

QSilver.hyper

COLLABORATORS

	<i>TITLE :</i> QSilver.hyper	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		November 2, 2022
<i>SIGNATURE</i>		

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	QSilver.hyper	1
1.1	Quick Silver v2.00 - FidoNet File Request Server	1
1.2	Quick Silver - Introduction	1
1.3	Quick Silver - Feature Overview	2
1.4	Quick Silver - Installation Instructions	3
1.5	Quick Silver - Command Options	4
1.6	Quick Silver - Command Options - Basic Options	4
1.7	Quick Silver - Command Options - File Handling	5
1.8	Quick Silver - Command Options - Security Options	5
1.9	Quick Silver - Command Options - Request Limits	5
1.10	Quick Silver - Command Options - Paths	6
1.11	Quick Silver - Command Options - Support Files	6
1.12	Quick Silver - Command Options - Option Flags	7
1.13	Quick Silver - Command Options - Miscellaneous	7
1.14	Quick Silver - Command Options - Reports	8
1.15	Quick Silver - Command Options - SYSOP	8
1.16	Quick Silver - Command Options - NODE	9
1.17	Quick Silver - Command Options - NAME	9
1.18	Quick Silver - Command Options - PRIORITY	9
1.19	Quick Silver - Command Options - NODELIST	9
1.20	Quick Silver - Command Options - LOGFILE	10
1.21	Quick Silver - Command Options - INBOUND	10
1.22	Quick Silver - Command Options - ACCTPATH	10
1.23	Quick Silver - Command Options - SYSACCT	11
1.24	Quick Silver - Command Options - DATABASE	11
1.25	Quick Silver - Command Options - SCREEN	11
1.26	Quick Silver - Command Options - SCREENPOS	12
1.27	Quick Silver - Command Options - HEADER	12
1.28	Quick Silver - Command Options - FIRSTCALL	12
1.29	Quick Silver - Command Options - MAXBYTES	12

1.30 Quick Silver - Command Options - MAXFILES	13
1.31 Quick Silver - Command Options - MAXMATCH	13
1.32 Quick Silver - Command Options - BAUDMAX	13
1.33 Quick Silver - Command Options - TIMEMAX	14
1.34 Quick Silver - Command Options - MAGIC	14
1.35 Quick Silver - Command Options - DEFCOMMENT	14
1.36 Quick Silver - Command Options - CNETDESC	15
1.37 Quick Silver - Command Options - DLGDESC	15
1.38 Quick Silver - Command Options - EXCELDESC	15
1.39 Quick Silver - Command Options - DEFAULTSEND	16
1.40 Quick Silver - Command Options - NOMATCHFILE	16
1.41 Quick Silver - Command Options - DIRECTORY	16
1.42 Quick Silver - Command Options - POINTS	17
1.43 Quick Silver - Command Options - POINTMSG	18
1.44 Quick Silver - Command Options - FAILPOINT	18
1.45 Quick Silver - Command Options - UNLISTED	18
1.46 Quick Silver - Command Options - UNLISTEDMSG	19
1.47 Quick Silver - Command Options - FAILUNLISTED	19
1.48 Quick Silver - Command Options - EXCLUDE	20
1.49 Quick Silver - Command Options - EXCLUDEMSG	20
1.50 Quick Silver - Command Options - FAILEXCLUDE	21
1.51 Quick Silver - Command Options - SECURE	21
1.52 Quick Silver - Command Options - SECURITYMSG	21
1.53 Quick Silver - Command Options - FAILSECURE	22
1.54 Quick Silver - Command Options - LOCALMSG	22
1.55 Quick Silver - Command Options - IGNORE	22
1.56 Quick Silver - Command Options - ALLOWNODE	23
1.57 Quick Silver - Command Options - DATABASE	24
1.58 Quick Silver - Command Options - OFFLINEMSG	24
1.59 Quick Silver - Command Options - UPDATEDL	24
1.60 Quick Silver - Registration Information	25
1.61 Quick Silver - Testing your Setup	25
1.62 Quick Silver - Revision History	25
1.63 v1.00	27
1.64 v1.01	27
1.65 v1.02	27
1.66 v1.03	27
1.67 v1.04	28
1.68 v1.05	28

1.69 v1.06	28
1.70 v1.07	28
1.71 v1.08	29
1.72 v1.09	29
1.73 v1.10	29
1.74 v1.11	29
1.75 v1.12	30
1.76 v1.13	30
1.77 v1.14	30
1.78 v1.15	30
1.79 v1.16	31
1.80 v1.17	31
1.81 v1.18	31
1.82 v1.19	31
1.83 v1.20	32
1.84 v1.21	32
1.85 v1.22	32
1.86 v1.23	33
1.87 v1.24	33
1.88 v1.25	33
1.89 v1.26	33
1.90 v1.27	34
1.91 v1.28	34
1.92 v1.29	34
1.93 v1.30	34
1.94 v1.31	35
1.95 v1.32	35
1.96 v1.33	36
1.97 v1.34	36
1.98 v1.35	36
1.99 v1.36	36
1.100v1.37	37
1.101v1.38	37
1.102v2.00	38
1.103Quick Silver - Tips & Tricks	38
1.104Quick Silver - Tips & Tricks - Access Levels	38
1.105Quick Silver - Tips & Tricks - Security	39
1.106Technical Support	40

Chapter 1

QSilver.hyper

1.1 Quick Silver v2.00 - FidoNet File Request Server

Quick Silver
The ARexx File Request Server
Copyright © 1992-1994 by BitPlane Enterprises

Introduction

Features

Installation

Command Options

Registration

Tech Support

Testing

Tips & Tricks

History

1.2 Quick Silver - Introduction

Welcome to Quick Silver, a fully functional FidoNet file request server written in ARexx and compiled with the Rexx Plus compiler by Dineen Edwards Group. This project began as an alternative to the few servers available to me when I first joined FidoNet. I've tried to look at as many servers as possible to provide the best selection of options available, while still maintaining the fast speed required for serving files. I hope you enjoy Quick Silver as much as I do.

