AM

MultiUser Account Manager
Version 1.1, released on 20.2.95

by Ingolf Koch

Overview

Are you running ↑`MultiUser`, the Multi User Security System for the Commodore Amiga? Then you surely have thought of a program providing an easy way to maintain the users and groups in your system.

Of course, such a program should have a graphical user interface which allows doing all necessary operations by a few mouse klicks or key strokes. Additionally, it should be localized i.e. speaking to you in your preferred language.

Then `AM` is the program you want. It was written to support the system operator's work by supplying an easy-to-use user interface based on ↑`MUI` (Magic User Interface) by Stefan Stuntz.

The following gives you a quick overview on `AM`'s facilities:

- localized (see ↑`Localization`)
- needs and makes extensive use of MUI 2.0 (`muimaster.library` version 7+)
- supports the following operations on the ↑`MultiUser` system files
  - create new users/groups
  - edit any characteristics of an existing user/group
  - delete users/groups
  - temporarily ban a user from the system
  - edit the '`MultiUser.config`' file
- automatically creates new $HOME directories and installs default files there by executing a user supplied script
- checks $HOME directories
- searches ↑`MultiUser` partitions for files with bad owner information and changes them
- supports ↑`AmiTCP office`, `work phone`, and `home phone` entries
- AmigaGuide® ↑`Online Help` system
- `ReadArgs()` and ToolTypes argument handling
- ↑`ARexx port`

Legal Stuff

There's not very much to say about this topic:

This program may be freely distributed i.e. without any charge except for a copying fee of maximum USD 5. It may not be included in commercial programs without prior written consent by the author.

This program may be included in shareware or public domain libraries as long as the price per disk does not exeed USD 5.

'AM 1.1' may be stored and made accessible in electronic networks such as the Internet, on anonymous ftp servers and in bulletin board systems.

'AM 1.1' may be stored on CD-ROMs (especially on "Fresh Fish", "Meeting Pearls", and "Aminet" CD-ROMs).

Storage and use of this program is only permitted for non military purposes, i.e. this program may not be used by military institutions or by non-military institutions co-operating with military institutions.

Although AM 1.1 has been thoroughly tested on various platforms there might still be some bugs in it. So use this program at your own risk. The author of AM 1.1 may not be made responsible for any damage caused directly or indirectly by AM 1.1.

1 Requirements

There are a few things you should have installed when you want to start `AM`:

- Of course, you need ↑`MultiUser` (Multi User Security System by Geert Uytterhoeven) :−)

- This implies that another requirement (AmigaOS 2.0+) is automatically satisfied.

- `AM` makes extensive use of the new features of ↑`MUI` 2.0. So make sure you have at least version 7 of `muimaster.library`. ↑`MUI` (Magic User Interface) is © by Stefan Stuntz (see the file '`ReadMe.mui`' for more info).

- If you want to use the locale facilities, you need at least AmigaOS 2.1.

- The partition scan function could use a lot of stack if you have a very deep directory tree. The minimal stack for `AM` can be roughly calculated by the formula (only valid for AmigaOS 3.0— might be less for 2.0 and 2.1):

$$min\_stack = \max(5000\,,\ 900 + 500 \cdot max\_depth)\,.$$

- To open the ↑`Options Window`, you need a screen with at least about 300 rows (or a very tiny font). Due to a bug in `muimaster.library` version 7, `AM` will hang if a window cannot be opened. This has been fixed since version 8 (thanks, Stefan).

- Although not required, it is recommended that you have at least a 68020 processor. The GUI would be very slow otherwise ;−)

- `AM` should be owned by `root` and can be made readable and executable for everyone. But *never* set the `u` flag!!!

2 Getting Started

You can start `AM` both from Workbench and CLI. To modify its standard behaviour see ↑`ToolTypes` and ↑`Command line arguments`.

If you start from CLI make sure you have enough stack before running `AM`. In this version, `AM` will not check and automatically increase stack size.

If `locale.library` is present in your system, `AM` will try to open the specified catalog (if you start from CLI and supply a language name) or the default language (set by Workbench's '`Prefs/Locale`').

If there's no `locale.library` or the desired catalog could not be opened, `AM` will talk to you in English (the builtin language).

Please refer to the chapter on ↑`Localization` for information on which languages are supported or on what to do if you want to translate `AM`.

After opening `locale.library`, `AM` checks if you are `root`. If not, you are prompted to enter `root`'s password. Note that `AM` refuses to work if you supply a wrong password :−)

If you are running `AM` for the first time (i.e. there's no '`ENV:mui/MU_AM.cfg`' file) the GUI will open with the default layout. You may want to change the layout and save it permanently (see ↑`Edit/Save` and your '`Prefs/MUI`' doc).

`AM` automatically finds your ↑`passwd` and ↑`MultiUser.group` files and loads them into its memory.

All error messages are localized, so if there's an error during startup, you'll see the reason in your preferred language (if there's a suitable catalog).

Have a look at the ↑`Requirements` chapter for more information on what is needed to run `AM`.

