

Gotcha

Thomas Bickel

COLLABORATORS

	TITLE : Gotcha		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY	Thomas Bickel	July 20, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Gotcha	1
1.1	Gotcha! v1.1	1
1.2	Limitations	1
1.3	Registration Form	2
1.4	Introduction	2
1.5	Bug Reports	3
1.6	Gotcha Echo	3
1.7	Features	3
1.8	Installation	4
1.9	Point Setup	4
1.10	Node Setup	5
1.11	The Modem	6
1.12	Nodelist	6
1.13	Custom Configurations	7
1.14	Basic Information	7
1.15	Forbid and Pickup	8
1.16	Keyboard	8
1.17	The Menu	9
1.18	Fax and Voice	10
1.19	ARexx	10
1.20	Dial Scripts	11
1.21	Call Options	12
1.22	Multiple Lines	13
1.23	OwnDevUnit Library	13
1.24	Configuration Commands	14
1.25	Embedded Percent Commands	14
1.26	Modem Strings	15
1.27	Commodity Info	15
1.28	Filenames	16
1.29	Flow Files	16

1.30 Compressed Mail	16
1.31 Outbound File Tagging	17
1.32 Inbound File Tagging	17
1.33 Filename Bumping	18
1.34 Acknowledgements	18
1.35 Requirements	18
1.36 Legal Stuff	18
1.37 Contact	20
1.38 Keywords	21
1.39 ADDMONITOR	23
1.40 ADDRESS	25
1.41 AFTERSESSION	25
1.42 AKAS	25
1.43 ALLOWREFUSING	26
1.44 ANSWER	26
1.45 AUTOSELECT	26
1.46 BACKGROUND	26
1.47 BANNER	27
1.48 BBS	27
1.49 BBSCHAR	27
1.50 BBSCOMMAND	27
1.51 BBSMESSAGE	28
1.52 BBSMODE	28
1.53 BOSSNUMBER	28
1.54 BUFFEREDLOG	29
1.55 CALL	29
1.56 CALLERID	29
1.57 CLOSEWINDOW	30
1.58 CONSCREEN	30
1.59 CONWIN	30
1.60 COST	30
1.61 CX_POPKEY	30
1.62 CX_PRIORITY	31
1.63 DIALER	31
1.64 DIALPREFIX	31
1.65 DIALSUFFIX	31
1.66 DIALTIMEOUT	31
1.67 DIRECTZAP	32
1.68 DISPLAYID	32

1.69 EMSI	32
1.70 ESCAPERESETSMODEM	32
1.71 FAXCOMMAND	33
1.72 FAXSTRING	33
1.73 FKEYs	33
1.74 FORBID	33
1.75 FREQUEST	34
1.76 GETCONFIG	34
1.77 HYDRA	34
1.78 ICONIFY	35
1.79 IMMEDIATE	35
1.80 INBOUND	35
1.81 INDIR	35
1.82 KEEP	36
1.83 KIWIPULSE	36
1.84 LISTEN	36
1.85 LOADINGMESSAGE	36
1.86 LOCATION	36
1.87 LOG	37
1.88 LOGFILE	37
1.89 LOGLEVEL	37
1.90 LOGLINES	37
1.91 MINBAUD	37
1.92 MODEMANSWER	38
1.93 MODEMDIALPOST	38
1.94 MODEMDIALPRE	38
1.95 MODEMHANG	38
1.96 MODEMHANGUP	39
1.97 MODEMINIT	39
1.98 MODEMRESETINTERVAL	39
1.99 NAME	39
1.100MODEMRESULTCODES	39
1.101NLDIR	40
1.102NOALLOWREFUSING	40
1.103NODELIST	41
1.104NODE	41
1.105NODIRECTZAP	41
1.106NOEMSI	41
1.107NOESCAPERESETSMODEM	41

1.108NOFORBID	42
1.109NOHYDRA	42
1.110NOKEEP	42
1.111NOKIWIPULSE	42
1.112NOLISTEN	42
1.113NONEMESSAGE	42
1.114NORETRYHANDSHAKE	43
1.115NOSTRIPDASHES	43
1.116NOSWEPULSE	43
1.117NOUNLISTEN	43
1.118NOWAZOO	43
1.119NOXMAIL	44
1.120NOZEDZAP	44
1.121NOZEDZIP	44
1.122ONCLOSEWINDOW	44
1.123OPENWINDOW	44
1.124OUTBOUND	45
1.125OUTDIR	45
1.126PASSWORD	45
1.127PHONENUMBER	45
1.128PICKUP	46
1.129QUIET	46
1.130QUIT	46
1.131REDIALDELAY	46
1.132REMMONITOR	46
1.133RESET	47
1.134RETRIES	47
1.135REXXNAME	47
1.136RINGCOMMAND	47
1.137RINGINGS	48
1.138RINGS	48
1.139RUN	48
1.140SCLEAR	48
1.141SCREENMODE	48
1.142SCREENTOBACK	49
1.143SCREENTOFRONT	49
1.144SERIALBAUD	49
1.145SERIALDEVICE	49
1.146SERIALFLAGS	49

1.147SERIALUNIT	49
1.148SHOWREXX	50
1.149SLINE	50
1.150SMODEM	50
1.151SPAWNMESSAGE	50
1.152SPAWN	51
1.153STARTSESSION	51
1.154TIMEOUT	51
1.155STRIPDASHES	51
1.156SWAIT	51
1.157SWEPULSE	52
1.158SWRITE	52
1.159SYSOP	52
1.160TASKPRI	52
1.161TIMESTAMP	52
1.162TURNAROUND	53
1.163UNICONIFY	53
1.164UNLISTEN	53
1.165USESYSFONT	53
1.166VERSION	53
1.167VOICECOMMAND	54
1.168VOICESTRING	54
1.169WAZOO	54
1.170XMAIL	54
1.171ZADZAP	54
1.172ZEDZIP	54
1.173ZMHMESSAGE	55
1.174Index	55

Chapter 1

Gotcha

1.1 Gotcha! v1.1

GOTCHA!

Version 1.2 beta 2

Copyright 1995/1996 Thomas Bickel; All rights reserved.

Requirements~~~~~	What you need for fun
Limitations~~~~~	Unregistered version limitations
Introduction~~~~~	About Gotcha!
Basic~Information~~~~~	How it works and how to use it
Keywords~~~~~	The keyword index
The~Support~Echo~~~~~	Infos about the Gotcha support echo
Contact~the~author~~~~~	! changed ! How to get in touch with the author
Legal~stuff~~~~~	Licence agreement, disclaimer and other stuff
Bug~Reports~~~~~	If you find a bug read this
Acknowledgements~~~~~	Acknowledgements
Registration~Form~~~~~	Registration Form

1.2 Limitations

Unregistered Version Limitations
~~~~~

First of all the unregistered version will open a little window each time you start it. You can close this window after a few seconds by clicking on the 'OK' gadget.

In addition to that the unregistered version has limits on how many calls it can make (outgoing and incomming). After five or six connections Gotcha will sooner or later quit.

---



## 1.3 Registration Form

### Gotcha Registration Form

~~~~~

Name: _____

Address: _____

Country: _____

Network-Addresses: _____

Computer Type: A __00

Processor: 680__

Kickstart Version: _____

Comment: _____

Price List:

Austrians: 200 öS

Rest of the World: 30 DM or 25 US\$ or 15 UK£

NOTE: Only Euro(!)cheques and cash will be accepted as payment.

Send this form along with the money to:

Thomas Bickel

Effingergasse 10/12

A-1160 Wien

Austria / Europe

1.4 Introduction

Features

A short list of some features

Installation

Some infos about the config file

Point~Setup

How to setup Gotcha for a Point system

Node~Setup

How to set up Gotcha for a node

The~Modem

What you need and how to set it up

The~Nodelist

What is that?

Custom~Configs

Set up configs for certain nodes only

1.5 Bug Reports

If you find a bug in Gotcha or one of the tools it would be quite nice if your sent a bug report.

If you send a bug report please give as much information as possible. Something like "Gotcha crashes, please fix it" does not help at all, there are lots of possibilities.

In addition to a exact description of the bug plus the circumstances it happens in, give the following informations (if you can):

What version of Gotcha are you using?
 Are you a registered user?
 What hardware are you using (CPU, memory, modem)?
 What patches are installed?
 Can the bug be reproduced on other amigas (models,configurations)?

The config file could be a help too (if it's Gotcha or GList that's buggy). And if the log shows something: Send the important parts.

1.6 Gotcha Echo

```

  _ _ _ _ _
 / _ | / _ \ _ _ / _ | | _ | / _ \
 | ( _ | ( _ ) || || ( _ | _ | _ |
 \ _ | \ _ / | _ | \ _ | _ | | _ | _ |

```

An echo setup to support and promote the Gotcha Amiga FTN mailer by Thomas Bickel. Echo name GOTCHA, available from:

```

2:259/15   - Home of the echo
2:259/26   - HUB, 259 NEC
2:259/66
2:250/366
2:250/347  - HUB
2:25/53    - MNL
2:442/403  - MNL

```

(Any other systems please let me know...)

For a feed contact 2:259/15

Also available is the related GOTCHA_AMY echo on Amiganet.

Lee Kindness, Trumpet Blower, cs2lk@scms.rgu.ac.uk Moderator of GOTCHA, GOTCHA_AMY.

1.7 Features

So what is this anyway....

Gotcha! is a mailer for nodes and points, meaning it will send out and receive mail and other files.

Features:

- * Does Hydra, DirectZap, ZedZap, ZedZip
- * EMSI and WaZoo handshakes
- * Is 5D aware
- * Makes use of the nodelist
- * Handles fax and voice calls
- * Tons of ARexx commands
- * Uses OwnDevUnit.library
- * Can handle more than 1 call at a time
- * Is a commodity
- * It's Faaaaaaaaaaaaast.....
- * Features Autoselect
- * Comes with a trapdoor.cfg converter

1.8 Installation

Gotcha is a CLI based application and can therefore not be run from Workbench. It accepts a large number of keywords that can either be set in a configuration file or passed as CLI-arguments.

Upon invocation Gotcha will look for its configuration file "Gotcha.config" in your "mail:" drawer. This file should contain your default settings, but should at least contain the BBS, ADDRESS and AKAS (if you have any) parameters. Points should also specify BOSSNUMBER and PASSWORD.

If you need different settings from what you specified in the configuration file you can either create another config file and pass its name as a CLI argument or list the parameters you wish to change as CLI arguments.

To tell Gotcha to use a different config file use the CONFIG keyword. E.g:
Gotcha CONFIG mail:gotcha.cfg2

To use no config file at all use: CONFIG "". If you do not use a config file then where possible default values will be used for all parameters, but you will need to specify other personal parameters as e.g. ADDRESS.

