier: **/etc/passwd**
   Mail delivery: [P]
   Mail domains: (an dieser Stelle angeben, welche Domains der Mailserver verwalten soll) hier: **foo.bmw.de**
   outgoing domain: hier: **foo.bmw.de**
   address format: [C]
   change Postmaster account:
      Postmaster password: (Passwort für den Mailserver-Administrator eingeben)
      Access domains: hier: foo.bmw.de
      Postmaster: (an dieser Stelle, die Kennung des Postmasters eingeben) hier: root@foo.bmw.de
   Administration Server Port: (an dieser Stelle den Port des Administration Servers angeben; diese Meldung erscheint nur, falls dieser bereits installiert wurde, ansonsten wird der AS mitinstalliert) hier: **8081**

6. Die Installation ist damit beendet

7. Durch den **Administration Server** kann man neben dem **Certificate Server** jetzt auch den **Mailserver** administrieren
   → **URL des AS** in einen Browser geben (hier**: http://fi22tf1.foo.bmw.de:8081**)
   → im Browser erscheint neben dem Certificate Server (falls installiert) auch der Mailserver:
   durch Anwählen des Mailservers erscheint ein neues Fenster, in dem z.B. neue mail-accounts, domains, etc. eingegeben werden können

8. Auf **Client-Seite** müssen **noch folgende Schritte** für jeden User durchgeführt werden:
   - im **Mailbox Messanger** des Netscape Communicators '**Preferences**' aus der **Menüleiste** wählen → neues Fenster mit Verzeichnisbaum wird geöffnet
   - unter '**Mail & Groups**' die Option '**Mail Server**' anwählen
     → in dem nun neu geöffneten Fenster folgende Angaben machen:
     Mail Server user name: (an dieser Stelle die Mail Server-Kennung des Users eingeben)
     outgoing mail Server: (an dieser Stelle, den host angeben, auf dem der Mail Server installiert wurde) hier: fi22tf1
     Incoming mail Server: (wie bei 'outgoing mail server') hier: fi22tf1

Probleme:

- Während des Testens sind im Zusammenhang mit dem Netscape Mailserver Probleme mit der **DNS-Auflösung** aufgetreten
  → Behebung: - auf **Client-Seite** (Windows NT) die Option '**DNS-Auflösung**' in der **Rubrik 'Netzwerk' ausschalten** und stattdessen '**Lmhosts-Datei**' wählen
  (folgendes Verzeichnis:**/Winnt/systems32/drivers/ets/Lmhosts**)
  - anschließend die Datei '**/Winnt/systems32/drivers/ets/Lmhosts**' wie folgt modifizieren: <host> <IP-Adresse>
  . . .
  z.B.: fi22tf4 160.50.48.165

3.6.3 Vor- und Nachteile

Vorteile	Nachteile
- Administration über gleiche Oberfläche wie CS - relativ komfortable Oberfläche - nützt alle Features des Netscape Messangers (z.B. Übertragung von Word-Dokumenten, Animationen, etc.)	- noch einige Bugs enthalten - grafische Oberfläche bei schneller Bedienung eher hinderlich (Maus ↔ Tastatur) - bei Installation 'über' einen bestehenden Mailserver, werden alle forwarding-Adressen gelöscht

3.6.4 Übersichtstabelle der wichtigsten Daten

URL-Adresse	http://home.netscape.com
Filename	ms20aix42.tar.gz
Filegröße	18 MB
Minimum-Plattenspeicher	40 MB
Minimum RAM	32 MB
Installationszeit	ca. 40 min.
anhalten/starten des Servers	*[INST_PATH]*/ns-home/ms/start bzw. stop
Dokumentation	Online-Dokumentation
Verfügbare Plattformen	- IBM AIX 3.25, 4.1, 4.2 - Solaris 2.4, 2.5 - HP-UX 9.05, 10.01, 10.10 - IRIX 5.3, 6.2 - Windows NT 4.0
Getestete Plattformen	IBM AIX 4.2

3.7 Remote Access Service (RAS)

Das Betriebssystem Windows NT 4.0 bietet für die sichere Kommunikation über das Post/Telefonnetz den **RAS-Dienst** an. Mit RAS hat der User die Möglichkeit **über ISDN** oder **Modem** auf das **Netz**, in dem der RAS-Server installiert ist, **zuzugreifen**, als **ob er direkt vor Ort** in das Netz eingebunden ist. Um eine sichere Kommunikation zu ermöglichen, wird der **Nachrichtenaustausch verschlüsselt**. RAS wurde jeweils auf Windows NT 4.0 Server und NT 4.0 Workstation installiert. Im BMW-Testfeld diente der fi22tf1 als RAS-Server und der Laptop als RAS-Client.

3.7.1 Voraussetzungen für eine erfolgreiche Installation

Folgende Voraussetzungen sind vor der Installation zu beachten:
- ISDN-Karte bzw. Modem bereits vorhanden
- ISDN-Karte bzw. Modem nicht bereits als NDIS-Dienstnutzer konfiguriert
- ggf. Windows NT CD-ROM

3.7.2 Installationsschritte

Grundsätzlich gibt es **zwei Möglichkeiten** den RAS-Dienst zu installieren:

a) durch Installation der Treiber der ISDN-Karte:

Während der Treiber-Installation der ISDN-Karte wird meist gleich zu Anfang gefragt, ob gewünscht wird, daß der RAS-Dienst mitinstalliert wird. Bejaht man diese Frage, werden durch interaktives Befragen des Users die RAS-relevanten Daten festgestellt. Dieses Vorgehen ist der zweiten Möglichkeit auf jeden Fall vorzuziehen.
Achtung: meist hat man **die Wahl zwischen NDIS- und RAS-Installation**. Wählt man zunächst NDIS und entscheidet sich dann doch zu einem späteren Zeitpunkt für RAS, müssen in den meisten Fällen die Treiber deinstalliert und anschließend wieder ohne NDIS installiert werden, bevor man RAS erfolgreich installieren kann.

b) durch Installation über Windows NT:

Dieser Weg ist nur dann erfolgreich, wenn **nicht bereits der NDIS-Dienst installiert** wurde. Falls dies der Fall ist, muß zunächst der NDIS-Dienst wieder entfernt werden. Folgende Schritte sind für die Installation von RAS durch Windows NT (4.0) durchzuführen:

1. In der **Systemsteuerung** das Symbol '**Netzwerk**' auswählen

2. Im neu geöffneten Fenster die **Karteikarte 'Dienste'** wählen
   → hier das Button '**Hinzufügen**' betätigen
   → in der nun angezeigten Liste '**RAS-Dienst**' auswählen und ggf. die Windows NT CD in das CD-ROM Laufwerk legen

3. Der **RAS-Setup wird gestartet** und es wird nach RAS-fähigen Geräten gesucht
   → nach Beendigung der Suchfunktion werden die **RAS-fähigen Geräte** (hier also die ISDN-Karte) **angezeigt**
   → das entsprechende **Gerät auswählen**

4. Das Button '**Konfigurieren**' betätigen und in dem neu geöffneten Fenster zwischen den Optionen '**eingehende Anrufe**', '**ausgehende Anrufe**' **und 'ein- und ausgehende Anrufe zulassen**' wählen (im Testfeld wurde die zuletzt aufgeführte Option gewählt)

