

CALLMEBACK v1.0

by Mark Kupferman

The Documentation

Copyright ©1990 Mark Kupferman, All Rights Reserved. This program and the associated documentation are both copyrighted by Mark Kupferman. Individual system operators are granted permission to use this program under the conditions of the shareware agreement located

in this document. Violation of these conditions are subject to reprisals in the form of criminal and civil court action on the basis of the copyright laws of the United States of America. Be scared. Be very scared.

Table of Contents

INTRODUCTION	1
SPECIAL NOTE TO REGISTERED KATIE USERS	4
GENERAL REGISTRATION FORM	5
CONFIGURING CALLMEBACK	6
General Options	7
Path to Second Sight	7
Path to Userlog	7
Path to Callerlog	7
Path to User Data File	8
Path to CallMeBack Text Folder	8
CallBack Bypass Code	8
Number of Times to Call User	8
Number of Chances for User to Get Password Right	8
Access Level for Sysop Commands	9
Name/Phone Number/Sysop of your BBS	9
Name of 'Welcome New User' File	9
Name of 'Welcome New LOSER' File	9
System Calling Range	10
Out Of Range Text File	10
Out of Range Options	10
General Options	10
Selecting a Range of Area Codes	11
Illegal Exchanges (Exchanges Not To Call)	12
Configuring Questions	13
The Header Fields	14
Phone Number Questions	14
Should I dial '1' to reach you?	14
Address Questions	14
Short Answer Questions	15
True/False Questions	15
Multiple Choice Questions	15
Summary of Sections 1 & 2	16
Configuring Access Levels	17
Opening Text File	18
Specific Questions To Ask	18
Configuration for Successful Call	18
Configuration for Unsuccessful Call	19
Suggestions	19
Finished.	19
SYSOP COMMANDS	20
Create Tab-Delimited Text File	20
Create 80 column Text File	20
Continue through Questions	20
Return to Second Sight	20
TEXT FILES USED & CREATED BY CALLMEBACK	21
User Instructions File	21
User Welcome File	21
Losers File	21
Good User Text File	21
StartUp File	22
Out Of Range Text File	22
PROGRAM LOGIC	23
INSIDE CALLMEBACK	24
MODEM COMMANDS	26
UPDATE NOTES	27

Introduction

What is this program, and why did I write it?....these are two very interesting questions which people always ask about my programs. Maybe it is because my programs don't tend to explain themselves outright. Oh well.

CallMeBack is a program that will verify your users, as well as maintain a database of their vital information. It verifies users by asking them for their phone number and then calling them back to make sure they are telling the truth. If they ARE telling the truth, their access is raised to whatever level you want to give them, and if they are not telling the truth, their access is lowered to whatever level you think 'losers' deserve. This is the basic concept of CallMeBack. Everything else is the icing on the cake.

What is the icing? Well, for those of you who prefer lists, here is a list of features of the program:

- **Sysop selected calling range** - the program allows you to select what kind of calls you want the program to make. You can tell the program to make only local calls, calls within the area code, calls within a set of area codes, and calls within the United States/Canada. At this time, CallMeBack does not support international callers (ie, callers who don't have 10 digit phone numbers) but it is a possible future enhancement. But then, who wants to verify callers from Hong Kong, anyways? Users who are outside of the calling range will receive a special text file of the sysop's devising.

- **Sysop configured display** - the sysop has complete control over the information sent to the user. All instructions & text files are written by the sysop, meaning that you will have complete control over how users are treated by the program. This also means that no two CallMeBacks will look the same. Think of it as a simple survey program that has the ability to make sure people are telling the truth.

- **256 levels of verification** - users of different levels can be verified differently with different options. For example, if you are putting CallMeBack on your already existing system and you wish to reverify all of your users, you may want users at higher levels to automatically return to the levels and time limits they already had. Thus, new users who are approved would get a standard user access (let's say 50) while your old users would be given back their old access level. This feature is explained more in the appropriate section of this document.

- **Sysop configurable data fields** - The program comes with twelve extra data fields which can be used to store questions of the sysop's own design. These are broken up into two short answer questions, 5 multiple choice questions, and 5 true/false questions. All of the database questions can be changed using RedEdit. It should be noted that one is not required to use any of the extra questions if one does not want to.

- **Will create 'Welcome' files and 'Loser' files** - CallMeBack will maintain a text file of new users (validated users) and a text file of losers (people who give the wrong phone number). you can add your own heading to these files, and then have them displayable to your users so they can either welcome new users (and see who they are) and cast plagues on those who try to enter your system untruthfully. The names of these files are, of course, configurable.

- **Sysop chosen bypass code** - For those special users who you trust, CallMeBack allows you to select a bypass code that will cause the program to automatically assume that they are good, honest users without having to call them back.

These are the major features (or at least, the most noticable). The program runs pretty much on it's own, and maintains a log of it's activity. If it runs into any trouble at all, it returns to host, keeping all things as they are.

CallMeBack was written because it seemed like a good idea. I have seen verification programs on IBM systems, and it seemed like an invaluable sysop utility that a lot of sysops would like to have available to them. After all, CallMeBack automates one of the most boring parts of sysoping - calling users just to see if they are who they say they are. Since CallMeBack calls everyone in the specified range, users that are called do not feel as if they are not trusted, since they know (depending on your text file) that everyone is called.

It should be admitted that CallMeBack does not allow for easy online configuration (i.e, there are no pretty dialog boxes to fill in as there are with Katie). This was done for three reasons. First, the program is not configured all that often, so it seemed like a waste of space to have the program filled with code and graphics that are hardly ever going to be used or seen. Second, there are quite a lot of fields to configure. If you are including all of the options, there are millions and millions of different ways to configure the program, since each starting user level can be configured differently. Third, I didn't really want to waste my time writing all of the dialog handlers required to configure the program, especially after considering points 1 & 2. Configuration is therefore done through the use of a special text file that you need to modify to match your needs.

A note on the previous statement. All of my programs are as sysop configurable as possible. This is because of my own personal preferences towards programs (I like to make any online program look and act like the way I want it to, and not how the programmer does) and also because that is the very concept which makes Second Sight BBSs so interesting to both sysops and users.

It is very frustrating to me when I go to all of this trouble - and believe me, it is trouble (hard coding options is a lot easier than allowing sysops to easily change them) - to give the sysop all of these options and then find that the sysop has basically left things the way they were when they unpacked them from the archive for the first time. This was especially true with Katie. I mean, come on....Katie allows the sysop not only to control how the menu looks, but it also allows them to control how the data is spit out to the user. All I ask is that the sysop use some creativity.

