

AmiPOP

COLLABORATORS

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

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	AmiPOP	1
1.1	AmiPOP	1
1.2	Introduction	1
1.3	What it is	1
1.4	What it's not	2
1.5	What you need	2
1.6	AmiPOP features	2
1.7	Options	3
1.8	Window	3
1.9	Gadgets	3
1.10	POPHost	4
1.11	Username	4
1.12	Password	4
1.13	Mail Box	4
1.14	Append to Mail Box	5
1.15	Delete Messages	5
1.16	Check Periodically	5
1.17	How Often	5
1.18	Check Now	5
1.19	Menus	6
1.20	hide	6
1.21	About...	6
1.22	Quit	6
1.23	Command Line	6
1.24	Username	7
1.25	Password	7
1.26	POPHost	8
1.27	MailBox	8
1.28	CheckPer	8
1.29	Time	8

1.30 Delete	8
1.31 Append	9
1.32 NoWindow	9
1.33 Port	9
1.34 Notify	9
1.35 cx_popkey	9
1.36 immediate	10
1.37 quiet	10
1.38 PubScreen	10
1.39 Once	10
1.40 Miscellaneous Notes	10
1.41 POP3 Daemon	11
1.42 Cautions	11
1.43 Quitting AmiPOP	11
1.44 Thanks	12
1.45 Bugs	13
1.46 To Do	13
1.47 A Note about Notify	13
1.48 Author Info	13
1.49 History	14

Chapter 1

AmiPOP

1.1 AmiPOP

AmiPOP - A POP3 Client

© 1993 Scott Ellis

Introduction

AmiPOP options

Miscellaneous notes

History

1.2 Introduction

Introduction-

What it is

What it's not

What you need

AmiPOP features

1.3 What it is

What it is -

AmiPOP is a POP3 Client for AmigaDOS. It was written directly from the POP3 RFC (RFC #1225). AmiPOP is only a simple transport mechanism for moving electronic mail messages between a POP-Host, typically a mainframe, and your local AmigaDOS system. The advantage of POP over other transports such as SMTP is that all E-mail can still be directed to the mainframe system. This is of primary importance when either a dialup line is used, which has a new IP address for each connection, or when a user uses many machines, and wishes to be able to transparently read mail on all of them.

1.4 What it's not

What it's not -

AmiPOP is not intended to be a do-everything-mail-reader-reminder-slicer-dicer program. AmiPOP is meant to simply transport mail from a host to your Amiga. That's it. Nothing else. The "Notify" function included in AmiPOP is intended to be a stopgap measure only, and was included as a request of an especially helpful beta-tester. The "Notify" function will never grow. If you want to do exotic things when you get new mail, use another utility, like Mike Schwarz's 'Mail-Minder', or Scott Ellis' (me) program 'Sabot'.

1.5 What you need

What you need -

AmiPOP requires either the AS225r2 TCP/IP package or the AmiTCP package, and a machine running a POP3 daemon. (More on this later)
AS225r2 and AmiTCP require AmigaDOS 2.0 or higher, and this is therefore required by AmiPOP as well.

1.6 AmiPOP features

AmiPOP features -

- AUISG compliant (or at least it tries to be!)
 - Makes heavy use of 2.0 (and higher) functions such as ReadArgs, utility.library, GadTools, and 2.0 functions in Amiga.lib.
 - Written from RFC 1225, not a port of a Unix or a PC application.
 - Small code size, and low system resource usage.
 - Commodities interface
-

1.7 Options

Options -

Window

Command line

1.8 Window

Window -

Gadgets

Menus

1.9 Gadgets

Gadgets -

Each gadget in the window may be activated from the keyboard by using the underlined character in the gadget label (case insensitive).

Activating a string or integer gadget enables the information to be immediately typed in, and the gadget can be de-activated either by the RETURN key, or by cycling to the next gadget with the TAB key.

After activating one of the string or integer gadgets, the TAB key allows cycling through each of the gadgets. The order is:

- POP Host
- How Often
- Username
- Password
- Mail Box

If "How Often" is disabled, then it is ignored, and the gadget is skipped during the TAB-Cycle.

POP Host

Username

Password

Mail Box

Append to MailBox

Delete Messages
Check Periodically
How Often
Check Now

1.10 POPHost

POP Host -

This is the address of the machine which is running the POP3 daemon. It can either be an IP address, a full name and domain, or just the name (if the site is in your inet:DB/Hosts file, or in your local domain).