2.1 Starting from Workbench

`AM` supports some ToolTypes:

- `LANGUAGE=`*language name*: Try to open the catalog for the specified language
- `PASSWDFILE`: Indicate that this Workbench project is a ↑`passwd` file
- `GROUPFILE`: Indicate that this Workbench project is a ↑`MultiUser.group` file
- `CONFIGFILE`: Indicate that this Workbench project is a ↑`MultiUser.config` file
- `LOGFILE`: Indicate that this Workbench project is an ↑`am.log` file
- `HELP=`*online help filename*: Use a different ↑`Online Help` file name (`%s` still supported)

Note: `PASSWDFILE`, `GROUPFILE`, `CONFIGFILE`, and `LOGFILE` do not take arguments. They determine the type of the file belonging to that project icon. *Never put these ToolTypes into program icon for* `AM`.

You can multiselect various project icons to tell `AM` to use them instead of the corresponding default files.

If you already have icons for files created by revision 1.0 of `AM` you should either delete these icons or add the appropriate tool type entry.

2.2 Starting from CLI/Shell

Template: *Language, P=PasswdFile/K, G=GroupFile/K, C=ConfigFile/K, L=LogFile/K, H=HelpFile*

- *Language*: Try to open the catalog for the specified language
- *PasswdFile*: Use this file as the ↑`passwd` file
- *GroupFile*: Use this file as the ↑`MultiUser.group` file
- *ConfigFile*: Use this file as the ↑`MultiUser.config` file
- *LogFile*: Use this file as the ↑`am.log` file
- *HelpFile*: Use this file as the ↑`Online Help` file (`%s` still supported)

Remember to set a sufficient stack size. See ↑`Requirements`.

3 The main window

If the system libraries are opened successfully `AM` opens its main window. This window consists of various areas:

3.1 The Users and Groups listviews and buttons

After loading the ↑`passwd` and ↑`MultiUser.group` files, the users and groups known to the system are displayed in two listviews. The left listview contains the users whereas you find the groups in the right listview.

For both users and groups, each entry in the according listview shows the UserID and GroupID as well as the UID and GID, respectively. The UID/GID will only be displayed, if there's enough space in the listview.

You may scroll the listviews via the propgadget or the keyboard. In the latter case, use the `TAB` key to step through the main window's `CycleChain` until the appropriate listview is the active one. To scroll up and down, hit the corresponding cursor key.

Below each of the listviews there are a few buttons which allow modification of the users/groups list. In most cases they operate on the currently activated user/group, so make sure you have an active entry in the corresponding listview.

To activate an entry, click once on the entry in the listview or use the cursor keys.

3.1.1 Creating a new user

Select the `NEW` button to create a new user. The ↑`Users Window` will open with most of the entries cleared. The UID entry already contains a valid number, namely the lowest available UID in the system (not currently occupied by another user)

You have to supply at least a valid ↑`UserID`, a valid ↑`UID (User)` and a valid ↑`Primary GroupID` to successfully add the new user to the system. The ↑`Primary GroupID` will initially be set to the value you supplied in the ↑`Default Primary Group` string gadget.

See ↑`Edit (User)`.

3.1.2 Editing a user's characteristics

If you press the `EDIT` button at the bottom of the `Users` listview in the ↑`Main Window` the
↑`Users Window` opens displaying the present characteristics of the selected user.

You may modify each of them but keep in mind that changing the ↑`UserID` automatically clears
the user's ↑`Password` and that changing the user's ↑`UID (User)` makes a ↑`Scan` of the whole file
system necessary to update.

(You may want to ↑`Scan` after changing the ↑`Primary GroupID`, too.)

3.1.3 Deleting a user

Pressing the `DELETE` button will automatically remove the selected user from the `Users` listview.

Caution: You are not asked to confirm this action!

If you find it necessary to open a confirmation requester before actually deleting the user, please
inform me about that (see ↑`Author`).

Strictly speaking, the user is not totally removed from the system at all. He remains in the
↑`passwd` file until you ↑`Save` your configuration. Additionally, the deleted user is kept in `AM`'s
internal memory because this information is needed when you ↑`Scan` the file system.

3.1.4 Temporarily disabling a user

Sometimes you might find it necessary to forbid a user's access to the system without deleting
information on this user. Usually, this is done by changing this user's password.

`AM` changes the encrypted password string of the selected user to `User_banned`. As it is (nearly)
impossible to find a password which will match this magic string after encryption, this is quite a
safe way of banning.

If `AM` detects a user whose encrypted password matches `User_banned` during ↑`Open` this user
will automatically be marked as banned.

Banned users are surrounded by asterisks * in the Users listview.

Note: If you ↑`Clear Password` a banned user's password the ban will be released.

Note: `AM` only remembers the original password until you leave the program or ↑`Open` the files. So if you ↑`Admit` the user again, the original password can only be restored if you did not leave `AM` or ↑`Open` the files before.

3.1.5 Releasing a ban

To permit a banned user to access the system again, click on the `ADMIT` button. In most cases the selected user's password will be cleared, so access is made possible after the next ↑`Save` operation.

The only exception of this rule is made if you did not ↑`Open` or ↑`Quit` between the ↑`Ban` and the `ADMIT` operations. If you by mistake ↑`Ban` a user you may safely click on `ADMIT` without destroying the user's password.

Note: Remember that the user should set his password immediately after ↑`Save`. You must reload the files to let `AM` know the new password.

3.1.6 The `Users` Window

The `Users` window is divided into two parts: On the lefthand side there are the user's main characteristics, on the righthand side you find the list of the user's secondary groups.