This is not a tremendous problem with CallMeBack. First off, different systems have different validation policies and different ways of greeting people, and these would have to be reflected in the text files and the questions. Also, I am also taking the further step of not including any sysop usable text files with the program. Thus, each of these text files will have to be written by the sysop. Of course, they do not have to be very long...in fact, you could even leave them out and the program would still run. However, it's nice to see when a sysop is getting as much out of the program as possible.

Let me offer a final note on the shareware fee. This program is \$5. If you use this program on your system, please send it. There are a couple of other projects I am working on which I think many people will find very useful (I'm not telling what they are) and I will not be happy if I call a lot of BBSs and find that they are running the program without having sent any money to support the time it took to develop it.

Lastly, if you have any questions, problems, etc. please feel free to let me know. I will be happy to do whatever is reasonably possible to get this program to function on your system.

However, I must ask that you not send the shareware fee until CallMeBack IS working. I will not be happy with people who send money, and then tell me that there is a problem and that I am obliged to fix it. If this happens, you will simply find your money returned to you minus the amount it cost me to send it back.

I know this sounds a little hard line. However, experience teaches. Which brings me to my last comment. The shareware fee entitles you to the use of the program AS IT IS. It does not entitle you to future versions that don't exist. Maybe I should clarify. As a registered user, you are entitled to use any future updates and enhancements that are released at no extra cost to yourself. Thus, if I update the program, you can use the update without sending me any more money. However, I reserve the right to decide whether or not to update the program and what features to include in the update. I will not be compelled under any circumstances to update this program unless I want to. Shareware means you try out the program, and then you pay for it. Since you have not tried out the nonexistent future enhancements, you did not pay for them. You have only paid for what you already have.

This does have a good side. Since you are entitled to use any future updates to this program, you can rest assured that I will not charge you extra money for any updates. I.e., if I give CallMeBack the ability to call up a user voice and carry on a conversation with them, you will not be asked to send me more money for this special feature (and no, it is very, very unlikely that CallMeBack will ever be able to talk with someone voice. That would require special hardware).

I should note that I usually update my programs based on response. I.e., if only two people send the shareware fee, it is unlikely that the program will be updated very often. If many people are interested, I will continue to enhance the program to include whatever features you think it should have.

Lastly, I would like to thank the individuals who provided input as to what this program should include. This especially includes the gang over at Newton's Corner BBS where I can usually be found, as well as other people around the country I have chatted with.

How to get in touch with me:

Mark Kupferman
258 West Newton Street, #3
Boston, MA, 02116-6435

Voice: (617) 859-7757
BBS: Tao of Telecommunications...(617) 536-6419
Network address: Give me a couple of weeks...

America Online: MSK
Delphi: SMUDGE
Prodigy: KJBJ40A

Registered users of any of my programs are given access to the support section on my BBS. When you register for CallMeBack you will receive the bypass code that will allow you instant access to the board and to the private section.

SPECIAL NOTE TO REGISTERED KATIE USERS

Because you are all such wonderful people, I have decided to allow you to have CallMeBack without the \$5 shareware fee. This is out of thanks for your continued support of my programs, and for your patience during 1989.

Thus, you need not send any money (although I would not stop you if you did). I only ask that you send me a copy of the CallMeBack registration form. I would also ask that on the back of the form you include a few comments as to how well Katie is running for you and if there are any future enhancements you would like to see. This will be considered payment enough for CallMeBack.

Once again, thank you for your support.

GENERAL REGISTRATION FORM

(please fill this out and mail it to me if you have not previously registered for the utility in question. This form can be used to register for Katie, CallMeBack, or both)

Name : _____
Address : _____
City : _____ State: _____ Zip : _____
Voice Phone # : _____
BBS Phone # : _____
BBS Name : _____

BBS Software type (circle) : Red Ryder Host WWIV Mansion

Program: ___ Katie v2.2z (\$14.00) ___ CallMeBack v1.0 (\$5.00)

___ Thank you for the program. The check included is \$_____ to cover the shareware fee for the program(s) above selected. I HAVE NOT included any disks in this letter.

___ Thank you for the program. The check included is \$_____ to cover the shareware fee for the program(s) above selected plus a little extra because I like to send strange people money. I HAVE NOT included any disks in this letter.

SEND NO DISKS!

Suggestions for possible future enhancements or other comments:

CONFIGURING CALLMEBACK

I will say before we begin that this is not going to be the easiest program you have ever configured in your life. It won't be the hardest, but it requires that you follow the instructions exactly, and that you do not go making changes to the configuration file without reading the instructions relevant to that line.

Included in this archive you will find a file named 'CallMeBack.Options'. This is a text file that is in a very particular format. Each line means something, and any missing line will cause the program to crash when you run it for the first time.

Ok. Let's start at the beginning:

Step 1: Set up your files.

The archive for CallMeBack should contain 3 files. One is the application, the second is the CallMeBack.options file, and the third is this documentation you are reading now. The documentation you can put wherever you like. The other two files (the application and the options file) should be placed in the same folder as Second Sight (note: from now on, Red Ryder Host will be referred to as Second Sight just so we can keep things up-to-date).

Next, create a folder with a rather short name anywhere you like. This will be the Text folder where all data files pertinent to the program are kept. You can put this folder anywhere you like, although make a note of what the path to it is. You will need to know it very shortly.

Finally, you will want to add an option to one of your menus to run CallMeBack. This can be either a cmd #50 or cmd #51, but I would suggest using a cmd #50 since the user is already going to have to enter their password a couple of times. Besides, whoever runs the program probably isn't going to have much access to your system anyways, since they have yet to be validated. Use your discretion.

Step 2: Configure CALLMEBACK.

Here is where we come to the more complicated part. You will need a text editor as well as a printed version of this documentation. It is important that you read this part carefully, since each of the options (and what they do) will be explained in this section.

The CallMeBack.Options file is divided into a number of different sections. We shall examine each section one at a time.

Section 1: General Options

The General options can be specified as all of the lines in the file from the first line to where it says '-----QUESTIONS-----'. In the file default file you now have, it should look something like this (without the line numbers):

```

1. KATHARINE:BBS:Red Ryder Host
2. KATHARINE:BBS:ONLINE:STORAGE:Userlog
3. KATHARINE:BBS:ONLINE:STORAGE:Callerlog
4. KATHARINE:BBS:ONLINE:CALLBACK:CallMeBack.Data
5. KATHARINE:BBS:ONLINE:CALLBACK:
6. SUPPORT
7. 3
8. 3
9. 200
10. Tao of Telecommunications II
11. 617
12. 5366419
13. Mark Kupferman
14. NEWUSERS.TXT
15. LOSERS.TXT
16. LOCAL
17. OUTRANGE.TXT
18. 0000000000000000
19. 0000000000000000
20. 2
21. 617
22. 508
23. 5
24. 555
25. 550
26. 950
27. 900
28. 200

```

PATH TO SECOND SIGHT

Line #1 is the path to your Second Sight application (the program that is to be executed when CallMeBack is finished). Although you could make this another application, I would be hard pressed to think one you would want to execute next without running Red Ryder Host. Therefore, just put the path to your Second Sight including the name.