1.3 Quick Silver - Feature Overview

Here is a basic look at the types of features that are available to you with Quick Silver:

SECURITY

- o Secure mode to limit requesting nodes to only those that you specifically give proper access.
- o Access levels can be assigned to calling nodes, file directories and Magic file names. Multiple access levels can be given to each as required.
- o Passwords can be assigned to file directories and Magic filenames.
- o The ability to specifically lock out a node or point from requesting files from your system.
- o Ability to lock out all points and/or unlisted systems from requesting files from your system.
- o Filter to check for file extensions that are not to be sent by Quick Silver to the requesting system. (.info .desc etc.)

ACCOUNTING

- o Full accounting and logging of all requests made to your system, tracking date of first request, latest request, number of requests made, number of files sent, number of bytes sent.

REPLY MESSAGES

- o FTS-001 compliant netmail reply messages are generated for the calling system to inform them of the results of their requests, as well as any security violations encountered.

ADVANCED REQUEST FEATURES

- o Update requests are supported for both standard UNIX and the new HRF (Human Readable Format) requests.
- o Wildcard pattern matching supported with the DOS standard ? and * characters, as well as the AmigaDos #? combination.

Example: Files.* Matches multiple characters
 Blah#?.LHA

 Foo?Bar.ZIP Matches single character

- o Database file for fast file location and security checking.

1.4 Quick Silver - Installation Instructions

The setup for Quick Silver is fairly straight forward, and should only take a few moments to accomplish. ←

- o Copy QSilver to your C: directory
- o Make a directory for Quick Silver's data and configuration files.
Example: "Mail:QSilver"
- o Copy "QSilver.CFG" to this directory
- o Make a subdirectory for Quick Silver's accounting files. QSilver will attempt to create this directory the first time it is run if the directory does not exist. If this fails, then QSilver will default to no accounting.

Example: "Mail:QSilver/Accounts"

- o If you wish to use the custom message files, then edit the included "QSilver.Header", "FirstTime.TXT", "Excluded.TXT", "Security.TXT", "Point.TXT" and "Unlisted.TXT" and place them in the Quick Silver data directory.
- o Tell your mailer/frontend how to call Quick Silver. You need to pass the following to Quick Silver:

```
[Input File] [Output File] [Configuration File] [Caller's Node Numbers]
[Caller's Baud Rate] [Caller's Name]
```

TrapDoor users:

```
Usage: FREQUEST "C:QSilver %i %o {Path to QSilver.CFG} \"%n %A\" %B %S"
```

```
Example: FREQUEST "C:QSilver %i %o QSilver:QSilver.CFG \"%n %A\" %B %S"
```

- o Edit the QSilver.CFG file to reflect your system's information
- o Copy the RPStart file to your WBStartup: directory for WB 2.0+
- o Copy the RPHi and RPInstall files to your C: directory
- o Copy the rexxplslib.library and rexxarplib.library to your Libs: directory.
- o Reboot your system so that the RexxPlus environment will be started for Quick Silver to run.
- o If you wish to use the optional DataBase lookup function, run the

DATABASE

utility prior to running Quick Silver.

1.5 Quick Silver - Command Options

List of commands available for use in Quick Silver's ↔
configuration file:

Basic Options

File Handling

Security

Request Limits

Paths

Support Files

Option Flags

Miscellaneous

Reports

1.6 Quick Silver - Command Options - Basic Options

Basic Options

Name

BBS Name

Node

FidoNet Node Number

Sysop

Sysop's Name

Priority

Task Priority

DefComment

Default File Comment

Options

1.7 Quick Silver - Command Options - File Handling

File Handling

Directory
File directory

Magic
Magic filename

Ignore
Ignored file extensions

Options

1.8 Quick Silver - Command Options - Security Options

Security

Secure
Secure mode

AllowNode
Allow nodes w/ access levels and byte/file limits

Exclude
Exclude nodes

Points
Allow/Disallow points request access

Unlisted
Allow/Disallow unlisted systems request access

Options

1.9 Quick Silver - Command Options - Request Limits

Request Limits

MaxBytes
Maximum bytes per request

MaxFiles
Maximum files per request

MaxMatch
Maximum wildcard matches per file per request

BaudMax
Maximum request limits per baud rate

TimeMax
Maximum request limits per time period

Options

1.10 Quick Silver - Command Options - Paths

Paths

AcctPath
Path to accounting files

Database
Path to Database file

Inbound
Path to inbound mail directory

Nodelist
Path to nodelist directory

LogFile
Path to Quick Silver's log file

Options

1.11 Quick Silver - Command Options - Support Files

Support Files

Text files sent to requesting node when the following conditions are met:

FailExclude
Excluded node

FailPoint
Point when NOPOINTS is set

FailSecure

Node not listed with ALLOWNODE when SECURE is set

FailUnlisted

Unlisted nodes when NOUNLISTED is set

FirstCall

Node on their first request

Header

Header for netmail reply message

DefaultSend

File sent with all good requests

NoMatchFile

File sent with all failed requests

OfflineMsg

File is found in DataBase but not online (CDRoms)

Options

1.12 Quick Silver - Command Options - Option Flags

Option Flags

CNetDesc

Use CNet Pro BBS internal file descriptions

DLGDesc

Use DLG BB/OS internal file descriptions

EXCELDesc

Use Excelsior! BBS internal file descriptions

SysAcct

Keep track of system accounting totals

Options

1.13 Quick Silver - Command Options - Miscellaneous

Miscellaneous

Screen

Open status window on public screen

ScreenPos
Location of status window on public screen

UpdateDL
External program to update DL counters

Options

1.14 Quick Silver - Command Options - Reports

Reports

Generate a local message when the following conditions are met:

ExcludeMsg
Excluded node makes a request

LocalMsg
Successful file request

PointMsg
Point requests when NOPOINTS is set

SecurityMsg
Node fails security check when SECURE is set

UnlistedMsg
Unlisted system requests when NOUNLISTED is set

Options

1.15 Quick Silver - Command Options - SYSOP

SYSOP

SYSOP defines the name that Quick Silver should send any
REPORTS

reports to. This can be defined as your name, handle, ↔
or perhaps
a co-sysop that you may assign to handle file requests.