Let's turn to the `Characteristics` part of this window. You may modify each of the entries as you like. Well, to be honest... nearly as you like (see ↑`UserID`, ↑`UID (User)`, ↑`Clear Password`).

3.1.6.1 The `UserID` entry

Every user *must* have a valid `UserID`. That's the name you use to login. Of course, the `UserID` must be unique. So `AM` does not allow you to specify a `UserID` which already exists.

Note: If the user has a ↑`Password` set and you modify his `UserID` the `password` will be cleared, as it is dependant on the `UserID`.

Some characters, for example |, are not allowed for a `UserID`. These will be rejected automatically by ↑`MUI`.

3.1.6.2 The user's password

If the user has a password set it will be displayed in the password text object. This is *not* the real password but the encrypted 11 character string (as it is not possible to decrypt it — better say: it's very difficult to decrypt it :−).

There are three ways for you to manipulate the password string:

- Clicking on the ↑`Clear Password` button will remove the password. *Now everyone can login as this user!*
- Modifying the ↑`UserID` clears the password, too. *Again, everyone can login as this user!*
- If you ↑`Ban` a user from the system the password will be set to the magic string `User_banned`. Remember that after clicking on ↑`Admit` the password will be cleared!

If there is a user with no password supplied you'll be informed about that in the ↑`am.log` when you ↑`Save`.

3.1.6.3 The user's UID

The UID is a number in the range [1..65535]. You *must* supply a valid (i.e. in the correct range and unique) UID for every user in the system.

If you try to set a wrong UID you'll be informed about that. Currently, the only way to see which UIDs are available is to look at the `Users` listview (see ↑`Listviews`).

If you change the UID of a user who is already known to the system (i.e. there's an entry for that user in the ↑`passwd` file) you should ↑`Scan` the ↑`MultiUser` formatted partitions on your hard drive, since the information on the files' owners must be updated.

Modifying a user's UID makes sense if you want to change the users hierarchy in your system imposed by the UIDs.

Unlike ↑`UserID`, modifying the UID does not clear the ↑`Password`.

3.1.6.4 The user's primary group

Every user belongs to at least one group in the system. The main group is also called `primary group`. You *must* supply a valid GroupID for the user's primary group.

To make this easier the GroupID may be entered by a `popup object`. Just click on the popup button to make a listview of currently available GroupIDs appear. You can then select the GroupID you wish by double clicking on the according entry in the listview.

Of course, you can enter the GroupID "by hand" in the string gadget. If you mistype the GroupID `AM` will refuse this string and open the listview automatically.

If the new primary group is already in the list of secondary groups it will be automatically removed from that list.

3.1.6.5 The user's real name

In this string gadget, you should supply the user's real name. You may use (nearly) any characters except | and , as these are the separators in the ↑`passwd` file.

3.1.6.6 The user's office

This string gadget is mostly intended for ↑`AmiTCP` users. You can enter the user's office here.

3.1.6.7 The user's work phone number

This string gadget is mostly intended for ↑`AmiTCP` users. It should contain the user's phone number at his/her place of work.

3.1.6.8 The user's private phone number

This string gadget is mostly intended for ↑`AmiTCP` users. It should contain the user's phone number at home.

3.1.6.9 The user's $HOME directory

The user's home directory ($HOME) is specified here. You may enter any syntactically correct path here—in other words: the path needn't exist. If you set the ↑`Set up new $HOMEs` switch non existing $HOMEs will be created and the necessary files will be installed there after ↑`Save`.

If you use the ASL file requester to select the $HOME directory `ASL` automatically appends a slash `/` to that string. You needn't worry about that as `AM` removes trailing slashes automatically.

Via the ↑`Check $HOMEs` switch you can tell `AM` if you want to be informed about $HOMEs which are not located on ↑`MultiUser` partitions.

See also ↑`Default Home`.

3.1.6.10 The user's default shell

You may enter the user's default shell here. Future versions of `AM` will supply a popup object for this.

3.1.6.11 Clear the user's password

Clicking on this button clears the password string. Since a user without a password is in a dangerous state you'll be warned about missing passwords in the ↑`am.log`.

3.1.6.12 Add a new secondary group

There are two ways of adding a new secondary group to the list:

- Type the group's name in the string gadget and hit `RETURN`
- Click on the popup button and select the new group by double clicking on the entry in the popup. (This is the more comfortable way.)

If the secondary group is already in the list or if it is the primary group nothing will happen.

If you supply an invalid GroupID a requester pops up and nothing will be added.

3.1.6.13 Delete a secondary group

Click on this button to remove the currently active secondary group from the listview. 'Active' means: the selected entry in the listview.

Note: There's no `UNDO` button. If you delete a secondary group by mistake, either reenter this GroupID or choose ↑`Cancel (User)` and start editing again.

3.1.6.14 Leaving the Users window by clicking on `OK`

The `OK` button is one way to leave the Users Window. If all important settings are valid (i.e. ↑`UserID`, ↑`UID (User)`, and ↑`Primary GroupID`) the new values are stored in `AM`'s memory.

If `AM` finds an invalid value it will complain and refuse to close the window.

3.1.6.15 Leaving the Users window by clicking on `CANCEL`

If you click on `CANCEL` all changes will be ignored, and the Users window is closed.

3.1.7 Creating a new group

Select the `NEW` button to create a new group. The ↑`Groups Window` will open with most of the entries cleared. The GID entry already contains a valid number, namely the lowest available GID in the system (not currently occupied by another group).