PATH TO USERLOG

Line #2 is the path to your Userlog. Please include the name.

PATH TO CALLERLOG

Line #3 is the path to your Callerlog. Although the callerlog is not written to at the moment, it seemed like a good idea to know where it is located for future versions. Make sure you include the name of your Callerlog.

PATH TO USER DATA FILE

Line #4 is the path to the file you wish to use as the data file for CallMeBack. In this file is stored the answers to all of the questions you have the program ask, including each user's name, address, phone number, etc. Although you can put this file anywhere you like, it is recommended that you put it in the new folder you created. Make sure you end the line with the name you want for the datafile (for those interested in accessing this datafile with other programs, the specs are included at the end of this documentation file).

PATH TO CALLMEBACK TEXT FOLDER

Line #5 holds the path to the folder where all text files should be stored. CallMeBack makes use of several text files...all of them will be stored in this folder, allowing you to only have to enter the file name of each text file. All files that are stored in this folder must be no longer than twelve characters, and it might be preferable to use the standard filename format that is used on other systems (i.e, 12345678.123). So, line 5 should contain the path to the folder you were asked to create previously. Although you have the option of specifying your host folder (or any other folder), please realize that this is a quick road to messiness.

CALLBACK BYPASS CODE

Line #6 hold the Bypass Code which you can give out to various users that you trust or that you don't want to have to go to the trouble of running through the verification (CallMeBack) routines. If they enter the appropriate bypass code the program will ask them the questions you specify, but will automatically raise their level and return to host without hanging up on the user and calling them back. If you only want the program to ask questions and NOT call anyone back, leave this line blank or leave the access code in the new user message. (Personally I think this would remove the point of having the program, but to each his own...) There are many uses to this code. For example, registered users of Katie and CallMeBack will receive a copy of the bypass code to the Tao of Telecommunications so that they can receive support without having to run through the CallBack procedure. The bypass code could also be given to members of a local users group. There are many different scenerios.

NUMBER OF TIMES TO CALL USER

Line #7 contains the number of times you want the program to call someone back before giving up and assuming that a wrong number has been entered. It is a good idea to leave this number as low as possible, since you do not want to be harrasing people. The default is 3. It might be better to use 2, and you might even want to use 1.

NUMBER OF CHANCES FOR USER TO GET PASSWORD RIGHT

Line #8 contains the number of trys you want to give the user to enter their password once a connection has been made by the system. This is the same as the option that is in the Second Sight configuration program. Thus, once CallMeBack reaches a modem, it will print the name of your system and ask for the user's password. If the user enters the right password, all is good. If they enter the wrong password, they will get as many chances as you specify to try again. Some people might as why this is necessary. Shouldn't the carrier be enough to prove that the person was telling the truth? No, it isn't. The user could give you the number of a local BBS system, which would also provide a carrier. So it is important that the user demonstrate that they are really the user in question.

ACCESS LEVEL FOR SYSOP COMMANDS

Line #9 contains the access level the user must have to access the sysop commands. It defaults at 200 and you can feel free to modify it to whatever you like. If the user has the level specified (or higher) they will automatically go to the special sysop menu instead of going through the questions, etc. The sysop menu will be explained later in this document. Until you are used to this program, it might be good to set this level to 255.

NAME/PHONE NUMBER/SYSOP OF YOUR BBS

Line #10 contains the name of your BBS. This is displayed when the system calls back the user as 'This is a CallMeBack from <system name>.' If you need a 'the' in the name, make sure you add it yourself.

Line #11 contains the area code of your BBS. This is an important field, since it will tell the system what should be considered a local call.

Line #12 contains the last 7 digits of your phone number. Although this is not presently used very much (the program does check to make sure it has not been told to call itself) it is just something nice for CallMeBack to know.

Line #13 contains the name of the sysop. This is also not used very much (in fact, it isn't used at all) but we must be prepared for future enhancements, eh?

NAME OF 'WELCOME NEW USER' FILE

Line #14 contains the name of the 'Welcome' file. This is a file you have the option of using on your system so that all new callers are appended to a file that can be displayed by your other users (should you wish them to be able to). The heading can contain whatever you want it to. For more information on this file, see the appropriate section of the documentation. However, whether you are going to use this option or not, you must specify a file name. The file will be stored in the data folder you specified. Please be sure you create a text file by whatever name you specify - even if you don't use it, it wouldn't hurt if it existed. This filename should be no longer than 12 characters.

NAME OF 'WELCOME NEW LOSER' FILE

Line #15 holds the names of the 'users' who did not give their real phone number. I like to think of it as the 'LOSERS.TXT' file, since that is what it contains. You are free to make this text file available to your users, and documentation as to how this feature works is explained along with how the Welcome file works. Whether or not you use this option, it is important that you specify some file name (12 characters) and that you create the file. This file will also be located in the text folder you specify.

SYSTEM CALLING RANGE

Line #16 contains the calling range. This field is a number between 0 and 4 which tells CallMeBack whether or not you only want to make calls to local users, calls to users who live in the same area code, calls to users within a range of area codes, or calls within the United States. So, this line would contain one of the following:

0 - If the line contains '0' the program will only call users that are in the local area code (the area code of your BBS). The program will not dial '1'. Thus, all calls made will be within your local calling range.

1 - If the line contains '1' then program will only call phone numbers that are in your area code. If a '1' has to be dialed, it will be dialed. Therefore, this will call only phone numbers in the local area code.

2 - This number is presently unused and represents nothing.

3 - If the line contains '3' the program will call any user that is in one of the area codes specified in a list you will create later on (starting at line #20). This is most useful if you want your system to call back any user in the state, even though the state might have more than 1 area code. For example, in Massachusetts there are 3 different area codes. I may want the program to call back all users who are in the state. I would later tell the system to call any phone number in area code 617, 508 and 413. This feature might be useful for BBSs located in New York City as well. The system will dial '1' for area codes that are not the same as your, and will dial 7 digits for the local callers (and '1' for out of range callers in your area code). Please see the information on setting up the range of callers listed with line #20.

4 - Although it would seem like a crazy thing to do, this command will call back any user in the United States/Canada (or any other place that can be reached by dialing '1' followed by a ten digit phone number. Think hard before you use this command.

OUT OF RANGE TEXT FILE

Line #17 contains the name of the text file that should be sent to the user if he/she is found to be out of the system's calling range. If a user is found to be out of range, they will be sent this text file and returned to Second Sight, with no change in either their access level or their time limit. This text file should explain to the user how you will go about validating out of range users, as well as anything else you would like them to know.

