

Miami_deutsch

COLLABORATORS

	<i>TITLE :</i> Miami_deutsch		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		February 12, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Miami_deutsch	1
1.1	Miami_deutsch.guide	1
1.2	Miami_deutsch.guide/NODE_DISCLAIMER	2
1.3	Miami_deutsch.guide/NODE_CONDITIONS	4
1.4	Miami_deutsch.guide/NODE_REGISTRATION	6
1.5	Miami_deutsch.guide/NODE_INTRODUCTION	7
1.6	Miami_deutsch.guide/NODE_REQUIREMENTS	8
1.7	Miami_deutsch.guide/NODE_INSTALLATION	9
1.8	Miami_deutsch.guide/NODE_MIAMIINIT	10
1.9	Miami_deutsch.guide/NODE_TOOLTYPES	11
1.10	Miami_deutsch.guide/NODE_MENU	12
1.11	Miami_deutsch.guide/NODE_CONFIGURATION	13
1.12	Miami_deutsch.guide/NODE_GUI_GENERAL	14
1.13	Miami_deutsch.guide/NODE_GUI_GENERAL_REGISTER	14
1.14	Miami_deutsch.guide/NODE_GUI_INTERFACE	15
1.15	Miami_deutsch.guide/NODE_GUI_INTERFACE_TYPE	16
1.16	Miami_deutsch.guide/NODE_GUI_INTERFACE_DEVICE	17
1.17	Miami_deutsch.guide/NODE_GUI_INTERFACE_SPEED	17
1.18	Miami_deutsch.guide/NODE_GUI_INTERFACE_CD	18
1.19	Miami_deutsch.guide/NODE_GUI_INTERFACE_PROTOCOL	18
1.20	Miami_deutsch.guide/NODE_GUI_INTERFACE_FLOW	19
1.21	Miami_deutsch.guide/NODE_GUI_INTERFACE_EOF	19
1.22	Miami_deutsch.guide/NODE_GUI_INTERFACE_SERIAL	19
1.23	Miami_deutsch.guide/NODE_GUI_INTERFACE_IP	20
1.24	Miami_deutsch.guide/NODE_GUI_INTERFACE_MASK	20
1.25	Miami_deutsch.guide/NODE_GUI_INTERFACE_GWAY	21
1.26	Miami_deutsch.guide/NODE_GUI_INTERFACE_MULTICASTS	21
1.27	Miami_deutsch.guide/NODE_GUI_INTERFACE_MAPPING	22
1.28	Miami_deutsch.guide/NODE_GUI_INTERFACE_MTU	23
1.29	Miami_deutsch.guide/NODE_GUI_INTERFACE_STP	23

1.30	Miami_deutsch.guide/NODE_GUI_INTERFACE_INACTIVITY	24
1.31	Miami_deutsch.guide/NODE_GUI_PPP	25
1.32	Miami_deutsch.guide/NODE_GUI_PPP_CHAP	25
1.33	Miami_deutsch.guide/NODE_GUI_PPP_CALLBACK	26
1.34	Miami_deutsch.guide/NODE_GUI_PPP_VJC	26
1.35	Miami_deutsch.guide/NODE_GUI_PPP_ACCM	27
1.36	Miami_deutsch.guide/NODE_GUI_PPP_QUICK	27
1.37	Miami_deutsch.guide/NODE_GUI_PPP_ESCAPE	28
1.38	Miami_deutsch.guide/NODE_GUI_PPP_DNSIPCP	28
1.39	Miami_deutsch.guide/NODE_GUI_PPP_TERMREQ	29
1.40	Miami_deutsch.guide/NODE_GUI_DIALER	29
1.41	Miami_deutsch.guide/NODE_GUI_DIALER_SCRIPT	30
1.42	Miami_deutsch.guide/NODE_GUI_DIALER_PHONE	30
1.43	Miami_deutsch.guide/NODE_GUI_DIALER_MAX	31
1.44	Miami_deutsch.guide/NODE_GUI_DIALER_DELAY	31
1.45	Miami_deutsch.guide/NODE_GUI_DIALER_RDELAY	31
1.46	Miami_deutsch.guide/NODE_GUI_DIALER_TEACH	31
1.47	Miami_deutsch.guide/NODE_GUI_DIALER_NAME	32
1.48	Miami_deutsch.guide/NODE_GUI_DIALER_CAPTURE	32
1.49	Miami_deutsch.guide/NODE_GUI_DATABASE	32
1.50	Miami_deutsch.guide/NODE_GUI_DATABASE_PROTOCOLS	34
1.51	Miami_deutsch.guide/NODE_GUI_DATABASE_SERVICES	34
1.52	Miami_deutsch.guide/NODE_GUI_DATABASE_HOSTS	35
1.53	Miami_deutsch.guide/NODE_GUI_DATABASE_NETWORKS	35
1.54	Miami_deutsch.guide/NODE_GUI_DATABASE_DOMAINS	35
1.55	Miami_deutsch.guide/NODE_GUI_DATABASE_DNSSERVERS	36
1.56	Miami_deutsch.guide/NODE_GUI_DATABASE_INETD	36
1.57	Miami_deutsch.guide/NODE_GUI_DATABASE_USERS	37
1.58	Miami_deutsch.guide/NODE_GUI_DATABASE_GROUPS	38
1.59	Miami_deutsch.guide/NODE_GUI_DATABASE_ARP	38
1.60	Miami_deutsch.guide/NODE_GUI_DATABASE SOCKS	38
1.61	Miami_deutsch.guide/NODE_GUI_DATABASE_IPFILTER	40
1.62	Miami_deutsch.guide/NODE_GUI_TCPIP	41
1.63	Miami_deutsch.guide/NODE_GUI_TCPIP_HOSTNAME	42
1.64	Miami_deutsch.guide/NODE_GUI_TCPIP_NAME	42
1.65	Miami_deutsch.guide/NODE_GUI_TCPIP_ICMP	43
1.66	Miami_deutsch.guide/NODE_GUI_TCPIP_BOOTP	43
1.67	Miami_deutsch.guide/NODE_GUI_TCPIP_VERIFYDNS	43
1.68	Miami_deutsch.guide/NODE_GUI_TCPIP_FAKEIP	44

1.69	Miami_deutsch.guide/NODE_GUI_TCPIP_TTCP	44
1.70	Miami_deutsch.guide/NODE_GUI_TCPIP_ADDDOMAIN	44
1.71	Miami_deutsch.guide/NODE_GUI_TCPIP_DOWN	45
1.72	Miami_deutsch.guide/NODE_GUI_TCPIP_PING	45
1.73	Miami_deutsch.guide/NODE_GUI_TCPIP_GETTIME	46
1.74	Miami_deutsch.guide/NODE_GUI_EVENTS	46
1.75	Miami_deutsch.guide/NODE_GUI_MODEM	48
1.76	Miami_deutsch.guide/NODE_GUI_MODEM_INIT	48
1.77	Miami_deutsch.guide/NODE_GUI_MODEM_EXIT	48
1.78	Miami_deutsch.guide/NODE_GUI_MODEM_PREFIX	49
1.79	Miami_deutsch.guide/NODE_GUI_MODEM_SUFFIX	49
1.80	Miami_deutsch.guide/NODE_GUI_MODEM_NULLMODEM	49
1.81	Miami_deutsch.guide/NODE_GUI_LOGGING	50
1.82	Miami_deutsch.guide/NODE_GUI_LOGGING_CONSOLE	50
1.83	Miami_deutsch.guide/NODE_GUI_LOGGING_FILE	50
1.84	Miami_deutsch.guide/NODE_GUI_LOGGING_SYSLOG	51
1.85	Miami_deutsch.guide/NODE_GUI_LOGGING_PHONE	51
1.86	Miami_deutsch.guide/NODE_GUI_LOGGING_PPP	51
1.87	Miami_deutsch.guide/NODE_GUI_WINDOWS	52
1.88	Miami_deutsch.guide/NODE_GUI_WINDOWS_REQQUIT	52
1.89	Miami_deutsch.guide/NODE_GUI_WINDOWS_REQOFFLINE	52
1.90	Miami_deutsch.guide/NODE_GUI_WINDOWS_REQERRORS	53
1.91	Miami_deutsch.guide/NODE_GUI_WINDOWS_DIALER	53
1.92	Miami_deutsch.guide/NODE_GUI_GUI	53
1.93	Miami_deutsch.guide/NODE_GUI_GUI_HOTKEY	54
1.94	Miami_deutsch.guide/NODE_GUI_GUI_SHOWICON	54
1.95	Miami_deutsch.guide/NODE_GUI_GUI_SHOWMENU	55
1.96	Miami_deutsch.guide/NODE_GUI_GUI_ONSTARTUP	55
1.97	Miami_deutsch.guide/NODE_GUI_GUI_ONLINEICON	55
1.98	Miami_deutsch.guide/NODE_GUI_GUI_OFFLINEICON	55
1.99	Miami_deutsch.guide/NODE_GUI_GUI_GUI	55
1.100	Miami_deutsch.guide/NODE_GUI_GUI_SWITCH	56
1.101	Miami_deutsch.guide/NODE_GUI SOCKS	56
1.102	Miami_deutsch.guide/NODE_GUI SOCKS_ENABLE	57
1.103	Miami_deutsch.guide/NODE_GUI SOCKS_SERVER	57
1.104	Miami_deutsch.guide/NODE_GUI SOCKS_MAXLOG	57
1.105	Miami_deutsch.guide/NODE_GUI SOCKS_AUTH	58
1.106	Miami_deutsch.guide/NODE_GUI_MISC	58
1.107	Miami_deutsch.guide/NODE_DIALERLANG	58

1.108Miami_deutsch.guide/NODE_AREXX	60
1.109Miami_deutsch.guide/NODE_ENVVARS	61
1.110Miami_deutsch.guide/NODE_ENVVARS_TZ	63
1.111Miami_deutsch.guide/NODE_EXCONFIG	64
1.112Miami_deutsch.guide/NODE_EXCONFIG_DIST	64
1.113Miami_deutsch.guide/NODE_EXCONFIG_PASSWORDS	68
1.114Miami_deutsch.guide/NODE_EXCONFIG_CLIENTS	69
1.115Miami_deutsch.guide/NODE_UTILITY	70
1.116Miami_deutsch.guide/NODE_UTILITY_ARP	71
1.117Miami_deutsch.guide/NODE_UTILITY_FINGER	71
1.118Miami_deutsch.guide/NODE_UTILITY_IFCONFIG	72
1.119Miami_deutsch.guide/NODE_UTILITY_MAPMBONE	73
1.120Miami_deutsch.guide/NODE_UTILITY_MRINFO	74
1.121Miami_deutsch.guide/NODE_UTILITY_MRROUTED	74
1.122Miami_deutsch.guide/NODE_UTILITY_MTRACE	75
1.123Miami_deutsch.guide/NODE_UTILITY_NETSTAT	77
1.124Miami_deutsch.guide/NODE_UTILITY_PING	81
1.125Miami_deutsch.guide/NODE_UTILITY_REMIND	82
1.126Miami_deutsch.guide/NODE_UTILITY_RESOLVE	84
1.127Miami_deutsch.guide/NODE_UTILITY_ROUTE	85
1.128Miami_deutsch.guide/NODE_UTILITY_SYSCTL	86
1.129Miami_deutsch.guide/NODE_UTILITY_TCPDUMP	89
1.130Miami_deutsch.guide/NODE_UTILITY_TRACEROUTE	92
1.131Miami_deutsch.guide/NODE_COMPATIBILITY	93
1.132Miami_deutsch.guide/NODE_RESTRICTIONS	93
1.133Miami_deutsch.guide/NODE_HISTORY	94
1.134Miami_deutsch.guide/NODE_FUTURE	95
1.135Miami_deutsch.guide/NODE_SUPPORT	95
1.136Miami_deutsch.guide/NODE_ACKNOWLEDGEMENTS	96

Chapter 1

Miami_deutsch

1.1 Miami_deutsch.guide

Miami

Dies ist die Dokumentation für Miami V3.0, ein integriertes TCP/IP-System für AmigaDOS. Copyright (C) 1996-1998 Nordic Global Inc. All rights reserved. Programm und Dokumentation von Holger Kruse.

Haftung

Rechtliche Informationen

Gebrauch / Verbreitung

Bedingungen für Gebrauch und Verbreitung

Registrierung

Shareware-Registrierung

Einführung

Einführung in Miami

Voraussetzungen

Benötigte Hardware und Software

Installation

Installation von Miami

MiamiInit

Einfacher Start mit MiamiInit

ToolTypes

ToolTypes für Miami

Menüs

Programmmenüs

Konfiguration

Manuelle Konfiguration

Wahlmodul-Befehlssprache	Beschreibung des Wahlmoduls
ARexx-Schnittstelle	Unterstützte ARexx-Befehle
Umgebungs-Variablen	Umgebungs-Variablen
Austauschen von Einstellungen	Im-/Exportieren von Einstellungen
Hilfsprogramme	Hilfsprogramme für Miami
Kompatibilität	Kompatibilität von Miami
Einschränkungen	Einschränkungen der aktuellen Version ↔
Entwicklung	Entwicklung von Miami
Die Zukunft	Die Zukunft von Miami
Unterstützung	Hilfe und neue Programmversionen
Danksagungen	Danksagungen

1.2 Miami_deutsch.guide/NODE_DISCLAIMER

Haftung

Miami WURDE ALS TCP/IP-PAKET FÜR AmigaOS ENTWICKELT, UM DEN AMIGA ÜBER MODEM ODER MITTELS EINER SCHNITTSTELLENKARTE MIT DEM INTERNET ZU VERBINDEN. OBWOHL ANSTRENGUNGEN UNTERNOMMEN WORDEN SIND, Miami SO KOMPATIBEL ZUM TCP/IP-STANDARD WIE MÖGLICH ZU MACHEN, KANN ICH TROTZDEM NICHT AUSSCHLIESSEN, DASS Miami FEHLER ENTHÄLT, DIE SCHÄDLICHE NEBENWIRKUNGEN AUF IHR SYSTEM ODER ANDERE ANGESCHLOSSENE RECHNER HABEN.

ICH LEHNE HIERMIT AUSDRÜCKLICH JEGLICHE RECHTLICHE ODER FINANZIELLE VERANTWORTUNG FÜR DIESE ODER ANDERE KONSEQUENZEN AUS DEM GEBRAUCH VON Miami AB. DAS BEINHÄLTET, BESCHRÄNKT SICH ABER NICHT AUF SCHÄDEN AN IHREM GERÄT, IHREN DATEN, ANDEREN MASCHINEN, DIE MIT IHREM AMIGA VERBUNDEN SIND, PERSONENSCHÄDEN, FINANZIELLE SCHÄDEN, ODER

IRGENDWELCHE ANDEREN NEBENWIRKUNGEN.

Miami WIRD IHNEN VON MIR SO ZUR VERFÜGUNG GESTELLT, WIE SIE ES VORGEFUNDEN HABEN, DAS HEISST ICH GARANTIERE NICHT, DASS Miami FÜR IRGEND EINEN BESONDEREN ZWECK GEEIGNET IST, ODER DASS ES NEUERE PROGRAMMVERSIONEN, FEHLERKORREKTUREN, ODER HILFE BEI DER FEHLERSUCHE GEBEN WIRD.

Miami basiert auf dem 4.4BSD V.2 TCP/IP Netzwerk-Code, in der Version, die von Walnut Creek auf CD-ROM vertrieben wird.

Der gesamte 4.4BSD-Code darf frei vertrieben werden, und steht unter Copyright von verschiedenen Autoren. Für Details zu individuellen Copyright- und Vertriebs-Bedingungen sehen Sie bitte in den Quelltexten nach, die von verschiedenen Quellen erhältlich sind, z.B. von dem 4.4BSD Lite-CD-ROM von Walnut Creek.

Der folgende Copyright-Hinweis bezieht sich auf das original-4.4BSD-Software-Paket:

Zitat Anfang

All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by The Regents of the University of California.

Copyright 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994 The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by the University of California, Berkeley and its contributors. 4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Zitat Ende

Bitte beachten Sie, daß sich dieser Copyright-Hinweis NICHT auf das Miami-Programmpaket bezieht. Miami ist NICHT frei verbreitbar, sofern das nicht an anderer Stelle ausdrücklich erlaubt ist. Siehe

Gebrauch / Verbreitung
für Details.

Einige der GUI-Module von Miami benötigen Magic User Interface (MUI). MUI steht unter Copyright von Stefan Stuntz.

Einige der GUI-Module von Miami benötigen die MUI-Custom-Class "Busy.mcc" von Klaus Melchior. Hier ist der dazugehörige Copyright-Hinweis:

Zitat Anfang

Busy.mcc is (c) 1994-1996 by Klaus 'kmel' Melchior

Zitat Ende

Einige der GUI-Module von Miami benötigen gtlayout.library von Olaf Barthel. Hier ist der zugehörige Copyright-Hinweis:

Zitat Anfang

Copyright © 1993-1996 by Olaf 'Olsen' Barthel Freely distributable.

Zitat Ende

Einige der GUI-Module von Miami verwenden Grafiken, die auf MagicWB basieren, und mit Erlaubnis der Autoren verwendet wurden. Hier ist der zugehörige Copyright-Hinweis. Mehr Informationen sind aus der Datei 'MagicWB.readme' ersichtlich, die dem Miami-Archiv beiliegt.

Zitat Anfang

Copyright © 1992-97, Martin Huttenloher

Zitat Ende

1.3 Miami_deutsch.guide/NODE_CONDITIONS

Gebrauch / Verbreitung

Miami ist Shareware. In diesem Fall bedeutet das, daß Sie ein persönliches Keyfile benötigen, um die volle Funktionalität von Miami zu nutzen.

Anwender bekommen ein persönliches Keyfile von mir, nachdem sie sich registriert haben. Das Keyfile darf anderen Anwendern nicht zur Verfügung gestellt werden. Die Weitergabe des Keyfiles an andere Anwender und die Benutzung von Keyfiles, die Sie nicht direkt von mir

zu Ihrem persönlichen Gebrauch bekommen haben, gelten als Software-Piraterie.

Keyfiles sind nicht übertragbar, und dürfen nicht an eine andere Person oder Organisation verkauft oder getauscht werden. Sie sind ausschließlich für die Person bestimmt, die sich registriert hat.

Die ausführbare Programmdatei "Miami" und die Programmdateien von Hilfsprogrammen dürfen in keiner Weise modifiziert oder gepatcht werden (auch nicht zum persönlichen Gebrauch), außer auf eine Art und Weise, die von mir ausdrücklich erlaubt wurde, um auf neue Versionen upzudaten. Der Gebrauch von gepatchten oder modifizierten Programmdateien gilt als Software-Piraterie.

Miami-Programmdateien dürfen nur in der vom Autor beabsichtigten Weise genutzt werden, d.h. sie dürfen nur von Amiga-Computern mit AmigaOS ausgeführt werden. Reassemblen, Reverse-Engineering oder das Übersetzen der Programmdateien sind ausdrücklich verboten.

Die Dokumentation und Programmtexte von Miami unterliegen den gleichen Copyright-Bestimmungen wie das Programm selbst. Das heisst, daß weder die Dokumentation noch die Programmtexte in irgendeiner Weise verändert oder übersetzt werden dürfen.

Um jegliches Mißverständnis zu vermeiden: SIE DÜRFEN die Miami-Programmtexte und -Dokumentation NICHT übersetzen, und die Übersetzungen dann vertreiben, es sei denn, Sie werden von mir ausdrücklich als offizieller Miami-Übersetzer benannt. Unauthorisierte Übersetzungen der Programmtexte oder Dokumentation sind illegal, verletzen mein Copyright, und werden von öffentlichen Software-Sites gelöscht.

Wenn Sie das Miami-Archiv vertreiben möchten, gelten die folgenden Bedingungen:

- * Der Verkaufspreis darf nicht höher sein als die Kosten für die für die Miami-Dateien benötigten Leerdisketten plus einer nominellen Kopiergebühr plus Kosten für Versand. Der Gesamtpreis darf nicht höher sein als 10 US\$ oder 15 DM oder dem entsprechenden Betrag in anderen Währungen.
 - * Wenn das Miami-Archiv als Teil einer CD-ROM-Sammlung von Public Domain- und/oder Shareware-Software vertrieben werden soll, dann darf der Verkaufspreis des CD-ROM 20 US\$, 30 DM oder den entsprechenden Betrag in anderen Währungen nicht überschreiten.
 - * Alle Teile des Programms und der Dokumentation müssen vollständig sein. Der Vertrieb von Teilen oder unvollständigen Archiven ist nicht gestattet. Der Vertrieb von Keyfiles ist nicht gestattet.
 - * Miami oder Teile davon dürfen normalerweise nicht in Verbindung mit oder als Teil von kommerzieller Software vertrieben werden. Auf Anfrage sind Lizenzbedingungen für den kommerziellen Verkauf von Miami von kruse@nordicglobal.com erhältlich. Allerdings sollten Sie nicht annehmen, daß Sie Miami kommerziell vertreiben dürfen, bis Sie von mir eine ausdrückliche, schriftliche Zusage erhalten.
-

- * Programm und Dokumentation dürfen nicht verändert werden.
Ausnahme (d.h. akzeptabel ist): die Verwendung von Archivierprogrammen wie LHA, wenn dabei garantiert ist, daß es möglich ist, die Originaldateien zu extrahieren.

1.4 Miami_deutsch.guide/NODE_REGISTRATION

Registrierung

Wenn Sie Miami häufig nutzen, Funktionen, die in der Demoversion nicht verfügbar sind, benötigen, oder Ihre Verbindung länger als eine Stunde aufrecht erhalten möchten, sollten Sie sich für Miami registrieren.

Um sich zu registrieren, starten Sie bitte das Programm MiamiRegister. Es erklärt die Registrierprozedur im Detail, und erlaubt Ihnen, sich im Dialog zu registrieren.

Wenn Sie aus irgend einem Grund MiamiRegister nicht starten können, kontaktieren Sie mich bitte bei kruse@nordicglobal.com.

Für Registrierungen innerhalb Deutschland wird derzeit der folgende Händler empfohlen:

ADX Datentechnik GmbH
Hallesdorferstr. 119
22179 Hamburg

Tel. 040 - 642 02656

Internet: <http://www.adx.de>
Infos zu Miami: miami@adx.de

Registrierte Anwender von ppp.device erhalten einen Rabatt beim Umstieg auf Miami, wenn Sie sich direkt beim Autor registrieren. Die Details werden in MiamiRegister erklärt.

Sonderangebote für Gruppenregistrierung (10 oder mehr Anwender auf einmal), Vorausbezahlung von Lizenzen und kommerziellen Wiederverkauf sind ebenfalls möglich. Kontaktieren Sie bitte kruse@nordicglobal.com für mehr Details.

Anwender, die sich bereits für Miami 2.x registriert haben, benötigen für Miami 3.0 neue Keyfiles. Die Upgraderegulung ist:

- * Wenn Sie Ihre Miami-Keyfiles nach dem 15. Juni 1997 erhalten haben, dann ist Ihr Upgrade auf Miami 3.0 kostenlos.
 - * Wenn Sie Ihre Miami-Keyfiles vor dem 15. Juni 1997 erhalten haben, benötigen Sie einen Upgrade-Code, um Ihre Keyfiles upzugraden. Upgrade-Codes sind von Nordic Global Inc. (für US\$ 12.00) erhältlich, und von einigen Registriersites.
-

Um Ihre Keyfiles upzugrade, starten Sie bitte MiamiRegister. Das Programm bietet Ihnen Upgradeoptionen an, und wird Ihnen auch mitteilen, ob für Sie beim Upgrade eine Gebühr entsteht.

1.5 Miami_deutsch.guide/NODE_INTRODUCTION

Einführung

Miami ist ein integriertes TCP/IP-System für AmigaOS, mit dem Sie auf sehr einfache Weise auf das Internet oder ein lokales Netzwerk zugreifen können, sowohl per Modem als auch per Schnittstellenkarte (z.B. Ethernet).

Miami basiert auf der aktuellen Version (4.4BSD V2) des offiziellen BSD-Netzwerk-Codes, plus einiger Erweiterungen Dritter (wie z.B. FreeBSD-T/TCP- und Path-MTU-Discovery-Code). Das heisst Miami ist ein "echter" und vollständiger TCP/IP-Stack, nicht eine Emulation, die nur Teile des TCP/IP-Standards unterstützt.

Die Programmierschnittstelle von Miami ist kompatibel mit der von AmiTCP 4.x verwendeten (bsdsocket.library), d.h. die meisten für AmiTCP 4.x geschriebenen und compilierten Programme funktionieren ohne Änderung und ohne neu-compilieren mit Miami.

Ausserdem enthält Miami ein eingebautes Wählmodul, das sowohl per Skript gesteuert, als auch interaktiv genutzt werden kann, eine Implementierung der (C)SLIP- und PPP-Protokolle, eine Schnittstelle für SANA-II-Treiber, eine graphische Benutzeroberfläche für Programmsteuerung und -konfiguration, die Client-seitige Implementierung des SOCKS-Protokolls für Proxy-Server, und viele andere Funktionen.

Miami hat zusätzlich eine eingebaute Implementierung von inetd, dem "Internet-Super-Server", mit diversen eingebauten Diensten einschließlich "fingerd" und "identd", eine eingebaute Implementierung von TCP:, dem AmigaDOS-Stream-Handler für TCP/IP, und eine eingebaute Implementierung der usergroup.library, dem Interface zur Verwaltung von Anwendern und Anwendergruppen.

Im Gegensatz zu anderen Mehrzweckstacks enthält Miami eine umfangreiche Unterstützung für Modem-basierte Internet-Anwahl. Die Konfiguration dafür ist so einfach wie möglich gehalten: die meisten Konfigurationsparameter werden von Miami automatisch ermittelt. Ausserdem unterstützt Miami die Verwendung von vorkonfigurierten Einstellungen für bestimmte Internet-Provider. Miami kann auch mit anderen Verbindungen außer Modem-Verbindungen genutzt werden, z.B. mit einer Ethernet-Karte, einer Arcnet-Karte oder einem Kabelmodem.