You have to supply at least a valid ↑`GroupID`, a valid ↑`GID (Group)` and a valid ↑`Manager UserID` to successfully add the new group to the system. The ↑`Manager UserID` will initially be set to the value you supplied in the ↑`Default Group Manager` string gadget.

See ↑`Edit (Group)`.

3.1.8 Editing a group's characteristics

If you press the EDIT button at the bottom of the Groups listview in the ↑Main Window the ↑Groups Window opens displaying the present characteristics of the selected group.

3.1.9 Deleting a group

Pressing the DELETE button will automatically remove the selected group from the Groups listview.

Caution: You are not asked to confirm this action!

If you find it necessary to open a confirmation requester before actually deleting the group, please inform me about that (see ↑Author).

Strictly speaking, the group is not totally removed from the system at all. It remains in the ↑MultiUser.group file until you ↑Save your configuration. Additionally, the deleted group is kept in AM's internal memory because this information is needed when you ↑Scan the file system.

3.1.10 The Groups Window

This window displays the group's Characteristics. You may modify each of the entries as you like. Well, to be honest... nearly as you like (see ↑GroupID, ↑GID (Group)).

3.1.10.1 The GroupID entry

Every group *must* have a valid GroupID. That's the group's symbolic name. Of course, the GroupID must be unique. So AM doesn't allow you to specify a GroupID which already exists.

Some characters, such as |, are not allowed for a GroupID. These will be rejected automatically by ↑MUI.

3.1.10.2 The group's GID

The GID is a number in the range [0..65535]. You *must* supply a valid (i.e. in the correct range and unique) GID for every group in the system.

If you try to set a wrong GID you'll be informed about that. Currently, the only way to see which GIDs are available is to look at the `Groups` listview (see ↑`Listviews`).

If you change the GID of a group which is already known to the system (i.e. there's an entry for that group in the ↑`MultiUser.group` file) you should ↑`Scan` the ↑`MultiUser` formatted partitions on your hard drive, since the information on the files' owners must be updated.

Modifying a group's GID makes sense if you want to change the groups hierarchy in your system imposed by the GIDs.

3.1.10.3 The UserID of the group's manager

Every group needs to have a manager responsible for this group. So you *must* supply a valid UserID for the group's manager.

To make this easier the UserID may be entered by a `popup object`. Just click on the popup button to make a listview of currently available UserIDs appear. You can then select the UserID you wish by double clicking on the according entry in the listview.

Of course, you can enter the UserID "by hand" in the string gadget. If you mistype the UserID `AM` will refuse this string and open the listview automatically.

3.1.10.4 The group's real name

In this string gadget, you should supply the group's real name. You may use (nearly) any characters except | as this is the separator in the ↑`MultiUser.group` file.

3.1.10.5 Leaving the Groups window by clicking on `OK`

The `OK` button is one way to leave the Groups Window. If all important settings are valid (i.e. ↑`GroupID`, ↑`GID (Group)`, ↑`Manager UserID`) the new values are stored in `AM`'s memory.

If `AM` finds an invalid value it will complain and refuse to close the window.

3.1.10.6 Leaving the Groups window by clicking on `CANCEL`

If you click on `CANCEL` all changes will be ignored, and the Groups window is closed.

3.2 The Info display

This frame displays information on what `AM` is doing.

If something has gone wrong you'll be informed about that here. Usually a requester will pop up first in this case, but you are reminded of the error by the error message displayed in the Info display.

3.3 Buttons in the main window

There are three buttons which start some of `AM`'s operations: `OPEN`, `SAVE`, and `SCAN`.

3.3.1 Load the passwd and group file

If you select the `OPEN` operation `AM` will check first whether the configuration is modified and should be saved first. If so a requester pops up and you may answer that you really want to load or that you've decided to abort.

All old information on users and groups (deleted ones, too) is discarded from the memory.

See also ↑`ARexx Open`.

3.3.2 Save the passwd and group file

AM does not just save the files. It checks first if the configuration contains serious errors. AM does not save if there was a load error before or if there's no user with UID 65535 (no root!).

In some situations AM will warn you (and you are asked if you want to proceed):

- If the user having the root UID has been changed
- If root's primary group's GID is not equal to 65535
- If you have modified one of the file names (see ↑`ARexx port`)

If you set the ↑`Backup to *.bak` switch in the ↑`Options Window` the old ↑`passwd` and ↑`MultiUser.group` files will be moved to '*filename*.`bak`' before writing anything. In case of a write error during the `SAVE` operation, AM will then do its best to restore the old files.

In some cases it may be necessary to ↑`Scan` after saving—if UIDs or GIDs have been changed. In those cases AM informs you about that.

See also ↑`ARexx Save`.

3.3.3 Scan MultiUser partitions

This is probably the most complex operation provided by AM. It is used to update the file system if you change certain settings, for example the UID. In such cases there might exist files on your ↑`MultiUser` formatted partitions which don't have a valid OwnerUID or OwnerGID.

Therefore AM opens a new window (the `Scan` window) and scans the whole directory tree (or better say 'trees' if you have more than one ↑`MultiUser` partition). All files which need to be updated are collected and displayed in the ↑`Scan Listview`. You can then select the files you want to modify and tell AM what to do with them.