The following CLI command MAY be enough if you are only a point: Gotcha CONFIG "" BBS "Point Prince" BOSSNUMBER "1-800-123456789" ADDRESS "1:273/69.69" PASSWORD "slayer"

1.9 Point Setup

To set up Gotcha for a Point system just follow these steps:

1. Copy the files to their appropriate directories.
All libraries to libs: The executables to a path where they will be found, the other files where you want them.
2. If you don't already have a mail directory and a "mail:" assign create a new subdirectory and assign mail: to it, then create

- the two subdirectories "mail:inbound" and mail:outbound".
3. Copy a example config to your mail: directory and rename it to "Gotcha.config" or leave out this step and create a new file with your text editor.
 4. Use GConfig (from the Gotcha archive) or a text editor to change change the configuration file to suit your needs.
Most important keywords are BBS, SYSOP, ADDRESS, BOSSNUMBER, and PASSWORD.
Depending on your modem and serial device setup you might have to change some things there too.
Make sure the ANSWER keyword is either not present or set to NO (ANSWER = NO) .
 5. That's it power up your modem and type "Gotcha" to call your boss.

1.10 Node Setup

A node is more complicated to set up than a point. Follow the steps listed in the Point setup section but set leave out the BOSSNUMBER parameter and set "ANSWER~~YES". You will also need to set other keywords which are important for nodes. In addition to that you will need to create a nodelist directory and assign "nodelist:" to it (and set NLDIR = "nodelist:"). Have a look at the GList docs and compile the nodelist.

The main idea in a node setup is to have Gotcha make outgoing calls to deliver and receive mail and receive incoming calls. Since people don't want to sit in front of there computers all day (and night long) the task of managing outgoing calls is accomplished by the use of GPoll, a utility that will scan your outbound directory for files (mail most of the time) that need to be delivered and tell Gotcha to call those nodes, and a cron utility. A Cron utility is a tool that executes other programmes at certain times.

Gotcha will make outgoing calls if it receives a Rexx "CALL" command (and that's exactly what GPoll will do if it detects a file that needs to be delivered: send Gotcha a CALL command). Unlike some other mailers Gotcha does not only handle one number (or address) at a time but multiply numbers. E.g. with "CALL 2:310/108!39:23/5" Gotcha will first call 2:310/108 and then 39:23/5. If one of the calls fails, i.e. if the line was busy or the session could not be completed successfully, Gotcha will wait REDIALDELAY seconds and then retry to to call the system or get a successful session. This will happen as many times as specified with the RETRIES keyword. Numbers will be removed from the call-list if the modem reported anything else than BUSY after the dialstring was sent to it or if the session was successful. The Number will also be removed if you specified NORETRYHANDSHAKE and the handshake failed.

While waiting for the redialdelay to elapse Gotcha is able to receive incoming calls. Also if there was already a CALL command sent to Gotcha, other CALL commands will be accepted and the new numbers appended to those received in previous CALL commands.

1.11 The Modem

Your modem must be Hayes compatible and be configured in a way that returncodes (such as BUSY, NO DIAL TONE, CONNECT, etc.) are given in words and not as numbers (verbal/numeric mode), CONNECT responds from the modem must include the connection speed (i.e. modem to modem speed). If your modem supports the CARRIER keyword, the CONNECT message is expected to include the computer to modem speed and the CARRIER message should report the modem to modem speed.

Be sure to set up your modem in a way that dropping DTR causes the modem to hang up. If this is not the case Gotcha will not be able to hang up in most cases! This can be achieved with AT&D2 on most modems.

You will also need a seven-wire cable.

Slow modems: Some modems require a delay between AT and the rest of the command. If this is the case with your modem place one or two tilde(s) ("~") after the AT wherever you use it in. e.g MODEMINIT "AT~~Z|"

Example USRobotics Courier HST Dual Standard V.34+ Settings:

```

B0 C1 E1 F1 M1 Q0 V1 X7
BAUD=57600 PARITY=N WORDLEN=8
DIAL=HUNT ON HOOK TIMER

&A3 &B1 &C1 &D2 &G0 &H1 &I0 &K1 &L0 &M4 &N0
&P0 &R0 &S0 &T5 &X0 &Y1 %N6

S00=000 S01=000 S02=043 S03=013 S04=010 S05=008 S06=002 S07=060
S08=002 S09=006 S10=007 S11=070 S12=050 S13=000 S14=001 S15=000
S16=000 S17=000 S18=000 S19=000 S20=000 S21=010 S22=017 S23=019
S24=150 S25=005 S26=001 S27=000 S28=008 S29=020 S30=000 S31=000
S32=009 S33=000 S34=000 S35=000 S36=000 S37=000 S38=000 S39=000
S40=000 S41=000 S42=126 S43=200 S44=015 S45=000 S46=000 S47=000
S48=000 S49=000 S50=000 S51=000 S52=000 S53=000 S54=064 S55=000
S56=000 S57=000

```

1.12 Nodelist

What is it

A nodelist is a text-file which contains (lists) all nodes of a certain Network (FidoNet, AmigaNet, etc). It stores various informations about each node, such as the sysops name the location and phonenumber. Gotcha can make use of the information stored in the nodelist, e.g look up a nodes phonenumber. To keep up with changes (e.g new nodes, changed phonenumbers) so called nodediffs are used. These files are much smaler than the nodelist itself and contain informations on how to change the nodelist to get it up to date.

How to use it

To use a nodelist with Gotcha you need to compile it with GList. Have a look at the GList docs for more informations. GList will also create the latest nodelists from the old one and the latest nodediff(s).

Points usually don't need a nodelist since they only call one or two numbers and those can easily be stored in Gotchas config file.

1.13 Custom Configurations

For certain nodes you may want to set up a special configuration, for example if you regularly call a node that cannot handle EMSI you will want to turn it off to speed up the session handshake.

Gotcha (or rather GList) allows you to set up such special configurations for nodes. So to turn off EMSI for say 1:300/999 you just have to specify

```
setconfig 1:300/999 "NOEMSI YES"
```

in "glist.config".

More than one entry for a node is OK too. You can also specify a pattern so if you want to use a different aftersession script for AmigaNet you could specify:

```
setconfig 39:* "AFTERSESSION s:amiganet.bat"
setconfig 40:* "AFTERSESSION s:amiganet.bat"
setconfig 41:* "AFTERSESSION s:amiganet.bat"
```

To list all or parts of your custom configurations use the "showxconfig" tool that comes with Gotcha.

For more informations on custom configurations have a look at the GList docs.

1.14 Basic Information

Forbid~and~Pickup	How to (not) get what you (don't) want
The~Keyboard	Alt and Function key assignment
The~Menu	Explains that
Fax~and~Voice	Fax and Voice support
ARexx	The ARexx port
Dial~Scripts	Dial script examples
Call~options	More about the CALL command
Multiple~Lines	Use it in a multiline enviroment
OwnDevUnit~Library	Gotcha supports it
Config~Commands	Some format infos
Embedded~%-commands	A list of those
Modem~strings	Special characters in modem strings explained
Commodity~Infos	Horay! It's a commodity
Filenames	2D and 4D filenames
Flow~Files	Explains these

Compressed-Mail~files Filename format and when they are sent
Outbound~file~tagging Filenote tags for outgoing files
Inbound~file~tagging Filenote tags for outgoing files
Filename~bumping Bumping filenames in the inbound directory

1.15 Forbid and Pickup

EMSI handshake allows the setting of certain flags that tell the other system what to send and what not to send. The features described here will only work if the handshake used at session startup was EMSI. Note that Gotcha can only ask the other system not to send certain files, the remote system may ignore our wishes.

With setting the PICKUP keywords Gotcha can be configured to ask the other system to - send files for all addresses (primary and akas)
- send files for the primary address only
- send no files at all

These are calling system options.

With the FORBID keywords Gotcha can be configured to tell the remote system to NOT send - file requests
- compressed mail
- all of the above

These are called system options.

The default is to receive and send everything (except filerequests if there is no requesthandler installed).

1.16 Keyboard

Gotcha has some nice features controlled with the keyboard.

Esc ... If Gotcha is idle in answer mode Esc will reset the modem i.e. send the MODEMINIT string to the modem. In a session Esc will abort the session as soon as possible. This has about the same effect as turning off your modem.

Alt-A ... Answer the phone. Gotcha will react as if the number of RINGS specified with the RINGS keyword had been received.

Alt-Q ... Quit. Gotcha will abort a possible redialdelay and exit.

Alt-X ... the same as Alt-Q

Alt-R ... Reset modem

Alt-S ... Toggle SHOWREXX. i.e. (don't) show received REXX commands.

Alt-W ... Abort a redial. i.e. remove all numbers that would be called after the redialdelay time had elapsed from the list.

HELP ... Show some information. Such as keyboard information, settings and

functionkey settings

In addition to those above 'Cursor Up' will bring up a little window with a string gadget. Strings entered in that window will be sent to the ARexx port shown in the second gadget, right below the stringgadget. Make sure that you specify only ARexx ports as a destination or you might crash your Amiga.

There are also standard <left Amiga> keys for available for most of the listed functions.

Function keys: Strings assigned to function keys via the FKEY~and~SFKEY keywords will be sent to Gotchas ARexx port. So something like

FKEY1 "NOEMSI NO"

FKEY2 "NOEMSI YES"

FKEY3 "RUN GPoll" is possible since NOEMSI and RUN are ARexx commands. See the keywords section for a list of all ARexx keywords.

1.17 The Menu

Project

Answer Phone		Act as if an incomming call had been detected
		and send the MODEMANSWER string to the modem.
Toggle ShowRexx		Will toggle the SHOWREXX setting
Reset Modem		Will cause the MODEMINIT string to be sent to
		the modem
Redial Abort		Will abort a redial, while in waiting for
		the REDIALDELAY to expire, in ANSWER mode.
Abort Session/Reset Modem		Resets the modem, or if in a session aborts
		the session.
Informations...		Will show some infos
Arexx sender...		Opens a window which lets you send ARexx
		commands

Iconify		If Workbench is open this will iconify Gotcha.

Hide		This will close the window, and is the same
		as iconify, except that there is no icon.

Quit		Quit Gotcha

Tools

Poll...		A built in version of GPoll
FReq...		A built in version of GFreq
NodeInfo...		Will open a requester that lets you browse throught


```
| the nodes of all compiled nodelists.
```

Settings

```
-----
| General...      | Edit settings such as SYSOP, ADDRESS, AKAS, etc
| Modem...        | Edit the modem settings
| Modem RC...     | Edit modem result codes settings
| IO...           | Edit I/O settings
| BBS...          | Edit BBS settings
| Commands...     | Edit the command settings
| Display...      | Edit display (screen, window, log,...) settings
| Flow Control... | Edit session flow control settings
| Function Keys...| Edit function keys settings
|-----|
| Save...         | Save current settings
|-----
```

F1-F12

```
-----
| Whatever you set FKEY1,~FKEY2,~etc to will be
| shown here.
```

SF1-SF12

```
-----
| Whatever you set SFKEY1,~SFKEY2,~etc to will be
| shown here.
```

1.18 Fax and Voice

Fax and Voice support

If your modem supports fax and/or voice connects you can configure Gotcha in a way that it will spawn external commands if a voice or fax connection is detected. To do so you will have to set MRC_FAX~and~MRC_VOICE to what your modem reports if such an connect is made and FAXCOMMAND and VOICECOMMAND to what you want to be executed. FAXCOMMAND and VOICECOMMAND both may contain embedded~%-commands the %?-command can be used to launch different commands depending whether it was an incoming or outgoing call.