Miami unterstützt verschiedene GUI-Module für die Konfiguration. Wenn Sie Miami bedienen (z.B. online oder offline gehen, oder Einstellungen verändern), muß ein GUI-Modul geladen sein. Sobald Miami online ist, ist es möglich, daß GUI-Module aus dem Speicher zu entfernen, um Speicherplatz zu sparen. Sie können das GUI-Modul

jederzeit wieder laden, wenn Sie Änderungen an Ihren Einstellungen vornehmen möchten.

Miami unterstützt derzeit die folgenden GUI-Module:

MUI

Dieses Modul benötigt MUI (Magic User Interface) 3.8 oder höher, und erzeugt eine Benutzerschnittstelle im typischen MUI-Stil.

MUIMWB

Dieses Modul ist identisch zum 'MUI'-Modul, hat aber Grafiken an einigen Stellen, nicht nur Text, und ist sorgfältiger designed worden. Es benötigt MUI 3.8 und einen Bildschirm mit mindestens 8 Farben mit der MagicWB-Farbpalette.

GTLayout

Dieses Modul generiert eine Benutzeroberfläche basierend auf GadTools, und benötigt Olaf Barthel's gtlayout.library Version 40 oder höher. Es benötigt MUI nicht.

Das empfohlene GUI-Modul ist 'MUI' oder (für Bildschirme mit vielen Farben) 'MUIMWB'. Das GTLayout-Modul kann auch verwendet werden, aber einige der Funktionen der MUI-Module (z.B. drag&drop-Sortieren von Datenbank-Einträgen) sind damit nicht verfügbar.

Vor dem Start sollten Sie sich zuerst

```
MiamiInit
```

```
. MiamiInit ist ein
```

Programm, das Miami für die meisten Anwender automatisch konfiguriert, einschließlich Wählskript, Passwort-Abfrage, IP-Adresse, DNS-Server, Netzmaske und allen anderen Konfigurationsparametern.

Nachdem Sie MiamiInit ausgeführt haben, sollten Sie Miami starten, die Konfiguration importieren, die Einstellungen speichern, und die Verbindung zu Ihrem Provider aufbauen.

1.6 Miami_deutsch.guide/NODE_REQUIREMENTS

Voraussetzungen

Um Miami zu benutzen, brauchen Sie:

- * einen Amiga mit OS 2.04 oder höher
- * MUI 3.8 oder höher, wenn Sie die MUI-Module verwenden wollen, oder alternativ gtlayout.library V40 oder höher für die GTLayout-Module.

Sie brauchen auch irgendwelche Hardware für die Netzwerkanbindung, und einen Rechner, zu dem Sie die Verbindung aufbauen möchten. Das können z.B. sein:

- * ein mit dem Amiga verbundenes Modem und eine Telefonleitung. Das Modem sollte einigermaßen Hayes-kompatibel sein. Das trifft auf die meisten modernen Modems zu. Ausserdem brauchen Sie dann einen SLIP- oder PPP-Zugang bei einem Internet-Provider. Wenn Sie nur einen Shell-Zugang haben, können Sie Miami damit auch verwenden, aber Sie müssen dann zunächst Slirp oder TIA bei Ihrem Provider installieren. In diesem Fall sollten Sie Ihren Provider zunächst um Erlaubnis fragen, und dann auch, wie und woher Sie Slirp oder TIA am besten bekommen können.
- * eine Ethernet-Karte, ein Kabelmodem, und einen Netzwerk-Zugang, wie oben beschrieben.
- * eine Ethernet-Karte, die Ihren Amiga mit einem lokalen Netz verbindet.

Beachten Sie, daß Miami ppp.device, appp.device, amipp.device oder (rh)(c)slip.device nicht benötigt. Die Protokolle PPP und (C)SLIP sind in Miami eingebaut, und zwar in Versionen, die effizienter und fortgeschrittener sind als die derzeit erhältlichen SANA-II-Treiber.

1.7 Miami_deutsch.guide/NODE_INSTALLATION

Installation

Zu Miami sind die folgenden Archive erhältlich:

Miami30main.lha

Das Hauptarchiv. Jeder Anwender braucht es.

Miami30-000.lha

Die 68000/010-Version von Miami. Sie brauchen dieses Archiv, wenn Ihr Amiga eine 68000 oder 68010 CPU hat.

Miami30-020.lha

Die 68020+-Version von Miami. Sie brauchen dieses Archiv, wenn Ihr Amiga eine 68020, 68030, 68040 oder 68060 CPU hat.

Miami30MUI.lha

Das MUI-Modul für Miami. Sie brauchen es, wenn Sie Miami zusammen mit MUI >=3.8 verwenden wollen.

Miami30GTL.lha

Das GTLayout-Modul für Miami. Sie brauchen es, wenn Sie Miami zusammen mit glayout.library V40 oder höher verwenden wollen.

Jeder Anwender benötigt das Hauptarchiv, eins der beiden CPU-Archive, und mindestens eins der beiden GUI-Modul-Archive. Sie können auch mehr als ein GUI-Modul installieren, wenn Sie möchten.

Downloaden Sie bitte die benötigten Archive, entarchivieren Sie sie, in das gleiche (temporäre) Verzeichnis, und starten Sie dann das beiliegende Installer-Skript, um Miami auf Ihrer Festplatte zu

installieren. Das Installer-Skript kann für eine Neu-Installation und für Updates benutzt werden.

Alle Dateien werden in einem einzigen Verzeichnis installiert, und keine Systemdateien oder Systemverzeichnisse werden von Miami verändert, mit einer Ausnahme:

Das Installer-Skript fragt Sie, ob Sie den "Miami:"-Assign erzeugen möchten, und fügt dann den benötigten Befehl zu Ihrer user-startup-Datei hinzu. Das ist notwendig. Wenn Sie diesen Schritt bei der Installation überspringen, müssen Sie den Assign manuell erzeugen, bevor Sie Miami starten. Ansonsten funktioniert Miami nicht korrekt.

1.8 Miami_deutsch.guide/NODE_MIAMIINIT

MiamiInit

MiamiInit ist ein Hilfsprogramm, das versucht, alle Konfigurationsparameter, die Miami für eine serielle Verbindung (SLIP oder PPP) oder eine SANA-II-Verbindung (Ethernet, Arcnet usw.) braucht, automatisch zu ermitteln, und das dann eine Konfigurationsdatei speichert, die später von Miami genutzt werden kann.

Nach dem Installieren von Miami sollten Sie zur Konfiguration Ihrer Verbindung zuerst MiamiInit starten und dem Dialog folgen. MiamiInit erzeugt dann die Verbindung, ermittelt alle notwendigen Parameter, und speichert diese zum Schluß.

MiamiInit unterstützt derzeit nur die häufig verwendeten Konfigurationen. Sehr ungewöhnliche Fälle, wie andere Datenformate als 8N1, Modems, die nicht kompatibel zum Hayes-Standard sind, oder 3-Draht-Modemkabel werden nicht unterstützt. Wenn Sie eine derartige, ungewöhnliche Konfiguration haben, sollten Sie Miami manuell konfigurieren, statt MiamiInit zu benutzen.

Bitte beachten Sie, daß je nach Konfiguration Ihres Netzes MiamiInit möglicherweise nicht alle Informationen ganz selbständig ermitteln kann. Es ist kein Fehler, wenn MiamiInit sie z.B. nach der IP-Adresse oder Netzmaske während der Konfiguration fragt. Das bedeutet nur, daß es auf dem Netz keinen Server gibt, der diese Informationen MiamiInit bereitstellt. In diesem Fall müssen Sie Ihren Netzanbieter oder Netzverwalter nach den fehlenden Informationen fragen.

Üblicherweise müssen Sie die meisten Informationen manuell eintragen, wenn Sie ein sehr kleines lokales Netz aufbauen, das nur aus Amigas und PCs besteht. Wenn Sie dagegen eine Verbindung zum Internet oder einem Firmennetz aufbauen, das dafür eingerichtet ist, neue Rechner automatisch zu konfigurieren, dann findet MiamiInit oft die meisten oder alle Informationen von einem Server.

1.9 Miami_deutsch.guide/NODE_TOOLTYPES

ToolTypes

Miami unterstützt die folgenden ToolTypes (beim Start von der Workbench) bzw. Parameter (beim Start von der Shell):

PACKETDEBUG

Schaltet den Paket-Level-Debugging-Modus ein. Wenn Sie "PACKETDEBUG=10" oder "PACKETDEBUG=20" angeben, erzeugt Miami eine Datei "Miami.debug" mit einem Hexdump aller verschickten und empfangenen Pakete. Sie sollten diese Funktion nur beim Debugging verwenden, nicht während des normalen Betriebs, da die Datei sehr schnell in der Größe anwächst, und das Protokollieren viel CPU-Zeit verbraucht. Ein Wert von 10 protokolliert nur Paketinhalte. Ein Wert von 20 protokolliert auch rohe Paketdaten (für PPP/SLIP).

DONTCONNECT

Wenn Sie Miami so konfiguriert haben, daß es beim Start automatisch eine Verbindung zu Ihrem Internet-Provider aufbaut, können Sie das mit dieser Option unterbinden. Das gibt Ihnen die Möglichkeit, vor dem Aufbau der Verbindung noch die Einstellungen zu verändern.

SETTINGS

Ein Projekt-Icon muss ein "SETTINGS"-ToolType haben, damit Miami es als eine Einstellungen-Datei erkennt. Von der Shell können Sie den Parameter "SETTINGS Dateiname" verwenden, um anzugeben, welche Einstellungen-Datei geladen werden soll.

IMPORTMIAMIINIT

"IMPORTMIAMIINIT=Dateiname" veranlaßt Miami, eine Einstellungen-Datei von MiamiInit zu importieren.

IMPORTASCII

"IMPORTASCII=Dateiname" veranlaßt Miami, eine Einstellungen-Datei vom ASCII-Format (Format für Vertrieb) zu importieren.

SAVESETTINGS

"SAVESETTINGS" veranlaßt Miami, die aktuellen Einstellungen als Standardeinstellungen zu speichern. Das ist besonders nützlich in Verbindung mit den ToolTypes "IMPORTMIAMIINIT" und "IMPORTASCII", um Einstellungen zu importieren, und in das Miami-Einstellungen-Format zu konvertieren.

AREXX

"AREXX=Dateiname" veranlaßt Miami, das angegebene AREXX-Skript beim Programmstart auszuführen.

GUI

"GUI=Name" teilt Miami mit, welches GUI-Modul verwendet werden soll. Dieses Argument hat Vorrang vor dem Wert, der in der Einstellungen-Datei gespeichert ist.

Versuchen Sie NICHT, undokumentierte ToolTypes zu verwenden !
Solche ToolTypes tun normalerweise nicht das, was Sie erwarten, und
können die Kompatibilität oder Leistung von Miami verringern.

1.10 Miami_deutsch.guide/NODE_MENUS

Menüs

Beschreibung aller Menüpunkte:

Projekt/Über...

Informationen über Miami anzeigen.

Projekt/Über MUI...

Informationen über MUI (Magic User Interface) anzeigen. Dieser
Menüpunkt erscheint nur, wenn Sie eins der MUI-Module verwenden.

Projekt/Iconifizieren

Alle Fenster von Miami iconifizieren. Beachten Sie, daß dieser
Menüpunkt für einige GUI-Module (z.B. GTLayout) gleichbedeutend
mit 'Projekt/GUI ausschalten' ist.

Projekt/GUI ausschalten

Iconifiziert alle Fenster von Miami, und entfernt das GUI-Modul
aus dem Speicher.

Projekt/Offline ohne aufzulegen...

Geht offline, ohne vorher die Modem-Leitung aufzulegen.

Projekt/Ende ohne aufzulegen...

Miami beenden, ohne vorher die Modem-Leitung aufzulegen.

Projekt/Ende...

Miami beenden.

Einstellungen/Laden...

Eine Einstellungen-Datei laden.

Einstellungen/Speichern

Einstellungen in der aktuellen Datei speichern.

Einstellungen/Speichern als...

Einstellungen in einer neuen Datei speichern.

Einstellungen/Speichern als Standard

Einstellungen in der Standarddatei speichern.

Einstellungen/Piktogramm erzeugen

Automatisch Piktogramm (Icon) erzeugen, wenn die Einstellungen
gespeichert werden.

Importieren von Vertrieb...

Einstellungen-Datei von Miami's Vertriebs-(ASCII-)Format

importieren.

Exportieren für Vertrieb...

Einstellungen-Datei in Miami's Vertriebs-(ASCII-)Format exportieren.

Importieren von MiamiInit V2...

Einstellungen-Datei von MiamiInit Version 2 importieren. Diese Funktion ist veraltet. Sie sollten stattdessen MiamiInit Version 3 verwenden, und die Import-Funktion 'Importieren von MiamiInit V3...'

Importieren von MiamiInitsANA2 V2...

Einstellungen-Datei von MiamiInitsANA2 Version 2 importieren. Diese Funktion ist veraltet. Sie sollten stattdessen MiamiInit Version 3 verwenden, und die Import-Funktion 'Importieren von MiamiInit V3...'

Importieren von MiamiInit V3...

Einstellungen-Datei von MiamiInit Version 3 importieren.

MUI Einstellungen...

MUI-Konfigurationsfenster öffnen. Dieser Menüpunkt erscheint nur, wenn Sie eins der MUI-Module verwenden.

1.11 Miami_deutsch.guide/NODE_CONFIGURATION

Konfiguration

Die Konfiguration von Miami erfolgt vollständig durch die graphische Benutzeroberfläche. Es müssen keine Konfigurationsdateien oder Environment-Variablen modifiziert werden.

Beschreibung der graphischen Benutzeroberfläche:

Allgemeines

Die 'Allgemeines'-Seite

Schnittstelle

Die 'Schnittstelle'-Seite

PPP

Die 'PPP'-Seite

Wählmodul

Die 'Wählmodul'-Seite

Datenbank

Die 'Datenbank'-Seite

TCP/IP

	Die 'TCP/IP'-Seite
Ereignisse	Die 'Ereignisse'-Seite
Modem	Die 'Modem'-Seite
Protokollierung	Die 'Protokollierung'-Seite
Fenster	Die 'Fenster'-Seite
GUI	Die 'GUI (Benutzeroberfläche)'- Seite ↔
Socks	Die 'Socks'-Seite
Sonstiges	Sonstige Schalter

1.12 Miami_deutsch.guide/NODE_GUI_GENERAL

Allgemeines

=====

Hier gibt es nicht viel, ausser dem offiziellen Miami-Logo und einem Schalter, um das Miami-Registrierprogramm zu starten.

Bei einigen GUI-Modulen (z.B. MUI) erscheint diese Seite, indem man auf "Allgemeines" im Listview klickt. Bei anderen GUI-Modulen ist im Hauptfenster immer der Inhalt der "Allgemeines"-Seite zu sehen, und alle anderen Seiten erscheinen in getrennten Fenstern.

Registrieren

Der 'Registrieren'-Schalter

1.13 Miami_deutsch.guide/NODE_GUI_GENERAL_REGISTER

Registrieren

Dieser Schalter startet das Programm MiamiRegister, mit dem Sie

einen Miami-Lizenzcode bestellen, Miami registrieren oder Ihre Registrierung upgraden können. MiamiRegister muß in dem gleichen Verzeichnis wie Miami sein, oder im Standard-Suchpfad der Shell.

1.14 Miami_deutsch.guide/NODE_GUI_INTERFACE

Schnittstelle

=====

Schnittstellen-Typ

Der 'Schnittstellen-Typ'-Schalter

Gerät / Nummer

Die 'Gerät'- und 'Nummer'-Schalter

Geschwindigkeit

Der 'Geschwindigkeit'-Schalter

Verwende CD

Der 'CD'-Schalter

Protokoll

Der 'Protokoll'-Schalter

Handshaking

Der 'Handshaking'-Schalter

EOF-Modus

Der 'EOF-Modus'-Schalter

Betriebsart

Der 'Betriebsart'-Schalter

IP-Typ / Adresse

Die 'IP'-Schalter

Netzmasken / Typ

Die 'Netzmasken'-Schalter

Gateway / Typ

Die 'Gateway'-Schalter

Multicasts

Der 'Multicasts'-Schalter

Umwandlung

Der 'Umwandlung'-Schalter

MTU

Der 'MTU'-Schalter

SANA-II-Parameter

Der 'SANA-II-Parameter'-Schalter

Inaktivität

Die 'Inaktivität'-Schalter

1.15 Miami_deutsch.guide/NODE_GUI_INTERFACE_TYPE

Schnittstellen-Typ

Dieser Schalter wählt den Typ der zu verwendenden Schnittstelle aus. Das genaue Aussehen der 'Schnittstelle'-Seite hängt davon ab, welcher Schnittstellen-Typ gewählt wird, d.h. es werden nur solche Schalter angezeigt, die für den ausgewählten Schnittstellentyp von Bedeutung sind.

Mögliche Typen:

interner serieller Port (PPP/SLIP)

Die eingebaute Standard-Implementierung von PPP oder (C)SLIP, in Verbindung mit Miami's eigenem Treiber für den im Amiga eingebauten Port. Dieser Schnittstellen-Typ kann nur mit dem im Amiga eingebauten seriellen Port genutzt werden, und benötigt keine zusätzlichen Treiber. Mit diesem Schnittstellen-Typ erreicht man i.A. besseren Datendurchsatz als mit 'seriell (PPP/SLIP)' und serial.device (oder einem Ersatz-Treiber).

seriell (PPP/SLIP)

Die eingebaute Standard-Implementierung von PPP oder (C)SLIP, in Verbindung mit einem beliebigen serial.device-kompatiblen Treiber. Dieser Schnittstellen-Typ benötigt keinen SANA-II-Treiber.

SANA-II Punkt-zu-Punkt

Ein SANA-II-Treiber für eine Hardware, die genau zwei Rechner verbindet, z.B. SLIP, PPP oder PLIP.

SANA-II Ethernet

Ein SANA-II-Treiber für eine Ethernet-Karte, z.B. die A2065-Karte, die Hydra-Karte, oder die Ariadne-Karte.

SANA-II "altes" Standard-Arcnet

Ein SANA-II-Treiber für eine Arcnet-Karte, z.B. die A2060-Karte. Diese Einstellung verwendet die normale "alte" RFC1051-Arcnet-Verkapselung, die in Amiga-Netzen häufiger verwendet wird als die "neue" RFC1201-Verkapselung. Sie sollten die "alte" Verkapselung verwenden, wenn Sie Ihren Amiga mit Rechnern vernetzen wollen, die AmiTCP/IP, I-Net 225 oder NetBSD 1.1 fahren.

SANA-II "neues" Arcnet

Ein SANA-II-Treiber für eine Arcnet-Karte, z.B. die A2060-Karte. Diese Einstellung verwendet die "neue" RFC1201-Arcnet-Verkapselung. Sie ist nicht kompatibel zu AmiTCP/IP oder NetBSD 1.1, aber Sie

brauchen diese Einstellung möglicherweise, wenn Sie Ihren Amiga mit anderen Plattformen wie Windows 95 vernetzen wollen.

SANA-II anderer Bus/Ring

Ein SANA-II-Treiber für eine Bus- oder Ring-Hardware ausser Ethernet oder Arcnet. Wählen Sie diese Einstellung, wenn Ihre Hardware mehr als zwei Rechner verbinden kann, aber nicht kompatibel zu Ethernet oder Arcnet ist.

1.16 Miami_deutsch.guide/NODE_GUI_INTERFACE_DEVICE

Gerät / Nummer

Für serielle Schnittstellen:

Geben Sie den Gerätenamen und die Unit-Nummer des seriellen Ports ein, an dem Ihr Modem angeschlossen ist. Für den im Amiga eingebauten seriellen Port verwenden Sie 'serial.device' '0' oder 'artser.device' '0'. Sie sollten derzeit nicht '8n1.device' verwenden, wegen Fehlern in diesem Treiber. Einige Anwender haben auch von Problemen mit 'BaudBandit.device' und 'v34serial.device' berichtet.

Für serielle Schnittstellenkarten verwenden Sie den Treiber, der mit der Karte mit geliefert wird, z.B. 'gvpser.device', mit der korrekten Unit-Nummer.

Für SANA-II-Schnittstellen:

Geben Sie den Gerätenamen und die Unit-Nummer Ihrer SANA-II-Hardware an. Die Unit-Nummer ist in den meisten Fällen 0.

1.17 Miami_deutsch.guide/NODE_GUI_INTERFACE_SPEED

Geschwindigkeit

(Diese Option erscheint nur für serielle Schnittstellen.)

Die Geschwindigkeit Ihres seriellen Ports. Für den internen seriellen Port sollten Sie 19200, 38400 oder (wenn Sie eine schnelle CPU und eine Grafikkarte haben) 57600 benutzen. Für serielle Schnittstellenkarten können Sie möglicherweise auch 115200 oder 230400 verwenden.

Verwenden Sie nicht 31250. Diese Geschwindigkeit ist ausschliesslich für MIDI reserviert und funktioniert mit den meisten Modems nicht.

Verwenden Sie auch nicht 14400, 28800 oder 33600. Ihr Modem mag zwar mit diesen Geschwindigkeiten Verbindungen zur Gegenseite aufbauen, aber unterstützt diese Geschwindigkeiten wahrscheinlich nicht für die

serielle Schnittstelle.

1.18 Miami_deutsch.guide/NODE_GUI_INTERFACE_CD

Verwende CD

(Diese Option erscheint nur für serielle Schnittstellen.)

Wenn "Verwende CD" aktiviert ist, dann verwendet Miami die "Carrier Detect"-Leitung Ihres Modems um festzustellen, ob Ihr Modem bereits mit der Gegenseite verbunden ist oder nicht.

Das kann nützlich sein, wenn Sie Ihren Amiga resetten ohne daß das Modem auflegt, damit Sie Miami neu starten können, ohne neu zu wählen.

Diese Option kann nur verwendet werden, wenn Ihr Modem so konfiguriert ist, daß die 'Carrier Detect'-Leitung nur aktiv ist, wenn eine Verbindung besteht.

Einige Modems haben Fabrikeinstellungen, die die "Carrier Detect"-Leitung immer auf "high" setzen, auch wenn das Modem nicht verbunden ist. Wenn das auf Ihr Modem zutrifft, müssen Sie entweder die Modem-Einstellungen ändern (normalerweise "AT&C1") und die Einstellungen dann mit einem Terminalprogramm im NV-RAM Ihres Modems speichern (normalerweise "AT&W"), oder "Verwende CD" abschalten.

Wenn Sie die Nullmodem-Einstellungen (konfiguriert auf der "Modem"-Seite) verwenden, dann bekommt dieser Schalter eine andere Bedeutung:

- * Wenn der Schalter aktiviert ist, wird das Wählskript überhaupt nicht ausgeführt.
- * Wenn der Schalter deaktiviert ist, wird das Wählskript ausgeführt, aber Miami wählt keine Telefonnummer an, d.h. der "ATDT..."-Befehl wird übersprungen, und die Telefonnummern-Liste wird bedeutungslos.

1.19 Miami_deutsch.guide/NODE_GUI_INTERFACE_PROTOCOL

Protokoll

(Diese Option erscheint nur für serielle Schnittstellen.)

Das Protokoll, das Ihr Internet-Provider verwendet. Derzeit unterstützt werden SLIP/CSLIP und PPP.

1.20 Miami_deutsch.guide/NODE_GUI_INTERFACE_FLOW

Handshaking.

(Diese Option erscheint nur für serielle Schnittstellen.)

Miami unterstützt derzeit zwei Handshaking-Methoden: Hardware-Handshaking (RTS/CTS) und Software-Handshaking (Xon/Xoff). Die Standardeinstellung ist Hardware-Handshaking, und es wird dringend empfohlen, das auch nicht zu ändern.

Wenn Sie Hardware-Handshaking nicht verwenden können (normalerweise wegen eines defekten Modems, Kabels, oder seriellen Ports), sollten Sie auf Software-Handshaking umschalten. Achten Sie dann aber darauf, dass der Initialisierungsstring Ihres Modems (im Wählmodul-Fenster) entsprechend angepasst wird. Ausserdem ist Software-Handshaking nur mit PPP möglich, nicht mit SLIP/CSLIP.

1.21 Miami_deutsch.guide/NODE_GUI_INTERFACE_EOF

EOF-Modus

(Diese Option erscheint nur für serielle Schnittstellen.)

Miami hat zwei Möglichkeiten, das Ende von ankommenden Datenpaketen zu erkennen: Die effizientere (mit geringerem CPU-Zeit-Verbrauch) verwendet das EOF_MODE-Flag. Allerdings funktioniert das nur, wenn der serielle Treiber den EOF-Modus unterstützt. Viele Treiber von Drittherstellern unterstützen diesen Modus nicht.

Normalerweise sollten Sie diesen Schalter auf "auto" stehen lassen, damit Miami die Standardeinstellung benutzt. Wenn Sie genau wissen, ob Ihr Treiber den EOF-Modus unterstützt, können Sie das auch hier einstellen.

1.22 Miami_deutsch.guide/NODE_GUI_INTERFACE_SERIAL

Betriebsart

(Diese Option erscheint nur für serielle Schnittstellen.)

Die Einstellungen für die Anzahl Datenbits und Parität. Für 99% aller Provider ist 8N1 korrekt. Einige wenige Provider (z.B. einige Dialin-Punkte für Compuserve) benötigen möglicherweise 7E1 oder 7O1.

Bitte beachten Sie, daß sich diese Einstellungen nur auf die Wähl- und Loginphase beziehen. Für das (C)SLIP/PPP-Protokoll verwendet Miami

immer 8N1, unabhängig davon, was Sie hier einstellen. Es ist absolut unmöglich, PPP oder (C)SLIP über eine 7-Bit-Leitung zu nutzen - mit jeder PPP/(C)SLIP-Implementierung. Das ist keine Einschränkung in Miami.

1.23 Miami_deutsch.guide/NODE_GUI_INTERFACE_IP

IP-Typ / -Adresse

Internet-Provider bieten üblicherweise zwei Arten von Internet-Verbindungen an: solche mit einer statischen IP-Adresse, die Ihrem Amiga permanent zugewiesen wird, und (häufiger) solche, bei denen Ihr Amiga bei jeder Neu-Verbindung eine andere, dynamische IP-Adresse erhält.