5. Im **RAS-Setup-Fenster** das Button '**Netzwerke...**' betätigen:
   → in dem neu geöffneten Fenster können Einstellungen bezüglich der Remote-Clients getätigt werden (z.B. welches Protokoll die Remote-Clients benützen dürfen, welche IP-Adresse sie zugewiesen bekommen, etc.)
   folgende Einstellungen wurden im Testfeld benützt:
   - Remote Clients zulassen mit TCP/IP und NetBEUI
   - bei TCP/IP-Konfiguration wurde der 'statische Adressenpool' mit folgenden Werten eingesetzt:
     Anfangsadresse: 192.168.207.99 Endadresse: 192.168.207.101

6. Mit Bestätigen der durchgeführten Einstellungen ist die **Installation beendet**

7. Um den RAS-Dienst auf **Server-Seite** zu **starten**, muß lediglich in der 'Start'-Leiste der Menüpunkt '**RAS-Verwaltung**' ausgewählt werden
   → in dem neu geöffneten Fenster können weitere Einstellungen bezüglich der Remote-Clients durchgeführt werden (z.B. welche Remote-Clients den Dienst nützen dürfen, zu welcher Uhrzeit, etc.)
   → in der Menüleiste des Fensters den Menü-Punkt '**RAS starten**' wählen

8. Auf **Client-Seite** (die Installation von RAS ist äquivalent durchzuführen) in der Startleiste unter den Menüpunkt '**Zubehör**' das '**DFÜ-Netzwerk**' wählen
   → in dem neugeöffneten Fenster die Telefonnummer des Hosts angeben, auf dem der RAS-Server läuft (hier: 382-40048)
   → daraufhin erfolgt die Anmeldung mit Kennung und Passwort wie gewohnt

Ursachen für eine nicht erfolgreiche Installation:

- Falls Windows NT **keine RAS-fähigen Geräte** findet, obwohl ein RAS-fähiges und funktionstüchtiges Gerät eingebaut/angeschlossen ist, wurde dieses Gerät bei der Treiber-Installation dem NDIS-Dienst zugewiesen
  → NDIS-Dienst entfernen und daraufhin RAS installieren
  → falls dies nicht hilft, müssen die Treiber deinstalliert und anschließend wieder mit dem RAS-Dienst installiert werden

3.7.3 Vor- und Nachteile

Vorteile	Nachteile
- RAS in Windows NT 4.0 standardmäßig enthalten - Remote-Client hat Zugriff auf Netz wie eine LAN-Komponente vor Ort	- RAS-Server bricht automatisch die Verbindung ab, falls für einen gewissen Zeitraum kein Kommunikationsverkehr stattfindet → dieses Zeitintervall ist leider nicht verstellbar

3.7.4 Übersichtstabelle der wichtigsten Daten

Dienstname	RAS
Installationszeit	ca. 15 min.
Dokumentation	Online-Dokumentation
Getestete Plattformen	Windows NT 4.0 Server und Workstation

3.8 Secure Shell (SSH) V1.2.20

Der Funktionsumfang von **Secure Shell (SSH) V1.2.20** besteht aus dem gesicherten Login auf einem Remote-Host, dem sicheren Ausführen von Kommandos auf einem entfernten Host und dem sicheren Kopieren von Dateien zwischen zwei Hosts. SSH ist also in erster Linie **als sicherer Ersatz** für die UNIX-Dienste **rlogin, rsh und rcp** konzipiert worden. Durch Verwendung eines RSA-IDEA-Hybridverfahrens zur Authentifizierung und Verschlüsselung der Kommunikation werden Sicherheitslücken wie z.B. IP-, Routing- und DNS-Spoofing geschlossen. Mit SSH ist auch die gezielte Vergabe von Zugriffsrechten möglich, das die Sicherheit eines Netzes weiter erhöht. Als besondere Zugabe besitzt SSH eine automatische Umleitung der Grafikausgabe, d.h. einerseits, daß die Grafikumleitung auch verschlüsselt wird und andererseits, daß der User die DISPLAY-Variable nicht mehr zu setzen braucht (für mehr Informationen über die Features und Befehle von SSH verweise ich auf die SSH-Dokumentation und Manual-Pages).
SSH steht für **UNIX-Systeme** als **nicht-kommerzielle Version** zur Verfügung, während es für **Windows-Systeme** (3.x, NT, 95) nur als **kommerzielle Version** angeboten wird (siehe auch Kapitel 3.9 dieser Ausarbeitung).
Installiert und getestet wurde SSH auf den Plattformen HP-UX10.X und IBM AIX 4.2.
Im BMW-Testfeld wurde SSH auf der LED und fi22tf1 installiert.
Im CIP-Pool des Instituts für Informatik der LMU München wurde SSH auf neptun, juno, minerva, merkur und mars installiert.

3.8.1 Voraussetzungen für eine erfolgreiche Installation

Für die Installation auf UNIX-Plattformen müssen folgende Voraussetzungen erfüllt sein:
- ein ANSI-C-Compiler muß bereits auf dem zu installierenden Host vorhanden sein (z.B. gcc oder cc)
- ausreichend Plattenspeicher (ansonsten bricht der Kompiliervorgang ohne Kommentar ab)
- globale (netzwerkweite) Installation muß unter ‘root’-Kennung erfolgen

3.8.2 Installationsschritte

SSH besteht aus zwei Teilen, die auf jedem Host, der SSH nützen will, vorhanden sein müssen:

1. **sshd**: SSH-Daemon (entspricht dem Server)
2. **ssh**: SSH-Klient (entspricht dem rlogin, rsh auf Client-Seite)

Die nachfolgenden Schritte beschreiben eine **globale Installation auf einem Host** unter ‘**root**’-Kennung:

1. Download-Adresse: **ftp://ftp.cert.dfn.de/pub/tools/net/ssh**

2. Filename: ‘**ssh.1.2.20.tar.gz**’

3. In ein temporäres Verzeichnis verschieben, dort **entpacken und extrahieren**
   → das Verzeichnis '**ssh1-2-20**' entsteht, indem sich u.a. folgende Files befinden:
   - '**README**' und '**INSTALL**' (enthalten Informationen über die Installation)
   - der **komplette source-code**

4. In das Verzeichnis '*[TEMP]*/ssh1-2-20' wechseln

5. **'./configure'** eingeben
   → es wird ein mit Autoconf generiertes configure-Skript erstellt (u.a. wird das Betriebssytem, der Compiler, etc. festgestellt), das dann die weitere Installationsroutine beeinflußt

   folgende **Standard-Konfigurationsoptionen** sind möglich:

--prefix=PREFIX	Pfad zur location, in dem die Files installiert werden (default: Unterverzeichnisse von /usr/local)
--exec_prefix=PREFIX	Pfad zur location, in dem die executable Files installiert werden (default: wie bei prefix)
--srcdir=DIR	Source-Code befindet sich im Verzeichnis DIR (default: Verzeichnis in dem sich 'configure' befindet)

   Aufrufbsp.:
   ./configure --prefix=/usr/stud/radisic --exec_prefix=/usr/stud/radisic/bin

   Dies heißt insbesondere, daß bei der Eingabe von **'./configure' alleine**, die Files '**ssh**' und '**sshd**' in '**/usr/local/bin**' und die **Manualpages** in '**usr/local/man/man1**' installiert werden. Die Nutzung der Konfigurationsoptionen sind vor allem für die lokale Installation gedacht.
   Zusätzlich gibt es weitere spezielle Optionen (wie z.B. Ausschalten bestimmter Verschlüsselungsvarianten) die im File '*[TEMP]*/ssh-1-2-20/INSTALL' beschrieben sind.