1.19 ARexx

ARexx

Gotcha features an powerful ARexx-port which allows easy control over Gotcha. While in answermode Gotcha will, except for the session handshake and for outgoing calls the time after the dialstring has been sent to the modem, always accept ARexx commands, even if it's just in a session and transferring files.

The port name will, if not changed to something else with the REXXNAME keyword, default to "GOTCHA". If a port with the given name already exists

Gotcha will add a dot and a number to the name and keep increasing the number until it is able to create a unique portname. I.e. if you invoke Gotcha three times with the default name unchanged the port names will be "GOTCHA", "GOTCHA.1" and "GOTCHA.2".

Nearly all commands that can be given in a configuration file can be sent to the ARexx port and will be understood. Exceptions are: BUFFEREDLOG, ANSWER, QUIET, REXXNAME, CX_POPKEY, CX_PRI and LOGLINES. If you need to change one of those you will have to terminate Gotcha and restart it with the new setting either as cli-argument or in a config file. In addition to those several other ARexx-only commands can be used to ease the writting of ARexx-programms.

NOTE: Serial parameters should only be changed while Gotcha is idle. Ie not making or receiving a call.

The most important ARexx command certainly is "CALL" which will make Gotcha call one or more nodes. There is an extra~section that describes more detailed all possibilities "CALL" has.

"GETCONFIG" will be quite handy if you need to change some settings but can not be sure what the current configuration is. Say you need to change the VOICECOMMAND:

```
OPTIONS RESULTS
```

```
ADDRESS "GOTCHA"
```

```
"GETCONFIG VOICECOMMAND"      /* get the configuration string */
```

```
oldcom = RESULT
```

```
...                             /* do what you need to do here */
```

```
oldcom                          /* and set back the old setting */
```

See the section about dial~scripts for some more ARexx information.

A small tool called GTell that comes with Gotcha may help you with sending ARexx commands. e.g 'GTell "call 123"' is the same as ' rx "ADDRESS 'GOTCHA';'call 123' " '.

See the GTools.guide for more information on GTell.

1.20 Dial Scripts

Dial Scripts

If you need to do more sophisticated dialing or want to use Gotcha as a dial-up tool to some other service you will need to write a ARexx script that will dial out and that tell Gotcha to start the file transfer. Something like this should work:

```
----- cut -----
ADDRESS "GOTCHA"
```

```

OPTIONS RESULTS
SIGNAL ON FAILURE
OPTIONS FAILAT 4          /* Gotcha will WARN if there's something wrong */

UNLISTEN                  /* so Gotcha can't interfere */
RESET                     /* reset the modem */

TIMEOUT 70                 /* timeout is 70 secs for connection */
'SMODEM AT~DT1234567|'    /* dial */

SWAIT "CONNECT"           /* and wait until we're connected */
SLINE
line = RESULT;

say "CONNECT" line;       /* show the speed */

TIMEOUT 25                 /* set the timeout to 25 secs */

SWAIT "login:"            /* wait for prompt */
SWRITE "Dan Aykroyd"||d2c(13) /* send the name */
SWAIT "password:"
SWRITE "John Belushi"||d2c(13)

SWAIT "connection ok"

STARTSESSION              /* and start the session */
LISTEN                    /* really! */

EXIT

FAILURE:
say "Could not connect"
EXIT

```

----- cut -----

1.21 Call Options

Gotcha has the ability to call more than one number at one time. I.e. you don't have to first tell Gotcha to call a number check wether Gotcha has made the call and then tell Gotcha to call the next number but you can specify a list of numbers Gotcha will call.

Numbers have to be seperated with a '!'.
 E.g. CALL "2:310/0!2:310/108!39:23/5!2:310/26" would first call 2:310/0

then 2:310/108 and so forth.

Gotcha will try to call the number, if the line is BUSY Gotcha will skip to the next number, if a connect is made a handshake will be attempted. If in this phase the modem responds with something else than a BUSY or a CONNECT/CARRIER string the number will be removed from the list.

If the handshake failes and NORETRYHANDSHAKE is YES the number will be removed from the list, else Gotcha will just skip this number, try to call again later and continue with the next number.

If the session is successful the number will be removed from the list else it will be skipped.

If all numbers in the list have been called and there are retries left Gotcha will wait RETRYDELAY time and then attempt to call the number that were not removed on the last attempt. This will happen as many times as specified with the RETRIES keyword.

Even if Gotcha is already busy calling numbers, i.e. if you already sent a CALL string to Gotcha and there are still numbers left to call from that previous CALL-command, new CALL commands can be sent to Gotcha. Those new numbers will be appended to the list of numbers to call. The number of numbers Gotcha can manage is aprox. 80.

Akas will be removed too. If you call 1:2/3 AKA 4:5/6 with a CALL "1:2/3!4:5/6" both numbers will be removed from the list after a successful session (and if the other side tells Gotcha about the AKA).

1.22 Multiple Lines

Gotcha can be used without any problems on multiple lines. Using it with an multiline BBS is no problem.

It is recommended to make Gotcha resident if you intend to use it more than once at a time. This will help to save some memory. The "r" attribute will have got lost during archiving with lha but Gotcha has been compiled with a special resident header and can therefore be made resident with the cli "resident" command.
E.g "resident mail:bin/Gotcha".

Gotcha will take care that its ARexx portname is unique but if you want to be sure that each line has the same port name all the times you should specify a port name for each invocation. Obviously at least different serial units or even different devices will need to be specified too. Those few changes to the config file can be given as command line parameters but it may be useful to copy the config file, make the changes to the new config file and specify this one as a command line parameter.

Note that if a config file is specified as a cli argument only that one will be read and the default "mail:gotcha.config" will not be read.

1.23 OwnDevUnit Library

OwnDevUnit.library is a freeware library that manages access to devices for different programs.

Say you have Gotcha running at serial.device unit 0 and want to make a call with Term. Without OwnDevUnit.library you will have to disable Gotcha and then make your call with Term. With OwnDevUnit.library Gotcha will be asked to release the device and Term can access it without the fear of Gotcha interfering. If Term is done it will tell

OwnDevUnit.library so and Gotcha will get back access to the serial device.

OwnDevUnit.library support is automatic. If the library can be opened, i.e it's in your libs: directory, it will be used.

1.24 Configuration Commands

All possible configuration and ARexx commands are listed in the keyword~index section. The keywords are neither position nor case sensitive. Keywords that can be used both as a config parameter (i.e. in a config file or as a cli argument) and as ARexx command are marked "Config & Rexx", ARexx only commands "Rexx".

If you wish to include spaces in a string parameter you must enclose the string in double-quotes. If you want to use double-quotes in a string you will need to escape the quote with a '\' (backslash). In this case backslashes need to be escaped too.

If you use GConfig you don't have to worry about all this.

```
Example: SYSOP "John Doe"
        SYSOP "John \"the quoted\" Doe"
        SYSOP "John \"the quoted\" Doe with a backslash \"
```

1.25 Embedded Percent Commands

In the AFTERSESSION, BBSCOMMAND, DIALER, FAXCOMMAND, FREQUEST, RINGCOMMAND, TURNAROUND and VOICECOMMAND command-strings the following case-sensitive embedded %-commands can be used.

```
%s ... Serial device name
%u ... Serial device unit
%f ... Serial device flags
%b ... Baudrate (computer to modem)
%B ... Baudrate (modem to modem)
%C ... Full CONNECT string from modem
%Z ... Zone number of the remote system
%N ... Net number of the remote system
%F ... Fido node number of the remote system
%P ... Point number of the remote system
%D ... Domain of the remote system
%n ... Complete FidoNet address of the other system
%a ... Aka of the remote system (one aka per %a)
%A ... All akas of the remote system (separated with space)
%S ... Name of Sysop of other system
%R ... Name of Gotchas ARexx port
%p ... Name of the public screen
%l ... Name of the logfile
%I ... Inbound path
%O ... Outbound path
%d ... 1 if inbound call, -1 if outbound call
```

```
%#?... see explanation below
%r ... Unique random number (a timestamp)
%% ... %
```

In order to make it possible to execute different commands or pass different arguments to commands on incoming than on outgoing calls the %? command was implemented. E.g TURNAROUND "echo \"Turnaround for this %?incoming#outgoing?call\"" would echo "Turnaround for this incoming call" for incoming calls and "Turnaround for this outgoing call" for outgoing calls.

For the FREQUEST command, the following commands are allowed in addition to the above:

```
%i ... name of the received .REQ file
%o ... name of the .RLO file the result should be stored in
```

1.26 Modem Strings

All commands sent to the modem such as DIALSUFFIX, MODEMINIT, MODEMHANG, MODEMANSWER accept some special characters in their strings.

```
~ ... Short delay (aprox. 1/3 of a sec)
^ ... Hang up, (i.e. drop DTR - wait - raise DTR)
| ... Send a carriage return character
\ ... Escape character, send next character unchanged
```

All other characters will be sent to the modem without any changes. If you need to send a "~", "^" or "|" character to the modem you will need to escape it with a backslash ('\').

E.g. if you want to send ~AT~ to your modem you have to use \~AT\~.

Take care if you need to have spaces in these strings you will have to escape the backslash (since the string must be put in double-quotes then).

E.g. " AT ~" -> " AT \~"

And even worse if you need to send a backslash to the modem:

```
' AT \' -> " AT \" -> " AT \\"
```

This will result in " AT \" when the config is read and when sent to the modem '\\' will become a single backslash again.

1.27 Commodity Info

Gotcha is a commodity. The popup key can be specified with CX_POPKEY command and the priority with CX_PRIORITY.

If the hotkey is pressed Gotcha will open its window nearly all the time immediately. Delays may happen if Gotcha is just in handshake phase with another system, just have some patience then.

If you click the close gadget Gotcha will hide rather than exit. If Gotcha has opened its own screen that one will be closed too if no other windows are opened on it.

If you have opened Workbench you can iconify Gotcha, it will use its info file "mail:def_gotcha.info" or, if that's not available, use the default tools icon.

1.28 Filenames

2-dimensional Filenames

2D filenames have the form 'aaaabbbb.ext'.
'aaaa' is the net number in hexadecimal notation.
'bbbb' is the node number in hexadecimal notation.
'ext' the file extension (e.g. 'PKT')

This format is rather old and usually files are stored with 4D filenames nowadays and are converted to 2D when transferred.

4-dimensional Filenames

4D filenames have the form 'zone.node.net.point.ext'. The advantage over 2D-filenames is that zone and point information is included in the file name, most mailers/tossers support this naming scheme.

Gotcha will scan the outbound directory for both filenames.

1.29 Flow Files

Flow Files

These files hold the path and name of files that should be sent to the other system. Each line refers to one file. The filenames listed may be preceded by a special character, indicating that some special processing needs to be done after sending the file.