Für serielle Schnittstellen:

Wenn Ihr Amiga eine feste IP-Adresse hat, wählen Sie "statisch" und geben Sie die IP-Adresse ein, die Ihnen Ihr Provider gegeben hat. Wenn Ihr Provider Ihnen eine dynamische IP-Adresse für jede Verbindung zuweist, dann wählen Sie "dynamisch", und Miami ermittelt die IP-Adresse automatisch, wenn Sie eine Verbindung aufbauen.

Wenn Sie TIA oder Slirp verwenden, müssen Sie "statisch" wählen, und die Pseudo-IP-Adresse eingeben, die TIA oder Slirp Ihrem Amiga zugewiesen hat. Mehr Informationen hierzu stehen in der TIA/Slirp-Anleitung.

Für SANA-II-Punkt-zu-Punkt-Verbindungen:

Wenn Ihr Amiga eine feste IP-Adresse hat, wählen Sie "statisch" und geben Sie die IP-Adresse ein. Wenn Ihnen die Adresse von einem BootP/DHCP-Server zugewiesen wird, dann wählen Sie "DHCP". Wenn der SANA-II/Treiber die IP-Adresse selbst ermittelt, dann wählen Sie "SANA-II".

Für SANA-II-Bus/Ring-Schnittstellen:

Wenn Ihr Amiga eine feste IP-Adresse hat, wählen Sie "statisch" und geben Sie die IP-Adresse ein. Wenn Ihnen die Adresse von einem BootP/DHCP-Server zugewiesen wird, dann wählen Sie "DHCP". Wenn Ihnen die Adresse von einem RArp-Server zugewiesen wird, dann wählen Sie "RArp".

1.24 Miami_deutsch.guide/NODE_GUI_INTERFACE_MASK

Netzmasken-Typ / -Adresse

(Diese Option erscheint nur für SANA-II-Bus-/Ring-Schnittstellen.)

Ihre Netzmaske muß korrekt konfiguriert werden, damit Miami weiß, welche Rechner auf dem lokalen Netz sind. Es gibt drei Möglichkeiten, die Netzmaske einzustellen:

statisch

Fragen Sie Ihren Netz-Administrator nach dem korrekten Wert, und geben Sie ihn ein.

DHCP

Miami versucht, den korrekten Wert von einem BootP/DHCP-Server zu ermitteln.

ICMP

Miami versucht, den korrekten Wert von einem Server per ICMP-Netmask-Discovery zu ermitteln.

1.25 Miami_deutsch.guide/NODE_GUI_INTERFACE_GWAY

Gateway-Typ / -Adresse

(Diese Option erscheint nur für SANA-II-Bus-/Ring-Schnittstellen.)

Ihr Standard-Gateway muß korrekt konfiguriert werden, damit Miami weiß, wohin Packets geschickt werden sollen, die nicht für Rechner auf dem lokalen Netz bestimmt sind. Es gibt drei Möglichkeiten, das Gateway einzustellen:

statisch

Fragen Sie Ihren Netz-Administrator nach dem korrekten Wert, und geben Sie ihn ein.

DHCP

Miami versucht, den korrekten Wert von einem BootP/DHCP-Server zu ermitteln.

ICMP

Miami versucht, den korrekten Wert von einem Server per ICMP-Router-Discovery zu ermitteln.

1.26 Miami_deutsch.guide/NODE_GUI_INTERFACE_MULTICASTS

Multicasts

(Diese Option kann nur in der registrierten Version verändert werden.)

Miami unterstützt Level-2-Multicasting, d.h. sowohl das Senden, als auch das Empfangen von Multicast-Nachrichten.

Wenn Sie Applikationen verwenden, die Multicasting nutzen (bisher gibt es noch keine), müssen Sie Multicasting in Miami einschalten. Die möglichen Einstellungen sind:

inaktiv

Multicasting ist ausgeschaltet.

als Broadcasts

Multicasts werden als Link-Level-Broadcasts (oder bei Punkt-zu-Punkt-Hardware: als normale Pakete) verschickt.

als Multicasts

Multicasts werden als Link-Level-Multicasts verschickt. Diese Option ist nur mit Ethernet möglich.

Hinweis: Multicasts sollten für eine Schnittstelle nur eingeschaltet sein, wenn Sie Ihren Multicast-Feed direkt von dieser Schnittstelle bekommen. Wenn Sie Ihren Multicast-Feed durch einen Tunnel mit MiamiMRouteD erhalten, sollten Sie normalerweise Multicasts auf der Seite 'Schnittstelle' ausschalten, da MiamiMRouteD Multicasting selbständig handhabt.

1.27 Miami_deutsch.guide/NODE_GUI_INTERFACE_MAPPING

Umwandlung

(Diese Option erscheint nur für SANA-II-Arcnet-Schnittstellen.)

Arcnet unterstützt zwei verschiedene Methoden, um IP-Adressen in Hardware-Adressen zu konvertieren:

Arp

Arp (Address-Resolution-Protocol) wird verwendet. Das ist die empfohlene Standard-Einstellung, und ist auch, was AmiTCP/IP verwendet.

direkt

Die untersten 8 Bits der IP-Adresse werden auf die Hardware-Adresse abgebildet. Das ist die Methode, die NetBSD 1.1 verwendet.

Wenn Sie mindestens einen NetBSD-1.1-Rechner in Ihrem Arcnet-Netz haben, können Sie die Konfiguration vereinfachen, indem Sie "direkte Umwandlung" wählen, anstatt auf allen Rechnern manuelle Arp-Einträge zu erzeugen.

In allen anderen Fällen sollten Sie auf allen Rechnern "Arp" wählen.

1.28 Miami_deutsch.guide/NODE_GUI_INTERFACE_MTU

MTU

(Diese Option erscheint nur für serielle Schnittstellen.)

Maximum Transfer Unit, d.h. die maximale Größe von Paketen, die "am Stück" übertragen werden. Der Standard für PPP ist 1500, der Standard für (C)SLIP ist 1006.

Für serielle Leitungen ist es normalerweise ratsam, die MTU auf einen geringeren Wert als den Standardwert einzustellen, um ein besseres Antwortzeitverhalten für interaktive TCP/IP-Anwendungen zu erhalten. Gute Werte sind:

- * für Modem-Geschwindigkeiten bis zu 19200 bps: MTU=296.
- * für Modem-Geschwindigkeiten über 19200 bps: MTU=552.

Bitte beachten Sie, daß ein Ändern des MTU-Wertes im Konfigurationsfenster nicht unbedingt auch zur Folge hat, daß die maximale Paketgröße wirklich auf diesen Wert verändert wird:

(C)SLIP kann den MTU-Wert überhaupt nicht aushandeln, d.h. der konfigurierte MTU-Wert beeinflusst nur die Größe von verschickten Paketen, nicht die Größe von empfangenen Paketen.

PPP hat Konfigurationsoptionen zum Aushandeln der MTU. Miami versucht immer, den hier eingestellten Wert auszuhandeln, aber es ist möglich, daß die andere Seite das nicht zuläßt. In diesem Fall muß Miami möglicherweise die Werte, die die andere Seite vorgibt, in eine Richtung oder auch in beiden Richtungen verwenden.

Beachten Sie auch: Für PPP ist der MTU-Wert nicht kritisch, d.h. Ihre Verbindung wird auch dann noch funktionieren, wenn der von Ihnen eingestellte Wert größer oder kleiner als der optimale Wert ist. Für (C)SLIP darf der eingestellte Wert jedoch nicht größer sein als der MTU-Wert bei Ihrem Provider.

1.29 Miami_deutsch.guide/NODE_GUI_INTERFACE_STP

SANA-II-Parameter

(Diese Option erscheint nur für SANA-II-Schnittstellen.)

Bei Klicken auf den Schalter "SANA-II-Parameter" öffnet sich ein Fenster mit SANA-II-Geräte-Einstellungen. Diese Einstellungen beinhalten:

- * Die Hardware-Adresse Ihres Gerätes, mit einer Option, diese zu erzwingen. (Nur für Bus-/Ring-Geräte.) Hardware-Adressen werden

als Bytefolge in hexadezimaler Schreibweise, getrennt durch `:'` angegeben, z.B. `01:23:45:67:89:ab`.

- * Der Link-Level-Pakettyp für IP-, Arp- und RArp-Pakete. (RArp ist nicht möglich mit Arcnet, und Arp ist nicht möglich mit Punkt-zu-Punkt-Geräten.)
- * Die MTU für das Gerät.
- * Die Anzahl der IORequests, die für IP- und Arp-Pakete verwendet werden sollen. (Arp ist nicht möglich mit Punkt-zu-Punkt-Geräten.)

In meisten Fällen sollten Sie alle diese Werte auf Standardwerte initialisieren, indem Sie auf "Parameter beim Gerät erfragen" klicken (nur möglich, wenn Miami offline ist). Allerdings können Sie Werte auch erzwingen, wenn das notwendig sein sollte, z.B. wenn Sie einen neuen Gerätetyp verwenden, zu dem Miami noch nicht die korrekten Standardeinstellungen kennt.

1.30 Miami_deutsch.guide/NODE_GUI_INTERFACE_INACTIVITY

Inaktivität

Einige Internet-Provider hängen auf, wenn auf der Leitung einige Zeit lang keine Aktivität ist, um zu verhindern, daß Anwender Leitungen blockieren, die sie nicht nutzen.

Mit den "Inaktivität"-Schaltern können Sie Miami so konfigurieren, daß es Leitungsaktivität simuliert, damit Ihr Provider nicht aufhängt, auch wenn Sie die Leitung gerade nicht benutzen.

Mit dem linken Schalter stellen Sie die Art der Aktivität ein: PPP ping oder ICMP ping. PPP ping benötigt weniger Bandbreite, aber funktioniert nur mit PPP, nicht mit (C)SLIP, und hat auch nicht bei allen Providern den gewünschten Effekt. ICMP ping braucht geringfügig mehr Bandbreite, aber sollte mit allen Providern funktionieren.

Wenn Sie (C)SLIP verwenden, wählen Sie ICMP ping. Ansonsten versuchen Sie es erst mit PPP ping, und wenn Ihr Provider dann immer noch aufhängt, mit ICMP ping.

Der Schalter rechts gibt die Anzahl Minuten zwischen aufeinanderfolgenden Pings an. Sie müssen damit etwas experimentieren. Typische Werte sind 9 oder 14, um ein Auflegen nach 10 oder 15 Minuten zu verhindern.

Hinweis: Sie müssen zuerst Ihren Internet-Provider fragen, ob er einen derartigen Aktivitäts-Simulator erlaubt. Einige Provider haben Nutzungsbedingungen, die den Gebrauch derartiger Simulatoren verbieten, d.h. durch den Gebrauch würden Sie die Nutzungsbedingungen verletzen. Ich lehne jegliche rechtliche oder finanzielle Verantwortung für die Folgen ab, die aus dem unberechtigten oder falschen Gebrauch des

Aktivitäts-Simulators entstehen.

Hinweis: Es gibt viele Gründe, warum ein Modem auflegen kann. Ein Grund ist ein Inaktivitäts-Timeout beim Provider, was durch diese Funktion verhindert werden sollte. Allerdings hängen manche Modems auch gelegentlich wegen Leitungsstörungen auf. Das läßt sich durch Software nicht verhindern.

Mit dieser Funktion können Sie das Auflegen bei Inaktivität verhindern. Einige Anwender möchten genau das Umgekehrte, d.h. das Auflegen bei Inaktivität erzwingen, um Telefonkosten zu sparen. Mit dem Hilfsprogramm

MiamiRemind
können Sie genau das tun.

1.31 Miami_deutsch.guide/NODE_GUI_PPP

```

====

PPP

PAP / CHAP-Passwort                Die 'PAP/CHAP'-Schalter

Rückruf                             Die 'Rückruf'-Schalter

VJC                                  Der 'VJC'-Schalter

ACCM                                 Der 'ACCM'-Schalter

Schnelle Neuverbindung              Der 'Schnelle Neuverbindung'-Schalter

Escape                               Der 'Escape'-Schalter

DNS mit IPCP ermitteln               Der 'DNS mit IPCP ermitteln'-Schalter

TermReq vor dem Auflegen             Der 'TermReq vor dem Auflegen'-Schalter

```

1.32 Miami_deutsch.guide/NODE_GUI_PPP_CHAP

PAP / CHAP-Passwort

PAP und CHAP sind Protokolle, die von PPP verwendet werden, um die Login-ID und das Passwort an einen PPP-Server zu übermitteln.

In den meisten Fällen sind die für PAP und CHAP verwendeten Daten für Login-ID und Passwort identisch zu denen im Wählskript. In diesem Fall aktivieren Sie "Wie im Wählmodul".

Wenn Ihr Provider für PAP/CHAP andere Daten für Login-ID und Passwort als im Wählskript benötigt, dann deaktivieren Sie "Wie im Wählmodul", und geben Sie stattdessen PAP/CHAP-Login-ID und Passwort manuell ein.

Registrierte Anwender, die MiamiSSL 1.2 oder höher installiert haben, können 'MS-CHAP erlauben' einschalten. Das erhöht die Kompatibilität mit einigen falsch konfigurierten Windows-NT PPP-Servern. Wenn diese Option ausgeschaltet ist, verwendet Miami PAP, wenn der Server MS-CHAP anfordert.

1.33 Miami_deutsch.guide/NODE_GUI_PPP_CALLBACK

Rückruf

(Diese Option ist nur in der registrierten Version anwählbar.)

PPP unterstützt Rückruf ('Callback', 'Dialback') mit dem CBCP-Protokoll. Wenn der Server Ihres Providers dafür konfiguriert ist, können Sie mit dem Provider aushandeln, daß er Sie zurück ruft, damit Sie Telefongebühren sparen.

Abhängig von der Konfiguration bei Ihrem Provider müssen Sie entweder 'CBCP, fest' wählen (in diesem Fall ruft der Provider zu einer fest vordefinierten Nummer zurück) oder 'CBCP, variabel' (in diesem Fall ruft der Provider bei der Nummer zurück, die Sie in dem Feld darunter eingeben).

'Min-Pause' ist die Wartezeit, die der Provider warten soll, bevor er Sie zurück ruft. Diese Pause sollte lang genug sein, um Ihrem Modem Zeit zu geben, die Leitung aufzulegen, und sich neu zu initialisieren.

'Max-Pause' ist die maximale Zeit, die Miami auf einen Rückruf warten soll, bevor es aufgibt.

1.34 Miami_deutsch.guide/NODE_GUI_PPP_VJC

VJC

Van Jacobsen Compression ist ein Verfahren zur Komprimierung der Header von TCP-Paketen. Diese Option sollte normalerweise angeschaltet sein, damit PPP automatisch die Benutzung von VJC aushandelt, und VJC benutzt, wenn die andere Seite zustimmt.

Allerdings unterstützen einige alte, defekte PPP-Server VJC nicht korrekt, so daß Sie VJC möglicherweise ausschalten müssen.

VJC hat keinerlei Einfluß auf die Datenkomprimierung von Ihrem Modem, d.h. Sie sollten VJC nicht abschalten, nur weil Ihr Modem MNP-5 oder V.42bis unterstützt. VJC kann unabhängig von MNP-5 oder V.42bis genutzt werden.

1.35 Miami_deutsch.guide/NODE_GUI_PPP_ACCM

ACCM

Das PPP-Protokoll unterstützt eine Liste von Kontrollzeichen, die während der Übertragung "escaped" werden müssen, d.h. die durch eine andere Zwei-Byte-Sequenz ersetzt werden. Diese Liste heisst ACCM (Asynchronous Control Character Mask).

Der Sinn dieser Liste ist es, PPP robuster auf Leitungen zu machen, die nicht vollständig 8-bit-transparent sind, und um zu verhindern, daß Software-Handshaking das PPP-Protokoll behindert.

Standardmäßig werden nur die Zeichen 17 und 19 (Xon/Xoff) escaped, so daß PPP mit Software-Handshaking genutzt werden kann. Wenn Sie PPP durch eine telnet-Verbindung betreiben, müssen Sie möglicherweise mehr Zeichen escapen. Jedes Zeichen, das in die Liste aufgenommen wird, reduziert den Datendurchsatz von PPP um ca. 0.8%.

Sie können die ACCM-Einstellungen direkt durch Eingabe einer 32-Bit-Maske ändern, oder bei einigen GUI-Modulen mit Hilfe des Popup-Schalters einzelne Kontrollzeichen getrennt aktivieren oder deaktivieren.

1.36 Miami_deutsch.guide/NODE_GUI_PPP_QUICK

Schnelle Neuverbindung

Normalerweise ist es mit Miami möglich, die Verbindung zum Provider neu aufzubauen (ohne neu zu wählen), wenn das Modem noch mit der Gegenseite verbunden ist, z.B. nachdem Sie den Amiga resettet haben, aber nur wenn der "Verwende CD"-Schalter auf der "Schnittstelle"-Seite aktiv ist.

Allerdings ist mit einigen Internet-Providern und PPP auch dann keine Neuverbindung (mit Neuaushandlung von PPP) möglich, wenn die Gegenseite beim Neuaushandeln von PPP auflegt.

"Schnelle Neuverbindung" hilft dann oft: Wenn "Schnelle Neuverbindung" aktiviert ist (auf "RAM" oder "Datei", dann versucht Miami nicht, PPP neu auszuhandeln, sondern übernimmt alle PPP-Parameter aus einem resetfesten RAM-Bereich, der vor dem Rebooten eingerichtet worden ist (für die "RAM"-Einstellung) oder aus einer Datei auf Festplatte (für die "Datei"-Einstellung). In den meisten Fällen ist es so möglich, nach einem Reset die Verbindung zum Provider neu aufzubauen.

Bitte beachten Sie: Wenn Sie die "Datei"-Einstellung verwenden, und Ihr Amiga crasht (aus irgendeinem Grund, z.B. durch ein defektes Commodity oder einen defekten Patch) während Miami die Neuverbindungs-Datei auf Festplatte speichert, ist es möglich, daß Ihre Festplatte invalidiert oder beschädigt wird. Die Ursache sind Fehler und Einschränkungen im Amiga-Filesystem.

Es ist daher sicherer, "RAM" zu verwenden, da Miami dann keine Datei auf Festplatte erzeugt. Allerdings funktioniert die "RAM"-Einstellung nur dann, wenn entweder gar kein Reset erfolgt ist, oder wenn nur ein "warmer" Reset erfolgt ist. Wenn Ihr Amiga sehr heftig abstürzt und Sie ihn "kalt" resetten müssen (wobei resetfeste Module überschrieben werden), dann werden auch die alten PPP-Parameter überschrieben, und "Schnelle Neuverbindung" funktioniert mit der "RAM"-Einstellung nicht.

1.37 Miami_deutsch.guide/NODE_GUI_PPP_ESCAPE

Escape

PPP kann mit der Gegenseite aushandeln, daß Zeichen in den Bereichen 0-31 und 128-159 escaped werden. Das wird in der ACCM konfiguriert.

Allerdings kann es auch Situationen geben, in denen andere Zeichen escaped werden müssen, z.B. 0xFF über rlogin-Verbindungen.

In diesem Fall geben Sie bitte die zweistelligen Hexcodes der zu escapenden Zeichen (getrennt von Leerzeichen) im Escape-Feld ein. Miami escaped diese Zeichen dann beim Senden.

Beachten Sie daß, im Gegensatz zur ACCM, dieses Escapen nur in einer Richtung geschieht: beim Senden. Wenn auf dem Rückkanal vom PPP-Server zu Miami ebenfalls Zeichen escaped werden müssen, muß das beim Server konfiguriert werden.

1.38 Miami_deutsch.guide/NODE_GUI_PPP_DNSIPCP

DNS mit IPCP ermitteln

Dieser Schalter ist normalerweise an. Das bedeutet, daß Miami versucht, IPCP-Erweiterungen zur automatischen Erkennung von DNS-Servern zu nutzen.

Leider unterstützen einige defekte PPP-Server diese Option nicht korrekt, lehnen sie aber auch nicht korrekt ab, sondern verletzen einfach das Protokoll. Wenn Sie Fehlermeldungen bzgl. des Link-Level-Protokolls mit Ihrem Internetprovider bekommen, müssen Sie diese Option möglicherweise deaktivieren.

1.39 Miami_deutsch.guide/NODE_GUI_PPP_TERMREQ

TermReq vor dem Auflegen

Diese Option sollte normalerweise eingeschaltet sein. Miami sendet dann eine LCP-TermReq-Nachricht an den Provider vor dem Auflegen der Leitung. Oft bewirkt das, daß der Provider zuerst auflegt, und Ihr dadurch Modem schneller auflegt, als das sonst der Fall wäre.

Allerdings gibt es einige PPP-Server, die LCP-TermReqs nicht korrekt unterstützen. Wenn Ihnen auffällt, daß das Auflegen der Leitung sehr lange dauert, dann schalten Sie diese Option ab, und versuchen Sie, ob das Auflegen dann schneller geht.

1.40 Miami_deutsch.guide/NODE_GUI_DIALER

Wählmodul =====	
Wählskript	Die 'Wählskript'-Liste
Telefonnummern	Die 'Telefonnummern'-Liste
Max Wiederholung	Der 'Max Wiederholung'-Schalter
Wiederholpause	Der 'Wiederholpause'-Schalter
Wahlpause	Der 'Wahlpause'-Schalter

Lernen

Der 'Lernen'-Schalter

Login-ID / Passwort

Die 'Login-ID' / 'Passwort'-Schalter

Speichern

Die 'Speichern'-Schalter

1.41 Miami_deutsch.guide/NODE_GUI_DIALER_SCRIPT

Wählskript

Die Liste im oberen Bereich der "Wählskript"-Gruppe enthält das Wählskript. Sie können die Einträge ändern, indem Sie sie anklicken, und dann im Textfeld darunter verändern.

Die Schalter am unteren Rand werden verwendet, um Einträge zum Wählskript hinzuzufügen, oder aus dem Wählskript zu löschen.

Für mehr Informationen zu der vom Wählmodul verwendeten Skriptsprache sehen Sie bitte unter
Wählmodul-Befehlssprache
nach.

Die Liste hat ein Kontextmenü, d.h. wenn Sie die rechte Maustaste über der Liste drücken, öffnet sich ein Menü, mit dem Sie das Wählskript von/nach ASCII-Format importieren/exportieren können.

1.42 Miami_deutsch.guide/NODE_GUI_DIALER_PHONE

Telefonnummern

Die "Telefonnummern"-Gruppe funktioniert ähnlich wie die "Wählskript"-Gruppe, hat aber zwei zusätzliche Schalter: "aktiv" und "deaktiv". Aktivierte Telefonnummern haben ein ">>"-Symbol neben ihnen. Nur aktivierte Telefonnummern werden beim Wählen benutzt.

In der Demoversion ist die Anzahl der Telefonnummern begrenzt auf drei. In der registrierten Version gibt es keine derartige Beschränkung.

1.43 Miami_deutsch.guide/NODE_GUI_DIALER_MAX

Max Wiederholung

Wenn keine Verbindung mit einer der angegebenen Telefonnummern aufgebaut werden kann, wartet Miami solange, wie unter

Wiederholpause

angegeben wurde, und beginnt dann wieder von vorne. Allerdings ist ←
die

Maximalzahl der Wiederholversuche begrenzt durch die Zahl im "Max Wiederholung"-Schalter. Danach gibt Miami auf und beendet das Wählmodul.

1.44 Miami_deutsch.guide/NODE_GUI_DIALER_DELAY

Wiederholpause

Wenn keine Verbindung mit einer der angegebenen Telefonnummern aufgebaut werden kann, wartet Miami solange, wie im

"Wiederholpause"-Schalter angegeben wurde, und beginnt dann wieder von vorne.

1.45 Miami_deutsch.guide/NODE_GUI_DIALER_RDELAY

Wahlpause

Dieser Wert gibt die Pause zwischen aufeinander folgenden Wahlversuchen an (also für verschiedene Telefonnummern). Normalerweise sollte dieser Wert null sein, damit Miami sofort die nächste Nummer wählt, wenn eine Nummer besetzt ist.

Allerdings benötigen einige Europäische Modems eine Minimalpause zwischen aufeinander folgenden Wahlversuchen. Wenn Sie eines dieser Modems besitzen, müssen Sie den Wert für "Wahlpause" groß genug für Ihr Modem einstellen.

1.46 Miami_deutsch.guide/NODE_GUI_DIALER_TEACH

Lernen

Der "Lernen"-Schalter startet das Miami-Wählmodul im interaktiven

Modus (d.h. ohne ein Wählskript auszuführen), speichert den gesamten Text, der vom Anwender eingegeben wird, und der vom Modem empfangen wird, und versucht dann, daraus ein geeignetes Wählskript zu erstellen.

In den meisten Fällen wird MiamiInit dazu verwendet, ein Wählskript zu erstellen, nicht "Lernen", aber wenn Ihr Provider die Anwahlprozedur irgendwann verändert, kann es bequemer sein, nur mit "Lernen" ein neues Wählskript zu erzeugen, statt mit MiamiInit eine komplett neue Konfiguration zu erstellen.

1.47 Miami_deutsch.guide/NODE_GUI_DIALER_NAME

Login-ID / Passwort

Die im Wählskript verwendeten Daten für Login-ID und Passwort. Wenn "Wie im Wählmodul" auf der PPP-Seite aktiviert ist, dann werden diese Daten auch für PAP/CHAP verwendet.

1.48 Miami_deutsch.guide/NODE_GUI_DIALER_CAPTURE

Speichern

Wenn Sie das "Speichern"-Schalter aktivieren, und im Eingabefeld einen Dateinamen eingeben, dann speichert das Wählmodul nach dem Wählen ein vollständiges Wählprotokoll in der angegebenen Datei.

1.49 Miami_deutsch.guide/NODE_GUI_DATABASE

Datenbank
=====

Die "Datenbank"-Seite entspricht den Dateien im "db"-Verzeichnis anderer Protokollstacks, d.h. sie erlaubt Ihnen, die meisten TCP-Einstellungen auf Ihrem System zu konfigurieren, z.B. welche Server gestartet werden sollen, welche Anwender das System kennt, und andere Dinge.

