

Datenkommunikation, Internet und Multimedia

Kapitel 11 Anwendungen 1 - Telnet, E-Mail, FTP, IRC, News

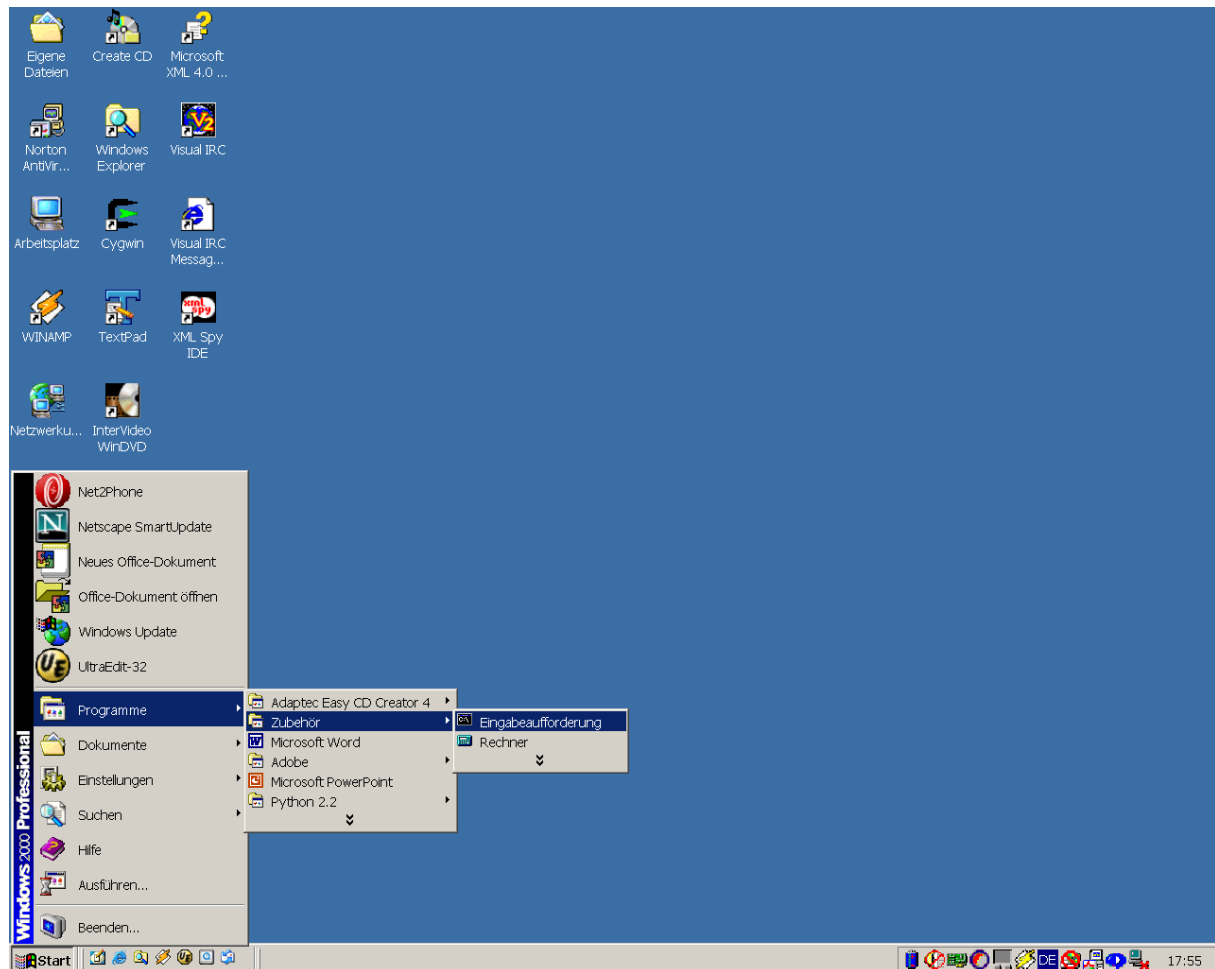
Prof. Dr. Nik Klever, Fachhochschule Augsburg

Einführung

Wir werden uns heute mit einigen grundlegenden Protokollen, nämlich Email, FTP, IRC, Telnet und News beschäftigen. Das nächste Mal werfen wir einen Blick auf WWW und XML und im dritten Teil der Anwendungen geht's dann um Multimedia. Wir beschäftigen uns zwar wieder mit Protokollen, die wir schon in den letzten Folgen kennengelernt haben, aber noch nicht im Detail. Jede Anwendung wie Email, FTP, IRC, Telnet und News benötigt für die Anwendungsschicht ein eigenes Protokoll und diese Protokolle schauen wir uns heute mal an.