Possible special characters are:

- ~ ... Don't send this file (has already been sent)
- @ ... Allow filenote tagging
- ... Delete this file without logging
- ^ ... Delete this file with logging
- # ... Truncate this file to zero length

Files with the following extensions are flow files: .FLO, .HLO, .CLO, .DLO

1.30 Compressed Mail

Compressed Mail files

These files are not sent automatically, i.e. Gotcha won't look for these files when scanning the outbound directory while in a session. To be sent out they must be listed in a flow file (#?.?LO, e.g. 35.23.7.34.FLO).

The filenames have the form <address>.<weekday><number>, where address is either a 4D filename (e.g. 22.7.44.0.MO0) or consists of 8 hex digits, the first 4 representing the difference between the net number of the originating and destination system, and the second 4 digits the difference between the node numbers. 'weekday' is two characters, the first two characters of a weekday (i.e. MO, TU, WE, TH, FR, SA, SU) and number is a hex digit to prevent duplicate filenames.

The creation of both compressed mail files, and the belonging flow files is usually the duty of a tosser (Spot, Traptoss, MailManager just to name a few).

Examples: 34.5.23.0.MO9, 001F0034.SU0

1.31 Outbound File Tagging

Outbound File Tagging

Some outgoing files need to be renamed, e.g. 4D filenames will be converted to 2D filenames and the extension of mail packets (.OUT, .DUT, .HUT, .CUT) will be changed to .PKT.

Sometimes (e.g. for points) a random 2D filename will be created (from the current time), now if a transfer failes those files would get a different name, thus making a resume impossible, to avoid this the newly created name will be recorded in the file's filenote with a 'Name' tag, so in case of a interrupted transfer the file will be sent under the same name again.

With this you can make Gotcha send files with any name you want e.g. with 'filenote mail:outbound/foo.doc "Name bar.txt;"' you can make Gotcha send 'foo.doc' as 'bar.txt' (of course 'foo.doc' needs to be listed in a flow file to be sent at all).

1.32 Inbound File Tagging

Inbound File Tagging

When Gotcha receives a file it will first give it a 'transfer name'. I.e. ç.87ED8ADFF.PKT.2.139.34.0. If the transfer was successfull the file will be renamed to it's original name (in this case 87ED8ADFF.PKT).

Gotcha will also set some filenote tags:

FileName	... the original filename (useful if the filename was bumped)
From	... the address of the node we got the file from
Length	... the length of the file

1.33 Filename Bumping

Filename Bumping

If a file has been received successfully Gotcha will rename the file to its original name (Gotcha received it under a special filename) it may however happen that a file with the same name already exists.

If the filename consists of 8 digits plus a dot plus a 3 character extension Gotcha will change the digit part until the filename is unique.

If the filename is not a 8 digit + dot + 3 char extension name Gotcha will add a ", " and a number and keep incrementing the number until the filename is unique.

1.34 Acknowledgements

Thanks got to
all registered users

everybody who sent bug reports and/or suggestions

Olaf 'Olsen' Barthel for releasing the Term and HydraCom Amiga sources.

The following names are either trademarks or the efforts of the person and/or company listed:

gtlayout.library by Olaf 'Olsen' Barthel

xprzedzap.library by Yves Konigshofer.

YooHoo by Wynn Wagner III.

ZModem by Chuck Forsberg.

The Licence Agreement was inspired by the FrontDoor Licence Agreement.

Fido and FidoNet are trademarks of Tom Jennings, Fido Software.

FrontDoor by Joaquim Homrighausen.

ARexx by William S. Hawes, Wishful Thinking Development Corp.

EMSI by the EMSC.

Excelsior! BBS by Sycom Design.

OwnDevUnit.library by Christopher A. Wichura

TrapDoor is a registered trademark of Maximilian Hantsch

and Martin Laubach.

1.35 Requirements

Gotcha requires OS 2.04 or better. That's it!

(Oh well, alright, you got me; I admit: some kb of memory and disk space will be needed too ;)

1.36 Legal Stuff

All Thomas Bickel products are trademarks of Thomas Bickel. Other brand and product names are trademarks or registered trademarks of their respective holders.

Copyright 1995/1996 Thomas Bickel; All rights reserved.

LICENCE AGREEMENT

This is a licence agreement between you, the user, and Thomas Bickel, hereafter "the author". The Gotcha Shareware software and information attached hereto, hereafter "Gotcha", is the property of the author. Read the terms and conditions of this licence agreement carefully before using the software. If you for any reason, whatsoever, cannot accept the conditions in this agreement, you are not permitted to use Gotcha.

You acknowledge and agree that Gotcha is a proprietary product of the author, protected by applicable copyright laws and international treaty provisions. You further acknowledge and agree that all rights, title, and interest in and to Gotcha are and shall remain with the author.

Gotcha may be used for a period of thirty (30) days on a trial basis to allow you to determine its suitability for your particular application. After this period you MUST register Gotcha.

Registration entitles you to a non-transferable licence to use Gotcha and any future versions of Gotcha for as long as you wish, subject to any special licensing conditions and/or applicable upgrade fees attached to future versions.

In the event that you are in violation of this licence agreement, you agree and accept that the author may cancel your registration and any rights to use Gotcha that you may have. In the case of a dispute over the licence agreement, you further agree and accept to fully reimburse the author for legal and other expenses resulting from the dispute, should the dispute be decided in favor of the author.

The author is in no way obligated to provide future versions of, or support for, Gotcha.

You may freely distribute and copy Gotcha provided that no fee is charged and the Gotcha distribution archive contains unmodified copies of the original files as produced by the author. No part of Gotcha may be modified, altered, reverse engineered, sold, or distributed in any form whatsoever which would involve some sort of trade without prior written permission from the author. Gotcha may be put on Aminet CDs.

In the case of any of the provisions of this agreement not being legally effective or losing its legal validity due to some subsequent circumstance, or in the case of a gap in this agreement coming into light, the legal validity of the remaining provisions will not be affected. An appropriate regulation is to take place of the ineffective provisions or fill the gap, which as far as legally possible, comes nearest to fulfilling the intentions the author had he considered the matter in question. If the invalidity of a provision is due to a scale of the performance contained

therein, then a legally permissible scale is to be inserted with due regard to the aforesaid.

DISCLAIMER

Gotcha is provided "as is", without warranty of any kind or fitness for a particular purpose, either expressed or implied, all of which are hereby explicitly disclaimed. The author only guarantees that Gotcha will occupy disk space.

In no event shall the author be liable to you or anyone else for any damages, including any lost profits, lost savings or other incidental or consequential damages arising out of the use or inability to use this software.

=====

The HYDRA protocol was designed by
Arjen G. Lentz, LENTZ SOFTWARE-DEVELOPMENT and
Joaquim H. Homrighausen
COPYRIGHT (C) 1991-1993; ALL RIGHTS RESERVED

Hydra protocol design and HydraCom driver:
Arjen G. Lentz
LENTZ SOFTWARE-DEVELOPMENT
Langegracht 7B
3811 BT Amersfoort
The Netherlands
FidoNet 2:283/512, AINEX-BBS +31-33-633916
arjen_lentz@f512.n283.z2.fidonet.org

Hydra protocol design:
Joaquim H. Homrighausen
389, route d'Arlon
L-8011 Strassen
Luxembourg
FidoNet 2:270/17
joho@ae.lu

=====

The xprzedzap.library is public domain and is not copyrighted. You may do with it as you please. It was written by Yves Konigshofer. See the xprzedzap.library documentation for more information.

=====

gtlayout.library:
Copyright (c) 1993-1995 by Olaf 'Olsen' Barthel
Freely distributable.

=====

1.37 Contact

The author of this software can be reached via e-mail:

2:310/72.34@FidoNet

Internet: tbickel@xpoint.at

Or via snail mail:

Thomas Bickel
Effingerg. 10/12
A-1160 Wien
Austria / Europe

1.38 Keywords

ADDMONITOR
ADDRESS
AFTERSESSION
AKAS
ALLOWREFUSING
ANSWER
AUTOSELECT
BACKGROUND
BANNER
BBS
BBSCHAR
BBSCOMMAND
BBSMESSAGE
BBSMODE
BOSSNUMBER
BUFFEREDLOG
CALL
CALLERID
CLOSEWINDOW
CONSCREEN
CONWIN
COST
CX_POPKEY
CX_PRIORITY
DIALER
DIALPREFIX
DIALSUFFIX
DIALTIMEOUT
DIRECTZAP
DISPLAYID
EMSI
ESCAPERESETSMODEM
FAXCOMMAND
FAXSTRING
FKEYs
FORBIDALL
FORBIDFREQ
FORBIDXMAIL
FREQUEST
GETCONFIG
HYDRA
ICONIFY
IMMEDIATE
INBOUND
INDIR
KEEPALL

KEEPXMAIL
KIWIPULSE
LISTEN
LOADINGMESSAGE
LOCATION
LOG
LOGFILE
LOGLEVEL
LOGLINES
MINBAUD
MODEMANSWER
MODEMDIALPOST
MODEMDIALPRE
MODEMHANG
MODEMHANGUP
MODEMINIT
MODEMRESETINTERVAL
MRC_BUSY
MRC_CARRIER
MRC_CONNECT
MRC_FAX
MRC_NOANSWER
MRC_NOCARRIER
MRC_NODIALTONE
MRC_RING
MRC_RINGING
MRC_VOICE
NAME
NLDIR
NOALLOWREFUSING
NODELIST
NODE
NODIRECTZAP
NOEMSI
NOESCAPERESETSMODEM
NOFORBIDALL
NOFORBIDFREQ
NOFORBIDXMAIL
NOHYDRA
NOKEEPALL
NOKEEPXMAIL
NOKIWIPULSE
NOLISTEN
NONEMESSAGE
NORETRYHANDSHAKE
NOSTRIPDASHES
NOSWEPULSE
NOUNLISTEN
NOWAZOO
NOXMAIL
NOZEDZAP
NOZEDZIP
ONCLOSEWINDOW
OPENWINDOW
OUTBOUND
OUTDIR
PASSWORD

PHONENUMBER
PICKUP
QUIET
QUIT
REDIALDELAY
REMMONITOR
RESET
RETRIES
REXXNAME
RINGCOMMAND
RINGINGS
RINGS
RUN
SCLEAR
SCREENMODE
SCREENTOBACK
SCREENTOFRONT
SERIALBAUD
SERIALDEVICE
SERIALFLAGS
SERIALUNIT
SFKEYs
SHOWREXX
SLINE
SMODEM
SPAWNMESSAGE
SPAWN
STARTSESSION
TIMEOUT
STRIPDASHES
SWAIT
SWEPULSE
SWRITE
SYSOP
TASKPRI
TIMESTAMP
TURNAROUND
UNICONIFY
UNLISTEN
USESYSFONT
VERSION
VOICECOMMAND
VOICESTRING
WAZOO
XMAIL
ZEDZAP
ZEDZIP
ZMHMESSAGE

1.39 ADDMONITOR

ADDMONITOR ARexx-port

Config & REXX

This adds a ARexx-port to the list of ports Gotcha sends status messages to.