1.11 Username

Username -

Username is your login name on the POP Host machine. It is used in determining which mailbox to look at, and which password to accept.

1.12 Password

Password -

This is the password for Username on machine POP Host. The Password gadget is a special string gadget which will NOT echo what you type. Instead, you will see a bullet (".") for each character you type. The Password gadget is more limited in its editing abilities than the other gadget, and will only allow you to delete characters from the end of the password, or clear the entire gadget.

1.13 Mail Box

Mail Box -

Mail Box is the FULL AmigaDOS path and filename that you would like the mail stored in on your Amiga. Most UUCP mail-readers require this be "UUMAIL:Username", but you are free to choose whatever you like.

1.14 Append to Mail Box

Append to Mail Box -

When checked, this gadget will append all incoming mail to the file listed in the MailBox gadget. If it is unchecked, new mail will overwrite the mailbox. This gadget is most useful in conjunction with the "Delete Messages" gadget.

1.15 Delete Messages

Delete Messages -

If Delete Messages is checked, it will delete the mail message on POP Host after it is downloaded. Caution is advised with this gadget, since once mail is deleted on the POP Host, there is no way to retrieve it. This gadget is best used with "Append to MailBox".

1.16 Check Periodically

Check Periodically -

When this gadget is marked, mail will be transferred from the POP Host on a time schedule. This schedule is defined in "How Often".

1.17 How Often

How Often -

This gadget defines the number of minutes that should elapse between queries for mail to POP Host. If Check Periodically is not checked, this gadget will be disabled, and have no effect.

1.18 Check Now

Check Now -

Check Now will immediately query POP Host for mail. It is primarily used for impatient people who cannot wait for the Check Periodically time to elapse, or who would like the mail checked manually, instead of periodically.

1.19 Menus

Menus -

Menu options can also be accessed via their key combination (Right Amiga key).

Hide

About...

Quit

1.20 hide

Hide -

This option will close the AmiPOP window, but not quit AmiPOP. The window may be reopened either via the hotkey (If specified) or with Commodities Exchange.

1.21 About...

About... -

Opens a requester with the AmiPOP authors name.

1.22 Quit

Quit -

Quits the AmiPOP program. This is identical to sending AmiPOP a Control-C character, or clicking on the close box in the AmiPOP window.

1.23 Command Line

Command Line -

Command line options maybe be given either on the command line when starting AmiPOP from a shell, or as ToolTypes when using AmiPOP from an icon.

AmiPOP uses ReadArgs to process command line and tooltype arguments passed to it.

Arguments:

Username
Password
POPHost
MailBox
CheckPer
Time
Delete
Append
NoWindow
Port
Notify
Cx_Popkey
Immediate
Quiet
PubScreen
Once

1.24 Username

Username -

This is identical to the Username gadget in the AmiPOP window. Specifying a username on the command line avoids having to enter it into the AmiPOP window.

1.25 Password

Password -

This is identical to the Password gadget in the AmiPOP window. Specifying a password on the command line avoids having to enter it into the AmiPOP window.

1.26 POPHost

POPHost -

This is identical to the POP Host gadget in the AmiPOP window. Specifying a POP Host on the command line avoids having to enter it into the AmiPOP window.

1.27 MailBox

MailBox -

This is identical to the MailBox gadget in the AmiPOP window. Specifying a MailBox on the command line avoids having to enter it into the AmiPOP window.

1.28 CheckPer

CheckPer -

When given, this switch specifies that the Check Periodically function should be enabled. It is identical to making Check Periodically enabled in the AmiPOP window.

1.29 Time

Time -

Time, followed by an integer will place a default value into the How Often gadget in the AmiPOP window. If CheckPer is not also given, or if the Check periodically gadget in the AmiPOP window is not active, this setting has no effect.

1.30 Delete

Delete -

When present, this switch enables the Delete Messages function of AmiPOP. It is identical to the Delete Messages gadget in the AmiPOP window.

1.31 Append

Append -

When present, Append activates the Append to MailBox feature of AmiPOP. It is identical to activating the Append to MailBox gadget in the AmiPOP window.

1.32 NoWindow

NoWindow -

When the NoWindow flag is given, it instructs AmiPOP not to open it's window my default. This is useful when running AmiPOP from WBStartup, or from a script. The AmiPOP window may later be opened by the hotkey, or via Commodities Exchange.