OUT OF RANGE OPTIONS

Line #18 contains out-of-range options. These are on off switches that are either '0' or '1', depending on whether or not you want the option turned on (1) or off (0). They go left to right, with left being option #1 and right being option #15. Presently, there are no out of range options. If you can think of any, please feel free to send your suggestions.

GENERAL OPTIONS

Line #19 contains general options for the system. As with the previous line, they are numbered 1 to 15 and presently do not do anything. If you have any suggestions, please feel free to send them. For those interested in why these switches exist in the form that they do, I will tell you that they represent boolean switches inside the information file the program will create when you run it the first time. In the future, these switches will do something. General options would control functions that are generally relevant to all users starting at any access level.

SELECTING A RANGE OF AREA CODES

CallMeBack will call a selected range of area codes that you select. As explained previously, this is most useful if you want the program to be able to call users who are nearby (such as in the same state) but are in a different area code. You can enter up to 5 area codes for the system to call.

These area codes are configured so that line #20 (the first line of the series) contains the total number of area codes that you want it to call. Then, each respective line should contain the area codes themselves. Therefore, if you wanted the program to call users in area codes '617', '508', and '413', you would enter the following:

Line #20 3 Line #21 617 Line #22 508 Line #23 413
--

It might help to understand the logic the program uses to read this part of the file. First, the program reads line 20. It then learns that you want CallMeBack to call 3 different area codes. The program will then read three lines (1 for each area code). It is important that you are in synch - you must fill as many lines as is specified in line 20, or you will run into a system error. Thus, if line 20 says that you have 3 area codes, the next three lines of the option file must contain area codes and nothing else.

If you are not going to use this feature, you must put a '1' in line #20 and your local area code in line #21. This is because the program will get confused if you tell it that there are no area codes. So, if you are not putting a '3' in line #16, you will put a '1' in line #20 and your local area code in line #21.

(at this point, it is going to be difficult to continue to refer to actual 'line numbers' since they are going to change depending on the way you configure your system. Therefore, the program will refer to the numbering of the default file. Note that these numbers are only for example purposes only, and that they may not be the same on your system.)

ILLEGAL EXCHANGES (EXCHANGES NOT TO CALL)

Obviously, there are certain phone numbers that your callers are not from. For example, none of your callers are from '555-1212' or in fact, any number in the '555' exchange. In Massachusetts, your callers are probably not from '550' (chat lines) or '911' (emergency services) or '411' (directory assistance). Many of these exchanges are built into the program and will automatically be rejected. Some you need to enter on your own. Listed below are the exchanges that CallMeBack will automatically reject:

Any exchange starting with a '0'
 Any exchange starting with a '1'
 '200' '300' '400' '500' '600' '700' '800' '900'
 '555'
 '911'
 '999'

In your own area code there may be a number of exchanges that simply do not have personal phone numbers attached. The system allows you to specify up to 15 of them that will be automatically rejected by the system.

Of course, this can be used for other purposes. If certain exchanges in your area code are considered long distance, you can mark these as 'illegal'. It should be noted that if a user enters any of these area codes the system will be considered 'out of range' and the appropriate text file will be displayed. Thus, any user from '555-1212' would be considered out of range.

This feature is configured in much the same way as was the previous feature. If we look at our example (now back quite a few pages) line #23 contains the number of illegal exchanges that are to follow. If there are 5 of them, lines #24 to #28 would contain the area codes themselves. If you have only 3 illegal exchanges, the first line would contain the number '3' and the following 3 lines would contain the 3 exchanges, 1 on each line.

If you do not have any illegal exchanges, then you should put '1' in the first line and put either '***' or '555' in the following line. It should be noted that these exchanges will not work in ANY AREA CODE - Thus, if you don't want the user to enter '555' in your area code, they will not be able to in any area code.

In the example, the system is being told not to call five exchanges. These are '555', '550', '950', '900', '200'. It should be noted that the last two are already coded into the program, and do not need to be in the option file. However, it will all work towards the same effect.

Section 2: Configuring Questions

CallMeBack provides the sysop with the ability to ask the user a number of questions. Some of them are standard, some of them configurable. The system will ask the user for a name, an address, and a phone number. It will also ask 5 short answer questions, 15 multiple choice questions, and 15 true/false questions, each of these at your devising. Although these actual use of this information is presently limited, the data is exportable to a tab delimited text file or a simple 80 column text files. More commands will be added based on the kind of data manipulation people who purchase the program want. Also, the format to the data structure used is in the end of this document.

The question section can be found in between the 'Questions' separator and the 'Control Functions for Access Levels' Separator. Here is our working example set:

```

1.  ----- QUESTIONS -----
2.  Unused
3.  Unused
4.  Enter your area code (xxx)      :
5.  First 3 digits of your phone # (xxx) :
6.  Last 4 digits of your phone # (xxxx) :
7.  Do I have to dial "1" to reach you (N/y)?
8.  First line of your address      :
9.  The city you live in            :
10. Your state                      :
11. Your zip code                   :
12. 4
13. If you operate a BBS, what is it called?
14. What is it's phone number?
15. If you are in a performing group, what is it called?
16. If you are in school (or work at one), where?
17. 2
18. Are you a registered Katie owner?
19. Are you a registered CallMeBack owner?
20. 2
21. What kind of computer do you use most?
22. 5
23. Amiga
24. Apple IIX
25. IBM
26. Macintosh
27. Other...
28. Which is your favorite area of the arts?
29. 4
30. Theatre
31. Dance
32. Music
33. Visual Arts

```

PLEASE NOTE THAT THE LINE NUMBERS YOU SEE ARE ONLY FOR REFERENCE PURPOSES - DEPENDING ON HOW YOU HAVE CHANGED THE OPTION FILE TO MEET YOUR OWN NEEDS YOUR OWN LINE NUMBERS WILL BE DIFFERENT FROM THE ONES YOU SEE ABOVE. HOWEVER, FOR DEMONSTRATION PURPOSES THEY WILL SERVE PERFECTLY.

THE HEADER FIELDS

If you look at our example, you will see that the first three lines really don't do much of anything. The first is the header line which simply separates the questions from the general options. Line #2 & #3 contain the word 'unused'. Originally, these were used to for the user to enter their first and last name. However, it seems like data management will work more effectively if we simply use the name the user already entered when they first logged onto your Second Sight system. Until I feel like changing the program, simply leave these lines to read 'Unused'.

PHONE NUMBER QUESTIONS

This is the first field where you will get to determine how a particular question is asked. Basically, lines #4 - #7 (in our example ONLY remember) contain the question that will be used to ask the user for their phone number. Each question can be up to 60 characters long. The first question (#4) is the question to ask the user their area code. The user can enter up to 3 characters.

The second phone number question (#5) is used to ask the user his/her exchange. Once again, the user can type no more than 3 characters. The third question is used to ask the user the last 4 numbers of their phone number, and this must be 4 characters long.