Usage: SYSOP [Sysop Name]

Example: SYSOP Roger Clark

1.16 Quick Silver - Command Options - NODE

NODE

Your 4-Dimensional FidoNet Node Number. Used to generate a proper netmail header for the reply message to the calling system. Enter the complete address to include Zone:Net/Node.Point

Usage: NODE [Node number]

Example: NODE 1:382/105.0

1.17 Quick Silver - Command Options - NAME

NAME

Your BBS name. This is used in the reply message generated to the calling node to identify your system.

Usage: NAME [BBS Name]

Example: NAME The Hotel California

1.18 Quick Silver - Command Options - PRIORITY

PRIORITY

Task priority you wish Quick Silver to run at. Default is 4. Running Quick Silver at a priority greater than 0 is recommended to allow a faster response when handling a request.

Usage: PRIORITY [Task Priority]

Example: PRIORITY 4

1.19 Quick Silver - Command Options - NODELIST

NODELIST

Path to your Nodelist Files. Required for nodelist support when using the

(NO)UNLISTED

keywords. Also will include data from the nodelist in your Quick Silver log file. Access to the nodelist is via the traplist.library by Martin J Laubach. This option defaults to the directory assigned to "Nodelist:"

Usage: NODELIST [Pathname]

Example: NODELIST DH0:Mail/Nodelist

1.20 Quick Silver - Command Options - LOGFILE

LOGFILE

Path to Quick Silver's log file. Omitting this keyword will tell Quick Silver not to keep a log of the file request results for your personal records.

Usage: LOGFILE [Pathname]

Example: LOGFILE DH0:Mail/QSilver/QSilver.Log

1.21 Quick Silver - Command Options - INBOUND

INBOUND

Path that your mail processor uses for inbound files. This is required if you specify any of the

LOCALMSG

,

SECURITYMSG

,

POINTMSG

,

UNLISTEDMSG

, or

EXCLUDEMSG

keyword in your config file.

The path defaults to "Inbound:".

Usage: INBOUND [Pathname]

Example: INBOUND Mail:Inbound

1.22 Quick Silver - Command Options - ACCTPATH

ACCTPATH

Path for Quick Silver's accounting files. Omitting this keyword will tell Quick Silver not to keep accounting data for calling systems. Quick Silver will attempt to create this accounting directory for you. If this fails, then Quick Silver will default to no accounting.

Usage: ACCTPATH [Pathname]

Example: ACCTPATH DH0:Mail/QSilver/Accounts

1.23 Quick Silver - Command Options - SYSACCT

SYSACCT

Tells Quick Silver to keep track of the system request totals. This is only available if

ACCTPATH

is defined for accounting purposes.

Usage: SYSACCT

Example: SYSACCT

1.24 Quick Silver - Command Options - DATABASE

DATABASE

Path to the optional DataBase file for Quick Silver. This file will speedup file location on most systems. By ommiting this keyword, Quick Silver will look in each directory until a match is found. You must use the

DATABASE UTILITY

to create a master

database file for this option to function.

Usage: DATABASE [Pathname]