Lists in this list will receive messages in the form:
"<STATUS><GOTCHA-REXXPORT>[<optional information>]"

Following states are defined yet (<STATE><optional information>):

GRX_CALLING <number> ... Gotcha is about to call <number>

GRX_CONNECT <connect string> ... There was a connect

GRX_HANDSHAKE ... Gotcha is about to enter handshake phase

GRX_XFER ... Files will be transfered now

GRX_CALL_OK ... The call was successful

GRX_CALL_FAILED ... The cal was not successful

GRX_IDLE ... Sent every MODEMRESETINTERVAL minutes

GRX_REDIALDELAY ... There are numbers left to dial entering delay

GRX_REDIALABORT ... Further retries have been aborted

GRX_REDIALEND ... Nothing left to dial

GRX_QUIT ... Gotcha has been terminated

GRX_CALL_INFO <online secs> <cost> ... Given after a call if secs<>0

GRX_INCOMING ... Incoming call detected

GRX_UNLISTEN ... UNLISTEN received

GRX_LISTEN ... LISTEN received

GRX_ODUNLISTEN ... OwnDevUnit requested the serial device

GRX_ODLISTEN ... Got the serial device back from OwnDevUnit

GRX_CALL_LIST <numbers> ... All numbers in the call/dial list

GRX_END_OF_CALL ... At the end of a call

GRX_HUMAN_CALLER ... A human caller was detected

GRX_FAX ... A fax call was detected

GRX_VOICE ... A voice connect was made

GRX_RI_NAME <name> ... Remote info name

GRX_RI_ADDRESS <address> ... Remote info address

GRX_RI_AKA <akas> ... Remote info akas

GRX_RI_SYSOP <sysop> ... Remote info sysop

```
GRX_RI_PLACE <place> ... Remote info location

GRX_RI_PHONENUMBER <phonenum> ... Remote info phonenum

GRX_RI_MAILER <mailer> ... Remote info mailer

GRX_RI_OFFER <offer> ... Remote info offer

GRX_RI_FLAGS <flags> ... Remote info flags

GRX_RI_END ... End of remote info messages
```

In a config file (cli arg) only one port can be specified, via the ARexx command a unlimited number can be added. Upon reception of ADDMONITOR the given port must already exist or it will not be added. Ports a message could not be sent to will be removed from the port list.

Example: ADDMONITOR "GOTCHA_MONITOR"

1.40 ADDRESS

```
ADDRESS zone:net/node[.point][@domain] Config & Rexx
```

This sets your main address.
If you have more then one address see AKAS.

```
Examples: ADDRESS 2:310/108@fidonet
          ADDRESS 39:23/5.3@AmigaNet
          ADDRESS 2:310/108.3
```

1.41 AFTERSESSION

```
AFTERSSESSION command-string Config & Rexx
```

The command executed after every session.

After every session with another node the command specified with this keyword will be executed. There may be embedded~%-commands in the string. The values such as %n (address of the remote system) will be substituted by the parameters of the session that just ended. To have no aftersession-command use AFTERSSESSION "" or omit the statement completely.

```
Examples: AFTERSSESSION "execute s:aftersession.bat"
          AFTERSSESSION ""
```

1.42 AKAS

AKAS akalist

Config & Rexx

During an EMSI-handshake not only your main address will be sent to the other system but (if available) also a list of "also-known-as" addresses. Using the AKAS keyword you can specify all of your AKA addresses.

The length of this string is limited to 1000 characters.

Example: AKAS "39:23/5.3@amiganet 27:47/11"
AKAS ""

1.43 ALLOWREFUSING

ALLOWREFUSING

Rexx

This is a synonym for NOALLOWREFUSING NO.

1.44 ANSWER

ANSWER on/off

Config

If answer is on Gotcha will hang around receiving and making outgoing calls. This really should be turned on if you are a node... For points this should be turned off. If turned off Gotcha will call the number(s) specified with BOSSNUMBER and then exit.

Examples: ANSWER ON
ANSWER OFF

1.45 AUTOSELECT

AUTOSELECT on/off

Config & Rexx

When AUTOSELECT is turned on and you have AKAS Gotcha will select the address that matches the remotes one best as your main address. Say you are ADDRESS 2:310/108.3 AKA 39:23/5.3, so if AUTOSELECT is turned on and you are calling 39:23/5, 39:23/5.3 will become main address and 2:310/108 will be your AKA.

Examples: AUTOSELECT ON
AUTOSELECT OFF

1.46 BACKGROUND

BACKGROUND on/off

Config & REXX

When BACKGROUND is turned on Gotcha will open it's screen behind all other screens.

Examples: BACKGROUND ON
BACKGROUND OFF

1.47 BANNER

BANNER bannerline

Config & REXX

This line (or file) will be sent to the other system when Gotcha answers a call. To specify a file use "<" as the first character.

Examples: BANNER "Welcome to TNT Amiga only BBS Line 2"
BANNER "<mail:welcome.txt"

1.48 BBS

BBS bbsname

Config & REXX

Name of this node which will be sent to the remote system while handshake.

1.49 BBSCHAR

BBSCHAR character

Config & REXX

This option is included to allow for other "drop-to-the-bbs" characters in addition to the ESC-key.

Examples: BBSCHAR !
BBSCHAR '!'
BBSCHAR \$21

1.50 BBSCOMMAND

BBSCOMMAND command-string

Config & REXX

This command which may include embedded ~%-commands will be executed whenever a human caller is calling your BBS.

This command will only be executed if the BBSMODE is SPAWN or EXIT.

Example: BBSCOMMAND = "bbs:excelbbs/bb -B %B -U 1 -L 1 -G -7 -S"

1.51 BBSMESSAGE

BBSMESSAGE text

Config & Rexx

With this keyword you can override the ZHMMESSAGE, NONEMESSAGE, SPAWNMESSAGE settings. If not empty instead of the texts specified with the other keywords this text will be sent right after the banner.

Examples: BBSMESSAGE "Sorry, this BBS is offline until 5:30"
BBSMESSAGE ""

1.52 BBSMODE

BBSMODE mode

Config & Rexx

There are four different modes to use Gotcha with(out) a BBS:

NONE ... There is no BBS. NONEMESSAGE will be displayed to human callers

SPAWN ... If a human caller enters two ESC characters the BBS will be launched. When the command returns the modem will be reseted and Gotcha will continue waiting for calls. SPAWNMESSAGE is displayed.

EXIT ... Same as SPAWN, but as soon as the command returns Gotcha will exit.

ZMH ... "Zone Mail Hour" There is a BBS, but at the moment no human callers are allowed. Gotcha will display ZMHMESSAGE.

Examples: BBSMODE SPAWN
BBSMODE ZMH

1.53 BOSSNUMBER

BOSSNUMBER number/address

Config & Rexx

With this keyword points can specify their bosses phonenumbers or address(es). If Gotcha is not in answer mode it will simply call the number(s) specified with this keyword and then exit. If you use a nodelist you can also specify addresses here. Note that this can also be used in answer mode. To make Gotcha call the numbers given with this keyword send "CALL BOSS" to Gotchas ARexx port.

Examples: BOSSNUMBER "2:310/108"
BOSSNUMBER "39:23/5!2:310/108"

1.54 BUFFEREDLOG

BUFFEREDLOG yes/no

Config & Rexx

If NO is selected every line written to the window will be written to the logfile immediately. If YES is selected the lines will first be written to a buffer and every ~512 bytes written to the logfile.

The advantage of turning buffering on is that it is much faster, however if there are crashes the important part may not be written to the log.

Examples: BUFFEREDLOG YES
BUFFEREDLOG NO

1.55 CALL

CALL number(s)

Rexx

If Gotcha is in answer mode you can place outgoing calls by sending ARExx CALL commands to it's rexx-port. Unlike other mailers Gotcha can handle more than one number/address at one time.

If Gotcha is already making outgoing calls it will attempt to append the new numbers to the list of old ones. This may fail if you specify to many numbers.

There is a limit of 1000 characters for this string.

If you are using addresses you must specify the zone since Gotcha uses the ':' character to distinguish addresses from phonenumber.

Please note that REDIALDELAY and RETRIES will also be used.

First Gotcha will try to call all nodes in the list. For successfull sessions the main address and all akas will be removed from the list. If lines were BUSY or sessions were not successfull Gotcha will wait REDIALDELAY seconds until it makes another attempt. This will happen RETRIES times.

Note that in answer mode Gotcha is able to receive incoming calls while it's waiting for another retry.

Examples:

"CALL 2:310/108!39:23/5" will call 2:310/108 and 39:23/5 and keep calling until either each call was successfull or no retries are left.

"CALL BOSS" will call the numbers specified with the BOSSNUMBER keyword.

1.56 CALLERID

CALLERID <filename>

Config & Rexx

If specified the callerid (received between first and second RING) will

be saved to <filename>.

Example: CALLERID "mail:cid"

1.57 CLOSEWINDOW

CLOSEWINDOW

Rexx

This Rexx-command will close the outputwindow.

1.58 CONSCREEN

CONSCREEN screenname

Config & Rexx

Specifies the screen Gotcha should open on. If no such (public) screen exists Gotcha will open an own screen, depending on the SCREENMODE.

Example: CONSCREEN "SPOT"

1.59 CONWIN

CONWIN Left/Top/Width/Height

Config & Rexx

This specifies the position and size of the window.

Example: CONWIN 10/10/500/200

1.60 COST

COST value

Config & Rexx

This value which should be given as units/minute, will be used if no nodelist is used to determine the cost of a call.

Example: COST 100

1.61 CX_POPKEY

CX_POPKEY string

Config

This specifies the commodity "Hotkey". If pressed and the window is closed the window will open.

Example: CX_POPKEY "rwaakey control esc"

1.62 CX_PRIORITY

CX_PRIORITY pri

Config

This is the commodity priority.

Example: CX_PRIORITY 0

1.63 DIALER

DIALER dialer-command

Config & Rexx

Use this if you have a custom dialer that performs some special magic for you. Embedded~%-commands can be used in the command.

If specified instead of dialing out the command will be executed and if it returns the session handshake will be attempted.

Examples: DIALER "callpcp"
DIALER ""

1.64 DIALPREFIX

DIALPREFIX string

Config & Rexx

Modem dial string such as "ATDT" or "AT~~DT~". This string will be sent to the modem whenever Gotcha wants to call out. The string can have special~modem~characters in it if needed.

Examples: DIALPREFIX "ATDT"
DIALPREFIX "AT~~D"

1.65 DIALSUFFIX

DIALSUFFIX string

Config & Rexx

This string will be sent to the modem after the DIALPREFIX string and the number have been sent. This usually is just a "|". The string can have special~modem~characters in it if needed.

Example: DIALSUFFIX "|"

1.66 DIALTIMEOUT

DIALTIMEOUT seconds

Config & Rexx

Specifies how long Gotcha will wait for a response (e.g. BUSY, NO DIAL TONE, VOICE, NO CARRIER, CONNECT) from the modem after it has sent the dialstring.

Example: DIALTIMEOUT 70

1.67 DIRECTZAP

DIRECTZAP

Rexx

This is the same as NODIRECTZAP = NO.