6. **'make'** aufrufen → es werden die executable files 'ssh' und 'sshd' erstellt (dies kann bis zu 35 min. dauern)

7. **'make install'** aufrufen → SSH wird endgültig auf dem host installiert:

   - 'ssh' wird in das Verzeichnis **'$exec_prefix/bin'** verschoben
   - 'sshd' wird im source-code-Verzeichnis erstellt
   - Die **Konfigurationsfiles '/etc/sshd_config'** und **'/etc/ssh_config'** werden angelegt (für Informationen über Einstellungsmöglichkeiten verweise ich auf die Manual-Pages; **Beispiele** für die beiden Konfigurationsfiles sind im **Anhang B** zu finden)
   - Die Hostschlüssel werden generiert und in **'/etc/ssh_host_key'** und **'/etc/ssh_host_key.pub'** abgelegt
   - Manual-Pages werden generiert und an entsprechender Stelle abgelegt

8. Damit ist die Installation für diesen Host beendet

9. Falls in der Netzwerkumgebung ein **shared binary Verzeichnis** vorliegt, so genügt die Eingabe des folgenden Befehls auf jedem Host im Netzwerk, um SSH dort zu installieren: **'make hostinstall'**
   ansonsten bleibt nur die Möglichkeit, die vorhergehenden Schritte zu wiederholen, um SSH auf jedem Host im Netzwerk zu installieren (siehe auch das Unterkaptitel „Installation im CIP-Pool" dieses Kapitels).

10. Mit **'*[INST_PATH]*/sshd'** (hier: **'usr/local/bin/sshd'**) wird der Daemon gestartet
    → es empfiehlt sich, den Daemon 'sshd' beim Starten des Betriebssystems zu aktivieren, z.B. durch Eintragen von

```
if [ -x /usr/local/bin/sshd ]; then
  /usr/local/bin/sshd && echo sshd
fi
```

    in eine geignete 'rc'-Datei (z.B: in '/etc/rc.local' oder '/etc/rc')

11. Um die Public-Host-Keys aller Maschinen, auf denen SSH installiert wurde, in einer Liste zur Verfügung zu stellen, folgenden Befehl eingeben:
    'make-ssh-known-host' → die Liste wird in der Datei '/etc/ssh_known_hosts' abgelegt
    dieses File kann man mit der 'rhosts'-Datei von rlogin vergleichen

Ersetzen von rlogin und rsh

SSH ist grundsätzlich dafür entwickelt worden, um rlogin und rsh vollständig zu ersetzen. Bei der Installation von SSH bietet sich die Möglichkeit, SSH mit den Namen rlogin und rsh zu installieren. In diesem Fall wird, wann immer die Möglichkeit besteht eine gesicherte Verbindung zu einem remote host aufzubauen, SSH ausgeführt und ansonsten das normale rlogin bzw. rsh (mit einer eingeblendeten Warnmeldung, daß die Kommunikation unverschlüsselt verläuft).
Folgendermaßen wird rlogin und rsh ersetzt:

1. Zunächst '**rlogin**' und '**rsh**' aus dem Verzeichnis '**/usr/bin**' in ein anderes Verzeichnis (z.B. '**/usr/lib/rsh**') verschieben
   → dies ist notwendig, da SSH default-mäßig in das Verzeichnis '/usr/bin' installiert wird und SSH symbolic links zu diesen Files legt
   → wird SSH in ein anderes Verzeichnis installiert, so braucht dieser Schritt nicht durchgeführt zu werden

2. '**./configure --with-rsh=*RSH-PATH* --programm-transform-name='s/^s/r/'** ' aufrufen
   - *RSH-PATH* : kompletter Pfad zu rsh und rlogin (hier: /usr/lib/rsh)

3. Ab hier erfolgt die Installation wie oben beschrieben

Problemlösungen

Bei auftretenden Problemen empfiehlt es sich den Server mit der '-d' Option zu starten und den Client mit der '-v' Option aufzurufen:

1. sshd -d (Debug-Modus)
2. ssh -v

Daraufhin werden alle Schritte die von Server und Client durchgeführt werden, im Ausgabefenster mitprotokolliert.

Administrationstools

- **ssh-keygen**: erzeugt RSA-Schlüssel (sowohl für Host, als auch für User für die RSA-Authentifizierung)

- **ssh-agent**: verwaltet private RSA-Keys
  → Challenges des Servers werden entschlüsselt
  → vereinfacht die Authentifizierung

- **ssh_add**: registriert neue Schlüssel beim ssh-agent

- **make-ssh-known-hosts**: erstellt eine Liste mit bekannten Public Host Keys

Mehr Informationen zu diesen Befehlen sind in den Manual-Pages zu finden.

Verfahren zur Authentifizierung des Klienten

Die Secure Shell bietet vier **verschiedene Verfahren zur Identifizierung des Klienten** an, die jeweils durch **Einträge in den Konfigurationsfiles '/etc/sshd_config'** und **'/etc/ssh_config'** explizit zugelassen oder unterdrückt werden können. Die (zugelassenen) Verfahren werden nacheinander bis zum ersten Erfolg bzw. bis zum endgültigen Mißerfolg versucht:

1. **Authentifizierung** auf der Basis von '**/etc/hosts.equiv**' bzw. '**.rhosts**'. Dieses Vefahren entspricht der Authentifizierung bei 'rlogin' und 'rsh'.
   → sehr unsicher und deswegen sollte dies ausgeschlossen werden

2. Kombination von Verfahren 1 mit **RSA-basierter Host-Authentifizierung**. Die bekannten öffentlichen Schlüssel der Klienten befinden sich auf dem Server-Host in den Dateien '**/etc/ssh_known_hosts**' und '**$HOME/.ssh/ssh_known_hosts**'. In einem Challenge-Response-Dialog weist der Klient die Kenntnis des privaten Schlüssels nach und damit implizit seine Identität. Dieses Verfahren ist **nur dann möglich**, wenn auf **beiden hosts** der Serverprozeß '**sshd**' **gestartet** wurde.
   → der **private Schlüssel** eines Hosts wird im File '**/etc/ssh_host_key**' gespeichert
   → '**/etc/ssh_host_key**' darf deswegen nur von **root lesbar** sein!

3. Reine **RSA-Authentifizierung des Users**. Dies ist das sicherste Verfahren, da der User nur seinen eigenen Schlüsseln vertraut. Um dieses Verfahren anwenden zu können, muß sich der User zunächst mit '**ssh-keygen**' sein eigenes RSA-Schlüsselpaar generieren (wird im file '**$HOME/.ssh/identity**' abgespeichert). Danach muß er den Public-Key auf dem Server-Host in die Datei '**$HOME/.ssh/authorized_keys**' eintragen. Wieder wird durch einen Challenge-Response-Dialog die Kenntnis des privaten Schlüssels des Users, und damit seine Identität, nachgewiesen.