Example: DATABASE DH0:Mail/QSilver/QSilver.DBase

1.25 Quick Silver - Command Options - SCREEN

SCREEN

Public Screen name to open status window on. This can only be used with WorkBench 2.0+. Please note that screen names are CASE SENSITIVE. Default is to not open a status window.

~~~~~

Usage: SCREEN [Screen Name]

Example: SCREEN TrapDoor

---



## 1.26 Quick Silver - Command Options - SCREENPOS

### SCREENPOS

Defines the X and Y coordinates of the upper-left corner for the status window. Only effective if

SCREEN keyword is used. Default is 100 75.

Usage: SCREENPOS [X-Coordinate Y-Coordinate]

Example: SCREENPOS 100 75

## 1.27 Quick Silver - Command Options - HEADER

### HEADER

Path and name of Header file for Quick Silver's report file. Omitting this keyword will create a report file with a generic header.

Usage: HEADER [Path to file]

Example: HEADER DH0:Mail/QSilver/QSilver.Header

## 1.28 Quick Silver - Command Options - FIRSTCALL

### FIRSTCALL

Path and name of the data file to be sent to first time callers. Omitting this keyword will tell Quick Silver not to send a file to first time callers.

Usage: FIRSTCALL [Path to file]

Example: FIRSTCALL DH0:Mail/QSilver/FirstTime.TXT

## 1.29 Quick Silver - Command Options - MAXBYTES

### MAXBYTES

Maximum number of bytes to be requested in one session. If the current file is large enough to exceed this limit, Quick Silver will skip this file and go on to the next, and an error message will be added to the report file to inform the caller of the problem. It is therefore advisable to set this value larger than the biggest file that is available for requesting. If this

---

keyword is omitted or set to zero, then no limit is imposed.

Usage: MAXBYTES [Number of bytes]

Example: MAXBYTES 750000

### 1.30 Quick Silver - Command Options - MAXFILES

#### MAXFILES

Maximum number of files to be requested in one session. Quick Silver will simply stop processing the request file when it reaches the limit set with this keyword. A message will be added to the report file telling the caller that the maximum file per call limit was exceeded. If this keyword is omitted or set to zero, then no limit is imposed.

Usage: MAXFILES [Number of files]

Example: MAXFILES 10

### 1.31 Quick Silver - Command Options - MAXMATCH

#### MAXMATCH

Maximum files to match with a single wildcard. Quick Silver will only send as many matches to a wildcard as you specify. If this keyword is omitted or set to zero, then all files that match the requested wildcard will be sent. NOTE: \*.\* is not accepted as a valid wildcard, nor is any other \*. wildcard. ie: \*.LHA would be rejected as a bad wildcard pattern. The AmigaDos pattern #? is equivalent to the MSDos \* character.

Usage: MAXMATCH [Number of matched files to send]

Example: MAXMATCH 5

### 1.32 Quick Silver - Command Options - BAUDMAX

#### BAUDMAX (Registered Mode Only)

This option allows you to set the max limits of files, bytes, and wildcard matches per baudrate of the caller. Quick Silver will set the limits to the baudrate that is equal to or higher than the caller's baudrate. If more than one BAUDMAX baudrate is higher than the caller's actual baudrate, then the limit that is closest to the caller's baud rate will be used. You may set as many BAUDMAX limits as you see necessary. If the two limits in the example

below were in your config file and a 9600 baud caller sent a file request, the limits for the 14400 baudrate would be used. If a caller tried a file request at 1200 baud then the limits for 2400 would be used instead.

Usage: BAUDMAX [Baudrate] FILES [number] BYTES [number] MATCH [bytes]

Example: BAUDMAX 14400 FILES 5 BYTES 1024000 MATCH 3  
 BAUDMAX 2400 FILES 2 BYTES 102400 MATCH 1

### 1.33 Quick Silver - Command Options - TIMEMAX

TIMEMAX (Registered Mode Only)

This option allows you to set the max limits of files, bytes, and wildcard matches per time period. Quick Silver will set these limits when a file request is received between the hours listed. Requests received outside of hours specified in the TIMEMAX option will receive the default limits.

Usage: TIMEMAX [Start] [Stop] FILES [number] BYTES [number] MATCH [bytes]

Example: TIMEMAX 04:00 10:00 FILES 5 BYTES 1024000 MATCH 3  
 TIMEMAX 22:00 04:00 FILES 2 BYTES 102400 MATCH 1

### 1.34 Quick Silver - Command Options - MAGIC

MAGIC

Defines a standard name for a file to be requested and may be followed by the PASSWORD keyword. Optional wildcard "\*" in the filepath will allow the most recent file that matches the pattern to be sent. Great for NODEDIFF's and other recurring files. Also supports access levels for allowing the same magic filename to send different files to different sites.

Usage: MAGIC [Magic Filename] [Path] {PASSWORD} {file password} {LEVEL}  
 {access level}

Example: MAGIC Files DH0:BBS/Files.LHA  
 MAGIC QSilver DH0:Beta/QSilver.LHA PASSWORD Secret  
 MAGIC AR.LHA DH0:ADSText/AR\*.LHA  
  
 MAGIC Special DH0:Files/Special/GoodStuff.LHA LEVEL 1  
 MAGIC Special DH0:Files/Special/GreatStuff.LHA LEVEL 2  
 MAGIC Special DH0:Files/Special/AwesomeStuff.LHA LEVEL 3-10

### 1.35 Quick Silver - Command Options - DEFCOMMENT

## DEFCOMMENT

Defines the comment that Quick Silver should use in its reply message if the filenote for a requested file is not set.

Usage: DEFCOMMENT [Comment]

Example: DEFCOMMENT No File Description Available

## 1.36 Quick Silver - Command Options - CNETDESC

### CNETDESC

Tells Quick Silver to try to use the internal CNet comments rather than the AmigaDos FileComment.

Usage: CNETDESC

Example: NOCNETDESC

## 1.37 Quick Silver - Command Options - DLGDESC

### DLGDESC

Tells Quick Silver to try to use the internal DLG comments rather than the AmigaDos FileComment.

Usage: DLGDESC

Example: NODLGDESC

## 1.38 Quick Silver - Command Options - EXCELDESC

### EXCELDESC

Tells Quick Silver to try to use the internal Excelsior BBS comments rather than the AmigaDos FileComment. If the Excelsior file comment cannot be located, then the AmigaDos file comment will be used instead. Some Magic Filenames may not be in the BBS file areas, and therefore not have the appropriate comment files available.

Usage: EXCELDESC

Example: EXCELDESC

---

### 1.39 Quick Silver - Command Options - DEFAULTSEND

DEFAULTSEND

Defines a file that will be sent with every file request. This file is sent only if the file request resulted in files being successfully requested and sent. If this keyword is omitted then no file will be sent. See also

NOMATCHFILE

.

New option with version 1.37 allows a configurable amount of days between the sending of this file. Default is 1 day, which sends only once per day.

Usage: DEFAULTSEND [Path] {DAYS} {days between sending}

Example: DEFAULTSEND BBS:NewFiles.LHA  
 DEFAULTSEND BBS:NewFiles.LHA DAYS 7

### 1.40 Quick Silver - Command Options - NOMATCHFILE

NOMATCHFILE

Defines a file that Quick Silver will send to the caller if it was unable to match any of the requested files. See also

DEFAULTSEND

.

Usage: NOMATCHFILE [Path]

Example: NOMATCHFILE BBS:MagicNames.TXT

### 1.41 Quick Silver - Command Options - DIRECTORY

DIRECTORY

Defines the directory path(s) to be searched for requested files and may be followed by the PASSWORD keyword. If the PASSWORD keyword is specified, then all files in that directory will require the password if they are to be successfully requested. It is advised that you list the paths that receive the most requests at the beginning of the list, as Quick Silver searches for files in the order of paths specified. This is not the case when you specify a

DATABASE

pathname. The DATABASE will alphabetize the files and search for them in a different manner. ↔

As of version 1.30, you may now specify an access level for each DIRECTORY (See ALLOWNODE). The number of levels is unlimited, and is used as follows.

Example: LEVEL 1-5,8,11-20

Where access would be given to nodes with access levels of 1 through 5, 8, and 11 through 20. PASSWORD and LEVEL are not position sensitive so you can list the LEVEL before or after the PASSWORD if you are using one for the DIRECTORY. By omitting the LEVEL keyword, a level of 0 is assigned and will match any access level given to any node.

Usage: DIRECTORY [Path] {PASSWORD} {dir password} {LEVEL} {range}

Example: DIRECTORY DH0:Files/Excelsior  
 DIRECTORY DH0:Files/Excelsior-Betas PASSWORD Confidential  
 DIRECTORY DH0:Files/Updates LEVEL 1-5,10-15,20,25  
 DIRECTORY DH0:Private PASSWORD Secret LEVEL 50

Version 1.4 of the DATABASE utility will now look for one additional keyword in the DIRECTORY string. This keyword is SUBDIR, and will tell the DataBase utility to look in all subdirectories of the listed directory for additional files. As of version 1.4, this only looks at one level of subdirectories. The next version should have support for unlimited directories. Keep in mind, adding the SUBDIR option will not change the way QSilver looks at directories for files if the DataBase utility is not being used. Using this option will also not allow the current version of Mercury (1.14) to recognise the additional directories. This will also be updated in the very near future.

Usage: DIRECTORY [Path] SUBDIR

Example: DIRECTORY CD0:AmiNet/OS20/ SUBDIR  
 DIRECTORY CD0:AmiNet/OS30/ SUBDIR

## 1.42 Quick Silver - Command Options - POINTS

POINTS

Allow points to request files from your system.

Usage: POINTS

NOPOINTS

Not allow points to request files from your system. If a point attempts to request a file while NOPOINTS is set, they will receive a reply message indicating this. You can also tell QSilver to send a local message to you when this happens by using the POINTMSG option.

Usage: NOPOINTS

Example: POINTS