Telnet

Beginnen wir nun mit dem Telnet-Protokoll, als einem der grundlegenden Protokolle der TCP/IP-Familie. Mit dem Telnet-Protokoll können wir den Ablauf von verschiedenen anderen Text-basierten Protokollen Schritt für Schritt nachvollziehen. Was bedeutet denn Telnet eigentlich und wozu wurde es entwickelt ? Telnet wurde entwickelt um ein Konsolen-Fenster von einem entfernten Rechner auf den eigenen Rechner zu übertragen. Früher gab es ja noch keine Windows-Oberflächen, die gesamte Steuerung und die Eingabe in Programme erfolgte über die Konsole . Konsole bedeutet dabei ein Gerät, welches Text über eine Tastatur eingeben und auch nur Text über einen Drucker oder später dann Bildschirm ausgeben kann. Wer heutzutage Windows-Betriebssysteme benutzt, kann über die Eingabeaufforderung ebenfalls ein Konsolen-Fenster starten. Jede Eingabe und Ausgabe in diesem Konsolen-Fenster ist dabei nur textorientiert. Das Konsolen-Fenster ist sicherlich ein Überbleibsel aus der DOS-Welt. Aber stellen Sie sich mal vor, Sie sind Außendienstmitarbeiter, Ihre Maus geht kaputt und Sie müssen aber unbedingt noch die letzten Aufträge zur Zentrale übertragen. Also starten Sie über Tastaturbefehle die Eingabeaufforderung (STRG-ESC -> Programme -> Zubehör -> Eingabeaufforderung)



und dann hierin das Filetransferprogramm ftp. Heutzutage wird zwar statt ftp das sicherere scp (secure copy) verwendet, aber vom Grundsatz her geht es genauso. Es gibt auch andere Konsolen-Fenster, die nicht standardmäßig in Windows-Betriebssystemen enthalten sind. Zum Beispiel wird mit dem Programmsystem Cygwin ein komplettes Unix-Betriebssystem unter Windows simuliert und dieses enthält ebenfalls ein Konsolen-Fenster. In einem Konsolen-Fenster kann man jetzt das Telnet-Programm starten und sich damit zu einem entfernten Rechner verbinden lassen. Erst dann erhält man über das Telnet-Protokoll ein Konsolen-Fenster des entfernten Rechners. Wenn wir jetzt Telnet auf den entfernten Rechner in der Fachhochschule Augsburg gestartet haben, kommt das Ergebnis bereits von dort, anschliessend müssen wir uns nur noch einloggen und dann haben wir ein Konsolen-Fenster vom entfernten Rechner hier bei uns. Und damit können wir jedes beliebige Programm starten oder beenden, Texte editieren, - alles was eben textorientiert möglich ist. Bilder oder Videos können mit Telnet nur rudimentär angeschaut werden. Eigentlich nur dann, wenn man den Zeichenvorrat der Terminals ausnutzt und ASCII-Animationen erstellt, also Animationen im American Standard Code for Information Interchange.