Der Auswahlswitcher über der Liste wird dazu verwendet, zwischen verschiedenen Teilen der Datenbank umzuschalten. Für jeden Teil der Datenbank erscheint eine Liste und ein Satz Eingabefelder, mit denen Sie den jeweils aktuellen Eintrag verändern können.

Mit dem Kontextmenü der Datenbank-Liste können Sie Teile der Datenbank von/nach ASCII-Format importieren/exportieren. Damit ist es z.B. möglich, Ihre alten db/#?-Dateien von AmiTCP/AS-225 mit Miami

weiter zu verwenden.

In der registrierten Version können Sie auch Teile der Datenbank sortieren, im/exportieren von/nach Clipboard, und die Datenbank mit einer ASCII-Datei mischen.

Mit den MUI-Modulen können Sie die Reihenfolge der Einträge in der Datenbank ändern, indem Sie sie mit der Maus seitlich aus der Liste heraus ziehen, und dann wieder zurück in die Datenbank, zur gewünschten Position ziehen. Für mehr Informationen hierzu sehen Sie bitte in der MUI-Dokumentation zum Thema "Drag&Drop" nach.

Jeder Eintrag in der Datenbank kann einzeln ein- oder ausgeschaltet werden. Eingeschaltete Einträge haben eine '>>'-Markierung an der linken Seite. Das Ein-und Ausschalten geht entweder durch einen Doppelklick (bei den meisten GUI-Modulen), oder mit den Schaltern 'Aus' und 'Ein'.

Jeder Eintrag in der Datenbank kann als "temporär" markiert werden, indem man den "Temp"-Schalter anklickt. Das bewirkt, daß der entsprechende Eintrag nicht mit abgespeichert wird, wenn man die Einstellungen speichert, und daß - in einigen Fällen - der Eintrag gelöscht wird, wenn man die Verbindung zum Provider neu aufbaut. Das kann nützlich sein, wenn einige der Einträge (z.B. dynamisch ermittelte DNS-Server-Adressen) nicht bei der nächsten Verbindung verwendet werden sollen.

Standardmäßig markiert Miami alle dynamisch ermittelten DNS-Server-Adressen und den dynamischen Hostnamen als temporär.

Teile der Datenbank:

Protokolle	Die 'Protokolle'-Seite
Dienste	Die 'Dienste'-Seite
Rechner	Die 'Rechner'-Seite
Netzwerke	Die 'Netzwerke'-Seite
Domains	Die 'Domains'-Seite
DNS-Server	Die 'DNS-Server'-Seite
InetD	Die 'InetD'-Seite
Anwender	Die 'Anwender'-Seite

Gruppen	Die 'Gruppen'-Seite
Arp	Die 'Arp'-Seite
Socks	Die 'Socks'-Seite
IP-Filter	Die 'IP-Filter'-Seite

1.50 Miami_deutsch.guide/NODE_GUI_DATABASE_PROTOCOLS

Protokolle

Liste aller unterstützten Protokolle (relativ zu IP), bestehend aus einem Protokollnamen, einer Protokoll-ID, und einer optionalen Liste von Aliases. Die Liste entspreche der "etc/protocols" oder "db/protocols" Datei in anderen Protokollstacks.

Diese Tabelle braucht fast nie verändert werden. Sie sollten nie Standardeinträge aus dieser Tabelle entfernen.

1.51 Miami_deutsch.guide/NODE_GUI_DATABASE_SERVICES

Dienste

Liste aller unterstützten Dienste (TCP oder UDP), bestehend aus einem Dienst-Namen, einer Dienst-ID, einem Protokollnamen, und einer optionalen Liste von Aliases. Die Liste entspreche der "etc/services" oder "db/services" Datei in anderen Protokollstacks.

Einige Anwenderprogramme erfordern Änderungen an dieser Tabelle. In der Regel müssen dann Dienste zu dieser Liste hinzugefügt werden. Allerdings sollten Sie nie Standardeinträge aus dieser Liste entfernen.

Insbesondere: Wenn Sie einen Dienst in InetD ausschalten wollen, sollten Sie das nicht tun, indem Sie den Eintrag aus der "Dienste"-Liste entfernen. Die korrekte Methode ist, den Eintrag aus der "InetD"-Liste zu entfernen, nicht aus der "Dienste"-Liste. Ansonsten könnten Sie später gelegentliche Fehler von anderen Programmen bekommen.

1.52 Miami_deutsch.guide/NODE_GUI_DATABASE_HOSTS

Rechner

Liste aller Rechnernamen (und zugehörigen IP-Adressen), bestehend aus einer IP-Adresse, einem Rechnernamen, und einer optionalen Liste von Aliases. Die Liste entspreche der "etc/hosts" oder "db/hosts" Datei in anderen Protokollstacks.

Miami fügt automatisch Einträge für "localhost" und für den Rechnernamen Ihres Amiga zu dieser Liste hinzu. Andere Einträge können manuell hinzugefügt werden, um Name->IP-Übersetzungen schneller zu machen. Allerdings sollten Sie nur Einträge für Namen hinzufügen, die unter Ihrer persönlichen Kontrolle sind. Fügen Sie nie Zuordnungen für andere Rechner auf dem Internet hinzu, weil Sie sonst diese Rechner nicht mehr kontaktieren können, wenn sich die IP-Adressen einmal ändern.

1.53 Miami_deutsch.guide/NODE_GUI_DATABASE_NETWORKS

Netzwerke

Liste aller Netzwerke, bestehend aus einem Netzwerk-Namen, einer Netzwerk-ID, und einer optionalen Liste von Aliases. Die Liste entspreche der "etc/networks" oder "db/networks" Datei in anderen Protokollstacks.

Diese Tabelle wird kaum noch genutzt, und ist nur für Kompatibilität mit alter Software und Diagnosesoftware vorhanden.

1.54 Miami_deutsch.guide/NODE_GUI_DATABASE_DOMAINS

Domains

Liste aller lokalen Domains, bestehend lediglich aus dem Domainnamen. Die Liste entspreche der "etc/domains" oder "db/domains" Datei in anderen Protokollstacks.

Diese Tabelle wird eigentlich nicht von TCP/IP benötigt, aber kann den Umgang mit Miami für den Anwender bequemer machen: Sie erlaubt dem Anwender, vollständige Rechnernamen abzukürzen, indem man nur den Namen des Rechners (ohne Domain) angibt, wenn man einen Rechner ansprechen will.

Beispiel:

Angenommen ein Rechner auf dem lokalen Netz heisst ex1.foo.edu, und

sie greifen häufig auf diesen Rechner zu. Wenn Sie foo.edu zur Domain-Liste hinzufügen, dann können Sie diesen Rechner auch nur mit ex1 ansprechen.

Hinweis: Das Abkürzen von Hostnamen mit dieser Methode funktioniert nur mit Namen, die durch DNS ermittelt werden, nicht mit Namen, die in der "Hosts"-Tabelle gefunden werden. Das bedeutet, wenn Sie z.B. die Domain "foo.edu" hinzufügen, einen Rechner "ex1.foo.edu" auf 10.0.0.1 haben, und auf diesen Rechner nur mit "ex1" zugreifen möchten, dann müssen Sie für diesen Rechner auch einen Eintrag in der "Hosts"-Tabelle erstellen (d.h. der "Hosts"-Tabellen-Eintrag sollte "10.0.0.1 ex1.foo.edu ex1" lauten).

1.55 Miami_deutsch.guide/NODE_GUI_DATABASE_DNSSERVERS

DNS-Server

Liste der DNS-Server, angegeben nur durch die jeweilige IP-Adresse.

DNS-Server werden benutzt, um zu logischen Rechnernamen die dazugehörige IP-Adresse zu ermitteln. Sie sollten mindestens einen DNS-Server in dieser Tabelle eingetragen haben, vorzugsweise einen DNS-Server bei Ihrem Provider, oder zumindest einen in der Nähe.

Wenn Miami beim Aufbau der Verbindung selbst DNS-Server findet, fügt es sie selbständig in diese Liste ein, und markiert sie als "temporär".

1.56 Miami_deutsch.guide/NODE_GUI_DATABASE_INETD

InetD

Liste aller Server die vom eingebauten InetD gestartet werden, bestehend aus einem Dienst-Namen (der einem Eintrag in der "Dienste"-Tabelle entspricht), einem Socket-Typ ("dgram" oder "stream"), einem Wartemodus ("wait", "nowait" oder "dos"), dem Anwender (bei AmigaDOS normalerweise "root"), dem Dateinamen des Servers, dem Prozessnamen des Servers, und einer Liste von Shell-Argumenten, die dem Server übergeben werden. Die Liste entspreche der "etc/inetd.conf" oder "db/inetd.conf" Datei in anderen Protokollstacks.

Der in Miami eingebaute InetD unterstützt die folgenden eingebauten Dienste: "daytime", "time", "echo", "discard", "chargen", "finger" und "auth". "auth" ist identisch zu "identd".

Server für andere (externe) Dienste können von InetD automatisch gestartet werden, wenn man einen entsprechenden Eintrag in dieser Tabelle vornimmt. Wenn Sie externe Server installieren möchten (z.B.

ftpd oder telnetd), dann sehen Sie bitte in der Dokumentation des Servers nach, welches genaue Format der "InetD"-Eintrag erfordert.

Aus Sicherheitsgründen wird empfohlen, daß Sie die "echo", "discard" und "chargin"-Dienste deaktivieren, da diese für Denial-Of-Service-Angriffe mißbraucht werden können.

1.57 Miami_deutsch.guide/NODE_GUI_DATABASE_USERS

Anwender

Liste aller Anwender im System, bestehend aus dem Anwender-Namen, dem Passwort, der Anwender-ID, der Gruppen-ID, dem Realnamen, dem Home-Verzeichnis, und dem Shell-Aufruf. Die Liste entspreche der "etc/passwd" oder "db/passwd" Datei in anderen Protokollstacks.

Normalerweise benötigen Sie nur einen Eintrag in dieser Datei (für Sie selbst), es sei denn, Sie verwenden Server wie ftpd oder telnetd, die anderen Anwendern Zugriff auf Ihren Amiga erlauben.

Passwörter werden in einem verschlüsselten Format gespeichert, und werden nicht direkt in der Liste angezeigt. In der Passwort-Spalte erscheint

`-`

wenn diesem Anwender kein Passwort zugeordnet ist, d.h. wenn Login ohne ein Passwort möglich ist.

`*`

wenn für diesen Anwender kein Login möglich ist.

ein zentriertes `x`

wenn ein gültiges Passwort für diesen Anwender existiert.

Die Prozedur, um das Passwort zu verändern, variiert für verschiedene GUI-Module. Für MUI und einige andere Module klicken Sie auf den "Passwort"-Schalter, um das Passwort zu verändern. Bei anderen Modulen müssen Sie das Passwort direkt in das Eingabefeld eingeben.

Wenn Sie das Eingabefeld leer lassen, wird dem Anwender kein Passwort zugeordnet (angezeigt als `-`). Wenn Sie nur das einzelne Zeichen `*` eingeben, werden Logins blockiert (angezeigt als `*`). In allen anderen Fällen wird der eingegebene Text als Passwort verwendet (angezeigt als zentriertes `x`).

Hinweis: Wenn Sie die Passwort-Datei von AmiTCP importieren, dann werden die Passwörter nicht mit übernommen, d.h. die Passwörter für alle Anwender werden als "leer" gespeichert, und müssen manuell neu eingegeben werden. Das ist notwendig, weil der in AmiTCP verwendete Passwort-Algorithmus aus rechtlichen Gründen nicht in Miami verwendet werden darf. Für mehr Informationen zu diesem Thema sehen Sie bitte in

Austausch von Passwoertern

nach.

1.58 Miami_deutsch.guide/NODE_GUI_DATABASE_GROUPS

Gruppen

Liste der Gruppen im System, bestehend aus einem Gruppennamen, einer Gruppen-ID, und einer optionalen Anwender-Liste. Die Liste entspreche der "etc/group" oder "db/group" Datei in anderen Protokollstacks.

Normalerweise benötigen Sie nur einen Eintrag in dieser Datei (für Sie selbst), es sei denn sie verwenden Server wie ftpd oder telnetd, die anderen Anwendern Zugriff auf Ihren Amiga erlauben.

1.59 Miami_deutsch.guide/NODE_GUI_DATABASE_ARP

Arp

Liste der manuellen Arp-Einträge im System, bestehend aus einer IP-Adresse und einer Hardware-Adresse. Die Hardware-Adresse wird in der üblichen Doppelpunkt-Hex-Notation angegeben (z.B. '01:23:45'). Die Liste entspreche der "etc/ethers" oder "db/ethers" Datei in anderen Protokollstacks.

Arp wird nur bei Bus-/Ring-SANA-II-Geräten verwendet, und Sie brauchen manuelle Arp-Einträge auch nur dann, wenn einer der Rechner in Ihrem Netz Arp nicht unterstützt.

1.60 Miami_deutsch.guide/NODE_GUI_DATABASE_SOCKS

Socks

Liste der SOCKS-Konfigurationseinträge im System, bestehend aus einem Protokolltyp, einem Befehl, einer Liste von Rechnern, einer Liste von Ports, und einer Liste von Proxies. Die Tabelle definiert, welcher Proxy-(SOCKS)-Server, wenn überhaupt, kontaktiert werden soll, abhängig davon, zu welchem Rechner und Port eine Verbindung aufgebaut wird.

Die meisten Anwender brauchen an dieser Tabelle keine Veränderungen vorzunehmen. Wenn Sie SOCKS überhaupt nicht verwenden, dann ignorieren Sie diese Tabelle einfach. Wenn Sie SOCKS verwenden, ist es in den meisten Fällen ausreichend, diese Tabelle leer zu lassen, und nur

einen SOCKS-Server in

Socks

zu konfigurieren. Sie brauchen in dieser

Tabelle nur Änderungen vornehmen, wenn Sie verschiedene SOCKS-Server für verschiedene Rechner oder Ports kontaktieren wollen, oder wenn Sie ein komplexes lokales Netz haben (mit mehrere Subnets), innerhalb des SOCKS-Firewalls.

Jeder Eintrag in dieser Tabelle definiert einen Filter für eine Verbindung oder einen Bind-Versuch, und eine Liste von Proxy-Servern, die kontaktiert werden sollen, wenn eine Verbindung dem Filter entspricht. Für jede Verbindung und jeden Bind-Versuch wird die Tabelle von vorne bis hinten durchsucht, und der erste passende Eintrag wird verwendet, d.h. die Reihenfolge der Einträge ist wichtig. Das Format jedes Eintrags ist folgendermaßen:

Typ

Definiert den Typ der Verbindung, die verwendet werden soll, wenn der Filter zutrifft. Gültige Einträge sind 'socks4' für eine SOCKS V4-Verbindung, 'socks5' für eine SOCKS V5-Verbindung und 'noproxy' für eine direkte Verbindung, ohne SOCKS.

Befehl

Dieses Feld ist Teil des Filters, und kann eine mit Kommata getrennte Liste von Buchstaben sein, ohne Leerzeichen dazwischen. Jeder Buchstabe entspricht einem Zugriffstyp: 'c': connect. 'b': bind. 'u': UDP. 'p': ping. 't': traceroute. '-' beliebiger Zugriff.

Hosts

Dieses Feld ist Teil des Filters, und kann eine Host-Definition in der folgenden Form sein: 'HostIP/Maske': trifft auf die Rechner zu, die von der IP-Adresse und Netzmaske bestimmt werden, z.B. '1.2.3.4/255.255.0.0'. '-': trifft auf alle Rechner zu. 'n1': entspricht 'n1.0.0.0/255.0.0.0'. 'n1.n2': entspricht 'n1.n2.0.0/255.255.0.0'. 'n1.n2.n3': entspricht 'n1.n2.n3.0/255.255.255.0'. '.domain.name': trifft auf alle Rechner zu, die in '.domain.name' enden. 'a.host.name': trifft genau auf den Rechner 'a.host.name' zu.

Ports

Dieses Feld ist Teil des Filters, und kann eine Port-Definition wie folgt sein: '-': trifft auf alle Ports zu. 'Name': trifft auf den angegebenen Dienst zu, z.B. 'ftp'. 'Zahl': trifft auf die angegebene Portnummer zu, z.B. '80'. '[100,1000]': trifft auf Ports 100 bis 1000 zu. '(100,1000)': trifft auf Ports 101 bis 999 zu. '(100,1000]': trifft auf Ports 101 bis 1000 zu.

Proxies

Dieses Feld definiert, welche Proxy-Server kontaktiert werden sollen für Requests, auf die der Filter zutrifft. Es kann eine Komma-getrennte Liste von Server-Einträgen sein. Jeder Server-Eintrag muß mit IP-Adresse oder Hostname angegeben sein, optional gefolgt von einem Doppelpunkt und einer Portnummer.

Diese Tabelle ist nur funktional, wenn 'SOCKS' in

Socks

eingeschaltet ist. Für Zugriffsarten, auf die kein Tabelleneintrag

zutrifft, wird der Standard-SOCKS-Server/Port verwendet, der in
Socks
definiert ist, mit dem SOCKS5-Protokoll.

1.61 Miami_deutsch.guide/NODE_GUI_DATABASE_IPFILTER

IP-Filter

(Diese Option erscheint nur in der registrierten Version.)

Diese Tabelle ermöglicht Ihnen, einige der bei Ihnen ankommenden IP-Pakete heraus zu filtern, oder System-Log-Einträge zu erzeugen, wenn derartige Pakete ankommen. Das erlaubt es Ihnen, einen sehr einfachen Firewall zu implementieren, oder Informationen zu erhalten, wenn jemand versucht, in Ihren Rechner "einzubrechen".

Die Tabelle besteht aus einer Folge von Regeln. Jedes ankommende Paket wird mit jeder der Regeln verglichen, in der Reihenfolge von oben nach unten. Die erste Regel, die auf das Paket zutrifft, bestimmt, ob das Paket ausgefiltert wird, und ob ein Log-Eintrag erzeugt wird. Weitere Regeln werden dann nicht mehr überprüft.

Jeder Tabelleneintrag besteht aus den folgenden Feldern:

- * Ein Protokoll, d.h. 'tcp', 'udp' oder '*' für ein beliebiges Protokoll.
- * Ein Dienst (ein Namen, der in der Dienste-Tabelle erscheint, '*' für 'ein beliebiger Port', oder '\$' für 'ein beliebiger Service-Port', d.h. ein Port, der nicht im Bereich von 1024-5000 liegt). Es ist auch möglich, einen Bereich direkt anzugeben, mit dem '/' als Trennzeichen zwischen dem ersten und letzten Dienst. '1/80' ist z.B. der Bereich von Port 1 bis Port 80.
- * Eine IP-Adresse (die Source-IP-Adresse des Pakets).
- * Eine Netzmaske (die den Bereich der IP-Adressen definiert).
- * Zwei Parameter, die bestimmen, was geschehen soll: Sie können Zugriff erlauben oder verbieten ("y" oder "n"), und optional einen Log-Eintrag erstellen ("y" oder "n").

Beachten Sie, daß Log-Einträge nur für 'tcp'-Dienste erstellt werden, nicht für 'udp'-Dienste.

Hier ist ein Beispiel für eine sinnvolle Startkonfiguration für den IP-Filter:

```
* * 127.0.0.1 (leere Maske) y n
tcp auth *.*.*.* (leere Maske) y n
* $ *.*.*.* (leere Maske) y y
```

Das bewirkt:

Die erste Zeile stellt sicher, daß alle lokal verschickten Pakete (d.h. von Ihrem Amiga zu Ihrem Amiga) ohne Logging erlaubt werden.

Die zweite Zeile erlaubt auch ankommende 'auth'-Anfragen ohne Logging. Das ist nützlich, weil 'auth'- ('identd'-) Anfragen von so vielen httpd-, ftpd- und ircd-Servern gestellt werden, daß sie vermutlich nicht mit Logeinträgen für jede derartige Anfrage belästigt werden möchten.

Die dritte Zeile erlaubt alle übrigen externen Anfragen, aber erzeugt jeweils einen Log-Eintrag, damit Sie erkennen, wenn jemand versucht, auf Ihren Rechner zuzugreifen. Es ist wichtig, daß Sie den Dienst als '\$' angeben, nicht als '*', und zwar weil ftp Rückwärts-Verbindungen (also vom Server zum Client) beim Upload und Download benutzt. Wenn Sie den Dienst als '*' angeben, erhalten Sie einen Log-Eintrag jedes mal wenn Sie eine Datei von/zu einem ftp-Server up/downloaden.

Alle übrigen Pakete (d.h. Pakete von aussen, die zu einem Port zwischen 1024 und 5000 geschickt werden), verwenden die implizite Standardregel, also Zugriff erlauben, und keinen Log-Eintrag erstellen.

1.62 Miami_deutsch.guide/NODE_GUI_TCPIP

TCP/IP

=====

Hostname

Die 'Hostname'- ↔
Gruppe

Realname / Username

Die 'Realname' und 'Username' ↔
'-Schalter

ICMP verwenden

Der 'ICMP verwenden'- ↔
Schalter

Verwende DHCP

Der 'DHCP-Schalter

DNS-Server überprüfen

Der 'DNS-Server überprüfen'- ↔
Schalter

Pseudo-IP

Der 'Pseudo-IP'- ↔
Schalter

T/TCP	Der 'T/TCP'- Schalter ←
Domain automatisch hinzufügen	Der 'Domain automatisch hinzufügen'- Schalter ←
'down' wenn Offline	Der ' 'down' wenn Offline'- Schalter ←
Schutz vor Dauer-Ping	Der 'Schutz vor Dauer-Ping'- Schalter ←
Zeit ermitteln	Die 'Zeit ermitteln'- Schalter ←

1.63 Miami_deutsch.guide/NODE_GUI_TCPIP_HOSTNAME

Hostname

In den meisten Fällen sollten Sie den Schalter "dynamisch" einschalten. Miami ermittelt dann automatisch den Hostnamen Ihres Amiga durch "reverse DNS lookup", nachdem die Verbindung aufgebaut wurde.

Einige Provider unterstützen allerdings kein "reverse DNS lookup", oder weisen ihren Kunden feste Hostnamen zu, die nicht im DNS gelistet sind. In diesem Fall schalten Sie "dynamisch" aus, und geben Sie den Hostnamen manuell ein.

1.64 Miami_deutsch.guide/NODE_GUI_TCPIP_NAME

Realname / Username

In diesen Eingabefeldern sollten Sie Ihren Realnamen (z.B. "Joe Smith"), und den Anwendernamen auf Ihrem Amiga (z.B. "jsmith") eingeben.

Obwohl Sie theoretisch hier beliebige Namen verwenden können, ist es üblich, die korrekten Namen zu verwenden, keine Phantasienamen.

Einige Programme fragen Miami nach Anwender-Informationen zu Ihrem Anwendernamen. Damit diese Programme korrekt funktionieren, sollten Sie darauf achten, daß im "Anwender"-Teil auf der "Datenbank"-Seite ein

Eintrag existiert, der dem hier angegebenen Usernamen entspricht.

1.65 Miami_deutsch.guide/NODE_GUI_TCPIP_ICMP

ICMP verwenden

Wenn dieser Schalter aktiviert ist, verwendet Miami ICMP-"ping"-Pakete, um die Gültigkeit von IP-Adressen, DNS-Servern usw. zu überprüfen.

Dieser Schalter sollte normalerweise eingeschaltet sein, weil er zusätzlich vor Fehlkonfigurationen schützt.

Allerdings kann es sein, daß Sie diesen Schalter ausschalten müssen, wenn Sie mit Hilfe eines TCP-Emulators wie TIA verbunden sind, da nicht alle TCP-Emulatoren ICMP unterstützen.

1.66 Miami_deutsch.guide/NODE_GUI_TCPIP_BOOTP

Verwende DHCP

Wenn Ihr Provider dynamische IP-Adressen verwendet, gibt es verschiedene Möglichkeiten für Miami, die korrekte (dynamische) IP-Adresse zu finden.

Für PPP-Verbindungen geschieht das normalerweise als Teil des PPP-Protokolls. Für (C)SLIP-Verbindungen ist das aber nicht möglich, d.h. (C)SLIP-Verbindungen verwenden manchmal ein Protokoll mit dem Namen "DHCP". Eine weitere Möglichkeit ist, die IP-Adresse aus dem Anwahl-Protokoll zu entnehmen.

Wenn Sie MiamiInit zum Konfigurieren der Leitung benutzt haben, dann lassen Sie diesen Schalter am besten in der Standardeinstellung. Wenn Sie Miami manuell konfiguriert haben, sollten Sie "DHCP" zunächst eingeschaltet lassen, und dann später testen, ob es auch ohne "DHCP" noch funktioniert.

Wenn Miami Ihre IP-Adresse auch ohne DHCP erkennen kann, sollte "DHCP" ausgeschaltet sein, da das Aufbauen der Verbindung dann schneller geht.

1.67 Miami_deutsch.guide/NODE_GUI_TCPIP_VERIFYDNS

DNS-Server überprüfen

Normalerweise versucht Miami, die IP-Adressen aller DNS-Server auf Gültigkeit zu überprüfen. Allerdings kann das Probleme mit einigen Internet-Providern verursachen, wenn deren DNS-Server eine schlechte Verbindung haben oder kurz nach Verbindungsaufbau noch nicht auf Anfragen reagieren.

Wenn Sie den Schalter "DNS-Server überprüfen" deaktivieren, dann überspringt Miami die DNS-Überprüfung beim Verbindungsaufbau.

1.68 Miami_deutsch.guide/NODE_GUI_TCPIP_FAKEIP

Pseudo-IP

Wenn Sie mit Hilfe eines TCP-Emulators wie TIA oder Slirp verbunden sind, und dieser Emulator Ihnen keine "echte" IP-Adresse zuweist, sondern eine Pseudo-IP-Adresse, dann sollten Sie diesen Schalter aktivieren.

Wenn dieser Schalter aktiviert ist, ermittelt Miami Ihren Hostnamen auf der Basis der Gateway-IP-Adresse, nicht ihrer lokalen Pseudo-IP-Adresse.

1.69 Miami_deutsch.guide/NODE_GUI_TCPIP_TTCP

T/TCP

(Diese Option ist nur in der registrierten Version anwählbar.)