1.33 Port

Port -

Normally, POP3 uses port 110 on the POP Host machine. When the Port switch is given, followed by an integer, that will be used in attempts to connect to the POP3 daemon on the POP Host, instead of port 110.

1.34 Notify

Notify -

The Notify switch will force AmiPOP to open a requester when it detects that you have new mail. By default, AmiPOP does not notify the user of new mail.

1.35 cx_popkey

Cx_Popkey -

This is the hotkey that AmiPOP will use. When pressed, AmiPOP will open it's window, if it is not already open. Any valid hotkey combination may be given. If Cx_Popkey is not specified, AmiPOP will not add itself to the input chain, and will not cause any system performance decrease.

1.36 immediate

Immediate -

The Immediate switch causes AmiPOP to immediately check for mail when it is started. The default is to wait the amount of time specified in the "How Often" gadget before checking.

1.37 quiet

Quiet -

When the Quiet flag is given to AmiPOP, AmiPOP will not open any requesters. This means that you will not be able to get any notice of errors, or get the Notify or About requesters.

1.38 PubScreen

PubScreen -

Giving AmiPOP the name of a public screen will cause the AmiPOP window to open on that public screen. If AmiPOP cannot open on the specified screen, or if no screen is specified, AmiPOP will open on the default public screen (normally WorkBench).

1.39 Once

Once -

The "Once" keyword causes AmiPOP to check for mail immediately, then exit.

1.40 Miscellaneous Notes

Miscellaneous Notes -

Pop3 Daemon

Cautions

Quitting AmiPOP

Thanks

Bugs

To Do

A Note about Notify

Author Info

1.41 POP3 Daemon

POP3 Daemon -

A POP3 Daemon does not need root privledges to install and operate. Any user may run a POP3 Daemon, so long as it is configured to use a non-blocked port (typically those above 1024).

A variety of POP3 Daemons may be found on via ftp on rohan.sdsu.edu.

1.42 Cautions

Cautions -

It is possible that not all of the bugs have been worked out of AmiPOP yet, and there are therefore some pitfalls which the user should keep in mind.

Foremost, Delete Messages WILL delete E-mail messages on POP Host, so be cautious. If AmiPOP encounters a problem during the transmission of a message, it will usually abort, and keep all the messages after and including the one with which it had problems with, on the POP Host, but there is always the possibility that the mailbox could be corrupted by a semi-functional POP3 Daemon on the POP Host, so be aware that your mailbox COULD get mangled.

If for any reason, AmiPOP must abort abruptly, it attempts to send the QUIT signal to the POP3 Daemon, but depending on the circumstances of the AmiPOP failure (i.e., power failure or general system crash), this may not be possible, and a brain-dead POP3 Daemon could be left running on the POP Host, preventing any further POP3 connections until that process is killed. Please review the CPU and Time limits for processes under Unix, as well as POP3 Daemon return codes.

1.43 Quitting AmiPOP

Quitting AmiPOP -

There have been many requests for a way to quit AmiPOP from a script. The simplest way to do this is to use a utility such as HandleCX, or CXKiller (available on Aminet), and stop AmiPOP that way. You may also send the AmiPOP process a Control-C signal to stop it, but quitting via commodities is often nicer. ;-)

1.44 Thanks

Thanks -

Thanks for help in creating AmiPOP are far and wide, and generally encompass the #amiga and #amigatech group on IRC. Specifically, the author would like to express his deepest gratitude to the following people:

Michael B. Smith - LOTS and LOTS of help with sockets, SMTP, NNTP, the OS, and just about everything.

Christopher A. Wolf - Chris provided invaluable help regarding the commodities and ReadArgs portion of this code, as well as general help on how in the heck to get this POP program working.

Ken C. Dyke - The very cool Password gadget edithook would have had me stumped forever, had it not been for a nice bit of code that Ken just happened to have laying around. :-)

Michelle R. Ketcham - She found my stupid mistakes, and also forced me to write AmiPOP in the first place.

Tim Wadsworth - Tim created the AmiPOP.guide file from my horrid AmiPOP.doc file, as well as applying the diffs send to me by Tomi Ollila for AmiPOP V1.6. Tim also compiled and tested the AmiTCP version of AmiPOP, and fiddled with the makefiles, and header files, to get things to work nicely for AmiTCP users. Loads of thanks Tim!

Tomi Ollila - Tomi ported AmiPOP V1.6 to AmiTCP, very soon after it was released, and then was so kind as to send me his diffs, so that AmiTCP support can be included in further AmiPOP revisions.