4. Authentifizierung des Klienten über das **UNIX-Userpasswort**. Die Übertragung des Passworts zwischen Klienten und Server erfolgt allerdings verschlüsselt.

Installation im CIP-Pool

Im folgenden werden die Schritte der Installation von SSH im CIP-Pool des Instituts für Informatik der LMU München beschrieben. Die Dienste rlogin, rsh und rcp sollen dabei durch SSH ersetzt werden.

Die Installation im CIP-Pool gliedert sich in die folgenden drei Schritte:
a) Installation auf dem ersten Host
b) Erstellung eines Shell-Scripts
c) Installation auf den restlichen Rechnern des CIP-Pools mit Hilfe des Shell-Scripts

a) Installation auf dem ersten Host (neptun)

1. login als '**root**' auf 'neptun'

2. '**ssh.1.2.20.tar.gz**' in temporäres Verzeichnis '**/tmp/SSH/**' verschoben und dort entpackt
   → source-code befindet sich im neuentstandenen Verzeichnis '**/tmp/SSH/ssh-1.2.20/**'

3. Anlegen eines neuen Verzeichnisses in '**/usr/local/bin/**' mit dem Namen '**remote**'

4. rlogin, rsh und rcp werden nach '**/usr/local/bin/remote/**' verschoben:
   - mv /usr/bin/rlogin /usr/local/bin/remote/rlogin
   - mv /usr/bin/rsh /usr/local/bin/remote/rsh
   - mv /usr/bin/rcp /usr/local/bin/remote/rcp

5. In das Verzeichnis '**/tmp/SSH/ssh-1.2.20**' gewechselt und folgender Befehl eingegeben:
   ./configure --with-rsh=/usr/local/bin/remote/rsh
   --program-transform-name='s/^s/r'

6. Folgenden Befehl eingegeben: **make all**
   → der komplette Source-Code wird kompiliert

7. Folgenden Befehl eingegeben: **make install**
   - damit wurden die folgenden Files im Verzeichnis **/usr/local/bin** generiert: ssh, ssh-add, ssh-agent, ssh-keygen, scp, make-ssh-known-hosts
   - außerdem wurden im Verzeichnis **/usr/local/bin** folgende Dateien **neu angelegt**, die die folgenden symbolic links zu den gerade generierten Files ssh und scp aufwiesen:
     rlogin → ssh; rsh →ssh; rcp → scp; slogin → ssh

8. Zusätzlich wurden **symbolic links** vom Verzeichnis **/usr/bin/** auf die Files ssh und scp gelegt, da nicht jeder CIP-Pool User das Verzeichnis /usr/local/bin im Suchpfad hat:
   ln -s /usr/local/bin/ssh /usr/bin/rlogin
   ln -s /usr/local/bin/ssh /usr/bin/rsh
   ln -s /usr/local/bin/ssh /usr/bin/rcp

9. sshd wird nach /usr/local/bin kopiert:
   cp /tmp/SSH/ssh-1.2.20/sshd /usr/local/bin/sshd

Damit wird jedesmal wenn ein User rlogin, rsh oder rcp aufruft, in Wirklichkeit ssh bzw. scp aufgerufen. SSH überprüft anschließend, ob auf dem remote host auch SSH installiert ist. Falls nicht, wird das entsprechende ‘unsichere’ Programm im Verzeichnis **/usr/local/bin/remote,** nach dem Anzeigen einer Warnmeldung, aufgerufen.

b) Erstellung eines Shell-scripts

1. Zunächst das Verzeichnis **/tmp/SSH/ssh-1.2.20** mit ‘tar’ zusammenfassen:
   tar -cvf ssh-pack.tar ssh-1.2.20
   → damit wurde der komplette Source-Code zusammengefaßt und zusätzlich wurden die ‘config’-Einstellungen für ‘make install’ übernommen
   → d.h. daß für jeden Host im CIP-Pool noch folgende Schritte durchgeführt werden müssen:
   - ‘ssh-pack.tar’ auf Host kopieren
   - ‘ssh-pack’ mit ‘tar -xvf ssh-pack.tar’ extrahieren
   - ‘make install’ aufrufen
   - Entsprechende symbolic links legen

2. Das Shell-Script hat also folgende Einträge:

```
mkdir /usr/local/bin/remote

mv usr/bin/rlogin  /usr/local/bin/remote/rlogin
mv usr/bin/rsh  /usr/local/bin/remote/rsh
mv usr/bin/rcp  /usr/local/bin/remote/rcp

tar -xvf /tmp/SSH/ssh-pack.tar
```

```
/tmp/SSH/ssh-1.2.20/make install

ln -s /usr/local/bin/ssh  /usr/bin/rlogin
ln -s /usr/local/bin/ssh  /usr/bin/rsh
ln -s /usr/local/bin/scp  /usr/bin/rcp
```

c) Installation auf den restlichen Hosts im CIP-Pool

Die Installation auf den übrigen Hosts könnte folgendermaßen durchgeführt werden:

- Mit '**rdist**' das Shell-Skript und 'ssh-pack.tar' auf jeden host im CIP-Pool kopieren
- Das Shell-Skript ausführen

Diese Vorgehensweise wurde im CIP-Pool auf den Rechnern neptun, juno, minerva, merkur und mars getestet.

Probleme:

- Das Perl-Skript '**make-ssh-known-hosts**' war auf HPUX10.20 **nicht ausführbar**
  → d.h. daß die öffentlichen Host-Schlüssel (in **'/etc/ssh_host_key.pub'**) der Rechner im CIP-Pool eigenhändig in einer Datei zusammengefaßt und auf jeden Host nach '**/etc/ssh_known_hosts**' kopiert werden müssen, damit die RSA-basierte Authentifizierung durchführbar ist.

3.8.3 Vor- und Nachteile

Vorteile	Nachteile
- kann ohne weiteres rlogin und rsh ersetzen - automatische Umleitung von X.11 - für User kann SSH transparent gemacht werden → kein Lernaufwand für den User durch gewohnte Umgebung - für UNIX-Systeme ist SSH kostenfrei	- für MS Windows Systeme sind nur kommerzielle Versionen von SSH vorhanden

3.8.4 Übersichtstabelle der wichtigsten Daten

URL-Adresse	ftp://ftp.cert.dfn.de/pub/tools/ssh
Filename	ssh.1.2.20.tar.gz
Filegröße	963 KB
Minimum-Plattenspeicher	20 MB
Installationszeit (pro Host)	ca. 60 min.
starten des Servers	*[INST_PATH]*/bin/sshd
Dokumentation	- Manual-Pages (werden mitinstal-liert) - *[TEMP]*/ssh-1-2-20/INSTALL - *[TEMP]*/ssh-1-2-20/README
Getestete Plattformen	IBM AIX 4.2, HP-UX 10.X

Die im CIP-Pool verwendeten **Konfigurationseinstellungen** für SSH (‘/etc/ssh_config’ und ‘/etc/sshd_config’) sind im **Anhang B** zu finden.