As stated many pages ago in the introduction, it is important to me that you make use of this ability to design your own questions. You have the ability to ask each of these questions in the way you would ask them, rather in the way I would ask them. Please use it. Change these lines to match the way you would like your system to appear.

SHOULD I DIAL '1' TO REACH YOU?

There is no easy way for the system to determine whether or not it needs to dial a '1' before it dials a number in the local area code. Obviously, the system will dial '1' if it has to call outside the area code. However, in some cases inside the area code the system will have to dial '1' to reach the user. Instead of forcing you to create a table of 'medium distance' exchanges, the program simply asks the user the answer.

If they enter the wrong answer, the system will not get through to them. Therefore, there is no advantage for a user to lie, since it will be assumed by the program that it has simply been given a wrong number. One can pretty much assume that if a user has to dial '1' to call your BBS that your BBS will have to dial '1' to reach them. You might want to phrase your question along those lines.

So, in line #7 of our example you would enter the question you would like the system to ask your user about whether or not your system will have to dial '1' to reach them. They have one character to reply and can either enter 'T' or 'Y' for yes or 'F' or 'N' for No (ie, you can form a True/False or a Yes/No question and the system will understand). If the user enters nothing, this will default to 'N'o.

ADDRESS QUESTIONS

The next four lines need to contain the questions used to ask the system where they live. The first line (#8) should ask them for their street address (254 Main Street), the second line should ask for their city (Boston), the fourth line should ask them for their state (they can enter up to two characters), and the fifth line should ask them for their zip code (5 characters). Phrase these questions any way you like. Note that it is advisable to leave a space after your question mark or colon to improve formatting.

SHORT ANSWER QUESTIONS

CallMeBack allows you to ask your users up to 5 different short answer questions. They have 20 characters to answer the questions (short really does mean 'short' here). You can ask anything you like....the data the user enters will be stored in their record.

First, you must tell the system how many short answer questions you are going to use. This is done in the same way you entered the range of area codes and the illegal exchanges. The first line beneath the zip code field (discussed previously) should contain how many short answer questions you are going to have (between 1 & 5). The next 1 to 5 lines would contain the questions you would like to ask. Here is an example. Let's say you want to ask 3 questions. You would tell CallMeBack this with the following:

```
3
Where do you go to school?
What is your favorite color?
What is your user group number?
```

As with everything else, you must have at least one short answer question. While this shouldn't be a problem to most people, some of you will just have to make one up. Such is life. If this is really a problem to anyone, please feel free to let me know.

TRUE/FALSE QUESTIONS

CallMeBack allows you to ask your users up to 15 different true/false questions. These questions can be answered either with a 'Y', 'N', 'T', or 'F'. Thus, you can have true/false questions and also yes/no questions.

True/false questions are entered the same way the short answer questions are. The first line contains the number of questions, followed by one to 15 questions. You must have at least one true/false question for the system to configure itself properly.

MULTIPLE CHOICE QUESTIONS

CallMeBack allows you to ask your user up to 15 multiple choice questions. Each of these questions can have up to 5 different choices. The system will keep track of the number of the option that the user selects (by storing it in their data record).

Multiple choice questions are entered differently from short answer questions and true/false questions. After all, not only is the question required, but so are the options. However, if you have been following how all of the other fields are entered you will understand how this works.

Each question can be up to 60 characters long. Each answer can be up to 30 characters long. First of all, you must enter the number of multiple choice questions that you will be asking. As before, you must have at least 1 multiple choice question or the system will get upset. The second line must contain the question. The line after that must contain the number of answers (ranging from 1 to 5). Since this is multiple choice, it might be advisable to enter at least two different choices. Next, you need to list the answers, one on each line. After that, you will enter the next question.

An example might serve us better here. Let us say you want to find out what kind of computer the user has. You only want to ask one question, and you want to offer four (4) different choices : Macintosh, Amiga, Atari, or IBM. So, you would have the following:

1.	1	# of questions
2.	What kind of computer do you have?	the question
3.	4	the number of choices
4.	Macintosh	choice #1
5.	Amiga	choice #2
6.	Atari	choice #3
7.	IBM	choice #4

If you had more than one question, you would change the '# of questions' to the total number you had and would repeat lines 2 through 7 until you had listed all of them.

SUMMARY OF SECTIONS 1 & 2

Before we move onto the final part of the configuration, let's quickly go over what we have done so far. We have configured all of our file paths, and we have set up our questions. It is at this point that we should check to make sure that there are no extra lines where there shouldn't be, and that everything you have set up is correct.

If there are any errors so far, they are probably in either the range of area codes, the illegal exchanges, or in the questions. Take a moment to go over the file so far to be sure that everything is set up correctly. I realize this may sound a little patronizing, but I would hate to have to deal 50 people who had configured their system to read 1 more question than they really entered. Once you have done this, please continue to the next page where we will configure the heart and soul of what this program does.

A note on questions: If you later think of a question that you forgot to add, the question can be easily added by deleting the information file. This is explained in the appendix.

Section 3: Configuring Access Levels

The last and final part of the configuration file controls the real driver of the system. It allows you to configure how different users at different access levels will be treated. Depending on how complicated your system is, this will either be quick or it will take a few minutes. There is no reason this should take a long time.

This part of the documentation will deal with how to edit the section of the configuration file that starts with “Control Functions for Access Levels” to the end of the configuration file.

CallMeBack allows you to configure the program so that callers who enter the program with different access levels are treated differently. You can display different text files to users of different access levels, as well as change their access to different levels. It is also possible to adjust their time limits and restriction flags.

If you skim over your file you will notice that there are 256 different ‘records’, each one looking approximately the same. They are. In the upper left hand corner of each record is the access level for that particular record. Therefore, we will go over just one of them. All of them work the same and have the same options.

This is the same record we will work with:

```

1. -----Level 10
2. 0
3. 000000000000000000000000
4. GENERAL.TXT
5. 11111111111
6. 20
7. 60
8. 000000000000000000000000
9. GOODUSER.TXT
10. 0000000000000000
11. 0
12. 10
13. 000000000000000000000000
14. 0000000000000000

```

THE HEADER (LINES #1, #2, #3)

The first line of each record contains the nothing important. It must exist, but it is mostly there to help you locate each separate record. The second line contains the access level. The third line contains the restriction information (left to right). Do not change these lines! Never, ever. In fact, just so nothing happens, leave lines #1 & 2 & 3 of each record exactly as you find them. This will insure that everything works nice and happily.

OPENING TEXT FILE

Line #4 contains the opening text file. This is the text file that will first be displayed when the program is run. If you wanted to, you could have a different text file for each access level. This way, users at different access levels will be treated differently by the program. For example, you might want to revalidate all of your old users, plus set up a system to validate new users. You could send a different text file to each.

