

**Sabot**

**COLLABORATORS**

	<i>TITLE :</i> Sabot		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 25, 2024	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1 Sabot</b>	<b>1</b>
1.1 Sabot	1
1.2 What	1
1.3 So	2
1.4 mail	2
1.5 nomail	2
1.6 oldmail	2
1.7 mailbox	3
1.8 iconname	3
1.9 xpos	3
1.10 ypos	3
1.11 clickcommand	3
1.12 clickstack	3
1.13 mailcommand	4
1.14 mailstack	4
1.15 dropcommand	4
1.16 dropstack	4
1.17 beeplots	4
1.18 countdelete	4
1.19 time	5
1.20 mailrdy	5
1.21 thingies	5
1.22 uses	6
1.23 thanks	6
1.24 author	7
1.25 history	7

---

# Chapter 1

## Sabot

### 1.1 Sabot

Sabot - A Mailbox Monitor

By Scott Ellis

About Sabot	Usage
Things to know	Thanks
About the author	History

### 1.2 What

What is Sabot?

Sabot is basically a clone of Mike Schwartz's MailMinder program, but but smaller and with added functionality. What Sabot does is watch a file (presumably your mailbox), and do things when the mailbox changes.

What Sabot always does is display an AppIcon on your Workbench screen showing you the status of the file being watched. If the file is empty it displays an icon illustrating an "empty mailbox". If the file is the same size an icon depicting old mail is displayed. If the mailbox gets bigger, then the icon changes to a "full mailbox" icon, until you look at the file. Very simple.

Is a silly AppIcon worth almost 6k of space on your system? Of course it is...but wait, that's not all! ;-) Sabot can also launch programs. Sabot can launch programs under a variety of circumstances, including:

- Mailbox got bigger
- Click on the AppIcon
- Drop files onto the AppIcon

Sounds exciting, eh?

---

## 1.3 So

So how do I use Sabot?

Sabot is simple to use. You start Sabot from either the command line, or from an Icon with appropriate ToolTypes. All options are optional, unless otherwise stated.

The following are valid options:

REQUIRED	OPTIONAL
Mail	XPos
NoMail	YPos
OldMail	ClickCommand
MailBox	ClickStack
IconName	DropCommand
	DropStack
	BeepLots
	CountDelete
	Time
	MailRdy

## 1.4 mail

Mail (required) -

This is the name of the icon you wish to be shown to indicate that you have new mail. (Name is WITHOUT the ".info" extension)

## 1.5 nomail

NoMail (required) -

This is the name of the icon you wish to be shown to indicate that you do not have new mail. (Name is WITHOUT the ".info" extension)

## 1.6 oldmail

OldMail (required) -

This is the name of the icon you wish to be shown to indicate that you have old mail. (Name is WITHOUT the ".info" extension)

---

## 1.7 mailbox

MailBox (MB) (required) -

This is the name of your mailbox file.

## 1.8 iconname

IconName (IN) -

This option controls what is printed under the AppIcon. If this option is omitted, the AppIcon will be named "MailBox".

## 1.9 xpos

XPos -

XPos is the horizontal position, in pixels, where you want the icon to appear. If left out, the icon will be free floating in the X direction.

## 1.10 ypos

YPos -

YPos is the vertical position, in pixels, where you want the icon to appear. If left out, the icon will be free floating in the Y direction.

## 1.11 clickcommand

ClickCommand (CC) -

ClickCommand is the command that gets executed when the AppIcon is double-clicked.

## 1.12 clickstack

ClickStack (CS) -

ClickStack is the stack that ClickCommand is run with. If ClickCommand was omitted, ClickStack has no effect.

---

## 1.13 mailcommand

MailCommand (MC) -

MailCommand is the command that is executed when new mail arrives.

## 1.14 mailstack

MailStack (MS) -

MailStack is the stack that MailCommand is run with. If MailCommand was omitted, MailStack has no effect.

## 1.15 dropcommand

DropCommand (DC) -

DropCommand is the command that is executed when icons are dropped on the AppIcon. This command is executed once for each icon dropped, and the name of the Icon is appended to the DropCommand.

## 1.16 dropstack

DropStack (DS) -

DropStack is the stack that DropCommand is run with. If DropCommand was omitted, DropStack has no effect.

## 1.17 beeplots

BeepLots -

Normally, Sabot executes MailCommand only once, when the first new mail message arrives. If you want MailCommand to be run for each new mail message, then use the BeepLots switch.

## 1.18 countdelete

CountDelete -

Some mail programs delete the mailbox file when there are no messages remaining, while others leave a zero length file. If

---

your mailer deletes the file, then specify the CountDelete switch so that Sabot will know that an empty file means there are no messages, not that the mailbox just isn't completely updated yet.

## 1.19 time

Time - (Sabot-Timer only)

This is the amount of time, in minutes that Sabot waits between looking at the mailbox for changes. Default is to check every 5 minutes.