### 1.43 Quick Silver - Command Options - POINTMSG

POINTMSG

Tells Quick Silver to create a netmail in your Inbound directory that is a copy of the reply sent to the requesting system. This will generate a message when a point tries to request from your system when you have NOPOINTS specified.

This option requires the INBOUND keyword to specify the directory so your mail processor can properly import the message.

Usage: POINTMSG

Example: POINTMSG

### 1.44 Quick Silver - Command Options - FAILPOINT

FAILPOINT

Defines the textfile that QSilver sends in the reply message to a caller if their system is a point and you have set the NOPOINTS option in your configuration file.

Usage: FAILPOINT [Path to File]

Example: FAILPOINT QSilver:FailPoint.TXT

### 1.45 Quick Silver - Command Options - UNLISTED

#### UNLISTED

Allow systems not listed in the Nodelist to request files from your system. This option is forced to NOUNLISTED in

SECURE  
mode.

Usage: UNLISTED

#### NOUNLISTED

Not allow systems not listed in the Nodelist to request files from your system. This option is forced to NOUNLISTED in SECURE mode.

Usage: NOUNLISTED

Example: UNLISTED

### 1.46 Quick Silver - Command Options - UNLISTEDMSG

#### UNLISTEDMSG

Tells Quick Silver to create a netmail in your Inbound directory that is a copy of the reply sent to the requesting system. This will generate a message when a node that is not listed in the Nodelist tries to request from your system when you have

NOUNLISTED  
specified. This option requires the

#### INBOUND

keyword to specify the directory so your mail processor can properly import the message.

Usage: UNLISTEDMSG

Example: UNLISTEDMSG

### 1.47 Quick Silver - Command Options - FAILUNLISTED

#### FAILUNLISTED

Defines the textfile that QSilver sends in the reply message to a caller if their node is not listed in the nodelist. See also

#### UNLISTEDMSG

.

---



Usage: FAILUNLISTED [Path to File]

Exclude: FAILUNLISTED QSilver:FailUnlisted.TXT

## 1.48 Quick Silver - Command Options - EXCLUDE

### EXCLUDE

Tells Quick Silver not to accept File Requests from the node or point specified after the keyword. Specify one node per EXCLUDE statement. Specify the full 4D address (Z:N/n.p) when listing the node number. Works with or without

SECURE

mode active, and

will generate a reply message to the caller informing them that they have been locked out of your system. If

EXCLUDEMSG

is also

specified, then a message will also be sent to you to inform you that an excluded node has tried to make a request from your system and will list all files that they tried to request. You may use wildcarding to specify the nodes that you wish to exclude from requesting. An asterisk (\*) will match any portion of a node number, and a question mark (?) will match a specific digit within the nodenumber. You can not mix the \* and ? within a section of a node number (zone, net, node, or point). So the node specification of 1:?4\*/3.0 would not be valid.

Usage: EXCLUDE [Node number]

Example EXCLUDE 1:234/56.0  
 EXCLUDE 2:345/67.8  
 EXCLUDE 1:456/\*.0  
 EXCLUDE 1:234/1??.\*

## 1.49 Quick Silver - Command Options - EXCLUDEMSG

### EXCLUDEMSG

Tells Quick Silver to create a netmail in your Inbound directory that is a copy of the reply sent to the requesting system. This will generate a message when a node that has been flagged with an

EXCLUDE

statement attempts to request

files from your system. This option requires the

INBOUND

keyword to specify the directory so your mail processor can ↔

properly import the message.

Usage: EXCLUDEMSG

Example: EXCLUDEMSG

## 1.50 Quick Silver - Command Options - FAILEXCLUDE

FAILEXCLUDE

Defines the textfile that QSilver sends in the reply message to a caller if their node is listed in an

EXCLUDE

statement, and

thereby locked out from your system. See also

EXCLUDEMSG

.

Usage: FAILEXCLUDE [Path to File]

Exclude: FAILEXCLUDE QSilver:FailExclude.TXT

## 1.51 Quick Silver - Command Options - SECURE

SECURE

Quick Silver will run in a secure mode where it will only allow nodes listed in the configuration file (see

ALLOWNODE

) to have

access to request files.

Usage: SECURE

## 1.52 Quick Silver - Command Options - SECURITYMSG

SECURITYMSG

Tells Quick Silver to create a netmail in your Inbound directory that is a copy of the reply sent to the requesting system. This will generate a message when a node tries to request from your system when

SECURE

mode is effect and the

node has not been added in your config file with the

ALLOWNODE

statement. This option requires the

---

INBOUND  
keyword to  
specify the directory so your mail processor can properly  
import the message.

Usage: SECURITYMSG

Example: SECURITYMSG

## 1.53 Quick Silver - Command Options - FAILSECURE

FAILSECURE

Defines the textfile that QSilver sends in the reply message  
to a caller if their node is NOT listed in an  
ALLOWNODE  
statement. Only applicable in  
SECURE  
mode. See also  
SECUREMSG.

Usage: FAILSECURE [Path to File]

Example: FAILSECURE QSilver:FailSecure.TXT

## 1.54 Quick Silver - Command Options - LOCALMSG

LOCALMSG

Tells Quick Silver to create a netmail in your Inbound directory  
that is a copy of the reply sent to the requesting system. This  
will generate a message only for successful requests. This option  
requires the

INBOUND  
keyword to specify the directory so your  
mail processor can properly import the message.

Usage: LOCALMSG

Example: LOCALMSG

## 1.55 Quick Silver - Command Options - IGNORE

IGNORE

Tells Quick Silver to ignore any files with the indicated  
filetype. Specify one file extension per IGNORE statement.

---

This keeps a system from requesting other files used by your BBS that may be in your file areas Uses AmigaDos pattern matching for wildcard support.

Usage: IGNORE [Pattern]

Example: IGNORE .INFO  
 IGNORE .DESC  
 IGNORE .BBS  
 IGNORE \_#?

## 1.56 Quick Silver - Command Options - ALLOWNODE

### ALLOWNODE

Defines nodes that are allowed to request files when Quick Silver is running in secure mode (See

SECURE

). Specify one node per

ALLOWNODE statement. Specify the full 4D address (Z:N/n.p) when listing the node number. An optional wildcard character "\*" or "?" can be used in place of a portion of the node number to allow all nodes or points that match the pattern to request files. You can also include an access level to be given to the node or point. This access level will only allow the node or point to request files from the directories that have the same access level. You may use wildcarding to specify the nodes that you wish to exclude from requesting. An asterisk (\*) will match any portion of a node number, and a question mark (?) will match a specific digit within the nodenumber. You can not mix the \* and ? within a section of a node number (zone, net, node, or point). So the node specification of 1:?4\*/3.0 would not be valid.

New with version 1.35, you can now specify

MAXFILES

and

MAXBYTES

for any specific node or point. These limits are on a ↔ per call

basis.

Usage: ALLOWNODE [node] {LEVEL} {levels} {MAXFILES} {files} {MAXBYTES} {bytes}

Example: ALLOWNODE 1:382/105.0 LEVEL 5-20  
 ALLOWNODE 2:254/521.0 LEVEL 3  
 ALLOWNODE 1:2606/\*. \* <- Any node or point in Zone 1, Net 2606  
 ALLOWNODE 1:2606/4??.\* <- Any node or point in Zone 1, Net 2606  
 and three digit node number starting  
 with a 4

ALLOWNODE 1:100/200.0 LEVEL 0 MAXFILES 5 MAXBYTES 100000

## 1.57 Quick Silver - Command Options - DATABASE

### DATABASE UTILITY

The DataBase Utility will read the directory information from your QSilver.CFG file and build a master database listing. This is an optional utility, so play with it and see if the speed increase is worth it. You need to specify a

DATABASE  
path in your QSilver.CFG file for the DataBase utility to work.

Usage: DATABASE [Path to QSilver.CFG]

Example: DATABASE QSilver:QSilver.CFG

## 1.58 Quick Silver - Command Options - OFFLINEMSG

### OFFLINEMSG

The OFFLINEMSG option allows you to specify a textfile to be appended to the reply message in the event that a requested file is found in the DATABASE, but is not found on the hard drive or CDROM.

Usage: OFFLINEMSG [Path to File]

Example: OFFLINEMSG QSilver:TextFiles/OffLine.TXT

## 1.59 Quick Silver - Command Options - UPDATEDL

### UPDATEDL

Allows you spawn another external program to update your BBS' download counters. The support for this option was made external due to the overwhelming diversity of software and the way they store their information. Future updates will include scripts for various BBS software. The external program will be spawned asynchronously so that QSilver can continue processing.

QSilver will pass all requested filenames with their complete paths to the called script or program in one long argument string. It's up to the called program to act upon this information accordingly.

Usage: UPDATEDL [Command to Execute]

Example: UPDATEDL RX REXX:UpdateDL.rexx

---

## 1.60 Quick Silver - Registration Information

This program is released as SHAREWARE. If you use Quick Silver for more than 30 days, you must register the program or stop using it. I do not like REQUESTORS as reminders, so you won't be annoyed these, but there is a delay loop imposed in the unregistered version. I am asking for only a \$15 registration. Please send your registration fee, payable to:

Roger Clark  
5508 Delwood Drive  
Austin, TX 78723 USA

## 1.61 Quick Silver - Testing your Setup

You can locally test Quick Silver to be sure that your configuration is setup properly by calling QSilver with all the required parameters. You can best do this by setting up an ALIAS in your S:Shell-Startup file.