The text file must (as with all text files) have a 12 character file name. The file(s) must be kept inside the text folder you specified oh-so-many pages ago. It should be noted that you can insert a page break within the next file by using the '␣' character (Shift/Option V). You should have at least one of these in your opening message, and it would be nice to have it at the end. Why? Well, it is through this little character that the user can enter the bypass code. When the program reaches one of these, it will say "Please hit <RETURN> or enter your bypass code." If the user enters the right code, the program will display "Bypass code accepted." and go on as usual except that it will not bother to call the user back, but will automatically raise the access level to whatever you specify for that particular level (to be explained very shortly).

SPECIFIC QUESTIONS TO ASK

This feature is not yet active. What it will do is allow you to control which questions you want the user to answer and which questions you don't want them to answer. This will be incorporated into the next version of the program.

CONFIGURATION FOR SUCCESSFUL CALL

Lines #6 through #10 control what happens to the user if they are successfully reached by the program or if they have used the bypass code. Line #6 contains the new access level for the user. Line #7 contains the users new time limit. Line #8 contains the users new restriction codes (unused in this version), line #9 contains the text file to display to the user before the system hangs up, and line #10 controls specific options.

The first three lines are pretty specific. These should contain the new access level, time limit, and restriction flags of the user that has been successfully called back. These new levels will be placed in the userlog. Restriction flags go from left to right (1-24) and it should be carefully noted that the user's restriction flags will be adjusted to equal what you enter on this line. Thus, before you have the user do anything that might change their restriction codes from some kind of a norm for the level they are on, it is a good idea to run this program first.

After the user has successfully called the user and raised their access level, it will display a text file to them telling them that they are welcome and that they are now validated users of your BBS. As with all other text files, this should be placed in the text folder you previously specified. If you use the '␣' character in the text file (Option/Shift 'V') the program will pause the text file (this should be done every 20 lines or so for nice formatting). After this text file is displayed, the user will be requested to hang up. This is done so that you will not be sponsoring their telephone call (although it doesn't hurt to call a user long distance for a couple minutes, an hour is a long time). Usually it is nice if you tell them that they are welcome to call right back at their new access and time limit, although that is completely up to you.

Line #10 contains the various options. Only one option is presently active (gotta love those future enhancements). This is the very left most option. As explained before, each one of these '0's represents a boolean switch that can be turned on (1) or off (0). Anyways, option #1 (on the left) is the only one that works right now. This option will control whether or not users who are beginning at the level specified at the top of the record (in our example '10') should be added to the Welcome file that you named at the top of the configuration file. Thus, you can have new users placed in this file (so everyone can give them a proper welcome) and old users who run the program will simply be verified. For more information on this option, please see the 'Explanation of Text Files' section of this documentation.

CONFIGURATION FOR UNSUCCESSFUL CALL

Lines #11 through #14 contain the information on what you want to happen to the user should they prove to be 'losers'. Line #11 contains their new access level (usually quite low), Line #12 their new time limit (usually quite low) and Line #13 their new restriction flags. Restriction flags go from left to right, and are represented with either '0's or '1's.

Line #14 contains the specific options available. Although these options are different from the 'good user' options, only one of them presently works. This is the one which controls whether or not the loser's name is placed in the 'Loser' text file, which you can make available to your users to read (and cast shame). This is option number 1 (the left most option) and is controlled by using our binary switching system ('1' for on, '0' for off). Please see the explanation of this text file in the appropriate section of the docs. (Which to tell the truth, I haven't written yet, but I know it will be really interesting).

SUGGESTIONS

If a user blows it the first time, give them another chance. For example, if they started with an access level of 5, bring them down to 4 and put them in the losers text file with a time limit of 5 minutes. Then, have a special opening text file for level 4 that explains to them that this is their last chance to get it right. If they get it right the second time, then bring them up to full access. If not, then throw them to a time limit of 0 and an access level of 0.

Note that the program does not know if you have already added the users name to the 'losers' text file. Thus, if you do lower their access to 4, make sure that level 4 is not set so that if they fail they are added to the losers text file again. While there is such a thing as a two time loser, it looks silly in a text file.

Finished.

Well, you're done. All that you have to do now is save all of your changes and run CallMeBack. What will happen is that CallMeBack will read the CallMeBack.Options file and create a 'CallMeBack.Info' file. If this doesn't happen (i.e., the system locks up or something like that) go back and check to make sure your Options file is correct.

You can now take your CallMeBack.Options file and put it in a safe place. Perhaps you might want to Stufit and put it away somewhere. You will not need to use it again unless you want to change your configuration.

You might be finished now. At this point you should test the program with a user or two. If everything works successfully, you are ready to face the masses. However, there is a chance that the modem command used to dial the user are not working. If this is a problem, don't cry yet! When I promise configurability, I mean it. You are simply going to have to locate your copy of ResEdit and skip to the appendix to find the appropriate STR# resources to change. If you still have a problem, send me mail and we will see what we can do to get you up and running.

Remember, do not send any money unless the program is running properly on your system (at least, as properly as you think it should)! It is simply not fair to hold me responsible.

See? It only took 15 pages to configure this program. Not bad, eh?

Sysop Commands

When you configured CallMeBack, you entered an access level to represent a sysop. Anyone who has an access level equal to or above this access level will automatically be transported to this menu when they run the program. It should be noted that this menu is not finished. There are many more commands that can and will be added to it in the future. I welcome your suggestions.

CREATE TAB-DELIMITED TEXT FILE

This command will create a text file called 'USERS.TAB' in your previously selected text folder. This file will contain all of the names, addresses, and the answers to all of the questions the users had previously answered earlier. You can take this file and import it into either a word processor or even a database or worksheet program. The data is sorted by the date of entry.

CREATE 80 COLUMN TEXT FILE

This command will create an 80 column text file that contains the name, address, and phone number of the user. This is useful if you want to keep a copy of your user's names and address online.

CONTINUE THROUGH QUESTIONS

This allows you to run through the questions like a regular user. It is most useful if you would like your co-sysops to enter their information even though you may not want the program to call them back. In effect, it has the same effect as a user entering the bypass code, except that they user doesn't have to enter the code.

RETURN TO SECOND SIGHT

This command will abort the program and return to Second Sight. You will probably use this command whenever you want to quit the program without going through the questions.

Once again, this menu is not finished. I am sure there are many interesting ways that you would all like the user data to be manipulated. This isn't a problem. Just send me your suggestions and I will see what I can do to impliment them.

TEXT FILES USED & CREATED BY CALLMEBACK

The following is a list and an explanation of all of the text files used and created by CallMeBack. It is useful to examine this information in order to give you some idea as to what each file needs to contain (or should contain) and of what value it will be to you. All text files should be kept in the Text File Folder you specified previously. If you would like to insert a pause in any text file the system sends out, insert a diamond character (Shift/Option "V") on its own line and the program will ask the user to hit <RETURN>.