1.68 DISPLAYID

DISPLAYID integer

Config & Rexx

Specifies the mode in which Gotcha should open it's screen. This ofcourse will only have an effect if Gotcha opens it's own screen at all (see SCREENMODE).

Examples:

DISPLAYID \$19000	;NTSC Hires
DISPLAYID \$29000	;PAL Hires
DISPLAYID \$19020	;NTSC Superhires
DISPLAYID \$29020	;PAL Superhires
DISPLAYID \$19004	;NTSC Hires Interlace
DISPLAYID \$29004	;PAL Hires Interlace
DISPLAYID \$19024	;NTSC Superhires Interlace
DISPLAYID \$29024	;PAL Superhires Interlace
DISPLAYID \$39024	;VGA Productivity
DISPLAYID \$39025	;VGA Productivity Lace
DISPLAYID \$39020	;VGA Productivity Dbl (AA)

1.69 EMSI

EMSI

Rexx

This is the same as NOEMSI = NO.

1.70 ESCAPERESETSMODEM

ESCAPERESETSMODEM yes/no

Config & Rexx

If you press the ESC key during a session Gotcha will stop the session immediately. If this option is YES the modem-init string will be sent too. If it is no then you will have to press ESC a second

time to reset it or wait until MODEMRESETINTERVAL does it for you.

Example: ESCAPERESETSMODEM NO

1.71 FAXCOMMAND

FAXCOMMAND command

Config & REXX

This command, which may contain embedded ~%-commands, will be executed whenever, on a outgoing or incoming call, the string specified in MRC_FAX is received.

Examples: FAXCOMMAND "fax %?letter1#letter2?"
FAXCOMMAND ""

1.72 FAXSTRING

FAXSTRING string

REXX

This is the same as MRC_FAX.

Example: FAXSTRING "FAX"

1.73 FKEYs

FKEY<x> rexx-command

Config & REXX

SFKEY<x> rexx-command

Assigns a rexx-command to a functionkey. Whenever a functionkey is pressed Gotcha will send the associated command to it's own REXX-port.

Examples: FKEY1 "NOEMSI NO" ; F1
FKEY2 "NOEMSI YES" ; F2
FKEY3 "CALL 123456" ; F3
SFKEY1 "NOWAZOO NO" ; Shift-F1
SFKEY9 "NODIRECTZAP YES" ; Shift-F9

1.74 FORBID

FORBIDALL yes/no

Config & REXX

FORBIDFREQ yes/no

FORBIDXMAIL yes/no

This selects what kind of mail you do not want to receive. In an EMSI handshake Gotcha will ask (!) the other system not to send theses files. The other end may however simply ignore this

(e.g. by setting NOALLOWREFUSING = YES).

```
FORBIDALL    Compressed mail and file attaches
FORBIDFREQ   File Requests
FORBIDALL    Anything (hold all traffic)
```

Example: FORBIDALL YES

1.75 FREQUEST

FREQUEST command-string

Config & Rexx

Sets the command to launch the filerequest-server. Like all external commands this one may have embedded~%-commands in the string.

Note that asynchronous execution is not allowed for this one, since right after the command returns the outbound dir will be rescanned for files to send to the remote system.

Example: "rx frequest.rexx %i %o %S"

1.76 GETCONFIG

GETCONFIG keyword

Rexx

With this ARExx-only command it is possible to get the current configuration for a certain keyword. So you can first get the current config, change it do some action and then set it back to the old value.

----- cut here -----

```
'GETCONFIG LOCATION'      /* get the current location */
config = RESULT
```

```
'LOCATION "My Place"'      /* set location to "My Place" */
```

```
'CALL 12345'              /* call out */
```

```
config                    /* set back the old config */
```

----- cut here -----

```
Examples: GETCONFIG AKAS
           GETCONFIG MRC_FAX
```

1.77 HYDRA

HYDRA

Rexx

This is the same as NOHYDRA = NO.

1.78 ICONIFY

ICONIFY

Rexx

This ARexx only command will iconify Gotcha if its window is open. I.e. if Workbench is loaded the window will be closed and an application icon will be put on the Workbench screen. If you want to use your own icon place a file called "def_Gotcha.info" in your mail: drawer.

1.79 IMMEDIATE

IMMEDIATE yes/no

Config & Rexx

If IMMEDIATE = YES is specified Gotcha will go directly and immediately to the session handshake. If in answer mode Gotcha will act as if it had detected an incoming call.

Defaults to NO.

Example: IMMEDIATE YES

1.80 INBOUND

INBOUND

Rexx

This is the same as INDIR.

1.81 INDIR

INDIR directory

Config & Rexx

This should be set to your inbound directory. Incoming mail and files will be stored here. Also temporary files will be created here during file transfers.

Example: INDIR "mail:inbound"

1.82 KEEP

KEEPALL yes/no Config & Rexx
KEEPXMAIL yes/no

This if set to YES this keywords disallow the sending of certain files during sessions.

KEEPALL surpresses the sending of any files.
KEEPXMAIL will surpress the sending of compressed mail files.

Example: KEEPALL YES

1.83 KIWIPULSE

KIWIPULSE yes/no Config & Rexx

Applies a special number translation for Kiwi people.

Example: KIWIPULSE YES

1.84 LISTEN

LISTEN Rexx

LISTEN is a synonym for NOUNLISTEN and NOLISTEN is a synonym for UNLISTEN.

1.85 LOADINGMESSAGE

LOADINGMESSAGE string Config & Rexx

This text will be displayed to human usern when the BBS is loaded.

Example: LOADINGMESSAGE "Please wait spawning the BBS..."

1.86 LOCATION

LOCATION string Config & Rexx

Sets the location of your node.

Example: LOCATION "Vienna; Austria"

1.87 LOG

LOG <level><text> Rexx

This Rexx command will put out a text with the given loglevel to the logwindow and logfile.

Example: LOG "1 This is a test"

1.88 LOGFILE

LOGFILE filename Config & Rexx

Sets the name of the logfile. Everything you see in Gotchas main window will also be saved to the logfile. The amount of information displayed/saved depends on the LOGLEVEL setting.

Example: LOGFILE "mail:Gotcha.log"

1.89 LOGLEVEL

LOGLEVEL level Config & Rexx

Sets the amount of logging information written to the logfile. There are 6 levels:

1 SILENCE	... nearly no output
2 MINIMUM	... some more infos
3 NORMAL	... this is quite nice
4 VERBOSSE	... nearly everything (IMHO the best)
5 MAXIMUM	... all there is
6 DEBUG	... even more (but nothing really needed)

Example: LOGLEVEL 5

1.90 LOGLINES

LOGLINES <num lines> Config

Specifies how many lines will be buffered in the scrollbar buffer of the main window. Minimum is 100, maximum 1000.

Example: LOGLINES 200

1.91 MINBAUD

MINBAUD baudrate

Config & REXX

Minimum baudrate to establish a connection at. If the connection baudrate is lower than specified with this keyword Gotcha will hang up immediately (on both incoming and outgoing calls).

Example: MINBAUD 2400

1.92 MODEMANSWER

MODEMANSWER string

Config & REXX

This string will be sent to the modem when the number of RINGS specified with the RINGS keyword have been received and Gotcha wants to answer the phone. The string can have all the special~modem~characters in it.

Example: MODEMANSWER "ATA|"

1.93 MODEMDIALPOST

MODEMDIALPOST string

REXX

Same as DIALSUFFIX.

1.94 MODEMDIALPRE

MODEMDIALPRE string

REXX

Same as DIALPREFIX.

1.95 MODEMHANG

MODEMHANG string

Config & REXX

This string will be sent to the modem whenever Gotcha wants to hang up. Using "~~~+++~~~ATH0|" is not recommended, you should configure your modem in way that it hangs up as soon as DTR is lowered. The recommended string for this setting is "^". The string can have special~modem~characters in it.

Example: MODEMHANG "^"

1.96 MODEMHANGUP

MODEMHANGUP string

Rexx

This is the same as MODEMHANG.

1.97 MODEMINIT

MODEMINIT string

Config & Rexx

This string will be sent to the modem whenever Gotcha wants to init the modem. The string may contain all the special~modem~characters. Gotcha expects every char to be echoed by the modem and after every carriage-return "|" a "OK" from the modem (if there was something sent to the modem before).

Example: MODEMINIT "AT~Z|"

1.98 MODEMRESETINTERVAL

MODEMRESETINTERVAL minutes

Config & Rexx

In answer mode the MODEMINIT string will be sent to the modem every so many minutes.

Example: MODEMRESETINTERVAL 15

1.99 NAME

NAME nodename

Rexx

This is the same as the BBS keyword.

1.100 MODEMRESULTCODES

MRC_BUSY string

Config & Rexx

MRC_CARRIER string

MRC_CONNECT string

MRC_FAX string

MRC_NOANSWER string

MRC_NOCARRIER string

MRC_NODIALTONE string

MRC_RING string

MRC_RINGING string

MRC_VOICE string

With these keywords you can configure Gotcha to what strings your

modem sends on various occasions.

MRC_BUSY	... Reported when the line is busy
MRC_CARRIER	... Not supported by all modems. The "modem to modem" speed
MRC_CONNECT	... The connection speed or (if possible) the computer to modem speed.
MRC_FAX	... Reported for Fax connects
MRC_NOANSWER	... Sometimes reported instead of NO CARRIER
MRC_NOCARRIER	... When carrier detect has failed
MRC_NODIALTONE	... If there is no dialtone
MRC_RING	... Rings on incoming calls
MRC_RINGING	... Rings on outgoing calls
MRC_VOICE	... For Voice connects

If the CARRIER result code is supported by your modem you should configure the modem in a way that the baudrate reported with this keyword reflects the "modem to modem" rate and the rate reported with CONNECT the "computer to modem" speed. This is e.g. quite nice for Hydra transfers since Hydra will use both values to calculate maximum block sizes and other things. However, if this is not possible it does not harm too much, actually on higher speeds (9600, 14400 for sure) the maximum block size will be at its limit anyway. If CARRIER is not supported the "modem to modem" speed should be reported with the CONNECT-message (as is usually the case) and the MRC_CARRIER keyword can be omitted.

Examples:

MRC_BUSY	"BUSY"
MRC_CARRIER	"CARRIER"
MRC_CONNECT	"CONNECT"
MRC_FAX	"FAX"
MRC_NOANSWER	"NO ANSWER"
MRC_NOCARRIER	"NO CARRIER"
MRC_NODIALTONE	"NO DIAL TONE"
MRC_RING	"RING"
MRC_RINGING	"RINGING"
MRC_VOICE	"VOICE"

These settings are also the default.

1.101 NLDIR

NLDIR directory

Config & REXX

If you want to use a nodelist with Gotcha set this to the directory where the compiled nodlist is stored. On more information on how to use nodelists with Gotcha read the GList documentation.

Example: NLDIR "mail:nodelist"

1.102 NOALLOWREFUSING

NOALLOWREFUSING yes/no

Config & REXX

This selects if the wishes of the other side in an EMSI handshake should be respected or not (see FORBID).

Example: NOALLOWREFUSING NO

1.103 NODELIST

NODELIST directory

REXX

This is the same as NLDIR.