E-Mail

Kommen wir nun zur nächsten Anwendung für heute, zu den Emails. Neben dem Browser die wohl zweitwichtigste Anwendung des Internets. Ich denke, fast jeder Zuschauer hat schon mal selbst eine Email geschrieben oder erhalten. Die Vorzüge wissen heute viele zu schätzen und daher ist die Email als Kommunikationsmittel inzwischen fast so populär wie das Telefon, - aber leider nicht ganz so einfach zu erklären. Wie ein Telefon funktioniert weiss heute fast jeder. Aber wie funktioniert es nun, dass eine Email vom Absender zum richtigen Empfänger kommt ? Beim Telefon reicht es eben, die physikalischen Eigenschaften in groben Zügen zu erklären um seine Wirkungsweise verstehen zu können. Für den Austausch von Emails müssen wir jedoch nicht nur die physikalischen Eigenschaften des Netzwerks, die in der ersten Schicht abgehandelt werden, erklären, sondern auch noch die Protokolle aller darüberliegenden Schichten und natürlich das Schichtenmodell selbst. Das meiste haben wir ja schon in den vergangenen Sendungen gehört, sodass wir uns heute wirklich nur auf die Protokolle der Anwendungsschicht konzentrieren können. Um per email kommunizieren zu können brauchen wir nämlich gleich mehrere Protokolle. Die bekanntesten und wichtigsten davon sind das **SMTP – Simple Mail Transfer Protocol** und das **POP – Post Office Protocol** oder das **IMAP – Internet Message Access Protocol**. Wenn wir uns eine Email mal genauer anschauen, dann hat ja jede einen Absender und einen oder mehrere Empfänger, die über die Email-Adressen angesprochen werden.

Store-and-Forward

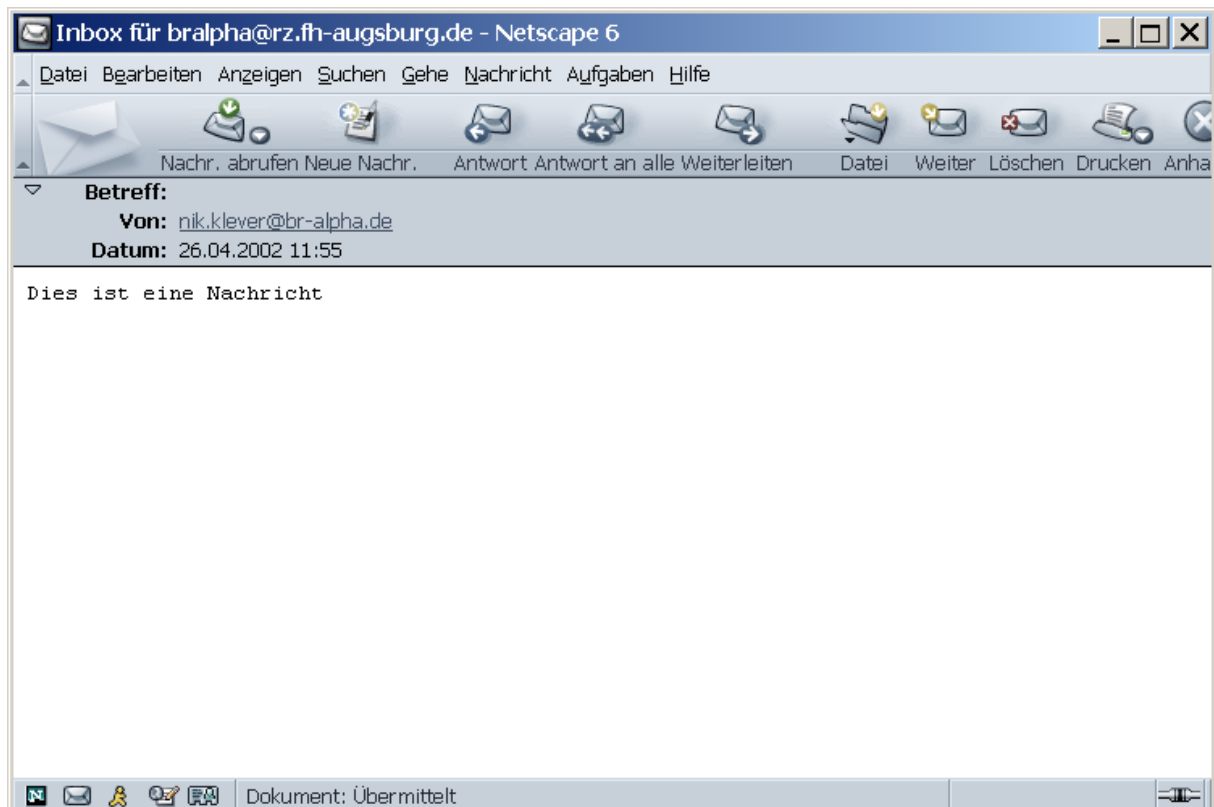
Beispielsweise werden Emails - im Gegensatz zu den Informationen, die man sich über den Browser holt - nicht generell vom Empfänger zum Sender geschickt, sondern im Store-and-forward- Modus versendet. Dies bedeutet, dass die Nachricht nicht von Absenderrechner direkt zum Empfängerrechner gelangt, sondern in Verteilerzentren (sog. Mail-Relays) gespeichert (also englisch Store) und von dort jeweils an das nächste Verteilerzentrum weitergeleitet wird - englisch forward -so erklärt sich also der Begriff Store-and-Forward. Soweit ist also noch alles vergleichbar mit der ganz normalen Briefpost. Aber schauen wir uns die Struktur der Email-Architektur noch auf weitere Ähnlichkeiten zur normalen Brief-Post hin an.