USER INSTRUCTIONS FILE

Although this really isn't mentioned in the configuration, CallMeBack will print a text file right before the system hangs up on the user (before it attempts to call them back). This file is called 'INSTRUCT.TXT' and if you wish to use it, you should place it in the text file folder. This file might contain information on exactly what is going to happen, thereby giving the user some idea as to what is expected of them. If this file does not exist the program will go immediately from asking the questions to hanging up the phone.

USER WELCOME FILE

This file is where you can have CallMeBack maintain a list of all of your new users that can be displayed by yourself and/or the rest of your users. What you do is provide a 'header' for the file which says something like 'Welcome the following new users to our BBS...' of as many lines as you like, and the program will append the names of the users to the bottom of the file as they are validated. The names of each user are indented 20 characters, so they will be somewhat centered on the page. The best way to handle this is to create the Welcome file, let 1 or two users log on, and take a look at the text file. Anything you add to the top of the file will remain in the file when the names of the users are appended. Names are only written to this file if this option is turned on in the option file.

If you want all of your users to see this file automatically when they log on, the only way to do this at the moment is to specify your Second Sight folder as your text folder. Then you can use one of your SYSTEMx.MSG files as your user welcome file. This is always a nice way to have your users greet all of the new callers.

LOSERS FILE

This file is much like the previous one, except that it contains users that were not successfully called back. You are allowed to add a header to the file, and the names of each loser will be appended to the bottom of the file. This option is only turned on if you change the appropriate zero to a one (this is explained in the user configuration file).

GOOD USER TEXT FILE

This file is sent to the user after they have entered their password correct and before the system once again hangs up on them or if they have entered the proper bypass code. In this file you should explain to the user that although they are being logged off, they are welcome to call right back and that they will now have full regular access to the system. It is possible to have a different Good User Text file for each user level. If you make this file longer than 20 lines, you should use the diamond character (Shift/Option "V") and the program will pause the listing and ask the user to hit <RETURN>.

STARTUP FILE

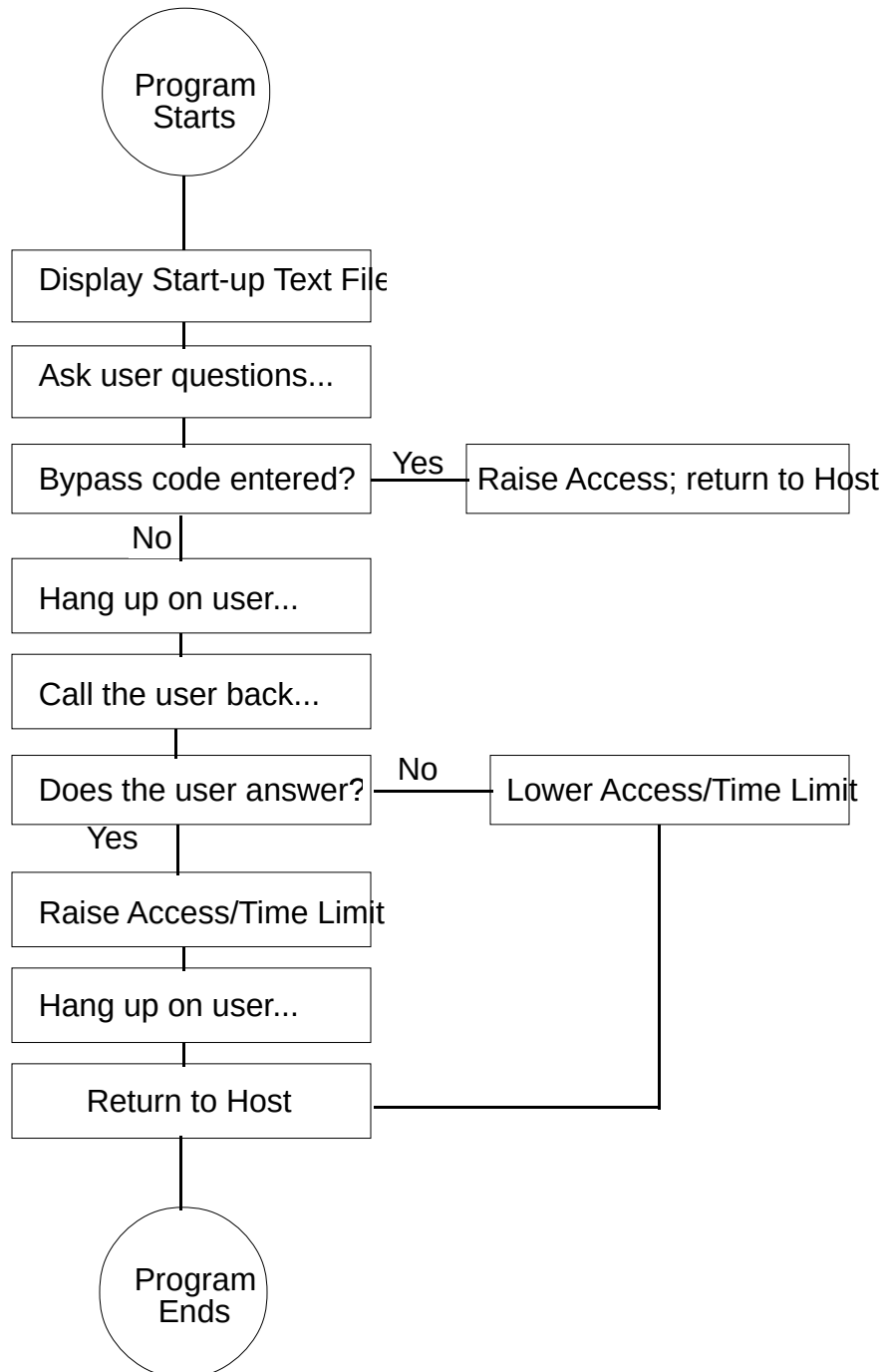
This is the text file that is sent to the user when they first run the program. It can be used to explain to the user your system policy/rules, as well as tell them why they need to answer the questions that are about to follow. At the end of this text file, it is usually a good idea to insert a diamond character (explained earlier) in order to offer the user the opportunity to enter either the bypass code or simply give them a chance to read the file before it scrolls off the screen.

OUT OF RANGE TEXT FILE

This is the file that is displayed to all users that are considered out of calling range for the BBS. It should explain to the user how you will go about validating them, or it can tell them that the board is only available to local users.

PROGRAM LOGIC

This section is here to give you some idea as to what the logic of the program is. It is certainly not technical, but it will hopefully show you the order of events on the system before, and probably make it easier for you to configure the program.