## 1.20 mailrdy

MailRdy -

Many mail utilities will send a message to the public port "T:MailRdy" when new mail arrives. If you wish to have Sabot create a message port with this name, use the MailRdy switch. When Sabot receives a signal at the T:MailRdy port, it will change the icon to indicate that new mail has arrived.

## 1.21 thingies

Things you should know-

I tried to make Sabot as small and simple as possible. Sabot does everything "by the book", or at least it tries to.

Commands that Sabot executes are run asynchronously, so if you drag 100 icons onto Sabot's AppIcon, it will start 100 new processes. That can make for a really slow system. ;-) Be careful, kids.

Sabot tries to be intelligent with its icon switching, but there is a limit. If the mailbox file size grows by less than 130 bytes, Sabot won't change the icon. This is to allow for mail programs, such as AmigaElm to add a few characters to the mail file, in order to mark messages as old, answered, etc. If you read your mailbox using a program not started by Sabot, and you don't actually edit the file, then Sabot cannot tell that you read it, and won't change the icon.

AmigaElm, when writing back the mailbox, writes to a temp file, deletes the original mailbox, then renames the temp file to the mailbox name. This small period when the mailbox is deleted causes some problems with Sabot. To eliminate these problems Sabot is now optimized to work with AmigaElm, and knows to ignore a non-existent file, but some mailers (dmail among them) would cause Sabot to not go back to a NoMail state because of this. So please specify the "COUNTDELETE" flag when using a mailer such as dmail.

---

Some filesystems, especially networked ones like Envoy, Parnet, EnlanDFS, and NFS do not support the file notification feature that Sabot relies on. In order to use Sabot with a mailbox on such a filesystem, you need to use the Sabot-Timer version, and specify how often to check the mailbox. Please note that the Sabot-Timer version may have a few quirks that make it slightly different from the "normal" version, such taking more CPU time (It basically takes as much as if you got new mail every 5 minutes with the "normal" Sabot). This is unavoidable.

## 1.22 uses

Suggested uses for Sabot -

I suggest having Sabot set up in the following manner:

ClickCommand runs your mailreader.  
DropCommand runs sendmail, to your favorite person.  
MailCommand plays a sample.

I've found these settings to be most useful.

## 1.23 thanks

Thanks To -

Michael B. Smith     Michael provided lots of help in writing Sabot, and was also kind enough to include it in his InetUtils package.

Klaus Alexander Seistrup     Klaus gave me some good ideas of changes to make to Sabot, and also gave me some nice MagicWB-looking icons to use with Sabot.

Colin Stuckless     Colin filled in the 1/4 of the globe that was hidden in Klaus' icons, so I could make new icons.

Kenneth Fribert     Suggested the OldMail idea first.

Oleg Rovner     Made sure these docs are readable. ;-)

Eric Sommer     Hacked Oleg's readable docs into this AmigaGuide database.

Christopher A. Wolf     Provided me with his great Commandline/Tooltype parsing code. Gotta love it. Thanks Chris!

Miscellaneous other people on IRC, who gave me ideas, and tidbits of code.

---

## 1.24 author

About the Author -

The author is...

Name: Scott Ellis  
E-mail: sellis@ucssun1.sdsu.edu  
Snail: 3640 Mt. Everest Blvd., San Diego, CA, 92111 (USA)

Feel free to contact me if you have any problems or suggestions.

P.S. It's called "Sabot" because I really liked when Kim Cattrall gave her ←  
little  
speech about sabotage in Star Trek VI: The Undiscovered Country. ;-)

## 1.25 history

History -

1.3 - First Public Release with InetUtils V1.1

1.4 - Added MailRdy option.  
(Suggested by Michael B. Smith)

Upped "ignored filesize change" to 200 bytes.

Added icon positioning (XPos and YPos) option.  
(suggested by Klaus Alexander Seistrup)

Removed redundant if(), and reorganized a bit of that code.

Reorganized compare(), to reduce the number of icon image changes.

1.5 - Fixed silly bug where the "HaveMail" icon was updated, even if it was  
already being shown.

1.6 - Reduced code size by changing linking with amiga.lib before sc.lib.

Added "OldMail" icon.  
(suggested by Kenneth Fribert)

Fixed bug where mailbox size wasn't getting set if the mailbox was deleted.

Added "CountDelete" tootype to help with deleted mailboxes.

1.7 - Sabot now shows "OldMail" if the mailbox is not empty when Sabot is ←  
started.  
Thanks to the hordes of people who reported this "bug". It was actually ←  
just  
an oversight.

1.8 - Integrated new ParseParameters() into Sabot. Reduced code size even more.

---

Created a new "Sabot Timer" that uses a polling method rather than file notification. Integrated it's source with the other version. To recreate the polling version, define "TIMER", or make "timer".

Documentation now in AmigaGuide format, thanks to Eric Sommer.

Put quotes around filenames that are dropped onto the AppIcon. This fixes problems with files from RAM: and files that have spaces in their names.

---