Auf dieser Folie sehen wir die Benutzer, dann die uns allen bekannten Mail-Programme, hier MUA, Mail User Agent genannt sowie die Mail-

Verteilungszentren, hier als MTA, Mail Transfer Agent bezeichnet. Auf der gegenüberliegenden Seite wieder ein MTA sowie ein MS, ein Message Store, ein Speicherplatz für Emails. Oft wird dieser auch Mailbox, also Briefkasten genannt. Wird eine Mail von einem Benutzer zum anderen geschickt, geht der Weg über den Mail User Agent zum Mail Transfer Agent, weiter zum nächsten MTA, dann zum übernächsten usw. - eben bis hin zum letzten MTA, an dem der Empfänger angeschlossen ist. Dabei wird für jede Verbindung das SMTP Protokoll verwendet. Im letzten MTA wird die Email dann an den Message Store des Empfängers ausgeliefert. Der Empfänger schaut dann mit seinem Mail User Agent immer wieder nach, ob in seiner Mailbox, eine Email vorhanden ist und holt sich diese dann mittels des POP- oder IMAP-Protokolls auf seinen eigenen Rechner ab.

SMTP

Wir fangen damit an, dass wir eine Telnet-Verbindung mit einem Mailrechner aufbauen und zwar zu Port 25, dem SMTP-Port. Mit diesem Telnet-Aufbau simulieren wir jetzt ein Mailprogramm wie zum Beispiel Microsoft's Outlook oder Netscape's eMail Programm. Wir geben Telnet, den Rechnernamen und 25 als Portnummer ein. Dann meldet sich das entfernte SMTP-Programm mit einer Statuszeile. Jetzt müssen wir uns mit HELO und dem Rechnernamen anmelden. Anschliessend geben wir MAIL FROM: und unsere Mailadresse und bei der nächsten Eingabe RCPT TO: die Empfänger-Mailadresse ein. Jedesmal wenn wir einen Befehl eingegeben haben, antwortet das entfernte System mit einer entsprechenden Statuszeile. Danach kommt noch das Schlüsselwort DATA und jetzt können wir unsere Nachricht eingeben. Mit einem Punkt in der ersten Spalte beenden wir die Nachricht. Als letztes geben wir noch QUIT ein und beenden somit die Verbindung. Und wie sieht diese Nachricht jetzt beim Empfänger aus ? Das schauen wir uns gleich mal an: klicken also unser Mailprogramm an, in unserem Fall Netscape , klicken auf "Nachrichten abholen" und haben hier das Ergebnis: eine sehr einfach aussehende Nachricht insbesondere fällt auf, dass kein Betreff-Feld und kein Empfänger-Feld vorhanden ist.