T/TCP (TCP for Transactions) ist eine Erweiterung zu TCP, die die Geschwindigkeit von Netzwerkanwendungen signifikant erhöhen kann, insbesondere für Web-Browser, wenn sowohl der Browser als auch der Server T/TCP unterstützen.

Registrierte Anwender sollten diese Option einschalten, um von dem Geschwindigkeitszuwachs zu profitieren. Allerdings haben einige wenige PPP-Server Probleme mit den erweiterten TCP-Paketten, die T/TCP erzeugt. Wenn Miami also nach dem Einschalten von T/TCP nicht mehr korrekt funktioniert, dann schalten Sie diese Option aus - oder suchen Sie sich einen neuen Provider.

1.70 Miami_deutsch.guide/NODE_GUI_TCPIP_ADDDOMAIN

Domain automatisch hinzufügen

Wenn dieser Schalter aktiviert ist, fügt Miami automatisch die Domain Ihres Hostnamen (d.h. alles hinter dem ersten '.') zu Miami's "Domains"-Datenbank hinzu.

Weder Miami noch irgendwelche Programme benötigen einen derartigen Eintrag, aber es kann für den Anwender bequemer sein, Abkürzungen für Rechnernamen zu verwenden. Für mehr Details sehen Sie unter

Die 'Datenbank'-Seite
nach.

1.71 Miami_deutsch.guide/NODE_GUI_TCPIP_DOWN

'down' wenn Offline

(Diese Option ist nur in der registrierten Version anwählbar.)

In der unregistrierten Version unterbricht Miami grundsätzlich alle TCP-Verbindungen, wenn die Schnittstelle offline geht. In der registrierten Version erhält Miami die Verbindungen in so einer Situation aufrecht. Das hat den Vorteil, daß Sie möglicherweise schnell eine neue Modem-Verbindung aufbauen, und Ihre bisherigen TCP-Verbindungen weiter nutzen können.

Der Nachteil ist, daß das Aufrechterhalten von TCP-Verbindungen bei einer Schnittstelle im Offline-Zustand auch zur Folge hat, daß eine Applikation nicht mehr erkennen kann, ob Miami online oder offline ist, d.h. wenn eine Applikation versucht, eine Verbindung aufzubauen, während die Schnittstelle offline ist, wird kein Fehler erzeugt, sondern die Applikation wartet nur auf einen Timeout.

Wenn Ihnen dieses Verhalten nicht gefällt, und Sie lieber hätten, wenn im Offline-Zustand Fehlermeldungen erzeugt werden, dann aktivieren Sie diese Option.

1.72 Miami_deutsch.guide/NODE_GUI_TCPIP_PING

Schutz vor Dauer-Ping

(Diese Option ist nur in der registrierten Version anwählbar.)

Miami enthält eine einfache Heuristik, um die Effekte von "Denial-of-Service-Attacks" zu vermindern, die durch Dauer-Pings ("ping

flooding") entstehen:

Wenn diese Option aktiviert ist, und ein Anwender versucht, Ihren Rechner mit Dauer-Pings zu belästigen (entweder mit sehr grossen Pings, oder mit sehr schnell aufeinander folgenden Pings), dann erzeugt Miami einen Log-Eintrag, um Sie davon zu informieren, und erzeugt dann für einige Zeit keine Ping-Antworten an diesen Anwender mehr, bis der Anwender für einige Zeit mit den Dauer-Pings aufgehört hat.

Beachten Sie, daß es unmöglich ist, die Dauer-Pings ganz zu verhindern, d.h. den Anwender davon abzuhalten, Ihre Modem-Kapazität zu verschwenden. Das einzige, was Miami in so einer Situation tun kann, ist, keine Antworten mehr zu schicken, und Sie zu informieren (damit Sie einen neuen Modemport anwählen können). Es ist Miami nicht möglich, die Verschwendung der Modembandbreite an sich zu verhindern. Das ginge nur durch einen Filter bei Ihrem Internet-Provider.

1.73 Miami_deutsch.guide/NODE_GUI_TCPIP_GETTIME

Zeit ermitteln

Wenn Ihr Amiga nicht mit einer batteriegepufferten Echtzeituhr ausgestattet ist, sollten Sie den Schalter "Zeit ermitteln" aktivieren, und Namen oder IP-Adresse eines Servers eingeben, der den "time"-Dienst unterstützt. Wenn Sie nicht sicher sind, welchen Server Sie eingeben sollen, versuchen Sie es mit dem Server, den Sie für Email oder News verwenden.

Wenn Sie diese Funktionen nutzen, dann achten Sie darauf, daß die Environment-Variable "ENV:TZ" korrekt eingestellt ist, d.h. normalerweise zu einem Text der Form "EST5" oder "EST4EDT" (während der Sommerzeit). Das ist wichtig, da der Server die Zeit im GMT-(UTC-) Format übermittelt, und dieses Format von Miami an die lokale Zeitzone angepaßt werden muß. Für mehr Informationen hierzu lesen Sie bitte

Zeitzone-Informationen

.

Verwenden Sie diese Funktion nicht, wenn Ihr Amiga eine batteriegepufferte Echtzeituhr hat, denn dann ist es möglich, daß beim Stellen der Zeit die Uhrzeit kurzzeitig rückwärts läuft. Das kann zu Problemen mit Programmen führen, die GetSysTime() für Berechnungen benötigen, und kann zu Abstürzen und anderen Schwierigkeiten führen.

1.74 Miami_deutsch.guide/NODE_GUI_EVENTS

Ereignisse

=====

Miami erlaubt es, auf verschiedene Ereignisse wie "offline", "online" usw. auf verschiedene Art zu reagieren, z.B. durch das Ausführen von ARexx-Skripts, das ikonifizieren des Miami-Fensters usw.

Die genauen Ereignisse, auf die Miami reagieren kann, sind:

Start

Programmstart.

Ende

Programmende.

aktives Offline

Abbauen der Verbindung, ausgelöst vom Anwender, z.B. durch klicken auf den "Offline"-Schalter, oder durch den ARexx-Befehl "OFFLINE".

passives Offline

Abbauen der Verbindung, wenn das Modem oder der Provider auflegen.

Online

Aufbauen der Verbindung, d.h. erfolgreicher Aufbau einer Verbindung zum Internet, einschließlich dem Start aller benötigten Protokolle.

erfolgloser Online-Versuch

Ein Versuch, eine Verbindung aufzubauen, ist aus irgend einem Grund fehlgeschlagen, z.B. weil alle Telefonleitungen besetzt waren, oder weil die maximale Anzahl Wiederholversuche erreicht wurde.

Miami kann in der folgenden Art reagieren. Nicht alle Optionen machen Sinn als Reaktion auf jedes Ereignis, so daß nur eine Untermenge aller Optionen für jedes Ereignis tatsächlich anwählbar sind:

ARexx

ARexx-Skript starten.

Shell

AmigaDOS Shell-Skript starten.

GUI ikonifizieren

Miami-Fenster schliessen, und stattdessen ein Piktogramm anzeigen.

GUI schliessen

Miami-Fenster schliessen, und GUI-Modul aus dem Speicher entfernen.

Auto-Online

Versuch, automatisch eine Verbindung aufzubauen.

Blinken

Bildschirm aufblinken lassen, oder Signalton ausgeben, wie in den Amiga-System-Preferences definiert.

Anzeigen

Miami-Fenster öffnen, und Piktogramm entfernen.

In der Demoversion von Miami sind die Optionen "ARexx" und "Shell" nicht verfügbar, und "auto-online" ist nicht als Reaktion auf "passives Offline" verfügbar.

Im Eingabefeld "Konsolename" können Sie die Ein-/Ausgabedatei angeben, die von ARexx und Shellskripten benutzt wird. Es sollte etwas in der Form "CON:1/1/400/100/Titel" sein.

1.75 Miami_deutsch.guide/NODE_GUI_MODEM

Modem

=====

Init-String

Der 'Init-String'-Schalter

Exit-String

Der 'Exit-String'-Schalter

Wählpräfix

Der 'Wählpräfix'-Schalter

Wählsuffix

Der 'Wählsuffix'-Schalter

Nullmodem

Der 'Nullmodem'-Schalter

1.76 Miami_deutsch.guide/NODE_GUI_MODEM_INIT

Init-String

Der Initialisierungsstring für Ihr Modem, normalerweise eingestellt von MiamiInit.

1.77 Miami_deutsch.guide/NODE_GUI_MODEM_EXIT

Exit-String

Der String, der beim Programmende zum Modem geschickt wird. Die

meisten Anwender benötigen diese Option nicht, aber sie kann nützlich sein, wenn mehrere Programme einen Modemport teilen, und das Modem auf Standardeinstellungen zurück gesetzt werden muß, wenn Miami beendet wird.

1.78 Miami_deutsch.guide/NODE_GUI_MODEM_PREFIX

Wählpräfix

Der von Ihrem Modem verwendete Befehl zum Wählen, d.h. der String, der der Telefonnummer vorangestellt wird. Normalerweise ist das "ATDT" oder "ATDP".

1.79 Miami_deutsch.guide/NODE_GUI_MODEM_SUFFIX

Wählsuffix

Der String, der an die Telefonnummer angehängt wird, um den Wählbefehl zu vervollständigen. Das ist normalerweise '\r' .

1.80 Miami_deutsch.guide/NODE_GUI_MODEM_NULLMODEM

Nullmodem

Miami nimmt normalerweise an, daß Sie ein Modem an Ihrem seriellen Port angeschlossen haben. Wenn Ihr Amiga mittels Nullmodem-Kabel direkt mit einem anderen Rechner verbunden ist, müssen Sie diesen Schalter aktivieren. Er verhindert daß Miami Modembefehle ("AT-Befehle") verschickt, und auf Antworten wie "OK" oder "CONNECT" wartet.

Wenn "Nullmodem" aktiviert ist, ändert sich die Bedeutung des "Verwende CD"-Schalters auf der "Schnittstelle"-Seite:

- * Wenn Ihr Rechner mit einem Computer verbunden ist, der eine Login-Sequenz benötigt, um SLIP/PPP zu starten, sollten Sie "Verwende CD" deaktivieren. Miami führt dann erst das auf der "Wählmodul"-Seite definierte Wählskript aus, aber ohne eine Telefonnummer zu wählen. Diese Option ist nützlich, wenn Sie mit einer Unix- oder Linux-Box verbunden sind, die einen getty mit Login/Passwort-Abfrage auf dem seriellen Port betreibt.
- * Wenn Ihr Rechner mit einem Computer verbunden ist, der auf dem seriellen Post direkt SLIP/PPP benutzt (z.B. ein anderer Amiga mit Miami), dann sollten Sie "Verwende CD" aktivieren. Miami

überspringt dann das Wählskript und versucht sofort, das Protokoll auszuhandeln.

1.81 Miami_deutsch.guide/NODE_GUI_LOGGING

Protokollierung

=====

Konsole

Der 'Konsole'-Schalter

Datei

Der 'Datei'-Schalter

syslog.library verwenden

Der 'syslog.library verwenden'-Schalter

Telefon-Log

Die 'Telefon-Log'-Schalter

PPP-Log

Die 'PPP-Log'-Schalter

1.82 Miami_deutsch.guide/NODE_GUI_LOGGING_CONSOLE

Konsole

Im diesem Eingabefeld können Sie den AmigaDOS-Dateinamen eines Fensters eingeben, das Miami für System-Log-Nachrichten verwendet. Diese Datei wird offen gehalten, nachdem die erste Systemnachricht ausgegeben wurde, d.h. Sie sollten die "CON:"-Optionen "/AUTO/CLOSE" verwendet, damit Sie das Fenster schliessen können, ohne daß alte Systemnachrichten verloren gehen.

1.83 Miami_deutsch.guide/NODE_GUI_LOGGING_FILE

Datei

In diesem Eingabefeld können Sie den AmigaDOS-Dateinamen einer Datei angeben, in der Miami System-Log-Nachrichten speichert. Wenn die Datei bereits existiert, fügt Miami neue Nachrichten am Ende an, d.h. alte Dateiinhalte werden nicht gelöscht.

1.84 Miami_deutsch.guide/NODE_GUI_LOGGING_SYSLOG

syslog.library verwenden

Wenn Sie diesen Schalter aktivieren, versucht Miami, auf syslog.library zuzugreifen, um das System-Log zu erzeugen. syslog.library ist Teil des SysLog-Pakets von Petri Nordlund.

1.85 Miami_deutsch.guide/NODE_GUI_LOGGING_PHONE

Telefon-Log

Miami kann Online- und Offline-Ereignisse protokollieren, um bei der Verwaltung von Telefonrechnungen zu helfen.

Mit den beiden "Telefon-Log"-Schaltern kann man das Telefon-Logging einschalten, und den Namen der Datei angeben, an der Miami neue Log-Einträge anfügt.

Im Moment wird nur das ASCII-Format unterstützt, mit den folgenden Einträgen:

```
Online: 27.07.1996 17:48:11 (5551234)
Passive offline: 27.07.1996 17:48:11
Active offline: 27.07.1996 17:48:11
Reconnect: 27.07.1996 17:48:11
```

Der "Online"-Eintrag enthält die gewählte Telefonnummer in "()". "Reconnect" tritt auf, wenn Miami eine Verbindung aufbaut, ohne neu zu wählen, z.B. nach einem Reset des Rechners.

Der Unterschied zwischen "passivem" und "aktivem" Offline ist, daß "aktives" Offline freiwillig ist, d.h. das Resultat eines "OFFLINE"-ARexx-Befehls, eines Klicks auf den "Offline"-Schalter usw. Ein "passives" Offline tritt auf, wenn das Modem auflegt, oder der Internet-Provider die Verbindung abbaut.

1.86 Miami_deutsch.guide/NODE_GUI_LOGGING_PPP

PPP-Log

(Diese Option ist nur in der registrierten Version anwählbar.)

Der PPP-Log-Schalter erlaubt Ihnen, eine Datei anzugeben, in der Miami die Phases des Verbindungsaufbaus von PPP protokolliert. Die Daten werden in leicht lesbarer Form protokolliert, d.h. nicht als Hexdump. Nur die Verbindungsaufbau-Phase, d.h. bis die LCP- und IPCP-Automaten den 'Open'-Status erreichen. Danach wird die Protokollierung beendet.

Die wichtigste Aufgabe des PPP-Logs ist es, Kompatibilitätsprobleme in PPP zu erkennen, und PPP-Optionen für einzelne Server zu optimieren.

1.87 Miami_deutsch.guide/NODE_GUI_WINDOWS

```

Fenster
=====

Ende-Requester                               Die 'Ende-Requester'-Schalter
Offline-Requester                             Der 'Offline-Requester'-Schalter
Fehler-Requester                             Der 'Offline-Requester'-Schalter
Wählmodul                                     Die 'Wählmodul'-Schalter

```

1.88 Miami_deutsch.guide/NODE_GUI_WINDOWS_REQQUIT

```

Ende-Requester
-----

```

Sie können konfigurieren, wann Miami einen Ende-Requester anzeigen soll:

- * immer
 - * wenn noch Programme laufen, die Miami benutzen
 - * wenn Miami online ist
- oder Kombinationen davon.

1.89 Miami_deutsch.guide/NODE_GUI_WINDOWS_REQOFFLINE

Offline-Requester

Wenn Sie diesen Schalter aktivieren, fragt Miami vor dem Auflegen der Leitung noch einmal nach.

1.90 Miami_deutsch.guide/NODE_GUI_WINDOWS_REQERRORS

Fehler-Requester

Normalerweise zeigt Miami einen Fehlerrequester an, wenn beim Wählen oder bei der Konfiguration der Verbindung ein Fehler auftritt. Wenn Sie diesen Schalter ausschalten, werden derartige Fehler ignoriert, und es erscheint kein Requester.

1.91 Miami_deutsch.guide/NODE_GUI_WINDOWS_DIALER

Wählmodul

Das Standard-Wählfenster hat drei Teile: Einen Hilfetext oben, einige Schalter darunter, und das Dialogfenster ganz unten. Mit den drei "Wählmodul"-Schaltern können Sie jeden dieser Teile einzeln einschalten oder ausschalten.

Wenn das Dialogfenster ausgeschaltet wird, zeigt Miami stattdessen eine einzelne Zeile an, in der der gerade ausgeführte Befehl erscheint.

Mit dem 'Fenster aktivieren'-Schalter teilen Sie Miami mit, daß Miami Wählfenster und Fehler-Requester automatisch aktivieren soll, wenn diese erscheinen.

1.92 Miami_deutsch.guide/NODE_GUI_GUI

GUI (Benutzeroberfläche)
=====

Diese Seite definiert die Einstellungen für die Benutzeroberfläche von Miami, d.h. Hotkeys, Iconifizierung und die zu verwendende Benutzeroberfläche.

Wichtig: Geben Sie Einstellungen, die die Benutzeroberfläche betreffen immer auf dieser Seite an, nicht in anderen Programmen. Auch wenn Sie MUI benutzen, verwenden Sie nicht die Iconify- und Hotkey-Funktion in den MUI-Einstellungen für Miami. Diese Funktionen

funktionieren nicht mit Miami, da Miami die Iconifizierung selbständig handhabt, ohne MUI.

Hotkey	Der 'Hotkey'-Schalter
Icon anzeigen	Der 'Icon anzeigen'-Schalter
Menue anzeigen	Der 'Menue anzeigen'-Schalter
Keine GUI beim Start	Der 'Keine GUI beim Start'-Schalter
Online-Icon	Der 'Online-Icon'-Schalter
Offline-Icon	Der 'Offline-Icon'-Schalter
GUI-Modul	Der 'GUI-Modul'-Schalter
Umschalten	Der 'Umschalter'-Schalter

1.93 Miami_deutsch.guide/NODE_GUI_GUI_HOTKEY

Hotkey

Hier können Sie den Commodities-Hotkey definieren, mit dem die Miami-Benutzeroberfläche iconifiziert und deiconifiziert werden kann. Es wird normale Commodities-Syntax verwendet, z.B. entspricht der Hotkey 'ctrl alt m' dem gleichzeitigen Drücken der Ctrl-Taste, einer der beiden Alt-Tasten, und der Taste 'm'. 'ctrl alt m' ist auch der Standardwert.

1.94 Miami_deutsch.guide/NODE_GUI_GUI_SHOWICON

Icon anzeigen

Wenn dieser Schalter aktiviert ist, wird auf der Workbench ein AppIcon angezeigt, wenn Miami iconifiziert ist.

1.95 Miami_deutsch.guide/NODE_GUI_GUI_SHOWMENU

Menue anzeigen

Wenn dieser Schalter aktiviert ist, wird ein Eintrag 'Miami' im 'Tools'-Menue der Workbench erzeugt, wenn Miami iconifiziert ist.

1.96 Miami_deutsch.guide/NODE_GUI_GUI_ONSTARTUP

Keine GUI beim Start

Wenn dieser Schalter aktiviert ist, lädt Miami das GUI-Modul nicht beim Programmstart, und öffnet das Hauptfenster auch nicht. Diese Funktion ist am nützlichsten, wenn man sie mit der Funktion 'automatisches Online beim Programmstart' verknüpft. Siehe Ereignisse für mehr Informationen hierzu.

1.97 Miami_deutsch.guide/NODE_GUI_GUI_ONLINEICON

Online-Icon

Hier können Sie das Icon ('.info'-Datei) angeben, das Miami als AppIcon verwendet, wenn Miami online ist. Der Standard (leeres Eingabefeld) ist ein in Miami eingebautes Icon.

1.98 Miami_deutsch.guide/NODE_GUI_GUI_OFFLINEICON

Offline-Icon

Hier können Sie das Icon ('.info'-Datei) angeben, das Miami als AppIcon verwendet, wenn Miami offline ist. Der Standard (leeres Eingabefeld) ist ein in Miami eingebautes Icon.

1.99 Miami_deutsch.guide/NODE_GUI_GUI_GUI

GUI-Modul

Mit diesem Schalter können Sie eins von mehreren installierten GUI-Modulen auswählen. Miami merkt sich Ihre Wahl, und speichert sie in der Einstellungen-Datei (wenn Sie die Einstellungen später speichern), schaltet aber nicht sofort auf das neue GUI-Modul um. Um das zu tun, klicken Sie auf

Umschalten

.

1.100 Miami_deutsch.guide/NODE_GUI_GUI_SWITCH

Umschalten

Wenn Sie auf diesen Schalter klicken, schaltet Miami auf das ausgewählte GUI-Modul um. (Was wirklich passiert ist: Miami wird iconifiziert, entfernt das aktuelle GUI-Modul, lädt das neue GUI-Modul, und deiconifiziert mit dem neuen Modul).

1.101 Miami_deutsch.guide/NODE_GUI SOCKS

Socks

=====

Auf dieser Seite können sie die SOCKS-Client-Unterstützung von Miami konfigurieren. Wenn Sie noch nie von SOCKS gehört haben, brauchen Sie es vermutlich nicht. SOCKS ist ein Proxy-System, das Rechnern innerhalb eines Firewalls erlaubt, Verbindungen zu Rechnern außerhalb des Firewalls aufzubauen.

Miami's SOCKS-Implementierung erlaubt es Amiga-TCP/IP-Clients, Verbindungen "durch" einen Firewall transparent aufzubauen, ohne daß dafür besondere Unterstützung in den Clients notwendig ist. Wenn Ihr Netzanbieter einen SOCKS-Firewall verwendet, dann fragen Sie ihn nach der IP-Adresse des SOCKS-Servers, und nach dem SOCKS-Usernamen und Passwort (falls der SOCKS-Server passwort-geschützt ist), und konfigurieren Sie Miami auf dieser Seite entsprechend.

Die Einstellungen auf dieser Seite sind die Grundeinstellungen für Ihren Rechner. Sie können SOCKS in mehr Detail unter

Datenbank/Socks
konfigurieren.

SOCKS aktivieren

	Der 'SOCKS aktivieren'-Schalter
Standard-SOCKS-Server	Die 'Standard-SOCKS-Server'-Schalter
Maximale Syslog-Stufe	Der 'Maximale Syslog-Stufe'-Schalter
Authentifizierung	Die 'Authentifizierung'-Schalter

1.102 Miami_deutsch.guide/NODE_GUI SOCKS_ENABLE

SOCKS aktivieren

Wenn dieser Schalter aktiviert ist, verwendet Miami SOCKS, um Verbindungen zu Rechnern aufzubauen, die nicht direkt durch eine Schnittstelle erreichbar sind. Sie müssen außerdem die IP-Adresse und den Port des SOCKS-Servers und, bei einigen SOCKS-Servern, Authentifizierung konfigurieren.

1.103 Miami_deutsch.guide/NODE_GUI SOCKS_SERVER

Standard-SOCKS-Server

Diese Schalter bestimmen die IP-Adresse und Portnummer des Standard-SOCKS-Servers in Ihrem Netz. Die Portnummer für SOCKS ist normalerweise 1080.

1.104 Miami_deutsch.guide/NODE_GUI SOCKS_MAXLOG

Maximale Syslog-Stufe

Mit diesem Schalter definieren Sie, wie viele Hilfsmeldungen Sie von SOCKS erhalten möchten. Der Schalter sollte normalerweise auf "none" oder "error" stehen. Höhere Einstellungen sind nützlich um zusätzliche Informationen bei der Fehlersuche zu erhalten.

1.105 Miami_deutsch.guide/NODE_GUI SOCKS_AUTH

Authentifizierung

Mit diesem Schalter geben Sie an, welche Authentifizierungsdaten an den SOCKS-Server geschickt werden. Die folgenden Einstellungen sind möglich:

keine

Keine Authentifizierungsdaten werden übertragen. Das funktioniert nur mit SOCKS-Servern, die keine Authentifizierung erfordern.

wie im Wählmodul

Miami übergibt dem SOCKS-Server die gleiche Username/Passwort-Kombination, die im Wählmodul eingestellt ist.

Username/Passwort

Miami übergibt dem SOCKS-Server die Username/Passwort-Kombination, die unten eingetragen wird.

1.106 Miami_deutsch.guide/NODE_GUI_MISC

Sonstiges

=====

Es gibt drei weitere Schalter in Miami, die noch nicht in den vorigen Abschnitten beschrieben worden sind:

- * "Online": Miami beginnt mit dem Wählen, und versucht, eine Verbindung aufzubauen.
- * "Offline": Miami bricht die Verbindung ab und legt auf.
- * Eine Liste oder eine Schalterreihe befindet sich auf der linken Seite des Miami-Fensters. Mit dieser Liste können Sie eine der Konfigurations-Seiten anwählen.

1.107 Miami_deutsch.guide/NODE_DIALERLANG

Wählmodul-Befehlssprache

Die folgenden Befehle werden vom Wählmodul unterstützt:

ABORT "Text1", "Text2", ...

Gibt eine Liste von Texten an, bei deren Empfang Miami den Wählvorgang ganz abbricht, z.B. "NO DIALTONE" vom Modem.

ASKPASSWORD

Erzeugt einen Requester, der Sie nach dem Passwort fragt.

DELAY Sekunden

Wartet die angegebene Anzahl Sekunden.

DIALNEXT "Text1","Text2",...

Gibt eine Liste von Texten an, bei deren Empfang Miami die nächste Telefonnummer wählt, z.B. "BUSY" vom Modem.

PARSEPASSWORD "Endzeichen"

Parsed alle vom Modem ankommenden Zeichen bis zu (nicht einschließlich) <Endzeichen>, und ersetzt dann das aktuelle Passwort durch diesen Text. Dieser Befehl kann nützlich sein für Einmal-Passwort-Systeme, die das Passwort für die nächste Verbindung während des Login-Vorgangs senden.

REDIAL "Text1","Text2",...

Gibt eine Liste von Texten an, bei deren Empfang Miami die Leitung auflegt, und die aktuelle Telefonnummer noch einmal wählt, z.B. "BUSY" vom Modem.

SAVECONFIG

Speichert die aktuelle Konfiguration (Einstellungen) auf Disk. Dieser Befehl wird normalerweise in Verbindung mit PARSEPASSWORD benutzt, um die Einstellungen mit dem neuen Passwort zu speichern.