3.9 F-Secure SSH Trial V1.0 (SSH-Client für Windows)

F-Secure SSH Trial V1.0 ist ein (kommerzieller) **SSH-Client für Windows NT** mit eigener Oberfläche. D.h. dem User wird mit diesem Software-Produkt die Möglichkeit gegeben, sich von einem Windows NT host aus auf eine UNIX-Maschine mit SSH-Server ferneinzuloggen. Allerdings ist die Graphikumleitung hier natürlich nicht möglich. Installiert wurde F-Secure SSH Trial V1.0 auf einer Windows NT 4.0 Server Plattform. Im BMW Testfeld wurde das Software-Produkt auf fi22tf3 und fi22tf4 installiert. Beim Testen diente der LED als SSH-Server-Host.

3.9.1 Voraussetzungen für eine erfolgreiche Installation

Es sind keine besonderen Voraussetzungen für die Installation auf Windows NT 4.0 bekannt.

3.9.2 Installationsschritte

1. Download-URL-Adresse: **http://www.datafellows.com**

2. Filename: '**sshwin10.zip**'

3. File in ein temporäres Verzeichnis **verschieben und dort entpacken**
   → u.a. entsteht das file '**setup.exe**'

4. In der Startleiste das Button '**Ausführen...**' betätigen
   → in dem neu geöffneten Fenster das **o.g. file mit kompletten Pfad** angeben und bestätigen
   → **Setup-Wizard** wird gestartet

5. Nach der Installation einiger files wird der '**SSH Wizard Key Generation**' gestartet
   → an dieser Stelle sind folgende Angaben zu machen:

   IDENTITY: (default-wert übernehmen)
   Comment: (hier am besten den Hostnamen eintragen) hier z.B.: fi22tf3
   Passphrase: (an dieser Stelle ein Passwort für das zu generierende Schlüsselpaar angeben)
   RSA-Key-length: (an dieser Stelle die Bitlänge des Schlüsselpaares angeben) hier: 2048

6. Nach Bestätigung der Angaben, werden die Primzahlen für die Schlüsselgenerierung errechnet
   → Achtung: je länger der Schlüssel, desto länger dauert die Generierung (fi22tf4 (486) benötigte über eine Stunde, um das Schlüsselpaar zu generieren)

7. Die Installation ist damit beendet

Bedienungshinweise

Nach dem Aufrufen von F-Secure hat der User auch hier die Möglichkeit anzugeben, nach welchem **Authentifikationsverfahren** der SSH-Server angesprochen werden soll. Falls die RSA-Authentifikation gewünscht wird, muß der eigene Public-Schlüssel in die Datei '**$HOME/.ssh/authorized_keys**' des Servers kopiert werden. Der **Public-Schlüssel** ist in der Menüleiste '**Edit—Properties—RSA-Identity**' zu finden und kann von dort aus in eine Datei kopiert werden.

3.9.3 Vor- und Nachteile

Vorteile	Nachteile
- rlogin auf einen UNIX-Rechner von einem Windows-Host aus mit SSH	- keine graphische Umleitung - nur kommerzielle Version vorhanden

3.9.4 Übersichtstabelle der wichtigsten Daten

URL-Adresse	http://www.datafellows.com
Filename	sshwin10.zip
Filegröße	590 KB
Minimum-Plattenspeicher	5 MB
Installationszeit (pro Host)	ca. 45 min. (von Schlüssellänge abhängig)
Dokumentation	Online-Hilfe
Getestete Plattformen	Windows NT 4.0 Server

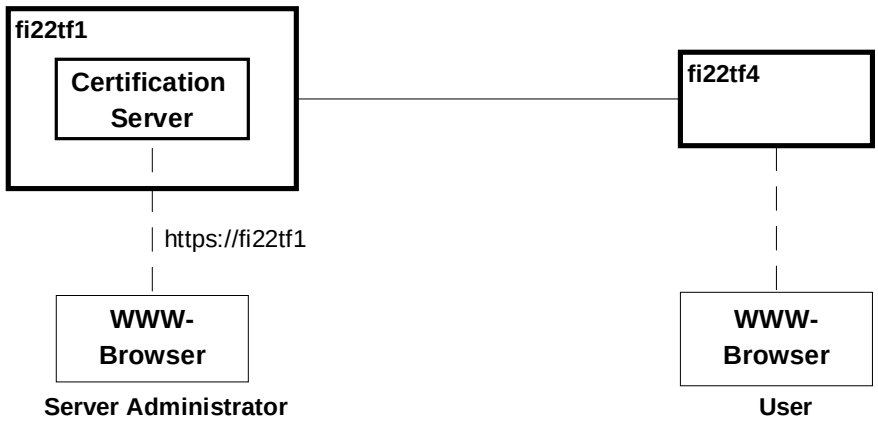
ANHANG

A. Zertifikatsausstellung mit dem Netscape CS V1.01

Im folgenden wird die Prozedur von der Beantragung bis zur Ausstellung eines Zertifikats mit dem Netscape Certificate Server V1.01 beschrieben. Es wird eine Beispielsitzung anhand des BMW-Testfeldes erläutert.

i) Beschreibung der Teilnehmer

Ein **User**, der eine Zertifikat beantragen will, sitzt im Testfeld am **Host fi22tf4**. Zur Beantragung des Zertifikats benötigt er einen WWW-Browser, wobei in diesem Fall der Netscape Navigator 4.0 PR 4 verwendet wird.
Zu den Aufgaben des **Server Administrators** gehört es, eingehende Zertifikatsbeantragungen zu prüfen und ggf. Zertifikate an die Antragsteller auszustellen. Weiterhin gehört zum Aufgabengebiet des Server Administrators die Verwaltung der Zertifikate, d.h. u.a. daß nicht mehr gültige Zertifikate als solche markiert werden. Der Server Administrator sitzt am **Host fi22tf1** und verwaltet den installierten CS ebenfalls über einen WWW-Browser (Netscape Navigator 3.01). Im Testfeld spricht der Administrator den CS über die URL https://fi22tf1 an, und meldet sich dort als ‘privileged’ an.



ii) Beantragung eines Zertifikats

1. User spricht den **CS** über den Browser an: URL: **https://fi22tf1**

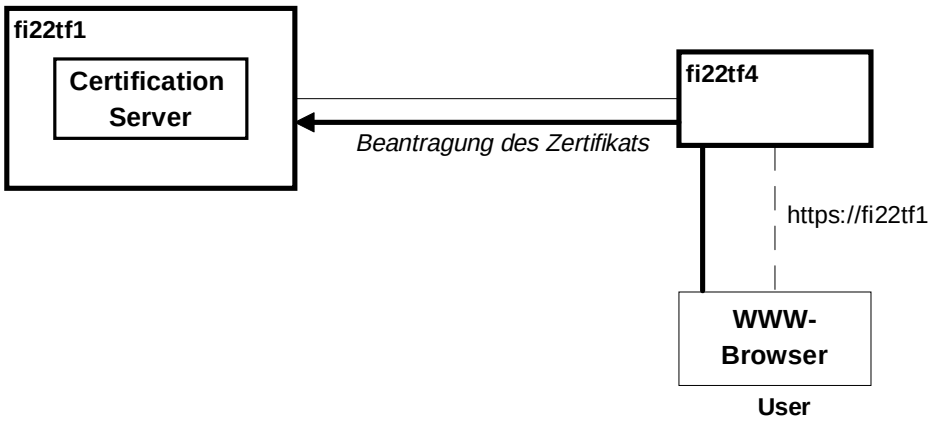
2. Nach Aufbau der ersten Seite des CS, meldet sich der User als ‘**public**’ an