1.104 NODE

NODE address

REXX

This is the same as ADDRESS.

1.105 NODIRECTZAP

NODIRECTZAP yes/no

Config & REXX

This selects if the DirectZap transfer protocol should be used for transfers.

Example: NODIRECTAZP YES

1.106 NOEMSI

NOEMSI yes/no

Config & REXX

This selects if an EMSI handshake should be (at least) attempted. EMSI is the most up to date handshake available in FidoNet sessions. If turned off a lot of features can not be used. e.g it's only possible in EMSI handshakes to exchange AKAs.

Example: NOEMSI NO

1.107 NOESCAPERESETSMODEM

NOESCAPERESETSMODEM

REXX

This is the same as ESCAPERESETSMODEM NO.

1.108 NOFORBID

NOFORBIDALL
NOFORBIDFREQ
NOFORBIDXMAIL

Rexx

This is the same as FORBID<what> NO.

1.109 NOHYDRA

NOHYDRA yes/no

Config & Rexx

This can be used to turn off Hydra transfers.

Example: NOHYDRA YES

1.110 NOKEEP

NOKEEPALL
NOKEEPXMAIL

Rexx

This is the same as KEEP<what> NO.

1.111 NOKIWIPULSE

NOKIWIPULSE

Rexx

This is the same as KIWIPULSE NO.

1.112 NOLISTEN

NOLISTEN

Rexx

This is the same as UNLISTEN.

1.113 NONEMESSAGE

NONEMESSAGE text

Config & Rexx

This message will be displayed whenever an user calls and the BBSMODE is set to NONE. This can be overridden with the BBSMESSAGE keyword.

Example: NONEMESSAGE "This is a mail only system -- please hang up"

1.114 NORETRYHANDSHAKE

NORETRYHANDSHAKE yes/no

Config & Rexx

Specifies whether Gotcha will retry to call this number if the handshake fails for some reason. Usually a number will only be removed from the dialing list if either the modem responded with something else than BUSY when dialing the number or the session was successful. It may however happen that the remote system has due to an error set up a wrong password for you. In this case you would be disconnected each time in the session handshake to prevent additional retries you can set this keyword to YES and whenever a handshake fails (for whatever reason) the number will be removed from the dialing list.

Example: NORETRYHANDSHAKE YES

1.115 NOSTRIPDASHES

NOSTRIPDASHES

Rexx

This is the same as STRIPDASHES NO.

1.116 NOSWEPULSE

NOSWEPULSE

Rexx

This is the same as SWEPULSE NO.

1.117 NOUNLISTEN

NOUNLISTEN

Rexx

This is the same as LISTEN.

1.118 NOWAZOO

NOWAZOO yes/no

Config & Rexx

Can be used to turn off WaZoo handshakes. Gotcha first tries to establish an EMSI handshake if this fails or Gotcha detects that the other side attempts a WaZoo handshake Gotcha will fall back to WaZoo. If you want your AKAs to be transferred always you should turn WaZoo off, so if an EMSI handshake fails no files at all will be sent but (if set) another attempt to do a EMSI handshake will be made at the next retry.

Example: NOWAZOO YES.

1.119 NOXMAIL

NOXMAIL yes/no

Config & Rexx

Enables/disables the setting of the XMA (compressed mail) flag for EMSI handshakes. This tells the other system if compressed mail should (not) be sent.

Example: NOXMAIL NO

1.120 NOZEDZAP

NOZEDZAP yes/no

Config & Rexx

Used to turn on/off the usage of the ZedZap protocol for file transfers.

Example: NOZEDZAP NO

1.121 NOZEDZIP

NOZEDZIP yes/no

Config & Rexx

Used to turn on/off ZedZap protocol in transfers.

Example: NOZEDZIP NO

1.122 ONCLOSEWINDOW

ONCLOSEWINDOW <HIDE|ICONIFY|QUIT>

Config & Rexx

With this keyword you can select what Gotcha will do if you press the window close gadget.

HIDE ... This is the default. Gotcha will hide.

ICONIFY ... Gotcha will iconify if Workbench is open.

QUIT ... Gotcha will quit.

Example: ONCLOSEWINDOW = QUIT

1.123 OPENWINDOW

OPENWINDOW

Rexx

If this command is received and the window is closed it will be opened. Gotcha sends this to its own Rexx-Port whenever the commodities hotkey is pressed.

Example: OPENWINDOW

1.124 OUTBOUND

OUTBOUND directory

Rexx

This is the same as OUTDIR.

1.125 OUTDIR

OUTDIR directory

Config & Rexx

This sets the directory where Gotcha will look for outgoing files.

Example: OUTDIR "mail:outbound"

1.126 PASSWORD

PASSWORD string

Config & Rexx

Specifies the password to be used in sessions.
If nodelist support is enabled the password will be fetched from there.
Note that if Gotcha is in answer mode and PASSWORD is set, it will be used whenever a node calls you have no password set up for via GList.
If PASSWORD is not set Gotcha will not care about passwords for incoming calls unless you have set up a password via GList.

Example: PASSWORD "secret"

1.127 PHONENUMBER

PHONENUMBER string

Config & Rexx

Sets your own phonenumber. If you don't want it to be sent out in sessions set this to "-Unpublished-".

Example: PHONENUMBER "43-456-7891234"

1.128 PICKUP

PICKUP < ALL | PRIMARY | NONE >

Config & Rexx

Select what mail and files to receive from the other system. This will only work with EMSI handshake sessions.

ALL ... everything there is
PRIMARY ... only files for your primary address not for AKAs
NONE ... nothing at all

Examples: PICKUP ALL
 PICKUP PRIMARY
 PICKUP NONE

1.129 QUIET

QUIET yes/no

Config

If set to YES Gotcha will not open a window or a screen but work in the background. The window can be opened with the OPENWINDOW Rexx command or by pressing the commodity hotkey.

Example: QUIET NO

1.130 QUIT

QUIT

Rexx

If Gotcha receives a QUIT Rexx command it will (guess what) quit if in answer mode. This is the same as pressing Alt-Q or Alt-X. If there is a session taking place it will quit as soon as it has ended.

1.131 REDIALDELAY

REDIALDELAY secs

Config & Rexx

Specifies the number of seconds to wait until the next dialattempt is made. If in answer mode Gotcha will receive calls while waiting for the delaytime to expire.

Example: REDIALDELAY 60

1.132 REMMONITOR

REMMONITOR ARexx-port

Rexx

This removes a port added with ADDMONITOR from the list.

Example: REMMONITOR "GOTCHA_MONITOR"

1.133 RESET

RESET

Rexx

This will reset the modem. The MODEMHANG and MODEMINIT strings will be sent to the modem and the counter for received RINGs will be set back to 0.

1.134 RETRIES

RETRIES integer

Config & Rexx

This specifies the number of attempts made to get a successful session. A number will be removed from the callinglist if either the modem returns something else than BUSY after the dialstring has been sent or the session was successful.

Example: RETRIES 10

1.135 REXXNAME

REXXNAME string

Config

This specifies the name of the REXXport. Defaults to "GOTCHA". For more information on Gotchas ARexx port see the ARexx~chapter in the basics section.

Example: REXXNAME "GOTCHA"

1.136 RINGCOMMAND

RINGCOMMAND commandstring

Config & Rexx

If in answer mode this command with may contain embedded~%-commands will be executed whenever Gotcha receives a RING from the modem.

1.137 RINGINGS

RINGINGS integer

Config & REXX

This specifies the maximum number of RINGINGS allowed for outgoing calls. If this number is reached Gotcha will abort the try and act as if DIALTIMEOUT had expired.

Note: Not all modems support this.

Example: RINGINGS 10

1.138 RINGS

RINGS integer

Config & REXX

Specifies the number of rings needed until the MODEMANSWER string is sent the modem. This works only in answer mode of course.

Example: RINGS 1

1.139 RUN

RUN cli-command

REXX

This is the same as SPAWN except that a run is put in front of the command to execute. So SPAWN "run list" is the same as RUN "list".

1.140 SCLEAR

SCLEAR

REXX

This one will clear the serial receive buffer.

1.141 SCREENMODE

SCREENMODE <mode>

Config & REXX

Specifies the screenmode Gotcha should use.

WORKBENCH ... open on Workbench
CUSTOM ... open an own (public) screen
PUBLIC ... if exists open on the public screen specified with the
CONSCREEN keyword, else open an own screen

The resolution for screens Gotcha opens can be set with the DISPLAYID keyword.

Example: SCREENMODE WORKBENCH

1.142 SCREENTOBACK

SCREENTOBACK

Rexx

Will move the Gotcha screen behind all other screens.

1.143 SCREENTOFRONT

SCREENTOFRONT

Rexx

Will move the screen Gotcha is on to the front.

1.144 SERIALBAUD

SERIALBAUD integer

Config & Rexx

This is the baudrate to talk to the modem (open the serial device with).

Example: SERIALBAUD 38400

1.145 SERIALDEVICE

SERIALDEVICE devicename

Config & Rexx

If you want to use another device you can select it with this keyword.

Examples: SERIALDEVICE "v34serial.device"
SERIALDEVICE "serial.device"

1.146 SERIALFLAGS

SERIALFLAGS integer

Config & Rexx

Some 3rd party serial devices need to get some flags passed when the device is opened.

Example: SERIALFLAGS 0

1.147 SERIALUNIT

SERIALUNIT integer

Config & REXX

If your modem is connected to something else than unit 0 change this as desired.

Example: SERIALUNIT 1

1.148 SHOWREXX

SHOWREXX yes/no

Config & REXX

Turns the display of rexxmessages on/off. If turned on rexxmessages will be displayed in the window and written to the logfile.

Example: SHOWREXX YES

1.149 SLINE

SLINE

REXX

This REXX commands will wait for a <cr> terminated line from the modem and return it in the rexx variable.

Example: SLINE

1.150 SMODEM

SMODEM string

REXX

Sends the given string to the modem. The given string will be run through the modem command processor. This allows special characters like "|", "^" and "~" (see Modem~Strings).

Example: SMODEM "~AT~H1~|"

1.151 SPAWNMESSAGE

SPAWNMESSAGE text

Config & REXX

This message will be displayed to human callers if the BBSMODE is SPAWN or EXIT. This can be overridden with BBSMESSAGE.

Example: SPAWNMESSAGE "Please press ESC twice to enter to BBS"

1.152 SPAWN

SPAWN cli-command Rexx

If this command is received Gotcha will execute the given cli-command and wait until it returns.

Example: SPAWN "echo \"Spawned from GOTCHA!\""

1.153 STARTSESSION

STARTSESSION Rexx

When Gotcha receives this command it will start a session waiting for the bannerline from the remote system.

1.154 TIMEOUT

TIMEOUT seconds Rexx

This command sets the timeout for the SLINE and SWAIT Rexx commands in seconds.

Example: TIMEOUT 10

1.155 STRIPDASHES

STRIPDASHES yes/no Config & Rexx

If set to YES Gotcha will strip dashes from phonenumbers before they are sent to the modem. Some modems don't like dashes in phonenumbers.