SEND "Text"

Sender <Text> zum Modem. Ein Linefeed/Carriage Return wird nicht automatisch angefügt. Miami erkennt die folgenden Standard-Kontrollsequenzen: \", \\, \r, \n. Ausserdem werden "\u" und "\p" unterstützt, um die aktuelle Login-ID oder das aktuelle Passwort zu senden.

SEENDBREAK

Sendet ein "Break"-Signal zum seriellen Port. Einige Terminalserver verwenden das Break-Signal, um in den Befehlsmodus umzuschalten.

SENDPAD "text",Länge

Schickt <text> zum Modem, aufgefüllt mit Leerzeichen, bis die in <Länge> angegebene Länge erreicht wird. Beispiel: 'SENDPAD "abc",5' würde "abc " senden.

SENDPASSWORD

Sendet das aktuelle Passwort, gefolgt von einem "\r".

SENDUSERID

Sendet die aktuelle Login-ID, gefolgt von einem "\r".

TIMEOUT Sekunden

Gibt an, wie lange Miami beim nächsten WAIT oder WAITPPP wartet, bevor das Wählskript abgebrochen wird.

WAIT "Text"

Wartet darauf, daß "Text" vom Modem empfangen wird.

WAITCONNECT

Wartet auf eine CONNECT-Meldung vom Modem, und den darauf folgenden

Text (normalerweise die Angabe der Leitungsgeschwindigkeit). Dieser Befehl ist identisch zu 'WAIT "CONNECT"', außer das Miami auch den der 'CONNECT'-Meldung folgenden Text in einen internen Puffer kopiert, und ihn später in der Statuszeile anzeigt. Auf diese Weise kann man mit vielen Modems erfahren, mit welcher Geschwindigkeit die Modemverbindung aufgebaut wurde.

WAITPPP

Wartet darauf, daß der Server auf PPP umschaltet.

Bei den Befehlen "ABORT", "DIAL" und "DIALNEXT" kann auch das Schlüsselwort "TIMEOUT" (ohne die Anführungszeichen) angegeben werden, statt eines Textes in Anführungszeichen, z.B.

```
ABORT "NO CARRIER",TIMEOUT
```

Das bedeutet, daß Miami das Wählskript abbricht, wenn ein Timeout auftritt. Andere Optionen sind, die aktuelle Nummer erneut anzuwählen, oder die nächste Nummer zu wählen, wenn ein Timeout auftritt.

1.108 Miami_deutsch.guide/NODE_AREXX

ARexx-Schnittstelle

Der Name des ARexx-Ports von Miami ist "MIAMI.1". Im Moment unterstützt Miami alle Standard-ARexx-Befehle für MUI-Applikationen ("QUIT", "HIDE", "DEACTIVATE", "SHOW", "ACTIVATE", "INFO", "HELP") und die folgenden zusätzlichen Befehle:

CHANGEDB

Neulesen der Datei "ENVARC:MiamiChangeDB", um die Einstellungen auf den neuesten Stand zu bringen. Bitte sehen Sie unter

Client-Einstellungen
für nähere Informationen nach.

GETCONNECT

Gibt den Verbindungstext zurück, der nach der 'CONNECT'-Meldung vom Modem zurückgegeben wurde. In den meisten Fällen enthält dieser Text einen Hinweis zur Übertragungsgeschwindigkeit.

GETCONNECTTIME

Gibt die Anzahl Sekunden, seit Miami die 'CONNECT'-Meldung vom Modem empfangen hat, in der 'result'-Variable zurück.

GETONLINETIME

Gibt die Anzahl Sekunden, die Miami online ist, in der 'result'-Variable zurück.

GETSETTINGSNAME

Gibt den Dateinamen der aktuellen Konfigurationsdatei zurück.

ISONLINE

Testet ob Miami online ist, und setzt den Fehlercode ("RC")

entsprechend. 1 bedeutet: Miami ist online. 0 bedeutet: Miami ist offline.

KILLGUI

Iconifiziert die Benutzeroberfläche, und entfernt das GUI-Modul.

LOADSETTINGS file/a

Lädt die angegebene Einstellungen-Datei.

OFFLINE

Abbau der Verbindung und auflegen. Gleiche Bedeutung wie das Anklicken des "Offline"-Schalters.

ONLINE

Aufbau der Verbindung. Gleiche Bedeutung wie das Anklicken des "Online"-Schalters.

QUITFORCE

Der "QUIT"-Befehl ist die sicherste Methode, um Miami von einem ARexx-Skript aus zu beenden, da Miami damit nur dann versucht, offline zu gehen, und das Programm zu beenden, wenn keine anderen ARexx-Skripte mehr laufen, um Deadlocks zu vermeiden. Der Nachteil davon ist, daß es Timing-Probleme geben kann, wenn Ihre ARexx-Kontrolle sehr komplex ist, und mehrere ARexx-Skripte umfaßt (insbesondere Skripte, die von anderen Ereignissen ausgelöst wurden), und eins oder mehrere dieser Skripte noch laufen, wenn der "QUIT"-Befehl gegeben wird: Miami würde sich dann weigern, das Programm zu beenden, obwohl es in einigen Situationen sicher sein kann, zu warten, und das Programm später zu beenden. In so einem Fall verwenden Sie den "QUITFORCE"-Befehl: damit wartet Miami, bis alle ARexx-Skripte abgearbeitet worden sind, und beendet das Programm dann. Warnung: dieser Befehl kann zu einem "Aufhängen" führen, wenn eins der aktiven ARexx-Skripts nie beendet wird, z.B. wegen einer Endlosschleife oder einem rekursiven Aufruf, d.h. der Befehl ist potentiell gefährlich, wenn eins der ARexx-Skripte fehlerhaft ist.

1.109 Miami_deutsch.guide/NODE_ENVVARS

Umgebungs-Variablen

Zeitzone-Information

Zeitzone-Information

Anwender brauchen keinerlei Umgebungsvariablen manuell zu setzen, um Miami zu benutzen. Trotzdem gibt die folgende Liste an, welche Umgebungsvariablen von Miami benutzt werden, falls Sie manuelle Änderungen vornehmen möchten:

DOMAIN, DOMAINNAME

Diese Variablen werden automatisch von Miami gesetzt, wenn Miami online geht. Sie werden auf die aktuelle Domain gesetzt (d.h. auf den Teil des Hostnamen, der dem ersten "." folgt).

HOME

Diese Variable wird automatisch von Miami gesetzt, wenn Miami online geht. Sie wird auf das Home-Verzeichnis des Anwenders konfiguriert, der auf der TCP/IP-Seite angegeben wurde. Home-Verzeichnisse können unter Datenbank/Anwender konfiguriert werden.

HOST, HOSTNAME

Diese Variablen werden automatisch von Miami gesetzt, wenn Miami online geht. Sie werden auf den konfigurierten Hostnamen gesetzt (bei statischen Hostnamen), oder auf den Hostnamen, der der aktuellen IP-Adresse entspricht, wie mittels Reverse-DNS-Lookup ermittelt. Wenn kein Hostname gefunden wurde, werden die Variablen auf die aktuelle IP-Adresse gesetzt.

MagicWB

Wenn kein GUI-Modul angegeben wird (vom Benutzer, in einer Einstellungen-Datei, oder in "ENV:MIAMI/GUI"), verwendet Miami entweder "MUI" oder "MUIMWB" als Standard-GUI. "MUIMWB" wird verwendet, wenn die "MagicWB"-Variable existiert, was darauf hinweist, daß MagicWB installiert ist.

REALNAME

Diese Variable wird automatisch von Miami gesetzt, wenn Miami online geht. Sie wird auf den Realnamen gesetzt, wie auf der TCP/IP-Seite konfiguriert.

SOCKETCONFIG

Diese Variable wird automatisch von Miami gesetzt, wenn Miami online geht. Sie wird von der Freeware-"socket.library"-Emulationslibrary benötigt (für I-Net-225-kompatible Software), und wird in einer Art und Weise gesetzt, die dazu führt, daß die Library korrekt funktioniert.

TZ

Diese Variable wird von Miami gelesen, um die aktuelle Zeitzone zu ermitteln. Sie sollte vor der Installation von Miami korrekt gesetzt werden. Für mehr Informationen sehen Sie bitte unter

Zeitzone-Information
nach.

USERNAME

Diese Variable wird automatisch von Miami gesetzt, wenn Miami online geht. Sie wird auf den Usernamen gesetzt, wie auf der TCP/IP-Seite konfiguriert.

MIAMI/GUI

Diese Variable sollte den Namen des Standard-GUI-Moduls enthalten (z.B. 'MUI', 'MUIMWB' oder 'GTLayout'). Sie wird automatisch bei der Installation gesetzt.

MIAMI/SSLIB

Diese Variable wird nur benötigt, wenn MiamiSSL installiert ist, und wird automatisch bei der Installation von MiamiSSL gesetzt. Sie sollte den Namen der SSL-Verschlüsselungsbibliothek enthalten, d.h. entweder 'Miami:Libs/miamisslintl.library' oder 'Miami:Libs/miamisslusa.library'.

1.110 Miami_deutsch.guide/NODE_ENVVARS_TZ

Zeitzone-Information

Die Umgebungs-Variable TZ muß folgendermaßen gesetzt werden:

Ausserhalb der Sommerzeit:

EST5

wobei EST der Name der Zeitzone ist, und 5 die *negative* Zeitdifferenz zu UTC ist (d.h. wenn Sie 2 Stunden östlich von UTC sind, muß der Wert -2 sein, nicht 2). In Europa, Asien und Australien ist dieser Wert normalerweise 0 oder negativ, in Amerika dagegen positiv. Siehe unten für Beispiele.

Während der Sommerzeit:

EST4EDT

EST, 4: gleiche Bedeutung wie oben. EDT ist der Name der Zeitzone während der Sommerzeit. Achten Sie bitte auch darauf, die Zahl (4 im Beispiel) um eine Stunde anzupassen, so wie es die Sommerzeit-Regelung erfordert.

Der Name der Zeitzone spielt eigentlich keine Rolle. Es ist nur wichtig, daß die Zahl stimmt, und daß die Zahl *exakt* mit dem vierten Zeichen beginnt.

Es ist FALSCH, irgendeine RFC-konforme Zeitangabe in ENV:TZ zu speichern, d.h. etwas in der Form "EST", "EST (-0500)" oder "-0500" funktioniert NICHT.

Einige Beispiele:

	Winter	Sommer
US-Westküste	PST8	PST7PDT
US-Ostküste	EST5	EST4EDT
England	WET0	WET-1WEDT
große Teile Westeuropas	MET-1	MET-2MEDT

1.111 Miami_deutsch.guide/NODE_EXCONFIG

Austauschen von Einstellungen

Die Miami-Einstellungen (Konfigurationsdaten) werden in einer IFF-Datei gespeichert, deren Format derzeit absichtlich nicht dokumentiert ist. Allerdings können Miami's Einstellungen in der folgenden Art und Weise ausgetauscht werden:

Format für den Vertrieb

Importieren/exportieren von ↔
Einstellungen für den Vertrieb

Austausch von Passwörtern

Austausch von Passwort-Dateien.

Client-Einstellungen

Sonder-Einstellungen für Clients

1.112 Miami_deutsch.guide/NODE_EXCONFIG_DIST

Format für den Vertrieb

=====

Miami hat eine Funktion, um Einstellungen in einem ASCII-Format zu exportieren, das für weitere Verbreitung und Vertrieb geeignet ist, z.B. zum Uploaden auf Aminet, oder zur Weitergabe an andere Anwender, die den gleichen Internet-Provider verwenden. Das Format kann auch von Internet-Providern dazu benutzt werden, vorkonfigurierte Miami-Einstellungen für neue Kunden zu erstellen.

Das ASCII-Dateiformat besteht aus einem Header, gefolgt von einer variablen Anzahl Parameter.

Beim Exportieren von Dateien nimmt Miami nur solche Parameter mit in die exportierte Datei auf, die sich auf den Provider beziehen, nicht solche, die sich auf die Systemkonfiguration des Anwenders beziehen, oder die in irgendeiner Form sicherheitsrelevant sind. Das heisst, man kann die Einstellungen gefahrlos exportieren und weitergeben, ohne daß das Risiko besteht, daß dabei geheime Informationen wie Passwörter mit heraus gegeben werden.

Beim Importieren von Dateien unterstützt Miami allerdings auch anwenderbezogene Informationen wie Passwörter, damit Provider Installationsskripte schreiben können, die den Anwender nach Login-ID, Passwort usw. fragen, und dann eine ASCII-Einstellungen-Datei schreiben, die alle von Miami benötigten Informationen enthält.

Um eine Vorstellung davon zu bekommen, wie eine derartige ASCII-Datei

aussieht, exportieren Sie einfach Ihre aktuellen Einstellungen. Das allgemeine Format ist:

- * ein 2-zeiliger Header. Beide Zeilen beginnen mit einem "\$"-Zeichen. Verändern Sie diesen Header nicht.
- * Eine variable Anzahl von Zeilen, die mit einem ";" beginnen. Diese Zeilen sind Kommentare und können frei verändert werden.
- * eine variable Anzahl von Zeilen mit Parametern.

Die meisten Parameter werden jeweils auf einer einzelnen Zeile angegeben. Diese Zeilen sehen folgendermaßen aus:

```
PARAMETER=Wert
```

Einige Parameter (z.B. das Wählskript) benötigen mehrere Zeilen. In diesem Fall wird folgendes Format verwendet:

```
PARAMETER=%
erster Wert
zweiter Wert
dritter Wert
%
```

Das heisst, daß ein einzelnes "%" einen mehrzeiligen Parameter anzeigt, und ein weiteres "%" als einziges Zeichen in einer Zeile das Ende der Liste anzeigt.

Die Parameter können innerhalb der Datei in beliebiger Reihenfolge auftreten. Sie sollten keine Annahmen machen, daß Miami die Parameter immer in einer bestimmten Reihenfolge speichert.

Liste der gültigen Parameter: Ein (m) zeigt einen mehrzeiligen Parameter an. Ein (i) bedeutet, daß der Parameter nur importiert, aber nicht exportiert wird. Ein (r) bedeutet, daß der Parameter nur in der registrierten Version unterstützt wird, und von der unregistrierten Version ignoriert wird. "(m)" und "(i)" sind nicht Teil der ASCII-Datei.

Wenn als Format "A / B" angegeben ist, bedeutet daß, das gültige Werte einzelne Zeichen sind: entweder "A" oder "B".

```
DEVNAME= (i)
Gerätename
```

```
UNIT= (i)
Gerätenummer
```

```
BAUD= (i)
Geschwindigkeit des seriellen Ports
```

```
PROTOCOL=
P / S (PPP oder SLIP)
```

```
FLOWCONTROL= (i)
H / S (Hardware- (RTS/CTS) oder Software- (Xon/Xoff))
```

Handshaking)

EOFMODE= (i)

Y / N / A (Ja / Nein / Auto)

SERMODE=

8N1 / 7E1 / 7O1

MTU=

Zahl

IPTYPE=

D / S (dynamisch oder statisch)

IP=

1.2.3.4

CD= (i)

Y / N (Verwende CD)

BOOTP=

Y / N (Verwende BootP)

INACTIVITY=

N / I / P (Inaktivitäts-Typ: nichts, ICMP, PPP)

INACTIVITYDELAY=

Minuten

PAPNAME= (i)

Username

PAPPWD= (i)

Passwort

PAPSAME=

Y / N

CALLBACKTYPE= (r)

NONE / CBCPFIXED / CBCPVARIABLE

CALLBACKPHONE= (i) (r)

phone_number

CALLBACKMINDELAY= (r)

integer

CALLBACKMAXDELAY= (r)

integer

ACCM=

000a0000

VJC=

Y / N

QUICKRECONNECT=

Y / N

DIALNAME= (i)
Login-ID

DIALPWD= (i)
Passwort

INITSTRING= (i)
Modem_Init_String

DIALPREFIX= (i)
Wähl_Präfix

DIALSUFFIX= (i)
Wähl_Suffix

DIALSCRIPT= (m)
Wähl_Skript

DIALNUMBERS= (i) (m)
Telefon_Nummern

DIALMAXREPEAT=
Max_Wiederholung

DIALREPEATDELAY=
Wiederholpause

DIALREDIALDELAY=
Wahlpause

HOSTDYNAMIC=
Y / N (Hostname dynamisch: ja / nein)

HOSTNAME= (i)
Hostname

REALNAME= (i)
Realname

USERNAME= (i)
Username

DOICMP=
Y / N

FAKEIP=
Y / N

TTCP= (r)
Y / N

DBHOSTS= (m)
Host_Datenbank

DBNETWORKS= (m)

Netzwerk_Datenbank

DBDOMAINS= (m)
Domain_Datenbank

DBDNSSERVERS= (m)
DNS_Server_Datenbank

1.113 Miami_deutsch.guide/NODE_EXCONFIG_PASSWORDS

Austausch von Passwörtern
=====

Miami erlaubt den freien Austausch von Dateien im db-Verzeichnis von Unix/AmiTCP, mit einer Ausnahme: Die passwd-Datei kann zwar importiert werden, aber dabei werden die Passwörter nicht übernommen, sondern gelöscht, d.h. die Passwörter müssen nach dem Importieren neu manuell eingegeben werden.

Der Grund dafür ist: AmiTCP (zumindest bis zur Version 4.3) verwendet den DES-Algorithmus zur Passwort-Verschlüsselung. DES ist ein kryptographisch starker Verschlüsselungs-Algorithmus, der unter die US-Export-Beschränkungen fällt. Ein Programm, das DES implementiert, darf nicht ohne individuelle Genehmigung aus den USA exportiert werden, und die US-Regierung erteilt derzeit keine derartigen Genehmigungen.

Das führt dazu, daß jeglicher Export von AmiTCP aus den USA illegal ist, einschließlich des Downloadens des AmiTCP-Archivs von einem FTP-Server in den USA zu einem Computer ausserhalb der USA. Aus diesem Grund darf AmiTCP nicht auf alle Aminet-Server upgeloadet werden, was die Verfügbarkeit von AmiTCP stark einschränkt.

Für Miami wäre die Situation noch schlimmer gewesen: da ich Miami in den USA entwickle (nicht in Finnland wie NSDi), hätte ich Miami niemandem ausserhalb der USA weitergeben dürfen, egal auf welche Art es vertrieben würde. Ich habe mich daher entschlossen, DES in Miami nicht zu verwenden, sondern einen anderen Verschlüsselungsalgorithmus, der nicht unter die US-Export-Beschränkungen fällt.

Miami verwendet eine iterierte Version von MD5 zur Passwort-Verschlüsselung. Dieser Algorithmus ist kryptographisch stark, d.h. es ist ausser der vollständigen Suche kein Algorithmus bekannt, um MD5 zu brechen, genau wie bei DES. Allerdings ist MD5, im Gegensatz zu DES, ein Einweg-Algorithmus, d.h. verschlüsselte Texte können nicht entschlüsselt werden, und deshalb fällt MD5 nicht unter die US-Export-Beschränkungen.

Das bedeutet, daß Miami frei in die USA importiert und aus den USA exportiert werden darf, auf Aminet und andere Server upgeloadet werden darf, und sowohl in den USA, als auch außerhalb der USA verwendet werden darf (es sei denn ein Land verbietet die Benutzung von MD5).

Es tut mir leid, daß Anwender deshalb möglicherweise mehrere

getrennte Passwort-Dateien verwalten müssen, aber ich sehe keinen anderen Weg, dieses Problem zu umgehen.

1.114 Miami_deutsch.guide/NODE_EXCONFIG_CLIENTS

Client-Einstellungen.

=====

Einige TCP/IP-Clients (also Programme, die unter Miami laufen sollen), z.B. `Amitalk`, erfordern Änderungen an den Dateien, die die meisten Protokollstacks im "db"-Verzeichnis speichern. In den meisten Fällen müssen Einträge zu den "services" oder "inetd.conf"-Dateien hinzugefügt werden.

Mit Miami können Sie die entsprechenden Änderungen direkt durch die graphische Benutzeroberfläche vornehmen, d.h. sie wählen die "Datenbank"-Seite an, wählen den korrekten Teil der Datenbank an (z.B. "Dienste"), und fügen die benötigten Einträge hinzu.

In manchen Situationen kann es bequemer sein, diesen Vorgang zu automatisieren, z.B. damit ein Installer-Skript eines TCP/IP-Clients alle benötigten Änderungen automatisch erledigen kann, ohne den Anwender damit zu belästigen. Mit Miami geht das folgendermaßen:

- * Zuerst fügen Sie eine Zeile am Ende der Datei "ENVARC:MiamiChangeDB" an, die folgendermaßen aussieht:

```
ADD services ntalk 518/udp
```

 oder

```
ADD inetd ntalk dgram udp wait root Servers:talkd (talkd)
```

 Wenn Miami gestartet wird, lädt es automatisch den Inhalt dieser Datei (wenn sie existiert), paßt die Einstellungen entsprechend an, und speichert die Einstellungen wieder ab.
- * Wenn Miami bereits läuft, wenn der Client installiert wird, und Miami die Einstellungen sofort ändern soll, dann sollten Sie Miami den "CHANGEDB" ARexx-Befehl schicken, nachdem Sie die o.a. Datei geändert haben.

Sie können auf diese Weise Einträge zu allen Datenbank-Tabellen von Miami hinzufügen. Allerdings werden aus Sicherheitsgründen nur die Tabellen direkt von Miami verändert, die von Clients häufig verändert werden müssen: 'inetd' und 'services'. Wenn eine Anwendung versucht, irgendeine andere Tabelle zu verändern (z.B. die 'users'-Tabelle), zeigt Miami zunächst einen Requester, und fragt den Anwender um Erlaubnis, die Änderung vorzunehmen.

Zusammenfassend: Um von einem Installer-Skript Miami automatisch für einen Client zu konfigurieren, verwenden Sie die folgenden Befehle:

```
echo >>ENVARC:MiamiChangeDB "ADD services ntalk 518/udp"
rx "address MIAMI.1;CHANGEDB"
```

Wenn Miami bereits läuft, werden die Änderungen sofort übernommen. Ansonsten übernimmt Miami die Änderungen beim nächsten Programmstart.

1.115 Miami_deutsch.guide/NODE_UTILITY

Hilfsprogramme

MiamiArp	MiamiArp
MiamiFinger	MiamiFinger
MiamiIfConfig	MiamiIfConfig
MiamiMapMBone	MiamiMapMBone
MiamiMRInfo	MiamiMRInfo
MiamiMRouted	MiamiMRouted
MiamiMTrace	MiamiMTrace
MiamiNetStat	MiamiNetStat
MiamiPing	MiamiPing
MiamiRemind	MiamiRemind
MiamiResolve	MiamiResolve
MiamiRoute	MiamiRoute
MiamiSysCtl	MiamiSysCtl
MiamiTCPDump	MiamiTCPDump
MiamiTraceRoute	MiamiTraceRoute

1.116 Miami_deutsch.guide/NODE_UTILITY_ARP

MiamiArp

=====

Anzeige und Kontrolle zur Adressumwandlung.

Gebrauch:

arp hostname

Zeigt den aktuellen Arp-Eintrag für <hostname> an.

arp [-n] -a

Zeigt alle aktuellen Arp-Einträge an. Wenn "-n" angegeben wird, werden alle Einträge nicht symbolisch, sondern numerisch angezeigt.

arp -d hostname

Löscht den Arp-Eintrag für <hostname>.

arp -s hostname hw_addr [temp] [pub]

Erzeugt einen Arp-Eintrag für <hostname> mit der Hardware-Adresse <hw_addr>. Der Eintrag ist permanent, es sei denn das Wort "temp" wird angegeben. Wenn das Wort "pub" angegeben wird, agiert dieser Rechner als Arp-Server für den angegebenen Host.

arp -f dateiname

Liest Befehle aus der Datei <dateiname> und führt sie aus.

1.117 Miami_deutsch.guide/NODE_UTILITY_FINGER

MiamiFinger

=====

MiamiFinger zeigt Informationen zu System-Anwendern an.

Gebrauch: MiamiFinger [-l] [anwender][@rechnername]

Optionen sind:

-l

Zeigt die Ausgabe im langen Format an (für nicht-lokale Rechner: schickt die "/W"-Option zum finger-Daemon).

Wenn kein Rechnername angegeben wird, wird "localhost" angenommen.

Wenn ein Anwender angegeben wird, werden Informationen zu diesem Anwender angezeigt. Ansonsten werden die Standardinformationen für den fingerd angezeigt, zu dem die Verbindung aufgebaut wird. In vielen

Fällen bedeutet das, daß allgemeine Systeminformationen und/oder eine Liste der eingeloggten Anwender angezeigt werden.

Diese Implementierung von MiamiFinger unterstützt T/TCP für schnellere finger-Antworten.

1.118 Miami_deutsch.guide/NODE_UTILITY_IFCONFIG

MiamiIfConfig
=====

Konfigurieren von Netz-Schnittstellen-Parametern

Hinweis: die meisten Optionen von MiamiIfConfig sollten derzeit nicht mit Miami benutzt werden, weil Miami die Werte bereits korrekt einstellt. Spielen Sie bitte mit diesem Hilfsprogramm nicht herum ! Sie sollten genau die Bedeutung der einzelnen Optionen kennen, bevor Sie versuchen, Schnittstellenoptionen zu verändern.

Die wahrscheinlich einzigen nützlichen Optionen derzeit sind "up" und "down", um Zugriff auf die Schnittstelle vorübergehend zu verhindern. Beachten Sie, daß das nicht dazu führt, daß das Modem auflegt. Abgesehen von diesen Optionen sollten Sie MiamiIfConfig nur benutzen, um sich Optionen anzusehen, nicht um sie zu ändern.