```
ALIAS TestFreq C:QSilver S:FreqFiles S:TestResults S:QSilver.CFG  
"1:382/105.0" 14400 Roger Clark <--( All on one line )
```

"S:FreqFiles" is a simple text file that contains a list of the files that you are testing, each listed on an individual line.

"S:TestResults" is the reply (.RLO) file that QSilver creates with a listing of all files to be sent back to the requesting system.

"S:QSilver.CFG" is your configuration file for Quick Silver.

"1:382/105.0" is your node number.

"14400" is the baudrate to simulate. Optionally, replace with "[" and add the baudrate manually when you test QSilver, ie: "TestFreq 28800"

"Roger Clark" is your name.

## 1.62 Quick Silver - Revision History

History of updates for each version of Quick Silver.

1.00

1.01

1.02

1.03

1.04

1.05

1.06

1.07

1.08

1.09

1.10

1.11

1.12

1.13

1.14

1.15

1.16

1.17

1.18

1.19

1.20

1.21

1.22

1.23

1.24

1.25

1.26

1.27

1.28

1.29

1.30

1.31

1.32

1.33

1.34

1.35

1.36

1.37

1.38

Time for something new!

2.00

## 1.63 v1.00

- First release to Beta testers.

History

## 1.64 v1.01

- Fixed password case sensitivity problem.
- Cleaned up code in some loops for better speed.

History

## 1.65 v1.02

- Added  
IGNORE  
keyword.
- Modified code to run when compiled with the "Rexx Plus Compiler".

History

## 1.66 v1.03

---



- Added  
EXCLUDE

History

## 1.67 v1.04

- Modified to run asynchronously with the Welmat/WPL mailers ↔  
by  
Russell McOrmond.

History

## 1.68 v1.05

- Cleaned up and optimized the code. Runs a little faster ↔  
now.  
- A few minor bugs have been caught and eliminated.

History

## 1.69 v1.06

- Added new code to generate Hex values for the MsgID kludge.  
- Fixed bug that would not allow unlimited files per session.  
- Added a few cosmetic changes to the text formatting.  
- Added

DEFAULTSEND  
keyword.

History

## 1.70 v1.07

- Fixed problem with  
EXCLUDE  
keyword.  
- Added

SCREEN  
keyword.

- Added tearline after First Time Letter.
- Fixed minor bug in MsgID calculations.

History

## 1.71 v1.08

- Added Update Request capability.
- Modified SCREEN function to not display requestors if SCREEN keyword is not specified. This keeps the compatibility with WB 1.3 still.
- Optimized a few more sections of code for better speed.

History

## 1.72 v1.09

- Added SCREENPOS keyword.
- Optimized routine for Update Requests for better speed.
- Added error checking to make sure MAGIC file paths are valid.

History

## 1.73 v1.10

- Added Wildcard pattern matching ( \* and ? )
- Now SHAREWARE! Please register.

History

## 1.74 v1.11

---

- Added message to report file to indicate failed requests ↔  
that  
involved wildcards when no match was found.

History

## 1.75 v1.12

- Added  
MAXMATCH  
keyword.  
- Made Quick Silver reject \*.\* as a wildcard.  
- Fixed minor bug in reply message routine.

History

## 1.76 v1.13

- Added  
DEFCOMMENT  
keyword.

History

## 1.77 v1.14

- Bob English provided updated code for MsgID kludge ↔  
generation.  
- Added  
PRIORITY  
keyword.

History

## 1.78 v1.15

---

- Added the TOPT kludge in the netmail headers by request (F. ←  
Perricone).
- Optimized a few more sections of the code for more speed.

History

## 1.79 v1.16

- Optimized the code AGAIN. I just keep finding little ←  
things to speed  
things up.
- Update requests were not working with Magic Filenames.

History

## 1.80 v1.17

- Added the Magic Filename reporting back to the QSilver.Log ←  
due to  
some routines being reorganized.
- Added support for UNIX timestamps in the update request
- Changed format of update request to +/-Uyymmdd[hhmmss]

History

## 1.81 v1.18

- Changed date format for log files to look more like ←  
TrapDoor's log  
(request by F. Perricone)

History

## 1.82 v1.19

- Added support for DLG's file comments. See  
DLGDESC  
.
- Compiled version now displays registration info in the status window

if it's active.

History

### 1.83 v1.20

- More minor bug fixes.
- Optimized the nodelist info lookup.
- Beginning work on optional DataBase file indexing

History

### 1.84 v1.21

- Added a DATABASE file for improved speed when locating files.
- Now only distributing the compiled version.

History

### 1.85 v1.22

- Fixed bug in routines that got the custom descriptions used by DLG BB/OS. This would only allow 1 file to be requested per session when DLGDESC was used (Chris Guertin).
- Added LOCALMSG keyword (Chris Guertin).
- Added SYSOP keyword.
- Fixed bug that would cause errors if the path for ACCTPATH ended with a ":" or "/".
- Added a delay loop for unregistered users. This is bypassed when you register Quick Silver and receive your keyfile.

History

---

## 1.86 v1.23

- Modified wildcard searches to include the AmigaDos "#?" ↔ pattern.
- Fixed bug in DataBase mode that wasn't properly handling password on a  
DIRECTORY  
path.

History

## 1.87 v1.24

- Added optional System Request Totals to the reply message.
- Added code to disable system requestors in case an invalid path is encountered (Mark Montminy).

History

## 1.88 v1.25

- Fixed small bug in System Request Totals routine.
- Included small script "MakeSystem.rexx" to total up all requests from existing accounting files.

History

## 1.89 v1.26

- Added  
NOMATCHFILE  
keyword.

History

---

## 1.90 v1.27

- Added wildcard option to  
MAGIC  
filenames. Will only send the most  
recent match to the wildcard. Great for NODEDIFFs! (Stace Cunningham)

History

## 1.91 v1.28

- Added support for Excelsior! style EX.BBSPaths file. Will ↔  
work with  
the existing DIRECTORY config keyword as well as seperately. See the  
EXCELPATHS keyword.

History

## 1.92 v1.29

- Had to remove code for the EX.BBSPaths file as it didn't ↔  
quite  
function the way I had anticipated. Will do some more work with the  
FBase.dat file to figure out the best way to deal with this. :-(  
- Finally found and fixed a small bug that would report an incorrect  
line number in the code where an error occured.

History

## 1.93 v1.30

- All additions for this version were suggested by Michael ↔  
Mehmet.  
[So now you know who to blame or thank! :-) ]  
- Added  
ALLOWNODE  
keyword.  
- Added  
EXCLUDEMSG  
keyword.  
- Added  
FAILEXCLUDE  
keyword.

- Added  
FAILPOINT  
keyword.
- Added  
FAILSECURE  
keyword.
- Added  
FAILUNLISTED  
keyword.
- Added LEVEL option to  
DIRECTORY  
keyword.
- Added LEVEL option to  
MAGIC  
keyword.
- Added POINTMSG keyword.
- Added  
SECURE  
keyword.
- Added  
SECURITYMSG  
keyword.
- Added  
UNLISTEDMSG  
keyword.
- Expanded wildcarding for  
EXCLUDE  
keyword.
- Distributing AmigaGuide® format documentation.

History

## 1.94 v1.31

- Fixed bug with  
ALLOWNODE  
and multiple access levels.
- Fixed bug with  
LOCALMSG  
that was not including the INTL or  
FMPT kludges in the netmail.

History

## 1.95 v1.32

- Fixed bug in source that would cause the system to crash ↔  
under some



- pretty strange situations.
- Finally got the last bug in the DLGDESC option fixed. Thanks to Andy Wasserman for helping me test this out.

History

## 1.96 v1.33

- Added EXCELDESC keyword to use Excelsior! BBS file comments.

History

## 1.97 v1.34

- Fixed potential bugs with Locale settings.
- Fixed a bug that was showing up on a few DLG systems.
- Optimized the code again in a few sections by moving some repetitive code into common function calls.

History

## 1.98 v1.35

- Added MaxFiles and MaxBytes flags to the ALLOWNODE statement.
- Added CNETDESC keyword. Thanks to Chris Brundell for help with the structures.

History

## 1.99 v1.36

---

- Added code to only send the  
DEFAULTSEND  
file on the first request  
session of each day.
- Added the  
BAUDMAX  
keyword.
- Added the  
TIMEMAX  
keyword.

History

## 1.100 v1.37

- Modified code to not send the  
DEFAULTSEND  
file if is was requested  
in the same session.
- Added DAYS option to the DEFAULTSEND keyword. Will let you select  
the number of days between the sending of the file specified.
- Added support for multiple AKAs. This will scan all the AKAs of the  
requesting site and find the first known node. This is useful if  
the requesting system doesn't use their FidoNet address as their  
primary address.
- Fixed small bug in the  
DATABASE  
utility. Now appends a "/" if the  
DIRECTORY path does not end in either a ":" or "/".

History

## 1.101 v1.38

- Added  
OFFLINEMSG  
keyword to report files in database but not online.
- Added  
UPDATEDL  
keyword to support updating of BBS download counters.
- Added  
SUBDIR  
option to the DIRECTORY command for use with the  
DATABASE utility.
- Added full support for points. QSilver will now also use the proper  
TOPT and FMPT kludges if the host or requesting system is running as  
a point. Seems a lot of European systems run as points and still

allow file requests, so here's the support you wanted.

History

## 1.102 v2.00

- Major rewrite of some code to enable Quick Silver to operate in the background and wait on an ARexx port for messages.
- ARexx port name "QSilver"
- ARexx commands supported: QUIT Shuts down Quick Silver  
CONFIG Rereads the default config file
- DataBase utility reworked for improved file location speed.
- See "Read.Me!" file for more details.

History

## 1.103 Quick Silver - Tips & Tricks

TIPS & TRICKS

Access Levels

Some basic hints as to how they work

Security

So you want to be as secure as Fort Knox?

## 1.104 Quick Silver - Tips & Tricks - Access Levels

ACCESS LEVELS

Access levels can be a bit tricky to get used to at first. You can set multiple levels for individual nodes, file directories, and magic filenames. These levels will be compared and if any of them match then access will be given to the calling system. Let's look at a few quick examples:

```
ALLOWNODE 1:382/105.0 LEVEL 1-5,7,15
```

This node has access to any directory or magic file that has an access level of 1,2,3,4,5,7 or 15. If any of these levels match, then the node will be given access to the appropriate files.

---

DIRECTORY DH0:Files/Games LEVEL 1,3-5,8,14

This directory will only be accessible to a node with an access level that matches either 1,3,4,5,8 or 14. The node listed above would be given access since it matches for levels 1,3,4 and 5.

DIRECTORY DH0:Files/Private LEVEL 6,10-14

This directory requires an access level of 6,10,11,12,13 or 14 in order for its files to be accessed. The node listed above would not be given access since it does not have any of the needed levels.

DIRECTORY DH0:Files/General

This directory has open access (Level 0) and can be accessed by any node that calls. The node listed above would be given access to this directory by default of the level 0 access requirement.

MAGIC FILES DH0:Files/FileList

This magic filename can be accessed by any node in the same manor as the above listed directory. You can specify multiple magic filenames each with a different level, and have a different file sent to requesting nodes based on their access levels.

MAGIC UPDATES DH0:Files/Updates/Update1.LHA LEVEL 1-5

MAGIC UPDATES DH0:Files/Updates/Update2.LHA LEVEL 6-10

MAGIC UPDATES DH0:Files/Updates/Update3.LHA LEVEL 11-15

These magic filenames, would send a different file to the requesting node based on their access level, while using the same magic name. If a requesting node has matching access levels for more than one magic filename, then the first match Quick Silver encounters will be sent.

ALLOWNODE 1:382/105.0 LEVEL 8-13

This node would match both Update2.LHA and Update3.LHA but would only be sent Update2.LHA since it was listed in the configuration file first.

## 1.105 Quick Silver - Tips & Tricks - Security

### SECURITY

The built-in security features of Quick Silver allow you to run an open-ended or restricted access system, based on your personal requirements. You can limit the nodes that are able to request files from your system by using the SECURE keyword in your configuration file. When this is in effect, only those nodes that you list with an ALLOWNODE keyword will be given access to your system. This is the most secure mode that Quick Silver offers.

If you wish to limit the nodes that have access to your file areas, but still allow any node to request certain files such as your full file listing, or new file listing, then DO NOT use the SECURE option, but take these steps instead.

---

1. List each node that you wish to have access to your file areas in an ALLOWNODE statement and give them the appropriate access to the file areas and magic filenames.