Now that this is all drawn out, I suppose it doesn't look as impressive as I had hoped. Hopefully the next version will be better drawn. If anyone else wants to draw the logic of this program, please feel free to send it to me and you will receive appropriate credit.

INSIDE CALLMEBACK

The following file formats can be used to access any and all of the data files created and used by CallMeBack. If you come up with anything useful, it would be wonderful if you could send me a copy of whatever it is. If you have any questions on how to use or read these fields, please let me know. I will be happy to do what I can.

CallMeBack was written in Turbo Pascal v1.1. It makes no use of any special commercial units or libraries, so should I ever release the source code (which I might...I have a tendency to do that...if you really want the source, send me the shareware fee and ask me politely for the code...in a moment of weakness I might let you have it) you will have no trouble editing it as long as you have Turbo Pascal.

USER DATA FILE FORMAT

The user data file is the file that contains all of the names and addresses of each of the users. The user data file contains one record for each caller, and they are one right after another (just like in the userlog). While I am not going to tell you how many bytes long this file is (figure it out yourself) I will describe each line for you.

CallerRec = record

DataPoint : LongInt;	Unused at the moment
DateOn : LongInt;	Date/Time user ran CallMeBack
FirstName : string[15];	User's first name
LastName : string[15];	User's last name
AreaCode : string[3];	User's area code
Office : string[3];	User's exchange
PhoneCode : string[4];	Last four digits of phone number
LocalNum : string[10];	Full phone number of user
InterNat : string[15];	For international phone numbers
Address : string[30];	User's street address
City : String[20];	User's city
State : String[2];	User's state
Zip : String[5];	User's zip code
Questions : Array[1..5] of String[20];	Answers to short answer questions
TrueFalse : Array[1..15] of Boolean;	Answers to true/false questions
MultChoice : Array[1..15] of signedbyte;	Answer to multiple choice questions
end; { CallerRec }	

CALLMEBACK INFO FILE FORMAT

This file contains all of the options you had previously specified in your CallMeBack.Option file. It is just one record long, and it contains a number of fields in a number of different formats. The extra formats are described after the main record.

CallBackType = record	
RRHPath : String[60];	Path to Second Sight
Userlog : String[60];	Path to Userlog
Callerlog : String[60];	Path to Callerlog
DataPath : String[60];	Path to User Data file
IndexPath : String[60];	Unused
TextPath : String[60];	Path to text file folder
ByPass : String[15];	Bypass code
BBSName : String[40];	BBS name
BBSArea : String[3];	BBS's area code
BBSPhone : String[7];	Last 7 digits of phone
BBSSysop : String[30];	BBS sysop's name
WelcomeOut : String[12];	Name of welcome file
LoserOut : String[12];	Name of Loser file
CallRange : Byte;	Calling range (0-4)
WhatToDo : Array[0..255] of WhatToDoType;	Explained below
Range : Byte;	Unused
OutOfRangeTxt : String[12];	Name of Out of Range text
OutOfRangeOpt : Array[1..15] of Boolean;	Out of range options
GeneralOpt : Array[1..15] of Boolean;	General options
Areas : Array[1..25] of string[3];	Selected area codes
BadExchange : Array[1..25] of string[3];	Illegal exchanges
GeneralQ : Array[1..10] of string[60];	Address/Phone questions
Questions : Array[1..5] of string[60];	Short answer questions
TrueFalseQ : Array[1..15] of string[60];	True/False questions
MultChoiceQ : Array[1..15] of MultChoiceType;	Explained below
SysopLevel : Byte;	Level to access sysop commands
CallTrys : Byte;	number of times to call user
PassTrys : Byte;	trys at password
end;	

SECURITY LEVELS FORMAT

This type is included in the CallMeBack.Info format. It contains the options and the user levels. There are 256 of these inside the program itself. This is really just a segment of the previous type.

WhatToDoType = record	
StartLev : byte;	User's starting level
StartRestr : RestrictType;	Unused
StartText : String[12];	Intro text file name
StartAsk : Array[1..12] of Boolean;	Unused
GoodLev : byte;	good access level
GoodTime : byte;	good time limit
GoodRestr : RestrictType;	Unused (restriction flags)
GoodText : String[12];	Good text file to display
GoodOption : Array[1..15] of Boolean;	Good options
BadLev : byte;	Bad access level
BadTime : byte;	Bad time limit
BadRestr : RestrictType;	Unused (restriction flags)
BadOption : Array[1..15] of Boolean;	Bad options
END;	

MULTIPLE CHOICE QUESTION FORMAT

This is another part of the CallMeBack.Info file that contains the multiple choice questions as well as their responses. It's pretty simple, so I won't bother to explain the meaning of each line.

```

MultChoiceType = Record
  Question : string[60];
  Answers  : Array[1..5] of string[30];
end;
```

ADJUSTING MODEM COMMANDS

All modems are not the same. Therefore, I have tried to make it possible for you to have some control over what commands are sent to the modem when CallMeBack tries to call the user. It should be noted that in order to make these adjustments you are going to have to drag out your copy of ResEdit.

The modem commands are stored in the 'STR#' resource. There is only one of them, so the right one should be easy to find. Here is what each line means:

Lines 1, 2, 3 - These three lines are sent to the modem before the computer starts to dial. You do not have to use all of them, but if you might want to put a few 'AT' commands on them just in case.

Line 4 - This is the actual string that comes before the phone number in the dialing. Usually, it would contain something like 'ATDT' or even 'ATDP'.

Line 5,6 - These lines contain the strings that your modem sends to represent no connection. For example, 'NO CARRIER' might be one of them. Another might be 'BUSY'. If you only want to use one word, then put some garbage in line 6 (i.e., something that is not likely to be sent by your modem.)

Line 7 - This is the string that means that the system should try the call again, without incrementing the counter. You might want to set this to 'NO ANSWER' or 'NO DIAL TONE' just in case another user has called your system before CallMeBack has finished the other user's validation procedure.

Line 8 - Contains the text sent by the modem to represent a connection. Usually, you will want this field to contain the word 'CONNECT'.

It is possible that the system will work the way it is configured now. You should try the program as/is before you start making changes.

- END OF CALLMEBACK DOCUMENTATION -

UPDATE NOTES

THIS PAGE WILL CONTAIN UPDATE NOTES, CHANGES, NEW FEATURES, ETC BETWEEN EACH VERSION OF THE PROGRAM. ALL NEW FEATURES WILL AUTOMATICALLY BE ADDED TO THE MAIN DOCUMENTATION, BUT THIS SECTION WILL EXPLAIN THE DIFFERENCES BETWEEN THE VERSIONS.

This is the first release version of the program, so there are no release notes. It should be noted that there is one previous version to this documentation, and that it should be destroyed as it contains errors.