Gebrauch: MiamiIfConfig interface [alias | -alias] [af [address [dest_addr]] [up] [down] [netmask mask]] [metric n] [arp | -arp] [broadcast address] [link0 | -link0] [link1 | -link1] [link2 | -link2]

interface

Derzeit entweder "lo0" oder "mi0"

alias/-alias

Die angegebene Adresse als Alias für eine existierende Adresse ansehen, d.h. die aktuelle Adresse nicht überschreiben.

af

Address-Familie: derzeit wird nur "inet" unterstützt.

address

Eine Protokoll-Level-Adresse. Für die Adress-Familie "inet" ist das eine IP-Adresse in Punkt-Notation (z.B. 123.45.67.89).

dest_addr

Die Protokoll-Level-Zieladresse. Wird nur für Punkt-zu-Punkt-Schnittstellen genutzt.

up/down

Markiert die Schnittstelle als "up" oder "down" (aktiv oder inaktiv).

netmask

Ändert die Netzmaske für die Schnittstelle.

metric

Ändert die Metrik (Priorität) für die Schnittstelle. Für einen Einzel-Schnittstellen-Stack wie Miami hat das keine Auswirkungen.

arp/-arp

Schaltet Arp für diese Schnittstelle ein oder aus. Diese Option sollte nicht mit Miami genutzt werden. Verwenden Sie stattdessen die Miami-GUI, um die Art der Adress-Umwandlung anzugeben.

broadcast

Setzt die Broadcast-Adresse für diese Schnittstelle.

linkx/-linkx

Setzt oder löscht die Link-Level-Flags 0, 1 oder 2. Diese Flags werden derzeit von Miami nicht genutzt.

1.119 Miami_deutsch.guide/NODE_UTILITY_MAPMBONE

MiamiMapMBone

=====

Multicast-Verbindungs-Anzeiger

Gebrauch: MiamiMapMBone [-r retry_cnt] [-t timeout_cnt] [-d debug_lvl] [-f] [-g] [start_router]

MiamiMapMbone versucht, alle Multicast-Router anzuzeigen, die vom angegebenen Multicast-Start-Router aus erreichbar sind. Wenn nicht anders angegeben, wird localhost als Start-Router verwendet.

Die Optionen haben die folgende Bedeutung:

-d debug_level

Gibt die Debug-Stufe an. Wenn die Debug-Stufe größer als der Standardwert 0 ist, werden zusätzliche Debugging-Meldungen erzeugt.

-f

Schaltet Flooding ein. Flooding erlaubt die rekursive Suche nach Nachbar-Multicast-Routern, und ist standardmäßig eingeschaltet, wenn der Start-router nicht angegeben wird.

-g

Erstellt einen Graph im GraphEd-Format.

-n

Schaltet die DNS-Suche für Multicast-Router-Namen aus.

-r retry_count

Gibt die Anzahl Wiederholversuche an. Der Standard ist 1.

-t timeout_count

Gibt die Anzahl Sekunden an, die auf eine Antwort auf eine Nachbar-Anfrage gewartet wird vor dem nächsten Versuch. Der

Standard ist 2 Sekunden.

1.120 Miami_deutsch.guide/NODE_UTILITY_MRINFO

MiamiMRInfo
=====

Zeigt Konfigurationsinformationen für einen Multicast-Router an.

Gebrauch: MiamiMRInfo [-d debug_level] [-r retry_count] [-t timeout_count] [multicast_router]

MiamiMRInfo versucht, Informationen zur Konfiguration des angegebenen Multicast-Router anzuzeigen. Wenn kein Router angegeben wird, wird localhost verwendet.

Die Optionen haben die folgende Bedeutung:

- d debug_level
Gibt die Debug-Stufe an. Wenn die Debug-Stufe größer als der Standardwert 0 ist, werden zusätzliche Debugging-Meldungen erzeugt.
- r retry_count
Gibt die Anzahl Wiederholversuche an. Der Standard ist 3.
- t timeout_count
Gibt die Anzahl Sekunden an, die auf eine Antwort auf eine Nachbar-Anfrage gewartet wird vor dem nächsten Versuch. Der Standard ist 4 Sekunden.

1.121 Miami_deutsch.guide/NODE_UTILITY_MROUTED

MiamiMRouteD
=====

IP-Multicast-Routing-Daemon

Gebrauch: MiamiMRouteD [-p] [-c config_file] [-d debug_level]

MiamiMRouteD ist ein Programm, daß Sie evtl. im Hintergrund starten müssen ("run MiamiMRouteD"), um Multicast-Daten zu empfangen, oder weiter zu leiten. Siehe unten für nähere Informationen.

Die Optionen haben die folgende Bedeutung:

- p
Startet MiamiMRouteD im 'non-pruning mode'. Sollte nur für Testzwecke verwendet werden.


```
-c config_file
    Gibt an, welche Konfigurationsdatei verwendet werden soll. Der
    Standard ist "Miami:MiMiMRouted.config".

-d debug_level
    Gibt die Debug-Stufe an. Der Standard ist 0 (keine
    Debug-Informationen).
```

MiMiMRouted ist ein sehr komplexes und leistungsfähiges Programm, mit dem Multicast-Daten empfangen und weitergeleitet werden können. Es wird mit einer getrennten Konfigurationsdatei konfiguriert. Das Format dieser Datei wird unten teilweise beschrieben. Da Miami nur eine einzige Schnittstelle unterstützt, werden nur die Funktionen beschrieben, die mit einer Schnittstelle Sinn machen. Die meisten Anwender brauchen MiMiMRouted vermutlich gar nicht.

Die beiden gebräuchlichsten Konfigurationen sind:

- * Sie empfangen Ihre Multicast-Daten direkt von einer Broadcast- oder Multicast-tauglichen Schnittstelle, z.B. Ethernet oder Arcnet. In diesem Fall verwenden Sie MiMiMRouted nicht. Schalten Sie stattdessen Multicasting in Miami ein, auf der "Schnittstelle"-Seite.
- * Sie empfangen Ihre Multicast-Daten durch einen IP-Tunnel, z.B. per PPP von Ihrem Netzanbieter. In diesem Fall schalten Sie Multicasting in Miami für Ihre PPP/SLIP-Schnittstelle aus, konfigurieren Sie MiMiMRouted für einen Tunnel zu Ihrem Netzanbieter (siehe unten), und starten Sie MiMiMRouted, nachdem Sie Miami gestartet haben.

Die Konfigurationsdatei für MiMiMRouted ist eine Standard-ASCII-Datei. Jede Zeile kann einen Befehl enthalten. Der einzige interessante Befehl derzeit ist der "tunnel"-Befehl, mit dem Sie einen IP-Tunnel zum Senden und Empfangen von Multicast-Datei konfigurieren können. Der wichtige Teil der Syntax ist:

```
tunnel <local-addr> <remote-addr>
```

Für <local-addr> können Sie eine IP-Adresse oder einen Schnittstellen-Namen (für Miami immer "mi0") angeben. <remote-addr> ist die IP-Adresse des Rechners auf der anderen Seite des Multicast-Tunnels, z.B. erstellt

```
tunnel mi0 1.2.3.4
```

einen Multicast-Tunnel zum Rechner 1.2.3.4.

1.122 Miami_deutsch.guide/NODE_UTILITY_MTRACE

MiMiMTrace

=====

Gibt den Multicast-Pfad vom Sender zum Empfänger aus

Gebrauch: MiamiMTrace [-g gateway] [-i if_addr] [-l] [-M] [-m max_hops] [-n] [-p] [-q nqueries] [-r resp_dest] [-s] [-S stat_int] [-t ttl] [-v] [-w waittime] source [receiver] [group]

MiamiMTrace ist ein Hilfsprogramm sehr ähnlich zu MiamiTraceRoute, aber für Multicast-Adressen, nicht Unicast-Adressen. Sehen Sie bitte unter

MiamiTraceRoute

für mehr Informationen zu TraceRoute nach.

"group" gibt die Multicast-Adresse an. "source" und "receiver" sind die Unicast-Adressen des Multicast-Pfads, der verfolgt werden soll. Wenn "goup" nicht angegeben wird, wird 224.2.0.1 benutzt. Wenn "receiver" nicht angegeben wird, wird localhost angenommen.

Die Optionen haben die folgende Bedeutung:

- g gateway
Sendet die Anfrage direkt per Unicast zum angegebenen Multicast-Router, statt Multicasting zu verwenden. Der Router muß der letzte Router auf dem Weg vom Sender zum Empfänger sein.
 - i if_addr
Verwendet die angegebene Adresse als lokale Schnittstellen-Adressen (bei einem Rechner mit mehreren Schnittstellen) beim Senden der Anfrage, und als Standard für den Empfänger und die Zieladresse für die Antwort.
 - l
Endlosschleife, und Ausgabe der Paketrage und der Verlust-Statistik für den Multicast-Pfad, nach jeweils 10 Sekunden (siehe auch '-S stat_int').
 - M
Antwort immer per Multicast senden, statt zuerst Unicast zu probieren.
 - m max_hops
Gibt die maximale Anzahl Hops an, die auf dem Weg zum Sender zum Empfänger verfolgt werden. Der Standard ist 32 Hops (unendlich für das DVMRP-Routing-Protokoll).
 - n
Gibt Hop-Adressen numerisch statt symbolisch und numerisch aus (spart einen DNS-Lookup pro gefundenem Router).
 - q nqueries
Gibt die maximale Anzahl Versuche pro Hop an. Der Standard ist 3.
 - p
Wartet passiv auf Multicast-Antworten von Traces, die von anderen initiiert werden. Funktioniert am besten mit einem Multicast-Router.
 - r resp_dest
Sendet die Antwort an den angegebenen Host, statt an den Host, auf dem MiamiMTrace läuft, oder an eine andere Multicast-Adresse, als
-

die, die für diesen Zweck reserviert ist (224.0.1.32).

-s

Erstellt die Ausgabe in kurzer Form: nur der Multicast-Pfad, nicht Paketrate und Verlust-Statistik.

-S stat_int

Gibt das Intervall zwischen aufeinanderfolgenden Statistiken in Sekunden an. Standard ist 10 Sekunden.

-t ttl

Gibt die 'time-to-live' (Anzahl Hops) für Multicast-Trace-Anfragen und Antworten an. Der Standard ist 64, außer für lokale Anfragen an die "all routers"-Multicast-Gruppe, die ttl=1 verwendet.

-v

Ausführliche Ausgabe. Gibt Hop-Zeiten und Statistik mit aus.

-w waittime

Setzt die Wartezeit für Antworten auf die angegebene Anzahl Sekunden.

1.123 Miami_deutsch.guide/NODE_UTILITY_NETSTAT

MiamiNetStat

=====

MiamiNetStat ist ein Programm zum Anzeigen von Konfigurationsparametern und Statistiken. Es ist fast identisch in Funktionalität zu der "netstat"-Version, die mit 4.4BSD ausgeliefert wird, hat aber einige zusätzliche Funktionen zur Anzeige von Link-Level-Statistiken.

Gebrauch:

- * MiamiNetStat [-AaDnN] [-f Adressfamilie]
- * MiamiNetStat [-dimnNrs] [-f Adressfamilie]
- * MiamiNetStat [-dnN] [-] [-I Schnittstelle]
- * MiamiNetStat [-s] [-] [-L Schnittstelle]
- * MiamiNetStat [-s] [-g]
- * MiamiNetStat [-p Protokoll]

Der MiamiNetStat-Befehl zeigt symbolisch die Inhalte verschiedener netzwerkbezogener Datenstrukturen an. Es gibt verschiedene Ausgabeformate für die auszugebenden Informationen, abhängig davon, welche Optionen man angegeben hat.

Mit der ersten Befehlsform erhält man eine Liste aller aktiven

Sockets für jedes Protokoll.

In der zweiten Form wird der Inhalt einer der Datenstrukturen angezeigt, entsprechend der angegebenen Option.

Mit der dritten Form zeigt MiamiNetStat Informationen zum Paketfluß auf der angegebenen Schnittstelle an.

Die vierte Form zeigt die Link-Level-Konfiguration oder (bei Angabe des "-s"-Flags) Link-Level-Statistiken für die angegebene Netzwerk-Schnittstelle an.

Die fünfte Form zeigt Informationen zu virtuellen Schnittstellen (für Multicasting) und Multicast-Routing-Statistiken an.

Die sechste Form zeigt Statistiken zum angegebenen Protokoll an.

Die Optionen haben die folgende Bedeutung:

- A Wird mit der Standardanzeige verwendet, und zeigt die Adressen aller zu den Sockets gehörenden Protokoll-Kontroll-Blöcke an. Wird zur Fehlersuche verwendet.
 - a Wird mit der Standardanzeige verwendet, und zeigt den Zustand aller Sockets an. Normalerweise werden die Sockets von Serverprozessen nicht angezeigt.
 - d Wird zusammen mit einer Schnittstellenoption (i oder I) verwendet, und zeigt die Anzahl der nicht weitergeleiteten Pakete an.
 - D Wird mit der Standardanzeige verwendet, und zeigt die Anzahl der übertragenen Bytes pro aktiver TCP-Verbindung an.
 - f Adressfamilie
Beschränkt Statistiken oder Adress-Kontroll-Block-Ausgaben auf die angegebene Adressfamilie. Nur die Adressfamilie "inet" wird derzeit erkannt.
 - g Zeigt die Tabelle der virtuellen Schnittstellen und die Multicast-Routing-Tabelle an. Zusammen mit der Option '-s' zeigt diese Option Multicast-Routing-Statistiken an. Beide Optionen sind nur sinnvoll wenn MiamiMRouteD läuft.
 - I Schnittstelle
Zeigt Informationen zur angegebenen Schnittstelle an.
 - i Zeigt Informationen zu allen konfigurierten Schnittstellen an.
 - m Zeigt Statistiken der Speicherverwaltungsroutinen an. (Das Netzwerk-Kernel hat einen eigenen Speicher-Pool.)
-

- n
Zeigt Netzwerk-Adressen als Zahlen an. (Normalerweise interpretiert MiamiNetStat Adressen, und versucht, sie symbolisch anzuzeigen.) Diese Option kann mit allen Ausgabeformaten verwendet werden.
- N
Zeigt Netzwerk-Adressen nur dann symbolisch an, wenn der symbolische Name ohne vorhergehende DNS-Anfrage bereit steht. Ansonsten wird die Netzwerk-Adresse als Zahl angegeben. Diese Option kann mit allen Anzeigeformaten benutzt werden.
- p Protokoll
Zeigt Statistiken zum angegebenen Protokoll an. Als Protokoll kann entweder ein Protokollname oder ein Alias angegeben werden. Eine leere Ausgabe bedeutet normalerweise, daß es keine interessanten Statistiken gibt. Das Programm gibt einen Fehler aus, wenn das Protokoll nicht erkannt wird, oder es für das Protokoll keine Statistikroutine gibt.
- r
Zeigt die Routing-Tabellen an. Wenn "-s" auch angegeben wird, werden stattdessen Routing-Statistiken angezeigt.
- s
Zeigt die Statistiken für alle Protokolle an. Wenn diese Option wiederholt wird, werden Ausgaben mit einem Wert von Null unterdrückt.

In der Standardanzeige werden für aktive Sockets beide Adressen, die Längen der Sende- und Empfangswarteschlangen (in Bytes), das Protokoll und der interne Protokollzustand angezeigt. Die Adressformate haben die Form "Host.Port" oder "Netzwerk.Port", wenn die Adresse eines Sockets ein Netzwerk, aber keine Hostadresse angibt. Die Host- und Netzwerkadressen werden mit Hilfe von Informationen aus den "Hosts"- und "Netzwerke"-Datenbanken symbolisch angezeigt, wenn möglich. Wenn kein symbolischer Name für die Adresse bekannt ist, oder die "-n"-Option angegeben wurde, wird die Adresse numerisch ausgegeben, je nach Adressfamilie.

Die Schnittstellen-Anzeige gibt kumulative Statistiken bzgl. der Anzahl der übertragenen Pakete, der Anzahl Fehler, und der Anzahl Kollisionen aus. Die Netzwerk-Adresse der Schnittstelle und die maximale Paketgröße ("mtu") werden ebenfalls angezeigt.

Die Routing-Tabellen-Anzeige gibt die verfügbaren Routen und deren Status an. Jede Route hat einen Zielhost oder ein Zielnetzwerk, und ein Gateway zum Weiterleiten der Pakete. Das "Flags"-Feld gibt einige binär gespeicherte Informationen zur Route aus.

- 1
RTF_PROTO1 Protokoll-spezifisches Routing-Flag #1 (derzeit unbenutzt)
 - 2
RTF_PROTO2 Protokoll-spezifisches Routing-Flag #2 (derzeit unbenutzt)
-

3	RTF_PROTO3 Protokoll-spezifisches Routing-Flag #3 (Bedeutung für TCP: Timeout für Route läuft).
C	RTF_CLONING Bei Benutzung neue Routen erzeugen
D	RTF_DYNAMIC Dynamisch (per redirect) erzeugt
G	RTF_GATEWAY Ziel nur über Zwischenrechner erreichbar
H	RTF_HOST Host-Eintrag (ansonsten für ein Netzwerk)
L	RTF_LLINFO Gültige Übersetzung von Protokoll- zu Link-Adresse
M	RTF_MODIFIED Dynamisch (per redirect) modifiziert
P	RTF_PRCLONING Route wird von Protokolls gecloned.
R	RTF_REJECT Host oder Netzwerk nicht erreichbar
S	RTF_STATIC Manuell hinzugefügt
U	RTF_UP Route benutzbar
W	RTF_WASCLONED Route wurde durch das Clonen einer anderen Route erzeugt.
X	RTF_XRESOLVE Externer Server übersetzt Protokoll- in Link-Adresse

Direkte Routen werden für jede Schnittstelle am lokalen Host erzeugt: Das Gateway-Feld für derartige Routen enthält die Adresse der Schnittstelle. Das refcnt-Feld gibt die Anzahl der aktiven Nutzer einer Route an. Verbindungsorientierte Protokolle verriegeln normalerweise eine einzelne Route für die Dauer der Verbindung, während nicht-verbindungsorientierte Protokolle die Route jedesmal neu ermitteln. Das Use-Feld gibt die Anzahl Pakete an, die über diese Route verschickt wurden. Der Interface-Eintrag gibt an, welches Interface für diese Route verwendet wird.

Mit der Option "-L" zeigt MiamiNetStat die Link-Level-Konfiguration an, z.B. den aktuellen Status der IPCP- und LCP-Unterprotokolle für PPP, für die angegebene Schnittstelle.

Mit der Optionskombination "-sL" zeigt MiamiNetStat Link-Level-Statistiken an, einschließlich Informationen zu

verschiedenen Pakettypen und Prüfsummenfehlern, für die angegebene Schnittstelle.

Derzeit unterstützt Miami nur zwei Schnittstellen:

lo0

Die "local loopback"-Schnittstelle

mi0

Die PPP/(C)SLIP-Schnittstelle unter Verwendung des in Miami eingebauten Schnittstellentreibers oder der aktuellen SANA-II-Schnittstelle.

1.124 Miami_deutsch.guide/NODE_UTILITY_PING

MiamiPing

=====

Schickt Pakete an Netzwerk-Rechner, und wartet auf die Antwort.

Gebrauch: MiamiPing [-Rdfnqrv] [-c count] [-i wait] [-l preload] [-p pattern] [-s packetsize] hostname

Optionen:

-c count

Nach Senden und Empfang von <count> Paketen abbrechen.

-d

Setzt die SO_DEBUG-Option auf dem benutzten Socket.

-f

Dauer-Ping ('flood ping'). Schickt Pakete so schnell ab, wie sie zurückkommen, aber mindestens hundert mal pro Sekunde. Für jeden gesendeten Ping wird ein "." angezeigt, und für jeden empfangenen Ping ein Backspace. Damit kann man schnell erkennen, wie viele Pakete verloren gegangen sind. Hinweis: Der Mißbrauch dieser Option für "Denial-of-Service-Attacks" ist illegal.

-i wait

Wartet <wait> Sekunden zwischen aufeinanderfolgenden Paketen. Der Standard ist, eine Sekunde zu warten. Diese Option ist inkompatibel zu "-f".

-l preload

Verschickt <preload> Pakete so schnell wie möglich, und geht dann zum normalen Verhalten über.

-n

Nur numerische Ausgabe.

-p pattern

Man kann bis zu 16 Füllbytes angeben, mit denen das zu verschickende Paket aufgefüllt wird. Das ist sinnvoll, um

datenabhängige Netzprobleme zu diagnostizieren. "-p ff" füllt z.B. das Paket vollständig mit binären Einsen auf.

-q

Stille Ausgabe. Nichts wird angezeigt, ausser der Statistik-Zeilen bei Start und Ende.

-R

Route protokollieren. Fügt die RECORD_ROUTE-Option in Ping-Pakete ein, und zeigt den Route-Puffer bei empfangenen Paketen an. Beachten Sie, daß der IP-Header nur Platz für neun derartige Routen hat. Viele Hosts ignorieren oder löschen diese Option.

-r

Normale Routing-Tabellen umgehen, und Daten direkt an einen direkt angeschlossenen Host schicken. Wenn der Host nicht mit einem direkt angeschlossenen Netz verbunden ist, wird ein Fehler zurück gegeben. Diese Option kann benutzt werden, um einen lokalen Rechner durch eine Schnittstelle anzupingen, die keine Route enthält (z.B. nachdem die Schnittstelle von routed abgemeldet worden ist).

-s packetsize

Gibt die Anzahl Datenbytes pro Paket an. Der Standard ist 56, was zu 64 ICMP-Datenbytes führt, wenn man es mit den 8 Bytes für den ICMP-Header kombiniert.

-v

Ausführliche Ausgabe. Empfangene ICMP-Pakete ausser Ping-Antwort-Paketen werden mit ausgegeben.

1.125 Miami_deutsch.guide/NODE_UTILITY_REMIND

MiamiRemind

=====

Einige Anwender sehen die automatische Warnung und das automatische Auflegen nach 30/60 Minuten in der Miami-Demoversion als nützliche Funktion an, um Telefon- oder Netzanbieter-Kosten zu sparen.

MiamiRemind ist ein Hilfsprogramm, mit dem ähnliche Funktionen in der registrierten Version verfügbar sind, aber zusätzlich zum normalen Warnen/Auflegen hat MiamiRemind die folgenden nützlichen Funktionen:

- * Die Anzahl der Warnungen, plus das Intervall zwischen Warnungen können frei konfiguriert werden.
- * Es ist möglich, nach einer bestimmten Anzahl Warnungen aufzulegen, nur eine bestimmte Anzahl Warnungen zu zeigen (ohne aufzulegen), oder immer weiter nach gleichen Zeitabständen Warnungen zu zeigen.
- * Zusätzlich zu festen Zeitintervallen ist es möglich, nach einer

bestimmten Dauer von *Inaktivität* Warnungen zu zeigen. Beide Arten von Warnungen (nach fester Zeit und nach Inaktivität) können gleichzeitig eingeschaltet sein.

- * Wenn man den Inaktivitäts-Timer direkt mit der "Auflegen"-Funktion benutzt, bekommt die Funktionalität "Auflegen nach Inaktivität", etwas das von Anwender wiederholt gewünscht worden ist.

Der Begriff "Inaktivität" ist für TCP/IP-Verbindungen schwer zu definieren. Die Standarddefinition, die MiamiRemind verwendet, ist "keine TCP-Daten". Mit dieser Definition benötigt MiamiRemind sehr wenig CPU-Zeit und Speicher.

Für Anwender, die detaillierte Definitionen von "Inaktivität" benötigen, hat MiamiRemind einen Parser und Compiler, der Ausdrücke verarbeitet, die im Format identisch sind zu denen, die MiamiTCPDump benutzt, d.h. Sie können z.B. Ausdrücke wie

```
"(tcp[13] & 3 != 0) or udp"
```

verwenden. Dieser Ausdruck würde alle TCP-SYN-Pakete, alle TCP-FIN-Pakete, und alle UDP-Pakete als "Aktivität" ansehen, aber keine anderen Pakete.

Dieser Parser/Compiler benötigt miamibpf.library und miamipcap.library, und benötigt daher etwas mehr Speicher und CPU-Zeit als die Standard-Definition "keine TCP-Daten".

```
Gebrauch: MiamiRemind [-f fixed_timer_spec]
                [-i inactivity_timer_spec]
                [-p pcap_spec]
```

Option "-f" gibt die Parameter für den festen Timer an, d.h. für den Timer, der startet, wenn MiamiRemind gestartet wird, unabhängig von Aktivität auf der Verbindung. Der Standard ist, diesen Timer auszuschalten.

Option "-i" gibt die Parameter für den Inaktivitäts-Timer an. Dieser Timer wird auf Null zurückgesetzt, wenn ein Paket übertragen wird, und dieses Paket als "Aktivität" angesehen wird. Der Standard ist, diesen Timer auszuschalten.

Option "-p" definiert den Ausdruck für Inaktivität, im MiamiPCap-Format (siehe Beispiele oben). Der Ausdruck sollte in doppelte Anführungszeichen (") eingeschlossen sein. Wenn dieser Parameter angegeben wird, verwendet MiamiRemind miamipcap.library und miamibpf.library zum parsen, compilieren und auswerten des Ausdrucks. Ansonsten wird die Standarddefinition "TCP-Daten" verwendet, und beide Libraries werden nicht benötigt.

"timer_spec" (für Optionen "-f" und "-i") ist ein Textausdruck, der aus Zahlen besteht, die Zeitintervalle beschreiben (angegeben in Minuten), getrennt von Kommas (",").

Jedes Zeitintervall im Textausdruck repräsentiert eine Pause zwischen aufeinanderfolgenden Ereignissen.

Ein "Ereignis" ist normalerweise ein Warn-Requester. Es ist aber auch möglich, vor einer Zahl den Buchstaben "D" anzugeben. Dann legt MiamiRemind die Leitung beim nächsten Ereignis auf. Wenn man den Buchstaben "L" angibt, geht MiamiRemind mit dem nächsten Ereignis in eine Schleife, d.h. verwendet das Zeitintervall immer wieder, für eine Folge von Ereignissen.

Beispiele:

MiamiRemind -f 30,D30

Dieser Aufruf ist identisch zum Verhalten der Demoversion von Miami, i.e. nach 30 Minuten wird eine Warnung angezeigt, nach weiteren 30 Minuten legt Miami auf.

MiamiRemind -f 30,20,L10

Zeigt eine Warnung nach 30 Minuten an, dann nach weiteren 20 Minuten, und dann wiederholt nach jeweils 10 Minuten. Die Leitung wird nie aufgelegt.

MiamiRemind -f 60,60 -i L10

Zeigt eine Warnung nach 60 Minuten an, und eine weitere Warnung nach weiteren 60 Minuten. Ausserdem wird nach Vielfachen von 10 Minuten Inaktivität (keine TCP-Daten) jeweils eine Warnung angezeigt.