Example: STRIPDASHES YES

1.156 SWAIT

SWAIT text Rexx

This Rexx only command will wait for the given text to appear at the serial port.

The timeout specified with TIMEOUT will be used, WARN (5) will be returned if the text was not received in the given time.

Example: SWAIT "connection ok"

1.157 SWEPUULSE

SWEPUULSE yes/no

Config & Rextt

If set to YES some special number translation for vanilla swedish modems will be made.

Example: SWEPUULSE NO

1.158 SWRITE

SWRITE string

Rextt

The given string will be sent to the modem without any changes.

Example: SWRITE "Let's go"

1.159 SYSOP

SYSOP name-string

Config & Rextt

Sets the name of the sysop of this node.

Example: SYSOP "John Doe"

1.160 TASKPRI

TASKPRI priority

Config & Rextt

Sets the priority of the Gotcha Task. Should only be changed to something else than 0 if there are problems.

Example: TASKPRI 1

1.161 TIMESTAMP

TIMESTAMP yes/no

Config & Rextt

If set to YES Gotcha will always use timestamps to convert 4D filenames such as 1.300.200.0.th0 to 2D filenames instead of net-node-2D filenames unless the remote system is a point or both systems are in the same net and have the same net number.

Example: TIMESTAMP NO

1.162 TURNAROUND

TURNAROUND command

Config & Rexx

This command with may include embedded~%-commands will be executed on session turnaround. This is when sides change, receiver becomes sender and the other was around. This command should not take too long to execute since timeouts may happen, the other side might hang up if this takes too long.

1.163 UNICONIFY

UNICONIFY

Rexx

This ARexx only command uniconifies Gotcha. I.e. removes the application icon and opens Gotchas window.

1.164 UNLISTEN

UNLISTEN

Rexx

UNLISTEN is a synonym for NOLISTEN. If UNLISTEN is received Gotcha will not send/read any bytes to/from the modem but wait silently until a LISTEN or NOUNLISTEN command is received. The serial device will not be closed.

1.165 USESYSFONT

USESYSFONT yes/no

Config & Rexx

If set to YES Gotcha will use the system screen font instead of the system default font for its screen. For windows proportional fonts will not be used.

Example: USESYSFONT NO

1.166 VERSION

VERSION

Rexx

Will return the version identifier. e.g "1.2".

1.167 VOICECOMMAND

VOICECOMMAND command-string

Config & Rexx

This command with may include embedded~%-commands will be executed whenever the string specified with the MRC_VOICE keyword is received from the modem on an incomming or outgoing call.

Example: VOICECOMMAND "VoiceAnswer ?incomming.txt#outgoing.txt?"

1.168 VOICESTRING

VOICESTRING string

Rexx

This is the same as MRC_VOICE.

1.169 WAZOO

WAZOO

Rexx

This is the same as NOWAZOO NO.

1.170 XMAIL

XMAIL

Rexx

This is the same as NOXMAIL NO.

1.171 ZADZAP

ZEDZAP

Rexx

This is the same as NOZEDZAP NO.

1.172 ZEDZIP

ZEDZIP

Rexx

This is the same as NOZEDZIP NO.

1.173 ZMHMESSAGE

ZMHMESSAGE text

Config & Rexx

This text will be displayed to human callers if the BBSMODE is set to ZMH. This can be overridden with BBSMESSAGE.

Example: ZMHMESSAGE "Zone Mail Hour -- please call later"

1.174 Index

Index of database Gotcha

Documents

Acknowledgements

ADDMONITOR

ADDRESS

AFTERSESSION

AKAS

ALLOWREFUSING

ANSWER

ARexx

AUTOSELECT

BACKGROUND

BANNER

Basic Information

BBS

BBSCHAR

BBSCOMMAND

BBSMESSAGE

BBSMODE

BOSSNUMBER

BUFFEREDLOG

Bug Reports

CALL

Call Options

CALLERID

CLOSEWINDOW

Commodity Info

Compressed Mail

Configuration Commands

CONSCREEN

Contact

CONWIN

COST

Custom Configurations

CX_POPKEY

CX_PRIORITY

Dial Scripts

DIALER

DIALPREFIX

DIALSUFFIX

DIALTIMEOUT

DIRECTZAP
DISPLAYID
Embedded Percent Commands
EMSI
ESCAPERESETSMODEM
Fax and Voice
FAXCOMMAND
FAXSTRING
Features
Filename Bumping
Filenames
FKEYs
Flow Files
FORBID
Forbid and Pickup
FREQUEST
GETCONFIG
Gotcha Echo
Gotcha! v1.1
HYDRA
ICONIFY
IMMEDIATE
INBOUND
Inbound File Tagging
INDIR
Installation
Introduction
KEEP
Keyboard
Keywords
KIWIPULSE
Legal Stuff
Limitations
LISTEN
LOADINGMESSAGE
LOCATION
LOG
LOGFILE
LOGLEVEL
LOGLINES
MINBAUD
Modem Strings
MODEMANSWER
MODEMDIALPOST
MODEMDIALPRE
MODEMHANG
MODEMHANGUP
MODEMINIT
MODEMRESETINTERVAL
MODEMRESULTCODES
Multiple Lines
NAME
NLDIR
NOALLOWREFUSING
NODE
Node Setup
Nodelist

NODELIST
NODIRECTZAP
NOEMSI
NOESCAPERESETSMODEM
NOFORBID
NOHYDRA
NOKEEP
NOKIWIPULSE
NOLISTEN
NONEMESSAGE
NORETRYHANDSHAKE
NOSTRIPDASHES
NOSWEPULSE
NOUNLISTEN
NOWAZOO
NOXMAIL
NOZEDZAP
NOZEDZIP
ONCLOSEWINDOW
OPENWINDOW
OUTBOUND
Outbound File Tagging
OUTDIR
OwnDevUnit Library
PASSWORD
PHONENUMBER
PICKUP
Point Setup
QUIET
QUIT
REDIALDELAY
Registration Form
REMMONITOR
Requirements
RESET
RETRIES
REXXNAME
RINGCOMMAND
RINGINGS
RINGS
RUN
SCLEAR
SCREENMODE
SCREENTOBACK
SCREENTOFRONT
SERIALBAUD
SERIALDEVICE
SERIALFLAGS
SERIALUNIT
SHOWREXX
SLINE
SMODEM
SPAWN
SPAWNMESSAGE
STARTSESSION
STIMEOUT
STRIPDASHES

SWAIT
SWEPULSE
SWRITE
SYSOP
TASKPRI
The Menu
The Modem
TIMESTAMP
TURNAROUND
UNICONIFY
UNLISTEN
USESYSFONT
VERSION
VOICECOMMAND
VOICESTRING
WAZOO
XMAIL
ZADZAP
ZEDZIP
ZMHMESSAGE

Buttons

Acknowledgements~~~~~
ADDMONITOR
ADDRESS
AFTERSESSION
AKAS
ALLOWREFUSING
ANSWER
ANSWER~~~YES
ARexx
ARexx~chapter
AUTOSELECT
BACKGROUND
BANNER
Basic~Information~~~~~
BBS
BBSCHAR
BBSCOMMAND
BBSMESSAGE
BBSMODE
BOSSNUMBER
BUFFEREDLOG
Bug~Reports~~~~~
CALL
Call~options
CALLERID
CLOSEWINDOW
Commodity~Infos
Compressed~Mail~files
Config~Commands
CONSCREEN
Contact~the~author~~~~~
CONWIN
COST
Custom~Configs

CX_POPKEY
CX_PRIORITY
Dial~Scripts
DIALER
DIALPREFIX
DIALSUFFIX
DIALTIMEOUT
DIRECTZAP
DISPLAYID
Embedded~%-commands
EMSI
ESCAPERESETSMODEM
extra~section
Fax~and~Voice
FAXCOMMAND
FAXSTRING
Features
Filename~bumping
Filenames
FKEY~and~SFKEY
FKEY1,~FKEY2,~etc
FKEYs
Flow~Files
FORBID
Forbid~and~Pickup
FORBID<what>
FORBIDALL
FORBIDFREQ
FORBIDXMAIL
FREQUENT
GETCONFIG
HYDRA
ICONIFY
IMMEDIATE
INBOUND
Inbound~file~tagging
INDIR
Installation
Introduction~~~~~
KEEP<what>
KEEPALL
KEEPXMAIL
keyword~index
keywords
Keywords~~~~~
KIWIPIULSE
Legal~stuff~~~~~
Limitations~~~~~
LISTEN
LOADINGMESSAGE
LOCATION
LOG
LOGFILE
LOGLEVEL
LOGLINES
MINBAUD
Modem~strings

MODEMANSWER
MODEMDIALPOST
MODEMDIALPRE
MODEMHANG
MODEMHANGUP
MODEMINIT
MODEMRESETINTERVAL
MRC_BUSY
MRC_CARRIER
MRC_CONNECT
MRC_FAX
MRC_FAX~and~MRC_VOICE
MRC_NOANSWER
MRC_NOCARRIER
MRC_NODIALTONE
MRC_RING
MRC_RINGING
MRC_VOICE
Multiple~Lines
NAME
NLDIR
NOALLOWREFUSING
NODE
Node~Setup
NODELIST
NODIRECTZAP
NOEMSI
NOESCAPERESETSMODEM
NOFORBIDALL
NOFORBIDFREQ
NOFORBIDXMAIL
NOHYDRA
NOKEEPALL
NOKEEPXMAIL
NOKIWIPULSE
NOLISTEN
NONEMESSAGE
NORETRYHANDSHAKE
NOSTRIPDASHES
NOSWEPULSE
NOUNLISTEN
NOWAZOO
NOXMAIL
NOZEDZAP
NOZEDZIP
ONCLOSEWINDOW
OPENWINDOW
OUTBOUND
Outbound~file~tagging
OUTDIR
OwnDevUnit~Library
PASSWORD
PHONENUMBER
PICKUP
Point~Setup
QUIET
QUIT

REDIALDELAY
Registration~Form~~~~~
REMMONITOR
Requirements~~~~~
RESET
RETRIES
RETRYDELAY
returncodes
REXXNAME
RING
RINGCOMMAND
RINGINGS
RINGS
RUN
SCLEAR
SCREENMODE
SCREENTOBACK
SCREENTOFRONT
SERIALBAUD
SERIALDEVICE
SERIALFLAGS
SERIALUNIT
SFKEY1,~SFKEY2,~etc
SFKEYs
SHOWREXX
SLINE
SMODEM
SPAWN
SPAWNMESSAGE
special~modem~characters
STARTSESSION
TIMEOUT
STRIPDASHES
SWAIT
SWEPULSE
SWEPULSE
SWRITE
SYSOP
TASKPRI
The~Keyboard
The~Menu
The~Modem
The~Nodelist
The~Support~Echo~~~~~
TIMESTAMP
TURNAROUND
UNICONIFY
UNLISTEN
USESYSFONT
VERSION
VOICECOMMAND
VOICESTRING
WAZOO
XMAIL
ZEDZAP
ZEDZIP
ZHMMESSAGE

ZMHMESSAGE