Jeremy Friesner - Jeremy made sure AmiPOP worked with the socket.library for AmiTCP...this saves me the trouble of having to get an AmiTCP port done before I release a new AmiPOP version. Thanks Jeremy!

Christopher A. Wichura - Christopher fixed up the AmiTCP sections of the code, and did the compile for this release 1.8 of AmiPOP's AmiTCP binaries ←

David P. Duggan - David did a WHOLE lot of work on AmiPOP, and basically rewrote the majority of pop_dopop.c, eliminating some longstanding bugs. Many many thanks to David for his work.

Hidehiko Ogata - Hidehiko spent time on making the retrieve() function work reliably. Many thanks!

Dan Murrell Jr. - Dan spend a lot of time helping me make sure things worked as they should, and offering suggestions for improvements.

I'm sure I'm forgetting lots and lots of people, and I hate that. They know who they are. ←

1.45 Bugs

Bugs -

None left that I know of.

1.46 To Do

To Do -

- Allow AmiPOP to get signals while it's transferring. Currently it can't be hidden/quit while a transfer is in progress. This is lame. ←

1.47 A Note about Notify

The current implementation of the "Notify" feature is done in such a way that the program is easy to maintain, and remains very small. I have had a LARGE number of requests to make the requester that "Notify" brings up be done in such a way that it will timeout, or will continue transferring files, even if the requester is not satisfied. I have looked into this prospect, and have concluded that it would make the code more cumbersome, and larger than need be. If users wish to have a reliable method of being notified when new mail arrives, I urge them to take a look at another program I wrote, which IMHO complements AmiPOP quite nicely. It is called "Sabot", and can do just about anything you want when new mail arrives. "Sabot" is most likely available from the location you received AmiPOP from. ← If not please contact me, and I will be happy to provide you with a copy of "Sabot".

1.48 Author Info

Author Info -

The author of AmiPOP may be contacted at:

Scott Ellis : sellis@rohan.sdsu.edu
sellis@gi.com
ScottE on IRC
3640 Mt Everest Blvd, San Diego, CA, 92111 (USA)

Please feel free to contact me, and let me know if you find any bugs in AmiPOP, or if you find any more efficient ways of doing things. I'm interested in hearing if anyone actually USES this program, or if it's just me. ;-)

The source and binary to this program are freeware. Do with it whatever you like, just remember to give me some credit if you get rich and famous.

The pop_dopop.c file, and miscellaneous other things are from David P. Duggan. Any thanks for the well-workingness should go to him as well. David put in quite a lot of time making AmiPOP perform flawlessly. All good stuff is his, all bugs are mine. ;-)

1.49 History

History -

1.0 - Initial release

1.1 - Fixed a few parts of the code where it was possible that we could do something with a window that wasn't opened yet.

Attempted to fix End Of Message problems.

1.2 - End of Message problem fixed some more. ;-) Still a big kludge.

Added busy pointer during dopop() routine.

Changed "Bummer" to "Ok" in most requesters.

Minor reorganization.

1.3 - Bug Fix.

1.4 - Released with INetUtils via Michael B. Smith. Thanks Michael!

1.5 - Major Code changes.

Variables now dynamically allocated to reduce executable size.

Adding of "From" header revamped.

Added forgotten PrintError() in ReadArgs() section! Oops!

Fixed longstanding bug of missing the first bit of a message when it was sent in the same packet as the +OK string. Fixed now. ↔

INetUtils release delayed due to circumstances beyond my control. V1.5 released with INetUtils-1.0.

1.6 - Moved 2 checkboxes over a tad, in hopes of avoiding conflicts with window border ↔

when using certain fonts.

Changed location of "Notify" window in code, and also make AmiPOP sendstat() after the "Notify" window is satisfied, in case new messages arrive.
Thanks to Stan Burton for that suggestion.

1.7 - Finally fixed the checkbox font rendering...really, honest, I did this ← time.

Reduced code size by performing some magic. (hehehe)

Added PubScreen keyword, which should have been in there from the beginning.

AmiTCP version added by Tim Wadsworth.

Minor header and makefile shanges by Tim Wadsworth.

AmiPOP.guide made by Tim Wadsworth. (revised by me)

1.8 - Added Async I/O routines to improve speed when writing the mailbox.
Routines courtesy Vertex/Martin Taillefer.

AmiPOP now correctly "Hides" when the window closegadget is clicked, rather than exiting.