While scanning, AM disables some buttons. Don't worry: They'll be enabled again when the scan is finished. The current directory is displayed in the text object surrounded by the `Scanning` frame. So you have an idea of where AM currently is and how long it will take to scan the rest. See ↑`Requirements`.

If you don't want to wait till `AM` has finished the scan you may click on the `ABORT` button. The scan process is aborted but you can modify the files already found.

If you click on the `BACK` button instead, the scan will be aborted as with `ABORT` but additionally the `Scan` window will be closed.

Each file displayed falls into exactly one of five categories which is shown in the ↑`Scan Listview`, too:

`Nobody:`	owned by Nobody (`OwnerUID = 0`)
`Deleted:`	owned by a user whom you deleted before
`Orphan:`	the file's OwnerUID is not known to the system
`UID:`	the UID of the file's owner was changed
`GID:`	the file's OwnerGID is different from the owner's primary GID

With the five buttons you can select/deselect the corresponding files in the ↑`Scan Listview` all at once. To explicitly select/deselect a single file click on the entry in the ↑`Scan Listview`.

Before you `START` modifying the selected files you should click on the ↑`Set Actions...` button to tell `AM` what to do with the selected files.

If everything is to your satisfaction use the `START` button to modify the selected files. Correctly processed files are removed from the ↑`Scan Listview`.

When the conversion is finished `AM` does not return to the ↑`Main Window` but lets you select other files, modify the actions by ↑`Set Actions...`, and process the newly selected files in a different way.

At any time you can return to the ↑`Main Window` by clicking on the `BACK` button.

Note: Scanning is only possible if `AM`'s internal memory reflects the contents of the *real* ↑`passwd` file used by ↑`MultiUser`. You could run into serious problems otherwise.

3.3.3.1 The listview in the Scan window

All files to be modified found during a scan of the ↑`MultiUser` partitions are displayed in the Scan listview. This listview consists of three columns namely the file name, the file's type, and the directory where the file is located.

The file's type is either ↑`Nobody`, ↑`Deleted`, ↑`Orphan`, ↑`UID`, or ↑`GID` and tells you about the reason why the file appears in the listview.

This listview is multiselectable, i.e. you may select as many entries at the same time as you like. You may select entries either by using the `File Selection` buttons or by mouse clicking on the corresponding entry. You need not use the multi select qualifier (such as the `SHIFT` key) for this multi selection.

3.3.3.2 The Nobody type

Files of type 'Nobody' are those with an OwnerUID of zero. Usually all files will be owned by Nobody if you've just installed ↑`MultiUser` in your system. So be prepared to have lots of entries if you run `AM` for the first time :−)

There may appear new 'Nobody' files later if someone creates a new file without logging in before.

3.3.3.3 The Deleted type

If you delete a user `AM` does not remove this entry from its internal memory. So `AM` is able to detect files which were owned by this deleted user and therefore don't have a valid owner now.

Note: Of course you must not leave `AM` between deleting the user and scanning. Otherwise `AM` would not be able to remember the deleted user's UID and the deleted user's files would be of type ↑`Orphan` or would even be owned by someone else who now has the original owner's UID!

3.3.3.4 The Orphan type

Orphan files are those files having an OwnerUID not equal to zero but without a user known to the system with this UID.

In most cases these files will appear if you delete a user from the system without scanning directly after that. After leaving `AM` no information about the deleted user can be recovered so files owned by that user are marked as orphan.

3.3.3.5 The UID type

AM allows you to change a user's UID. After doing so, all files owned by that user will no longer have a valid OwnerUID as this must be adjusted to the new value, too.

Note: Although you can give another user the UID 65535 (the root UID), under no circumstances files owned by root will change their OwnerUID as they should be owned by the new root user (i.e. keep the OwnerUID of 65535).

3.3.3.6 The GID type

Usually a file's OwnerGID should match its owner's primary GID. Files having a different OwnerGID will be added to the ↑`Scan Listview`, as you might have changed the owner's primary group and want to adjust the OwnerGID.

But there might be situations in which the different OwnerGID is intended to make the file accessible to members of a group different from the owner's primary group. Then you should keep such files deselected.

Quick note: This will be improved a lot in the future. At the moment, it might be better not to use this. . . Sorry.

3.4 The main menu

If you are in the ↑`Main Window` you can start some of AM's operations by selecting the appropriate menu item. But there's even more in the main menu:

3.4.1 The Project menu

From this menu you can select the ↑`Open`, ↑`Save`, and ↑`Scan` operations (the same as clicking on the corresponding button in the ↑`Main Window`), as well as open the ↑`Options Window`, get information about the ↑`Author` and the translator (`ABOUT...` item) and ↑`Quit` the program.

3.4.2 The Edit menu

This menu is very much like the standard 'Prefs' menu of the same name. AM automatically stores some of your settings in a private configuration file in ENV: when you ↑Quit the program. This file is loaded automatically during startup or explicitly (see below).

The saved/restored settings comprise the ↑AM Options and the ↑Set Actions... part.

You'll find the following items in the EDIT menu:

3.4.2.1 Load configuration from ENVARC:

Selecting the Edit/Last Saved item causes AM to read the configuration data from the file 'ENVARC:mui/MU_AM.cfg'.

3.4.2.2 Load configuration from ENV:

Selecting the Edit/Restore item causes AM to reread the configuration data from the file 'ENV:mui/MU_AM.cfg'.