Ein Betreff-Feld haben wir auch nicht eingegeben, aber die Empfängeradresse wurde doch mit dem Befehl RCPT-TO eingegeben? Wie hängt dies zusammen? Dies liegt daran, dass im **SMTP Protokoll** nur an die Mailbox adressiert wird und der eigentliche Inhalt der Email wird noch durch einen anderen RFC beschrieben, nämlich den RFC 2822, dem Internet Message Format. In diesem RFC wird beschrieben, wie der Teil, der nach dem DATA Befehl kommt, aussehen sollte. Hier in dieser Folie ist aufgezeichnet, wie es richtig hätte lauten müssen. Der Aufbau des Textes muss immer in dieser Weise erfolgen, man darf also nicht "Betreff:" statt "Subject:" schreiben, da ansonsten das Mailprogramm des Empfängers damit nichts anfangen kann. Jedes Mailprogramm muss sich an die Konventionen in diesem RFC halten.

POP 3

Das Post Office Protocol haben wir vorhin schon mal gebraucht, als wir "Nachrichten abrufen" angeklickt haben. Schauen wir uns dieses Protokoll jetzt nochmal über Telnet an. Dazu machen wir wieder eine Telnet Verbindung zu unserem Mailserver auf, allerdings diesmal nicht zu dem SMTP Port 25 sondern zu Port 110, also zu dem POP Protokoll zugehörigen Port. Ebenso wie das SMTP Programm meldet sich auch das entfernte POP Programm mit einer Statuszeile, anschliessend müssen wir uns mit dem Befehl USER authentifizieren und mit PASS

das entsprechende Passwort eingeben. Dann erhalten wir mit dem Befehl LIST eine Auflistung aller im Message-Store vorhandenen Nachrichten. Mit RETR und der entsprechenden Nachrichten-Nummer, die im LIST-Befehl angeführt war, können wir uns dann die entsprechende Email herunterladen und mit dem Befehl DELE und der Nachrichten- Nummer auch vom Message Store löschen, was wir aber jetzt nicht tun. Alle email-Programme müssen also mindestens zwei Protokoll-Sprachen sprechen: Das SMTP Simple Mail Transfer Protocol und das eben besprochene POP Protokoll. Das Internet Message Access Protocol, kurz IMAP bildet lediglich eine Alternative zum Post Office Protocol und wird hier jetzt nicht näher angesprochen.

FTP

Kommen wir jetzt also zum **Filetransferprotokoll FTP**. Im Gegensatz zu den bisherigen Anwendungen ist das FTP Protokoll etwas komplizierter, da es zwei TCP-Verbindungen aufbauen muss: eine Verbindung zur Steuerung, in denen die FTP-Kommandos übertragen werden und eine reine Datenverbindung. Dies kann man in etwa vergleichen mit ISDN, dort haben wir auch mit dem D-Kanal einen Steuerungskanal und mit den B-Kanälen die Datenkanäle. Wir können also mit Telnet nur die Steuerungsverbindung aufbauen und keine Dateien bzw. Directorylistings übertragen, sondern nur allgemeine Statusmeldungen. Verwenden wir das normale, im Betriebssystem enthaltene FTP-Programm und bauen eine FTP-Verbindung zum FTP-Server der FH Augsburg auf. Wenn wir keinen eigenen Account auf dem entfernten Rechner besitzen, so ist meistens der Account "anonymous" freigegeben. Als Passwort müssen wir unsere Mailadresse eingeben und dann haben wir Zugang zum öffentlichen Bereich des FTP-Servers vergleichbar mit dem öffentlichen Bereich eines Webservers. Wenn wir jetzt eine Datei von einem bestimmten Ordner herunterladen wollen, müssen wir uns erst auf diesen Ordner setzen mit CD /pub/doc/rfc und anschliessend GET rfc_index.txt eingeben. Jetzt wird die Datei übertragen. Für das FTP-Programm muss man sich allerdings ein paar Kommandos merken, die man aber mit HELP abfragen kann.

IRC

Fahren wir jetzt im heutigen Programm fort und sehen uns eine weitere wichtige und inzwischen auch sehr populäre Anwendung an, nämlich IRC, das Internet Relay Chat oder einfach nur Chat wie z.B. der BR-online Chat.