AmiPOP now lists the current MailBox in the Commodities Exchange information window.

AmiPOP now doesn't delete a message from the POP Host if there is a problem writing it to the local mailbox.

Now use the sprintf() in amiga.lib, to chop 1.7k off the executable size.
Minor changes to the sprintf() format definitions needed to accomodate this.

Added QUIET and IMMEDIATE options.

Integrated new ParseParameters() code from Chris Wolf. Thanks!

AmiTCP code now integrated into main AmiPOP code. Thanks Chris Wichura!

1.9 - Now uses `__AMIGADATE__` for version string.

Date compiled now appears in "About" requester.

Fixed minor bug with finding "From " in message headers.

No longer chops off last CR at end of message.

Removed some pointless sprintf()'s.

No longer relies on SAS's autoopen/close stuff.

Modified pop.h so that the AmiTCP version compiles easier.

1.10 - Updated ASyncI/O code to latest version.

Moved GTMN_NewLookMenu to the right place, so now the menu's really

do look good. ;-)

Attempts to fix weird "From " header insertion problems. Thanks to David Duggan for lots of help in tracking this down.

Commodities information changed. Now AmiPOP name is based on mailbox, not username, as it should have always been. Also, the Information field shows 'Mailbox for user@host' to be more descriptive.

AmiPOP now uses nearly RFC-822 compliant headers. AmiPOP doesn't insert timezone information, but other than that, they are RFC compliant. Many thanks to Michael B. Smith for some nice efficient code for this.

Changed all sprintf() references to DoFmt() references. This saved a little space, since the sprintf() from amiga.lib doesn't get linked in.

- 1.11 - Fixed a glaring bug with parsing "From " headers once more. This bug is really starting to piss me off. ;-)
- 1.12 - Fixed longstanding problem with the Commodities interface which could cause an enforcer hit or a crash when starting AmiPOP a second time with the same parameters. Thanks to Kai Iske for help in tracking this down ↔

- 1.13 - Changed error messages to be more descriptive.

Added "Once" keyword to make AmiPOP quit after checking mail.

Changed behaviour of starting AmiPOP twice with the same parameters. Now, this causes the AmiPOP process with those same parameters to exit.

- 1.14 - AmiPOP no longer gets messages in reverse order. This "feature" had been in there since the the beginning. I wonder why I never noticed it? Thanks to Erwin Zwart for informing me of this.

AmiPOP doesn't send the "HELO" greeting command anymore. Apparently this ↔ was causing some problems with certain POP3 daemons....hope nobody is using a ↔ braindead daemon that needs it! Scott Thomason found this oddity. Thanks Scott!

Messed with message truncation problems once more. I'm still stumped... ↔ please... someone take a look at this code and see where the loophole is!

- 1.15 - It's fixed! "From " problems, and message combining are all fixed, thanks to the wonderful work of David Duggan in fixing those longstanding bugs. All AmiPOP users (Especially me!) owe him a great big "Thank you!". David put in a lot of time (and trashed messages) to get this code working.

Thanks to Hidehiko Ogata for looking into the "From " problem as well. He did find an error, but David's changes eliminated the need for Hidehiko's ↔ fix.

I've gone back to the old way of having AmiPOP open it's window if you try and run it twice with the same parameters. There were quite a few

users who wanted this the other way, but I just got too many questions and complaints about the change to warrant keeping it. Besides, AUISG says it should open the window.

If the Password field is blank, the password is not sent. This should help some people using Kerberos.

- 1.16 - Added in the total number of messages in the titlebar during transfers, so that it now says "Retrieving x/y". Thanks go out to Dan Murrell Jr. for reminding me about this.

Added WinX and WinY keywords to determine where the window appears. Kai 'MagicCX' Iske has been bugging me about this forever, since it should have been in v1.0. ;-) Thanks Kiske.

Made the zipped window zip to the same coordinates as the original window under version 3.x of the OS. 2.x users will get the old behaviour.

Changed all references to "MailDir" to "MailBox" to help out with some confusion as to whether it should be a path, or a path and filename. Thanks to Stephanie Sullivan for reminding me of the "MailDir" problems.

AmiPOP now correctly handles long connect messages. Before it was assuming the initial message would fit in BUFSIZE. This was wrong. ;-) Thanks to NTT (Sorry, can't remember your real name) for finding this bug.

The AmiPOP window now blocks input while a transfer is in progress. I started make it respond to input, but decided blocking was easier. ;-)