This file is automatically loaded at startup and saved on ↑Quit.

3.4.2.3 Save configuration to ENVARC:

Selecting the Edit/Save item makes AM permanently save the configuration data to the file 'ENVARC:mui/MU_AM.cfg'.

There's no need for a menu item which saves to ENV: as this is automatically node on ↑Quit.

3.5 Quit the program

Pressing the QUIT button or selecting the QUIT item from the ↑Project menu terminates AM. Before really exiting AM checks if the files need to be saved and informs you about that by popping up a requester.

You can still continue to work with all information in `AM`'s memory at this point. If you decide to terminate `AM` all information on deleted users is discarded. See ↑`Deleted`.

4 The `Options` Window

Many ↑`MultiUser` and `AM` features can be controlled in this window. To distinguish between these two parts the window is divided into MultiUser.config Options and AM Options.

This window may be too tall for your screen (see ↑`Requirements` how to change this).

4.1 The '`MultiUser.config`' Options

The ↑`MultiUser.config` file contains some information on how ↑`MultiUser` should behave in varoius situations.

This file is automatically loaded into `AM`'s memory when you open the ↑`Options Window`. The current state of the switches are displayed in the left part of the window.

↑`MultiUser` provides a way of freezing some users' passwords. With the strings and sliders in the `Allow password change...` part you can select which users are allowed to change their password.

All users with a UID of the displayed number or higher as well as users whose primary GID is the displayed number or higher can change their password.

The `OPEN` button reloads the '`MultiUser.config`' file without checking if you have altered the state of some entries.

With the `SAVE` button you write the '`MultiUser.config`' file.

Both `OPEN` and `SAVE` use the filename you supplied in your ↑`Command line arguments` or in the ↑`ToolTypes` entry. The filename defaults to '*MultiUser_Group_Dir*/MultiUser.config'.

4.2 Some helpful features

In this part of the ↑`Options Window` you can control some of `AM`'s features. They belong to four different categories: $HOMEs, Files, Miscellaneous, and Defaults.

All these settings are saved automatically to 'ENV:' when you ↑Quit AM. If you want to store them permanently use the ↑Edit/Save menu item.

4.2.1 Set up new $HOME directories

If a $HOME directory does not exist when you ↑Save it will be created, and the script file '*MultiUser_passwd_dir*/am_create_home.script' will be executed (if it exists).

This script is executed after logging in as the new user. The current directory is the *MultiUser_passwd_dir*.

So you could create a '*MultiUser_passwd_dir*/default_home' directory containing all files to be installed in the new $HOME. If you have a MAssign HOME: %h running you could put a Copy default_home/#? HOME: CLONE ALL command into the 'am_create_home.script'.

In case you want to install files having a different owner just call (from script) Logout (you are root then), Login as the new user (you don't have to supply a password!), and proceed with the installation.

Note: Remember to make the 'default_home' directory and its contents readable by everyone.

Note: Due to a small bug in MultiUser, automatic creation of home directories might fail in some cases. For some reason, MultiUser does not react on the file notification caused by the write to the ↑passwd and ↑MultiUser.group files. So MultiUser does not recognize the new users, and AM cannot log in as these new users.

As a workaround, I suggest to delete the new home directory and ↑Save again until MultiUser knows about the new users.

4.2.2 Check $HOME directories

If set, AM will check during ↑Save if the users' $HOME directories are located on ↑MultiUser partitions.

4.2.3 Create icons for new $HOME directories

If set, `AM` puts an icon for newly created $HOME directories. Its image will be the default drawer icon. Its owner is the corresponding user, so he can adjust the dimensions of the drawer's window.

4.2.4 Create icons for the system files

If set, `AM` creates icons for the ↑`passwd`, ↑`MultiUser.group`, ↑`MultiUser.config`, and ↑`am.log` files which are owned and only accessible by root. Their image is taken from `AM`'s image, but they are `Project` icons with a default tool of `AM`. Other ↑`ToolTypes` are set according to the files' types.

4.2.5 Safe saves

If set, `AM` first moves the ↑`passwd` and ↑`MultiUser.group` files to '`*.bak`' when you ↑`Save`. So '`passwd.bak`' and '`MultiUser.group.bak`' represent the status before saving.

If there is an error during the write process, `AM` tries to recover from this situation by restoring the old contents from these '`*.bak`' files.

4.2.6 Turn on/off logging

This switch turns logging on/off. If selected, important information is appended to ↑`am.log` during ↑`Save` and ↑`Scan`.

4.2.7 Freezing the display

This switch is intended mostly for those of you who must work with slow machines. To make life a bit easier you can turn this switch off. When you ↑`Scan` the ↑`MultiUser` partitions the listview is not updated every time a new entry is added to that list.

Of course this will only save reasonable time if there are many files to be added.

Maybe this feature will be improved in future releases.

4.2.8 The default primary group

Every newly created user's ↑`Primary GroupID` will initially be set to the group you supply here.

4.2.9 The default group manager

Every newly created group's ↑`Manager UserID` will initially be set to the user you supply here.

4.2.10 The default $HOME directory

This string gadget describes the string which is copied to a newly created user's ↑`Home` field. For example you could enter here the directory in which all the $HOME directories are located.