MiamiRemind -i D30

Legt die Leitung nach 30 Minuten Inaktivität (keine TCP-Daten) auf.

MiamiRemind -i D20 -p "tcp or udp"

Legt die Leitung nach 20 Minuten Inaktivität auf. "Inaktivität" bedeutet hier: "keine TCP- oder UDP-Daten".

MiamiRemind wird automatisch beendet, wenn die Schnittstelle offline geht (egal aus welchem Grund), wenn Miami versucht, das Programm zu beenden, wenn das Programm ein Ctrl-C erhält, oder wenn beide Timer ausgeschaltet werden.

Die einfachste Art, MiamiRemind zu verwenden, ist es dirkt von Miami zu starten, wenn Miami online geht, d.h. mittels "run >nil: Miami: MiamiRemind [options]" in einem Shell-Skript, das von Miami gestartet wird (konfiguriert in Ereignisse->Online).

1.126 Miami_deutsch.guide/NODE_UTILITY_RESOLVE

MiamiResolve

=====

Ermittelt zu einem Hostnamen die IP-Adresse oder umgekehrt.

Gebrauch:

MiamiResolve ip_address

Ermittelt zu `ip_address` den Hostnamen und alle IP-Adressen, und zeigt diese an.

`MiamiResolve host_name`

Ermittelt zu `host_name` alle IP-Adressen, und zeigt diese an.

`MiamiResolve -s port_number`

Ermittelt zu einer Postnummer den zugehörigen Servicenamen.

`MiamiResolve -s service_name`

Ermittelt zu einem Servicenamen die zugehörige Portnummer.

1.127 Miami_deutsch.guide/NODE_UTILITY_ROUTE

`MiamiRoute`

=====

Manuelles Manipulieren der Routing-Tabellen.

Gebrauch: `MiamiRoute [-nqv] command modifiers args`

Optionen:

`-n`

Verhindert, daß Host- und Netznamen symbolisch angezeigt werden in der Ausgabe. (Die Umwandlung zwischen symbolischen Namen und den numerischen Äquivalenten kann recht zeitaufwendig sein, und kann das korrekte Funktionieren des Netzes erfordern. Es kann also ratsam sein, diese Umwandlung auszuschalten, insbesondere wenn man sich in der Reparatur eines Netzwerkfehlers befindet).

`-q`

Unterdrückt jegliche Ausgabe.

`-v`

Gibt zusätzliche Details aus.

Befehle:

`add`

Route hinzufügen

`flush`

Alle Routen entfernen. Seien Sie sehr vorsichtig bei der Benutzung dieses Befehls. Er entfernt auch einige von Miami's Standard-Routen. Wenn Sie das dann nicht wieder hinterher manuell reparieren, müssen Sie Miami neu starten, um normal weiter arbeiten zu können.

`delete`

Löscht eine einzelne Route.

`change`

Ändert Charakteristika einer Route (z.B. das Gateway).

get

Zeigt zu einem Ziel die zugehörige Route an.

monitor

Zeigt kontinuierlich die Änderungen an den Routing-Tabellen an, sowie fehlgeschlagene Route-Suchen und vermutete Netsplits. Hinweis: ohne eine Implementierung von "routed" ist dieser Befehl nicht sehr nützlich.

Der MiamiRoute-Befehl wird normalerweise mit einem Einzel-Interface-Stack wie Miami nicht benötigt, und ist sehr komplex und schwierig zu benutzen. Für eine vollständige Beschreibung aller Optionen sehen Sie bitte in der BSD-Dokumentation des "route"-Befehls nach.

Die vermutlich einzige nützliche Anwendung des "MiamiRoute"-Befehls derzeit ist das Ansehen von Routen, z.B. um die Paketlaufzeiten oder MTU-Werte zu erhalten. Um das zu tun, verwenden Sie die Syntax:

```
MiamiRoute get hostname
```

Um sich die vollständige Routing-Tabelle anzusehen, verwenden Sie den Befehl "MiamiNetStat -r", nicht MiamiRoute.

1.128 Miami_deutsch.guide/NODE_UTILITY_SYSCTL

MiamiSysCtl

=====

MiamiSysCtl erlaubt Ihnen das Ansehen und Ändern einiger interner Variablen von Miami.

Gebrauch:

```
MiamiSysCtl [-n] variable
```

Zeigt den Wert von <variable> an.

```
MiamiSysCtl [-n] -w variable=value
```

Ändert den Wert von <variable> auf <value>.

```
MiamiSysCtl [-n] -a
```

Zeigt eine Liste aller Variablen.

```
MiamiSysCtl [-n] -A
```

Zeigt eine Liste aller Variablen, plus zusätzliche Informationen zu Domains, auf die nicht durch MiamiSysCtl zugegriffen werden kann.

Optionen:

```
-n
```

Zeigt nur numerische Ausgaben an.

Erläuterung aller Variablen:

net.inet.ip.forwarding/redirect

Diese Optionen haben keine Bedeutung in einem Einzel-Schnittstellen-Stack wie Miami.

net.inet.ip.ttl

Gibt die Standard-ttl (time-to-live) für von Miami verschickte Pakete an. Sollte auf dem Standardwert 64 stehen.

net.inet.ip.rtxpire/rtminexpire/rtmaxcache

Kontrolliert Timing und Größe für Route-Cloning. Sie sollten diese Werte nicht ändern.

net.inet.ip.sourceroute

Gibt an, was mit Paketen passieren soll, die eine IP-Source-Route-Option enthalten. Diese Option ist nur für Router wichtig.

net.inet.ip.pathmtudisc

Gibt an, ob Path MTU Discovery ein- oder ausgeschaltet sein soll (1/0). Der Standard ist, es einzuschalten, aber wenn Ihre Netzanbindung durch alte, defekte Router läuft, und Sie Probleme mit TCP-Verbindungen haben, dann versuchen Sie, diese Option auszuschalten.

net.inet.icmp.maskrepl

Gibt an, ob Miami eine Netzmaske als Antwort auf ICMP-Masken-Anfragen verschicken soll. Wenn die Netzmaske korrekt konfiguriert ist und diese Option eingeschaltet wird, dann kann ein anderer Rechner auf dem lokalen Netz in MiamiInitsANA2 oder Miami automatisch die korrekte Netzmaske per ICMP erkennen.

net.inet.tcp.rfc1323

Schaltet die RFC1323-TCP-Erweiterungen ein. Diese Erweiterungen kollidieren mit PPP/SLIP-VJC, sollten also nur für Bus-Schnittstellen eingeschaltet werden, nicht für Punkt-zu-Punkt-Schnittstellen.

net.inet.tcp.rfc1644

Schaltet T/TCP ein.

net.inet.tcp.mssdflt

Setzt die normale maximale Segmentgröße für TCP. Wenn Ihre Interface-MTU größer ist als 552, sollten sie diesen Wert erhöhen, um Path MTU Discovery besser zu nutzen.

net.inet.tcp.rttdeflt

Diese Option kontrolliert das Zeitverhalten von TCPs Retransmission und sollte nicht verändert werden.

net.inet.tcp.keepidle/keepintvl

Diese Optionen kontrollieren das Zeitverhalten von TCPs Keep-Alive-Timer und sollten nicht verändert werden.

net.inet.tcp.sendspace/recvspace

Diese Optionen definieren die Standardgröße von TCPs Sende- und

Empfangsfenstern und sollten nicht verändert werden.

net.inet.tcp.bulkftp

Reserviert für zukünftige Verwendung, und derzeit ungenutzt.
Nicht ändern !

net.inet.udp.checksum

Schaltet UDP-Prüfsummen bei allen abgeschickten Paketen ein. Diese Option sollte immer eingeschaltet sein.

net.inet.udp.maxdgram/recvspace

Diese Optionen kontrollieren Grenzwert für UDP-Pakete und sollten nicht verändert werden.

dns.cache.size

Kontrolliert die Größe von Miami's eingebautem DNS-Cache.

dns.cache.flush

Wenn man diesen Wert auf 1 setzt entleert Miami den eingebauten DNS-Cache.

dns.cache.enabled

Schaltet Miami's eingebauten DNS-Cache ein oder aus. Der Standardwert ist 2, d.h. alle Host-Einträge werden gecached. Wenn die Variable auf 1 gestellt wird, werden nur Host-Einträge mit einer einzigen IP-Adresse gecached. Auf diese Weise werden Probleme mit 'round-robin'-IP-Adress-Tausch vermieden. Wenn diese Variable auf 0 gesetzt wird, ist Miami's DNS-Cache komplett ausgeschaltet. Der Cache sollte nur ausgeschaltet werden, wenn Sie eine sehr schnelle Verbindung zu einem lokalen DNS-Server haben.

dns.cache.split

Diese Variable ist normalerweise 0, d.h. Miami verwendet einen einzigen, kombinierten DNS-Cache für vorwärts- und rückwärts-DNS-Suche. Wenn Sie diese Variable auf 1 setzen, verwendet Miami getrennte Caches für vorwärts- und rückwärts-DNS-Suche. Dadurch werden Ausgaben von z.B. MiamiNetStat langsamer, aber es garantiert die 'korrekte' rückwärts-Ermittlung der Hostnamen, die zu einer IP-Adresse gehören (mittels PTR-Lookup).

inetd.retrytime

Gibt die Verzögerung an, nach der InetD versucht, für einen Socket erneut bind() aufzurufen, wenn das beim ersten Mal nicht geklappt hat.

inetd.toomany

Gibt die maximale Anzahl Verbindungen an, die InetD innerhalb des definierten Zeitintervalls akzeptiert.

inetd.cntintvl

Gibt das zu inetd.toomany gehörende Zeitintervall an.

inetd.maxbultin

Gibt die maximale Anzahl der Prozesse an, die für interne Server in InetD erzeugt werden sollen.

inetd.processpri

Gibt die Prozess-Priorität für Server an, die von InetD gestartet werden. Der Standard ist -5. Sie sollten diesen Wert erhöhen, wenn Sie CPU-intensive Hintergrundprozesse verwenden (z.B. den RC5-Challenge-Client). Ansonsten erhalten Ihre Server nie CPU-Zeit.

socket.maxqlen

Diese Option gibt die Länge der Socket-Verbindungs-Warteschlange für einen listen()-Parameter von 5 an. Der Standard ist 7, aber wenn Sie mit einem sehr schnellen Netz verbunden sind, und ausreichend Speicher haben, sollten Sie diesen Wert erhöhen, um die möglichen Effekte von SYN-Flooding zu reduzieren.

1.129 Miami_deutsch.guide/NODE_UTILITY_TCPDUMP

MiamiTCPDump

=====

MiamiTCPDump ermöglicht es, Netzwerk-Pakete zu filtern und dann auszugeben.

Gebrauch: MiamiTCPDump [-adflnNOqStvx] [-c count] [-F file] [-i interface] [-r file] [-s snaplen] [-T type] [-w file] [expression]

Optionen:**-A**

Wird in Verbindung mit '-x' verwendet: Pakete werden in ASCII und als Hexdump angezeigt.

-a

Versuche, Netzwerk-Adressen und Broadcast-Adressen in Namen umzuwandeln.

-c count

Beenden nach <count> Paketen.

-d

Compilierten Packet-Matching-Code in lesbarer Form ausgeben, und dann beenden.

-dd

Compilierten Packet-Matching-Code als Programmfragment ausgeben. dann beenden.

-ddd

Compilierten Packet-Matching-Code als Dezimalzahlen plus Anzahl ausgeben.

-f

Nicht-lokale Internet-Adressen numerisch ausgeben, nicht symbolisch.

- F file
<file> wird als Eingabe für den Filter-Ausdruck verwendet. Ein weiterer Filter-Ausdruck in der Kommandozeile wird ignoriert.
- i interface
Verwende <interface> (derzeit "lo0" oder "mi0"). Wenn nicht angegeben, sucht MiamiTCPDump auf der System-Schnittstellen-Liste nach der konfigurierten Schnittstelle (außer loopback) mit der niedrigsten Nummer. Bei gleichen Nummern wird die erste Schnittstelle verwendet. Derzeit ist das Resultat dieser Suche immer "mi0".
- l
Zeilenpufferung für die Standardausgabe einschalten. Nützlich, wenn man sich die Daten während der Ausgabe ansehen möchte.
- n
Verhindert, daß Adressen (d.h. Hostadressen, Portnummern usw.) in Namen umgewandelt werden.
- N
Verhindert, daß Domainnamen bei Hostnamen mit angegeben werden. Z.B.: wenn man dieses Flag angibt, gibt MiamiTCPDump "nic" statt "nic.ddn.mil" aus.
- O
Packet-Matching-Code-Optimizer nicht starten. Das ist nur sinnvoll, wenn Sie einen Bug im Optimizer vermuten.
- q
Kurze Ausgabe. Gibt weniger Protokollinformationen aus, d.h. die Zeilen sind kürzer.
- s snaplen
Übernimmt <snaplen> Bytes von jedem Paket statt des Standards (68). 68 Bytes sind angemessen für IP, ICMP, TCP und UDP, aber können Protokollinformationen von Nameservern oder NFS-Paketen zu früh abschneiden (siehe unten). Pakete, die wegen einer zu kleinen <snaplen> angeschnitten wurden, werden in der Ausgabe mit "[proto]" angezeigt, wobei <proto> der Name des Protokolls ist, bei dem das Paket abgeschnitten wurde. Beachten Sie, daß eine größere <snaplen> die Verarbeitungszeit pro Paket erhöht, und die effektive Puffergröße verringert. Sie sollten <snaplen> auf den kleinsten Wert stellen, der noch alle Protokollinformationen enthält, an denen Sie interessiert sind.
- S
Gibt absolute anstelle von relativen TCP-Sequenznummern aus.
- T type
Erzwingt, daß selektierte Pakete als Typ <type> interpretiert werden. Derzeit bekannte Typen sind
- * rpc (Remote Procedure Call)
 - * rtp (Real-Time Applications protocol)
-

- * rtcp (Real-Time Applications control protocol),
- * vat (Visual Audio Tool),
- * wb (distributed White Board).

- t
Uhrzeit nicht auf jeder Zeile ausgeben.
- tt
Unformatierte Uhrzeit auf jeder Zeile ausgeben.
- v
Etwas ausführliche Ausgabe. Es werden z.B. die time-to-live und die type-of-service-Informationen in einem IP-Paket mit ausgegeben.
- vv
Noch ausführlichere Ausgabe. Es werden z.B. zusätzliche Felder von NFS-Antwort-Paketen mit ausgegeben.
- w file
Speichert die Pakete in <Datei>, statt sie zu analysieren. Das erlaubt eine spätere Auswertung mit der Option "-r". Die Standardausgabe wird verwendet, wenn <file> als "-" angegeben wird.
- x
Gibt jedes Paket (ausser dem Link-Level-Header) hexadezimal aus, bis zu <snaplen> pro Paket.

<expression> gibt an, welche Pakete ausgegeben werden sollen. Wenn keine <expression> angegeben wurde, werden alle Pakete ausgegeben. Ansonsten werden nur die Pakete ausgegeben, auf die <expression> zutrifft.

Die Syntax von <expression> ist zu umfangreich und komplex, um sie hier vollständig zu dokumentieren. Für eine vollständige Beschreibung der Syntax und aller Details des Ausgabeformats sehen Sie bitte in der Dokumentation der frei erhältlichen BSD-Version von "tcpdump" nach. Hier sind einige Beispiele für gültige Ausdrücke:

- "host sundown"
Gibt alle Pakete aus, die beim host "sundown" ankommen oder von dort abgeschickt werden.
 - "host helios and (hot or ace)"
Gibt alle Pakete aus zwischen "helios" und "hot" oder "ace".
 - "ip host ace and not helios"
Gibt alle IP-Pakete aus zwischen "ace" und jedem Host außer "helios".
 - "tcp[13] & 3 != 0"
Gibt die Start- und End-Pakete (SYN und FIN) für jede TCP-Verbindung aus.
 - "icmp[0] != 8 and icmp[0] != 0"
Gibt alle ICMP-Pakete aus, die nicht Echo-Requests/Replies sind
-

(also keine Ping-Pakete).

1.130 Miami_deutsch.guide/NODE_UTILITY_TRACEROUTE

MiamiTraceRoute
=====

Gibt den Weg aus, den Pakete zu einem Rechner nehmen.

Gebrauch: MiamiTraceRoute [-m max_ttl] [-n] [-p port] [-q nqueries]
[-r] [-s src_addr] [-t tos] [-v] [-w waittime] host [packetsize]

Optionen:

- m max_ttl
Gibt die maximale time-to-live (maximale Anzahl Schritte) aus für abgeschickte Pakete. Der Standard ist 30.
 - n
Gibt Adressen numerisch statt symbolisch und numerisch aus (spart einen DNS-Lookup pro Gateway entlang des Pfads).
 - p port
Gibt den Basis-UDP-Port an, der in Paketen verwendet wird (Standard ist 33434). MiamiTraceRoute hofft, dass niemand auf UDP-Ports <port>+nhops-1 beim Zielhost Verbindungen annimmt (damit eine ICMP PORT_UNREACHABLE-Nachricht zurückgegeben wird, um das Tracing zu beenden). Wenn jemand auf Verbindungen in diesem Portbereich wartet, kann diese Option benutzt werden, um einen ungenutzten Portbereich auszuwählen.
 - q nqueries
Setzt die Anzahl Packets pro "ttl" auf <nqueries>. Standard ist drei.
 - r
Normale Routing-Tabellen umgehen, und Daten direkt an einen direkt angeschlossenen Host schicken. Wenn der Host nicht mit einem direkt angeschlossenen Netz verbunden ist, wird ein Fehler zurück gegeben.
 - s src_addr
Verwendet die angegebene IP-Adresse (angegeben als IP-Adresse, nicht als Hostname) als Quell-IP-Adresse in verschickten Paketen. Auf Rechnern mit mehr als einer IP-Adresse kann man so eine andere IP-Adresse erzwingen, als die von dem Interface, auf dem das Paket verschickt wird. Wenn die angegebene IP-Adresse keine gültige IP-Adresse für diesen Host ist, wird ein Fehler zurückgegeben, und nichts wird verschickt.
 - t tos
Setzt den type-of-service in Paketen auf den angegebenen Wert (Standard ist 0). Der Wert muß im Bereich von 0 bis 255 sein. Diese Option kann verwendet werden um zu erkennen, ob verschiedene
-

types-of-service zu unterschiedlichen Pfaden führen würden.

-v

Ausführliche Ausgabe. Empfangene ICMP-Pakete ausser TIME_EXCEEDED und UNREACHABLE werden ausgegeben.

-w

Gibt die maximale Wartezeit für eine Antwort an. (Standard ist 3 Sekunden).

1.131 Miami_deutsch.guide/NODE_COMPATIBILITY

Kompatibilität

Bisher funktioniert Miami mit allen AmiTCP-Clients und -Servern, mit denen es getestet wurde, mit einer Ausnahme:

Die AmiTCP-4.x-Version von "telnet" funktioniert normalerweise nicht mit Miami. Das liegt daran, daß diese "telnet"-Version einige undokumentierte Eigenschaften von "TCP:" ausnutzt, die nicht von Miami emuliert werden können.

Es gibt dafür zwei Lösungen:

- * Verwenden Sie eine andere telnet-Version, z.B. die Version, die im Aminet in comm/tcp erhältlich ist, ein Terminalprogramm zusammen mit telser.device, oder "napsaterm" im telnet-Modus, oder den neuen graphischen telnet-Client "AmTelnet", erhältlich von www.vapor.com
- * Installieren Sie die Version des "inet-handler", die AmiTCP 4.0demo beiliegt, erzeugen Sie einen entsprechenden Mountlist-Eintrag für "TCP:", und geben Sie "mount TCP:" ein bevor Sie Miami starten (wobei natürlich immer noch auf den Miami-Stack zugegriffen wird), statt die in Miami eingebaute Version von "TCP:" zu verwenden.

1.132 Miami_deutsch.guide/NODE_RESTRICTIONS

Einschränkungen

Die Demoversion hat die folgenden Einschränkungen:

- * Nach 60 Minuten legt Miami die Leitung auf. SANA-II-Verbindungen werden nach 30 Minuten unterbrochen.
 - * Es ist nicht möglich, TCP-Verbindungen aktiv zu halten, wenn das Modem auflegt.
-

- * Die "Ereignisse"-Optionen "Auto-Online nach passivem Offline" und das Starten von ARExx-Skripten sind nicht verfügbar.
- * Die Anzahl der Telefonnummern im Wählmodul ist beschränkt auf drei.
- * Telefon-Logging ist nicht möglich.
- * Die Fenster-Optionen sind nicht anwählbar.
- * Multicasting und T/TCP sind nicht aktiv.
- * Der IP-Filter steht nicht zur Verfügung.
- * Dauer-Ping-Schutz ist nicht anwählbar.
- * Das Sortieren, Mischen und der Clipboard-Im/Export der Datenbank sind nicht möglich.
- * PPP-Rückruf ist nicht anwählbar.
- * Die Paket-Monitor-Funktion (für externe Paket-Monitore wie MiamiTCPDump) funktioniert nicht.
- * System-Log-Ereignisse können nicht an syslog.library exportiert werden.
- * Die Hilfsprogramme MiamiIfConfig, MiamiRoute, MiamiSysCtl und MiamiTCPDump, sowie die Libraries miamibpf.library und miamipcap.library können nicht verwendet werden.
- * MS-CHAP-Unterstützung ist nicht verfügbar.

Das GTLayout-GUI-Modul hat einige Einschränkungen, verglichen mit der MUI-Version:

- * Drag&drop-Sortieren in Listviews ist nicht möglich.
- * Mit aktuellen Versionen von gtlayout.library (<=43.6) ist das Layout in einigen Windows nicht sehr hübsch: es scheint etwas unnötigen Leerraum im unteren Fensterbereich zu geben. Das ist vermutlich ein Fehler in gtlayout.library, der hoffentlich in neueren Versionen behoben wird.

1.133 Miami_deutsch.guide/NODE_HISTORY

Entwicklung

Version 3.0

Release-Version

- * Zu viele Änderungen, um sie hier anzugeben. Viele Teile des Programms wurden komplett neu geschrieben. Keyfiles V3 werden

benötigt.

Version 2.1p

Zwischenversion

- * Letzte offizielle Release-Version vor 3.0. Hauptsächlich Fehlerkorrekturen, und kleinere Änderungen verglichen mit vorherigen Versionen.

1.134 Miami_deutsch.guide/NODE_FUTURE

Die Zukunft

Meine kurzfristigeren Pläne für Miami beinhalten:

- * Ein neues API zur Handhabung von Online/Offline-Übergängen, kontrolliert von Clients.
- * Ein vollständig neues API ("ANDI") für wesentlich einfacheren, protokoll-unabhängigen Zugriff auf TCP/IP-Funktionen von Anwendungen.

ISDN ist ein anderer Schwerpunkt. Externe ISDN-Terminal-Adapter werden schon vollständig unterstützt. Das gleiche gilt für die ISDN-Master-Karte in den meisten Modi, mit Ausnahme von "synchronem PPP im HDLC-Modus". Dieser Modus erfordert neue Treiber und ein neues API, aber einige Leute arbeiten daran...

Die einzige größere (absichtliche) Einschränkung von Miami ist, daß nur eine einzige Schnittstelle genutzt werden kann. Ein Nachfolgeprogramm "Miami Deluxe" ist für später im Jahr 1998 geplant. Meine derzeitigen Pläne für Miami Deluxe beinhalten Unterstützung für PPP-Dial-in-Verbindungen, mehrere Schnittstellen, und vermutlich viele andere Funktionen, die für Router-Betrieb nützlich sind, wie Socks-Daemon-Unterstützung, IP-Masquerading, und Firewall-Funktionen.

Es wird für registrierte Miami-Anwender einen Rabatt bei der Registrierung von Miami Deluxe geben.

1.135 Miami_deutsch.guide/NODE_SUPPORT

Unterstützung

Es gibt verschiedene Möglichkeiten, technische Unterstützung und Updates zu bekommen:

Email

kruse@nordicglobal.com

normale Post

Holger Kruse
c/o Nordic Global Inc.
P.O. Box 780248
Orlando FL 32878-0248
USA

WWW

<http://www.nordicglobal.com/Miami.html>

Mailing-Listen

Schicken Sie "SUBSCRIBE miami-announce-ml" oder "SUBSCRIBE miami-talk-ml" im Text einer Email an "Majordomo@nordicglobal.com".

1.136 Miami_deutsch.guide/NODE_ACKNOWLEDGEMENTS

Danksagungen

Herzlichen Dank an

- * Die Tester früher Alpha- und Beta-Versionen Karl Bellve, Mike Fitzgerald, Adam Hough, Daniel Saxer, Stefan Stuntz und Oliver Wagner.
 - * Karl Bellve und Daniel Saxer für Ihre große Unterstützung.
 - * NSDi für das erste frei erhältliche TCP/IP-Protokoll-System für AmigaOS und eine sehr brauchbare Programmierschnittstelle.
 - * James Cooper, Steve Krueger und Doug Walker für das SAS/C Entwicklungssystem und Ihre große Unterstützung.
 - * Stefan Stuntz für sein hübsches User-Interface-Paket MUI.
 - * Klaus Melchior für die MUI custom class "Busy.mcc".
 - * Robert Reiswig für das Ausleihen von Computerhardware.
 - * der University of California für ihre erfolgreiche Arbeit an dem exzellenten BSD-Netzwerk-Code.
 - * Reinhard Spisser und Sebastiano Vigna for ihre Amiga-Portierung von "makeinfo".
 - * Paul Trauth, dem Sieger des Miami-Logo-Contest, for seine hübsche Image-Sammlung.
 - * John Pszeniczny for seine Variationen des "Miami"-Logos.
 - * Martin Huttenloher und Stefan Stuntz für die Erlaubnis, MagicWB-Grafiken in Miami zu benutzen.
 - * Roman Patzner für neu designete Icons.
-

- * Olaf Barthel für die `gtlayout.library`.
 - * allen Anwendern, die sich dazu entscheiden, sich für Miami zu registrieren.
-