3. Jetzt bietet sich dem User im Menü die Möglichkeit ein Zertifikat zu beantragen: Button ‘**Get a Certificate**’ im Seitenmenü betätigen

4. Folgende Angaben sind vom User nun zu machen:
   Your Name: (der Name des Users) z.B. Hans Mueller
   Organization Unit: (der Name der Abteilung) z.B. FI-22

Organization: (Name der Firma, Gesellschaft, etc.) z.B. BMW
Country: z.B. DE
E-mail: (e-mail-Adresse des Users) z.B. muellerha@foo.bmw.de
additional comments: (an dieser Stelle ist unbedingt noch einmal die e-mail-Adresse des Antragstellers zu wiederholen, falls noch die Version 1.0 des CS eingesetzt wird; hierbei handelt es sich um einen Bug der Vorgängerversion)
E = muellerha@foo.bmw.de

→ mit dem Betätigen des Buttons '**submit request**' wird der Antrag abgeschickt



iii) Prüfen des Antrags

Der **Administrator** muß alle neu eingegangenen Anträge prüfen.

1. Button '**List Certification Signing Requests**' betätigen

2. Button '**Run Query**' betätigen
   → es werden alle neuen Anträge aufgelistet

3. Zertifikat, das der Administrator prüfen will, auswählen
   → drücken der entsprechenden Nummer

4. Da durchaus mehrere Administratoren Zertifikate prüfen und ausstellen können, muß zunächst ein Zertifikat einem Administrator zugewiesen werden
   → Button '**Assign to me**' betätigen

5. Neues Frame mit allen vom Antragsteller gemachten Angaben, erscheint
   → an dieser Stelle prüft der Administrator die Angaben z.B. durch Anruf in der entsprechenden Abteilung des Antragstellers (es muß auf jeden Fall sichergestellt werden, daß es sich tatsächlich um die Person handelt, die den Antrag gestellt hat)

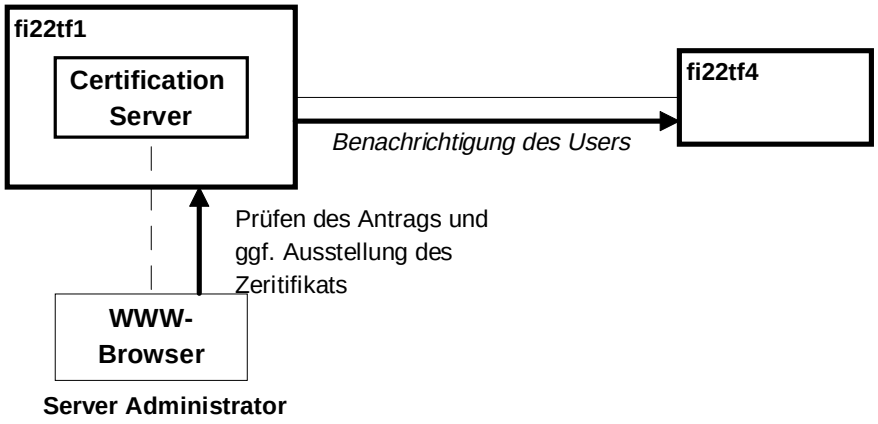
6. Falls NCS V1.0 eingesetzt wird, muß aufgrund des **Bugs dieser Version des CS**, folgender Schritt durchgeführt werden:
   bei '**Subject Name**' hinter dem letzten Eintrag folgende Zeile eingeben:
   E = <e-mail-Adresse des Antragsstellers> hier:
   E = muellerha@foo.bmw.de

7. Button '**Include Subject Key Identifier**' betätigen

8. An der Stelle '**Select an Operation to perform on this Request**' folgendes auswählen: '**Issue this Certificate**'

9. Zum Schluß das Button '**Perform this Operation**' drücken

10. In dem nun neu aufgebautem Frame hat der Administrator die Möglichkeit den Antragsteller per e-mail über die Ausstellung des Zertifikats zu benachrichtigen



iv) Importieren des Zertifikats

Der **Antragsteller** wird über die Auststellung des Zertifikats **benachrichtigt** und muß dieses in seinen Browser importieren.

1. **e-mail des CS-Administrators** kommt an
   → e-mail **enthält Hyperlink** zum Zertifikat (nur bei Mail-Clients, die Hyperlinks unterstützen, z.B. Netscape Messanger)

2. Durch **Betätigen des Hyperlinks** wird im Netscape Navigator die WWW-Seite, die das Zertifikat enthält, geöffnet

3. Am Ende der Seite befindet sich der Hyperlink '**Importing this Certificate to a Navigator**'
   → durch Betätigen dieses Hyperlinks wird das Zertifikat in den Browser importiert

4. Während des Importierens wird ein neues Fenster mit Informationen über das ausgestellte Zertifikat und die CA geöffnet
   → diese Information durch Betätigen des Buttons '**OK**' bestätigen

5. Im darauffolgenden Fenster **unbedingt** das Button '**Save As**' drücken
   → damit wird das Zertifikat abgespeichert und man kann es in einen anderen Browser importieren

6. Zum Schluß erfolgt eine Meldung über das **erfolgreiche Importieren** dieses Zertifikats

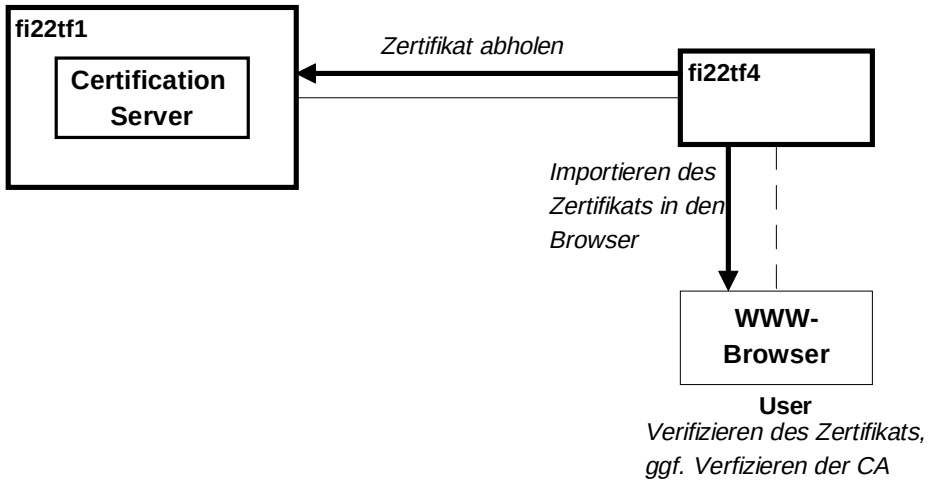
7. Das gerade **importierte Zertifikat** muß anschließend noch vom User **verifiziert werden**
   → den '**Security**'-Button des Netscape Navigators oder Messangers drücken
   → im neugeöffneten Fenster den '**Certificates / Yours**'-Link betätigen
   → das eigene Zertifikat auswählen und zur Verifizierung '**verify**' drücken
   → ein neugeöffnetes Fenster bestätigt die **erfolgreiche Verifizierung** des Zertifikats

   Falls die **Verifizierung als nicht erfolgreich** bestätigt wird, so wurde die CA noch nicht verifiziert:
   - '**Security**'-Fenster durch drücken des **'Security'-Buttons** öffnen
   - Den '**Certificates / Signers**'-Link betätigen
   - Die entsprechende CA auswählen (hier: CA-FI - fi22tf1.foo.bmw)
   - '**Edit**'-Button drücken

   → im neugeöffneten Fenster folgende Kästchen ankreuzen:
   '**Accept this Certification Authority for certifying network sites**' und
   '**Accept this CA for certifying e-mail users**'
   → diese Angaben mit '**OK**' bestätigen

   Unbedingt auch den Punkt 'Troubleshooting' (S.57) beachten!