4.2.11 The Actions window

After pressing the `SET ACTIONS...` button a window will open containing three register groups (↑`Nobody`, ↑`Deleted`, and ↑`Orphan`) with lots of buttons.

For each of these three types you can tell `AM` what to do with files which fall into this category.

Two types of Actions are supported:

- `CHANGE` means that the file's owner is changed to the owner given in the string gadget below, and the file's protection bits are modified as stated in the `Protection` part of this column.
- `DELETE` means that the file will be deleted. The owner string gadget and the `Protection` area are disabled in this case.

If you choose `CHANGE` you must supply the new owner's UserID in the string gadget. The most comfortable way is to use the popup. `AM` *will not close this window if there is an illegal UserID.*

In the `Protection` area you state how each of the file's protection flags should be modified. All ↑`MultiUser` supported flags are listed here.

- U: Only clearing and keeping is allowed here. *Keeping this flag unmodified is very very dangerous!!!* Default is 'Clear'.

– `SPA`: These flags may either be set, cleared or kept unmodified. Default is 'Keep'.

– `RWED` for owner: These flags may either be set, cleared or kept unmodified. Default is 'Keep'.

– `RWED` for group members or others: You have the choice between 'Set', 'Clear', 'Keep' and 'Owner'. If you select 'Owner' the corresponding flag of the owner's `RWED` part is copied. This is done *after* altering the owner `RWED` flags.

Use the `OK` button to close this window. Please note again that the string gadgets must contain valid UserIDs if they are not disabled. *If there is an invalid setting in one of the string gadgets for the new owner, `AM` does not close the window but displays the popup list for that entry!*

All settings in this window are saved automatically to '`ENV:`' when you ↑`Quit AM`. If you want to store them permanently (of course you want to do so :−) use the ↑`Edit/Save` menu item.

4.2.12 The Files requester

When you click on this button a requester pops up showing the file names currently in use for the following types of files:

4.2.12.1 The MultiUser passwd file

This is the file where nearly all information on the users is stored (all but the secondary group information). See the ↑`MultiUser` doc for more info.

This filename can be changed at run time by an ↑`ARexx ChPasswd` call.

4.2.12.2 The MultiUser group file

This is the file where all information on the groups is stored. See the ↑`MultiUser` doc for more info.

This filename can be changed at run time by an ↑`ARexx ChGroup` call.

4.2.12.3 The MultiUser config file

This is the file where you can control the various ↑`MultiUser` features. See the ↑`MultiUser` doc for more info.

4.2.12.4 `AM`'s log file

The log file informs you about some important things. The following information appears in the log file:

- The file names when you ↑`Save`
- Users who don't have a password set when you ↑`Save`
- Files modified by ↑`Scan`
- Errors encountered during ↑`Save` or ↑`Scan`

The default place for the log file is in the same directory as the passwd file (determined by a `muGetPasswdDirLock()` call).

See also ↑`Logging`.

5 AM's ARexx port

Besides the standard ↑MUI ARexx commands, AM understands some additional ones which are usually not needed very often as they are mostly intended for testing purposes:

The ARexx port's name is MU_AM.

If you find it useful to add more power to the ARexx port please contact the ↑Author.

5.1 The ARexx ChPasswd command

This command is used to change the name for the ↑passwd file at run time. If the new file already exists AM uses the full path to the file.

Note: This command does not load the new file into its memory.

Template: *FileName/A*

Example:

```
rx "ADDRESS MU_AM ChPasswd 'foo'"
```

Please note the single quotes around 'foo' preventing ARexx from changing it to uppercase 'FOO' which IMHO looks a bit PC-ish.

5.2 The ARexx ChGroup command

This command is used to change the name for the ↑MultiUser.group file at run time. If the new file already exists AM uses the full path to the file.

Note: This command does not load the new file into its memory.

Template: *FileName/A*

Example:

```
rx "ADDRESS MU_AM ChGroup 'bar'"
```

Please note the single quotes around 'bar' preventing ARexx from changing it to uppercase 'BAR' which IMHO looks a bit PC-ish.

5.3 The ARexx `Open` command

Sending this command to the `MU_AM` port is the same as clicking on the ↑`Open` button in the ↑`Main Window`.

5.4 The ARexx `Save` command

Sending this command to the `MU_AM` port is the same as clicking on the ↑`Save` button in the ↑`Main Window`.

6 AM's online help system

Since AmigaOS 3.0 (V39+), `intuition.library` provides an online help support: Place the mouse pointer over a portion of the window (a button, string, or even the background) and press the `HELP` key on your keyboard.

If you supplied a valid help file (and have at least AmigaOS 3.0) you can make use of this feature to get quick help. `AM` then searches for the help file in the following order:

```
(1) the user supplied help file (if any)
(2) 'PROGDIR:docs/%s/am1_1.guide'
(3) 'HELP:%s/am1_1.guide'
(4) the user supplied help file (if any); %s replaced by english
(5) 'PROGDIR:docs/english/am1_1.guide'
(6) 'HELP:english/am1_1.guide'
```

In (1), (2), and (3), the `%s` is replaced by the name of the language you want `AM` to run in. See also ↑`ToolTypes` and ↑`Command line arguments`.

7 Information on `AM`'s development

7.1 Short information on the author

`AM` was written by:

```
Ingolf Koch
Wellenkampstraße 38
D−32791 Lage
Germany
Phone: 05232/2700
```

If you call from other countries replace the leading 0 by the appropriate prefix for Germany.
My email address is `ingolf@mathematik.uni-bielefeld.de`.
Additionally, I can often be found on IRC. My nick name is `Balin`.