```
ALLOWNODE 1:382/105.5 LEVEL 5
ALLOWNODE 1:382/105.* LEVEL 4
ALLOWNODE 1:382/1??.* LEVEL 3
ALLOWNODE 1:382/*.* LEVEL 2
ALLOWNODE 1:*/*.* LEVEL 1
```

2. Use the global wildcard to assign access level 0 to all other nodes. This will let any other node call into your system, but only get access to file areas or magic files that have level 0 access. Make sure this is the LAST allownode statement in your configuration file, as Quick Silver reads the statements in a most specific to most general order.

```
ALLOWNODE */*.* LEVEL 0
```

3. Assign appropriate access levels to all other magic files and directories that you don't want to give general access to. This will keep all nodes with level 0 out, and only allow those nodes that you have specifically given access to be able to request those files. You can then assign level 0 to your magic filename for your file listings and anyone can request them without having access to the rest of your system.

4. New to version 1.35 is the ability to add file and byte limits to individual systems without effecting other systems. Use the additional MAXFILES and MAXBYTES keyword to define the limits for each node. You can use either or both of these keywords as needed. Omitting them in the ALLOWNODE definition will use the global limits.

```
ALLOWNODE 1:234/567.0 LEVEL 1 MAXFILES 5 MAXBYTES 250000
```

## 1.106 Technical Support

TECHNICAL SUPPORT

Technical support for Quick Silver is available through the following:

Netmail to Roger Clark at the following FTN addresses

```
FidoNet : 1:382/105.0      FamilyNet: 8:3000/14.0
AmigaNet: 40:201/1.0      DarkNet  : 11:2/1.0
```

Internet mail addressed to: rclark@infomail.com

Voice support at the following numbers:

USA, Puerto Rico, Virgin Islands and Canada (Registered Users Only):  
1-800-928-9736

---

Worldwide:  
1-512-928-4955