Benutzen wir erstmal ein IRC-Programm, wie zum Beispiel mIRC, ksirc, ircII, ircToons oder GamersIRC oder irgendein anderes und schauen uns die Anwendung mal an. Wir müssen uns erst einen IRC-Server aussuchen, auf dem wir chatten wollen, zum Beispiel irc.leo.org. Sobald wir diesen Server ausgewählt haben, erscheint dort automatisch die Portnummer, sehen Sie ! Im allgemeinen ist dies 6667. Wenn wir uns nun dorthin verbinden, sind wir automatisch angemeldet, können uns aber noch mit einem Nickname identifizieren. Anschliessend gründen wir einen eigenen Chatchannel indem wir /JOIN bralpha eingeben, um jetzt auch ganz normalen Text als Nachrichten in diesem Chat zu verbreiten. Nun kann sich jeder andere auch an diesem Chat beteiligen. Dies kann man jetzt gleich mal mit Telnet durchspielen und eine Telnet-Verbindung zu irc.leo.org auf Port 6667 aufbauen. Anschliessend geben wir zur Identifikation einen Nickname über das Kommando NICK telnet ein sowie registrieren uns mit dem Befehl USER telnet 0 * :Telnet Chat. Daraufhin erscheint die Anmeldenachricht und wir können uns anschliessend mit dem Kommando JOIN #bralpha an dem Chatchannel von eben beteiligen. Um jetzt eine Nachricht über den Chat zu schicken, müssen wir das Kommando PRIVMSG #bralpha : Dies ist eine Nachricht eingeben. Im Chatfenster von mIRC sehen wir dann, dass der Benutzer mit dem Nickname Telnet "Dies ist eine Nachricht" eingegeben hat.

Was ist denn jetzt eigentlich der Unterschied zum IRC-Programm, dort haben wir doch auch den Befehl JOIN eingeben müssen ? Dort war jedoch die Anmeldung nicht ganz so kompliziert oder eine ganz normale Nachricht muss eben immer mit PRIVMSG CHATNAME :NACHRICHT eingegeben werden. Die Kommandos müssen eben genau dem IRC-Protokoll der RFCs 2810-2813, welches wir in dieser Folie auszugsweise sehen, entsprechen; ansonsten gibt es eine Fehlermeldung. In der Telnet-Simulation müssen wir uns also immer strikt an das Protokoll halten, während in den entsprechenden Programmen eine für die Benutzer verständliche Oberfläche um die Protokolle herumprogrammiert worden sind. Mit Telnet können die meisten Anwendungsprotokolle – so wie sie standardisiert sind – manuell, also Schritt-für-Schritt simuliert werden.

News

Bei News handelt es sich um Neuigkeiten, Nachrichten, die quasi an einem elektronischen schwarzen Brett veröffentlicht werden. Dort stehen verschiedene Rubriken, die wiederum in weitere Unterrubriken ziemlich tief verschachtelt sind, zur Verfügung. So ist zum z.B. comp.lang.cpp die Rubrik, die sich mit allen Neuigkeiten und Fragen rund um die

Computersprache C++ beschäftigt. Dabei steht *comp* für die Computerrubrik und *lang* für die Unterrubrik Language also Sprache und *cpp* dann für die Unterrubrik C++. Hier kann nun jeder eine Nachricht oder Frage einstellen, die dann von allen, die ebenfalls diese Rubrik abonniert haben, gelesen werden kann. Sind Email-Listen nicht das Gleiche? Fast – bei einer Email-Liste muss man eingetragen sein, das Lesen von Newsgroup-Rubriken erfolgt anonym. Eine Email-Liste verschickt die Nachrichten an alle in der Liste eingetragenen Mitglieder als Email während eine News-Nachricht auf einen News-Server eingestellt wird, der die Nachricht dann an die News-Server weiterleitet, die ebenfalls diese Rubrik verteilen. Der dritte Unterschied liegt in der Aufbewahrung der Nachrichten, bei einer Email-Liste liegt dies im Ermessen des Empfängers, bei den Newsgroups werden die Nachrichten auf dem News-Server nur für eine festgesetzte Zeit vorgehalten und anschliessend automatisch gelöscht. Im Endergebnis – also der Verteilung von Nachrichten an eine grosse Menge von Nutzern – ist jedoch kein Unterschied vorhanden. Deswegen ist eine Entscheidung für oder gegen Newsgroups bzw. für oder gegen Email-Listen mehr eine emotionale denn eine rationale.