   - Die Schritte zur Verifizierung des Zertifikats wiederholen



v) Signieren und Verschlüsseln von e-mails

1. Netscape Messanger aufrufen

2. '**New Message**' Button drücken

3. Adresse, etc. wie gewohnt ausfüllen

4. Neben den Zeilen für die Adressen befinden sich **drei kleine Buttons**
   → das letzte Button drücken ('**Sending Options**')
   → in der nun erweiterten Darstellung kann der User folgende Kästchen ankreuzen:
   '**Signed**' : die Nachricht wird signiert
   '**Encrypted**' : die Nachricht wird mit dem Public-Schlüssel des Empfängers verschlüsselt
   → diese Einstellung ist natürlich nur dann sinnvoll, wenn man im Besitz des Public-Schlüssels des Empfängers ist
   der **eigene Public-Key** wird in der 'Signed'-Einstellung **dem Empfänger mitgeschickt**, so daß dieser zukünftige Nachrichten an den User verschlüsseln kann

5. Nachdem die Nachricht verfaßt wurde, den **'Send'-Button** betätigen
   → falls eine eingestellte Option nicht möglich ist (z.B. 'Encrypted' wegen fehlendem Schlüssel des Empfängers), so wird dies durch eine Fehlermeldung angezeigt

Troubleshooting

Bei Testläufen mit dem CS im Rechnerraum des Lehrstuhls war es **nicht möglich** die **eingerichtete CA** in dem derzeit aktuellen **Netscape Communicator V4.03** unter 'Security/Certificates/Signers' zu **verifizieren**, da diese in der dargebotenen Liste nicht angezeigt wurde.

Lösung:
- Die CS-Seite aufrufen (hier: https://hpheger9)
- Im linken 'public'-Seitenmenü befindet sich der Hyperlink '**Accept this Authority in your navigator**' → diesen Hyperlink betätigen
- Die nun folgenden Optionen nach eigenem Ermessen bestätigen und ggf. verändern
- Unter 'Security/Certificates/Signers' des eigenen Navigators wird die CA nun angezeigt undf kann verifiziert werden

B. Beispieldateien für die SSH-Konfiguration

Im folgenden sind die im CIP-Pool verwendeten SSH-Konfigurationfiles notiert.

Konfigurationsangaben für den SSH-Client in '/etc/ssh_config':

```
# This is ssh client systemwide configuration file.  This file provides
# defaults for users, and the values can be changed in per-user configuration
# files or on the command line.

# Configuration data is parsed as follows:
#  1. command line options
#  2. user-specific file
#  3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for various options

# Host *
# ForwardAgent yes
# ForwardX11 yes
RhostsAuthentication yes
RhostsRSAAuthentication yes
RSAAuthentication yes
# TISAuthentication no
PasswordAuthentication yes
FallBackToRsh yes
# UseRsh no
# BatchMode no
# StrictHostKeyChecking no
IdentityFile ~/.ssh/identity
#   Port 22
#   Cipher idea
#   EscapeChar ~
```

Konfigurationsangaben für den SSH-Daemon in '/etc/sshd_config':

```
# This is ssh server systemwide configuration file.

Port 22
ListenAddress 0.0.0.0
HostKey /etc/ssh_host_key
RandomSeed /etc/ssh_random_seed
ServerKeyBits 768
LoginGraceTime 600
KeyRegenerationInterval 3600
PermitRootLogin yes
IgnoreRhosts yes
StrictModes yes
QuietMode no
X11Forwarding yes
X11DisplayOffset 10
FascistLogging no
PrintMotd yes
KeepAlive yes
SyslogFacility DAEMON
RhostsAuthentication no
RhostsRSAAuthentication yes
TISAuthentication no
RSAAuthentication yes
PasswordAuthentication yes
PermitEmptyPasswords yes
# UseLogin no
# PidFile /u/zappa/.ssh/pid
# AllowHosts *.our.com friend.other.com
# DenyHosts lowsecurity.theirs.com *.evil.org evil.org
# Umask 022
# SilentDeny on
```

Für die **RhostRSAAuthentication** muß auf beiden Hosts der Daemon 'sshd' laufen!

C. Boot-Dateien des CS auf der hpheger9

Zu den Aufgaben der Installation des Netscape Certificate Servers V1.01 am Lehrstuhl gehörte auch, den CS und die dafür notwendigen Prozesse jeweils beim Booten des hosts hpheger9 automatisch zu starten.
Im folgenden Kapitel sind die am Bootvorgang des CS-hosts hpheger9 beteiligten Dateien dokumentiert.

1. /sbin/rc3.d/S980nssvr:
Die Datei ‘/sbin/rc3.d/S980nssvr’ enthält einen symbolic link auf die Datei ‘/sbin/init.d/netscape-servers’.

2. /sbin/init.d/netscape-servers:

```
#!/sbin/sh

# start/stop Informix daemon for CS

# Allowed exit values:
#    0 = success; causes "OK" to show up in checklist.
#    1 = failure; causes "FAIL" to show up in checklist.
#    2 = skip; causes "N/A" to show up in the checklist.
#           Use this value if execution of this script is overridden
#        by the use of a control variable, or if this script is not
#        appropriate to execute for some other reason.
#       3 = reboot; causes the system to be rebooted after execution.
# Input and output:
#    stdin is redirected from /dev/null
#
#    stdout and stderr are redirected to the /etc/rc.log file
#    during checklist mode, or to the console in raw mode.

PATH=/usr/sbin:/usr/bin:/sbin:/usr/local/sbin:/soft/bin
export PATH

# NOTE: If your script executes in run state 0 or state 1, then /usr might
#    not be available.  Do not attempt to access commands or files in
#    /usr unless your script executes in run state 2 or greater.  Other
#    file systems typically not mounted until run state 2 include /var
#    and /opt.

rval=0

# Check the exit value of a command run by this script.  If non-zero, the
# exit code is echoed to the log file and the return value of this script
# is set to indicate failure.

set_return() {
     x=$?
     if [ $x -ne 0 ]; then
           echo "EXIT CODE: $x"
           rval=1 # script FAILed
     fi
}

case $1 in
'start_msg')
     # Emit a _short_ message relating to running this script with
     # the "start" argument; this message appears as part of the checklist.
     echo "Starting the Netscape Servers"
     ;;

'stop_msg')
     # Emit a _short_ message relating to running this script with
     # the "stop" argument; this message appears as part of the checklist.
     echo "Stopping the Netscape Servers"
     ;;
```

```
'start')
     # source the system configuration variables
     if [ -f /etc/rc.config ] ; then
           . /etc/rc.config
     else
           echo "ERROR: /etc/rc.config defaults file MISSING"
     fi

     # Execute the commands to start your subsystem
     # Check if starting the Informix daemon was successful
     if sh /usr/informix/start.sh
     then
           if /usr/ns-home/start-admin
           then
                 if /usr/ns-home/slapd-hpheger9/start
                 then
                       /usr/ns-home/cms-hpheger9/cms_start /usr/ns-home/cms-
                       hpheger9/start /etc/cms.txt
                       if [ "$?" != 0 ]
                       then
                             rval=1
                       fi
                 else
                       rval=1
                 fi
           else
                 rval=1
           fi
     else
           rval=1
     fi

     tail /usr/informix/online.log
     ;;

'stop')
     # source the system configuration variables
     if [ -f /etc/rc.config ] ; then
           . /etc/rc.config
     else
           echo "ERROR: /etc/rc.config defaults file MISSING"
     fi

     # Execute the commands to stop your subsystem

     /usr/ns-home/cms-hpheger9/stop
     /usr/ns-home/slapd-hpheger9/stop
     /usr/ns-home/stop-admin
     sh /usr/informix/stop.sh
     #fi

     tail /usr/informix/online.log
     ;;

*)
     echo "usage: $0 {start|stop|start_msg|stop_msg}"
     rval=1
     ;;
esac

exit $rval
```