Please note that my email address will change in March or April '95 as I will probably move to
Jena. If you want to be informed about my new address let me know!

And don't forget to send me a postcard if you like this program!

7.2 The development crew

There are quite a few people who had great influence on `AM` being what it is now. On the one
hand, there are the translators:

- **Paul Dolk** (Dutch catalog)
- **Håkon Enger** (Norwegian catalog)
- **Christofer Kull** (Swedish catalog)
- **Gaël Marziou** (French catalog)
- **Tomasza Nideckiego** (Polish catalog)
- **Goncalo Valverde** (Portuguese catalog)

They also did a lot of testing (and bug reporting :−). Thank you! (The German 'translation'
was done by the ↑`Author` himself.)

On the other hand, there are additional testers who helped me very much with their bug reports, criticism, and suggestions (and the list of suggestions is growing and growing. . .):

- **Reza Elghazi**
- **Walter Francis**
- **Michael Hohmann**
- **Krunoslav Njers**
- **Bernhard Seefeld**
- **Horst Weber**

Thanks again for your work, guys.

7.3 Lots of thanks

Besides the `AM` development ↑`Crew` I'd like to thank the following people

- Geert Uytterhoeven for ↑`MultiUser`
- Stefan Stuntz for creating ↑`MUI`, answering lots of questions and bearing my frequent (positive and negative) criticism
- Matt Dillon for DICE
- Markus Illenseer (ill) and other people on IRC (#AmigaGER) for their help
- Gaël Marziou for YAK
- Stefan Becker for ToolManager

I'm sure there are many more people I should mention here but they don't come to my mind just now. . .

7.4 Known bugs

There are some bugs I'm aware of. They are not really caused by `AM` but can make the work with `AM` difficult in some cases.

- There is a bug in version 7 of `muimaster.library` making `AM` hang if a window could not be opened. This has been fixed since version 8. (See ↑`Requirements`.)

- Sometimes ↑`MultiUser` does not recognize that the ↑`passwd` or ↑`MultiUser.group` files have been changed. In this case ↑`Set up new $HOMEs` does not work properly.

If you find additional bugs: don't hesitate to inform the ↑`Author`.

7.5 Program history

```
1.1:    20-Feb-95

        * New: "Check $HOMEs" now works
        * New: Options window replacing and extending the Options menu
        * New: MultiUser.config supported
        * New: new entries "office", "work phone", and "home phone" (AmiTCP)
        * New: Tooltypes supported
        * New: ReadArgs() argument utilization
        * New: Online AmigaGuide® help system
        * New: Some own ARexx commands supported (ADDRESS MU_AM)
        * Safer saves (*.bak file support)
        * Description for am_create_home.script added to the doc
        * Set Actions... window now with register groups

        - Bug fixed: Popup button in User window was not disabled correctly
        - Bug fixed: Popups sometimes were not correctly reenabled


1.0:    29-May-94

        * First public release
```

8 Translating `AM` to another language

Currently, there are eight languages available for `AM`: English (the builtin), Dutch, French, German, Norwegian, Polish, Portuguese, and Swedish (see ↑`Crew`). If you'd like to translate `AM` to another language, please fill in the translated strings in the blank '`am.ct`' file which comes with this distribution.

In many cases, a look at the catalog descriptor ('`am.cd`') will help you translate the strings in the right way. You can then compile the new catalog using Commodore's `CatComp` utility (or similar).

But please don't spread your private catalog versions! If you have a new catalog or just want to improve an already existing one, you should contact the ↑`Author` of `AM`.

You can even translate the documentation. It is written in `Texinfo`, a simple TeX dialect. Via `makeinfo`, it can be converted to an AmigaGuide® file, but you can also run TeX with this file to produce a printed document.

Your translation should be based on the English '`am1_1.texinfo`' file. There are lots of hints for the translation and compilation in it.

9 Information on related products

9.1 Short information on MultiUser

MultiUser is a (or better say 'the') Multi User Security System. It was created by Geert Uytterhoeven for the Commodore Amiga computer. It works with AmigaOS 2.0+ and is implemented as a normal DOS file system. So it works with (nearly) all programs.

For more information see the '`MultiUser.doc`' which comes with the MultiUser distribution.

9.2 Short information on MUI

MUI is an abbreviation for Magic User Interface. It was created by Stefan Stuntz for the Commodore Amiga computer. To utilize MUI at least a 68020 or better is recommended, your computer will be very slow otherwise. . .

For more information see the '`ReadMe.mui`' which comes with this distribution of `AM`.

9.3 Short information on AmiTCP

AmiTCP is freely distributable TCP/IP networking package (better say: was—from version 4.0 it isn't free any longer). In many cases ↑`MultiUser` is used in combination with AmiTCP. But unfortunately there are (still) differences in the passwd and group file formats.

Currently `AM` supports only the ↑`MultiUser` formats, but you may set ↑`Office`, ↑`Work phone`, and ↑`Home phone` information for a user (which is a feature of AmiTCP).

`MultiUser 2.0` will have a passwd format compatible with AmiTCP. So future versions of `AM` will be usable both for MultiUser and AmiTCP.

10 Index

{No value for "tocstring"}