3. /usr/ns-home/cms-hpheger9/cms-start:
Bevor der CS tatsächlich gestartet wird, müssen normalerweise beim Aufruf von '/usr/ns-home/cms-hpheger9/start' die Passwörter für die CS-Schlüssel und die CS-Datenbank in der Shell eingegeben werden. Um dies zu automatisieren, wird in '/sbin/init.d/netscape-servers' stattdessen das Skript '/usr/ns-home/cms-hpheger9/cms-start' aufgerufen, der die Schlüssel aus einem File einliest und an die CS-Startdatei weitergibt. Folgendermaßen wird 'cms-start' aufgerufen:

cms-start *file1 file2*

file1: der vollständige Pfad der tatsächlichen CS-Startdatei
hier: /usr/ns-home/cms-hpheger9/start
file2: der vollständige Pfad der Datei, die die Passwörter enthält
hier: /etc/cms.txt (in diesem Fall haben alle Schlüssel dasselbe Paßwort)

cms-start:

```
#!/soft/bin/expect -f

# open cms-password-file and read pwd (second argument)
set x [open [lrange $argv 1 1] "r"]
set pw [read $x]
close

# set location of the real (original) CS-start file (first argument)
set x [lrange $argv 0 0]
send_user "x: $x\n"
spawn [lrange $argv 0 0]

# send the pwd read from the cms-password-file
expect {
     "Password for Certificate Signing Key File:" {
           send "$pw\r"
     }
}
expect {
     "Password for Certificate Server Database:" {
           send "$pw\r"
     }
}
expect {
     "Key File Password:" {
           send "$pw\r"
           exp_continue
     }
     "listening.*\n" {
           send_user "$argv0: succeeded.\n"
     }
     eof {
           send_user "$argv0: failed.\n"
     }
}

wait
```

D. Abkürzungsverzeichnis

AS	Administration Server
CA	Certification Authortiy
CS	Certificate Server
DNS	Domain Name System
DS	Directory Server
HTTP	Hypertext-Transfer-Protocol
[INST_PATH]	Pfad zum Verzeichnis, in dem die Software letztendlich installiert wurde
IP	Internet Protocol
ISDN	Integrated Services Digital Network
MIME	Multipurpose Internet Mail Extension
NCS	Netscape Certificate Server
PGP	Pretty Good Privacy
RAS	Remote Access Service
RFC	Request for Comment
TCP	Transmission Control Protocol
[TEMP]	temporäres Verzeichnis, in dem das gepackte File als erstes verschoben und schließlich entpackt worden ist
SSH	Secure Shell
SSL	Secure Socket Layer

Literaturverzeichnis

[Eckert1]
Eckert, C.: Betriebssysteme. Skript zur gleichnamigen Vorlesung, Institut für Informatik, LMU München, 1997

[Eckert2]
Eckert, C.; Geiger, J.: Sichere Rechensysteme. Skript zum gleichnamigen Praktikum, Institut für Informatik, TU München, 1997

[Quarterman94]
Quarterman, John S.; Carl-Mitchell, S.: The Internet Connection: System Connectivity and Configuration. Addison-Wesley, 1994

[Reis97]
Reiser, Helmut: Sichere TCP/IP-basierte Kommunikation bei der BMW AG. Diplomarbeit am Lehrstuhl Prof. Dr. Hegering des Instituts für Informatik, LMU München, 1997

[Rose93]
Rose, Mitchell T.: The Internet Message: Closing the Book with Electronic Mail. PTR Prentic Hall, 1993

[Scheller94]
Scheller, M.; Boden, K.-P.; Geenen, A.; Kompermann, J.: Internet: Werkzeuge und Dienste. Springer Verlag Berlin Heidelberg, 1994

RFC:

[RFC 1521]
Borenstein, N.; Freed, N.: RFC 1521: MIME (Multipurpose Internet Mail Extensions) part one. September 1993

[RFC 1522]
Moore, K.: RFC 1522: MIME (Multipurpose Internet Mail Extensions) part two. September 1993

[RFC 1847]
Galvin, J.; Murphy, S.; Crocker, S.; Murphy, S.: RFC 1847: Security multiparts for MIME: Multipart/signed and multipart/encrypted. October 1995

RFCs können als .txt-file im Leo-Archiv u.a. per Stichwörter gesucht werden:
URL: http://www.leo.org/

Internet Drafts:

[S/MIME]
Dusse, S.: S/MIME Message Specification: PKCS Security Services for MIME, <draft-dusse-mime-msg-spec-00.txt>. Internet Draft, NWG, September 1996

[SSL]
Freier, Alan O.; Karlton, Philip; Kocher, Paul C.: The SSL Protocol Version 3.0. Internet Draft, March 1996

URLs:

PGP:
MIT Distribution site for PGP: http://web.mit.edu/network/pgp.html

SSH:
Secure Shell, Gesicherte Kommunikation über unsichere Netze:
http://www1.tu-chemnitz.de/~hot/ssh/ssh.html

Secure Shell Page der University of Florida:
http://www.cis.ufl.edu./help-system/ssh/

SSH Page: http://www.cs.hut.fi/ssh/

SSL:
Netscape Standards: http://home.netscape.com/newsref/std/

SSLeay ftp-Archiv Mainz: ftp.uni-mainz.de/pub/internet/security/ssl/SSL

The SSLP Reference Implentation Project (Bristol):
http://www.cs.bris.ac.uk/~bradley/Documents/project2.html

Saga of Developing a free SSL: http://petrified.cic.net/~altitude/ssl/

S/MIME:
S/MIME by RSA: http://www.rsa.com/rsa/S-MIME/

S/MIME: Anatomy of a Secure E-mail Standard:
http://www.ema.org./html/pubs/mmv2n4/s-mime.htm

S/MIME Implementation Guide:
ftp://ftp.rsa.com/pub/S-MIME/IMPGV2.txt