Operator Accounts Without Root Access

Special Notices

Information in this document is correct to the best of our knowledge at the time of this writing. Please send feedback by fax to "AIXServ Information" at (512) 823-4009.

Please use this information with care. IBM will not be responsible for damages of any kind resulting from its use. The use of this information is the sole responsibility of the customer and depends on the customer's ability to evaluate and integrate this information into the customer's operational environment.

About This Document

The following menush script allows specialized access to root functions. System administrators can establish accounts to allow other users to perform functions, such as backups, killing hung processes, or managing print jobs. While these functions require root authority, it is often undesirable to allow access to all root functions. This document applies to versions 3.2 and 4.1.

Menush Script

The menush script provides a menu of options and is invoked as an operator account in place of a shell. The first line below is not a comment and is required for the shell to work.

```
#!/bin/ksh
# This script is a menuing program that can be used to
# give operators access to superuser level commands,
# such as backup,   shutdown, kill, etc. without giving
# them a root shell.  This works by making this script the initial
# program of the users so that when they log in,
# they have UID 0, but do not get a shell prompt.  It
# requires a user defined startup file in the HOME
# directory of the "operator" account called .menushrc,
# which is defined as follows:
#
#    Menuname
#    1=Option Name=/path/command=prompt=page
#    99
#
# where Menuname is a one word name for the menu, followed
# by up to 15 option lines consisting of a option number,
# text for the option, command to run, prompt text or the
# word none, and the word yes or no to indicate if the
# output of the command should be piped through
# the pg. command.  The 99 at the end of the options is
```

```
# required to signal the end of the option lines.
#

# First we setup a trap that disables break (^-C) that the
# wiley operator might try to get a root shell.
trap "" 2

# Now define a function to process the user selected option.
process_option()
{
    ITEXT=$(grep "¬$resp=" $HOME/.menushrc | cut -d'=' -f2)
    PROMPT=$(grep "¬$resp=" $HOME/.menushrc | cut -d'=' -f4)
    CMD=$(grep "¬$resp=" $HOME/.menushrc | cut -d'=' -f3)
    PG=$(grep "¬$resp=" $HOME/.menushrc | cut -d'=' -f5)
    if [ "$CMD" != "" ]
    then
|       if [ "$PROMPT" != "none" ]
        then
            echo " "
            echo -n "\t\t\t$PROMPT"
            read input
            echo $(date)" "$ITEXT" "$input >>$HOME/menush.log
            if [ "$PG" != "yes" ]
            then
                eval $CMD $input
            else
                eval $CMD $input | pg
            fi
        else
            echo $(date)" "$ITEXT >>$HOME/menush.log
            if [ "$PG" != "yes" ]
            then
                eval $CMD
            else
                eval $CMD | pg
            fi
        fi
    else
    echo $(date)" "$resp" Invalid option" >>$HOME/menush.log
        echo "\t\t\tInvalid option"
        sleep 2
    fi
}

# Now we parse the $HOME/.menushrc file to get the valid
# menu options.  If this file does not exist, exit
# immediately.  If we find the file, use it to display
# the menu and co ntinue to do so until the user
# selects option 0 to exit.
  if [ -r $HOME/.menushrc ]
  then
    IFS='='
```

```
    resp="99"
    while [ "$resp" != "0" ]
    do
        exec < $HOME/.menushrc
        read menuname
        clear
        echo "\t\t\t\t$menuname Menu"
        echo " "
        read inum iname icmd iprompt ipg
        while [ $inum != "99" ]
        do
            echo "\t\t$inum\t$iname"
            read inum iname icmd iprompt ipg
        done
        echo "\t\t0\tExit"
        echo " "
        exec <&1
        echo -n "\t\t\tSelection: "
        read resp
    case $resp in
       "0") exit;;
       "1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9"|"10"|"11"|"12"|"13"|"14"|"15")
            process_option;;
       *)   echo $(date)" "$resp" Invalid option" >>$HOME/menush.log
            echo "\t\t\tInvalid option"
            sleep 2;;
    esac
    done
else
    echo "\t\tNo .menushrc found.  Exiting."
    sleep 2
    exit
fi
```

Setup Steps

1. Type this script in and save it as /usr/sbin/menush.
2. Run the following command to set permissions on the file.

   ```
   chmod 0750 /usr/sbin/menush
   ```

3. Create an operator user account and change the UID and GID to 0 in /etc/passwd. This can be done by:
   - Use SMIT to create an operator account using the defaults with the exception of the LOGIN user? prompt. Type "false" in this field.
   - Once SMIT has created the user account, edit /etc/passwd and change the operator line from:

     ```
     operator:*:201:10::/u/operator:/bin/ksh
     ```

201 and 10 are numbers assigned by the SMIT command:
operator:*:**0**:**0**::/u/operator:/**usr**/**sbin**/**menush**

NOTE: The fields above marked in bold are the ones that must be changed in /etc/passwd. Do not put these marks in /etc/passwd.

- Change directory to the home directory of the new account. Create the menu definition file that will be used by the menushscript. The .menushrc format is as follows:

```
Menuname
1=Option text=command=prompt=page
99
```

Where Menuname is the one word name for this menu, such as "Operator", followed by up to 15 option definition lines, followed by the number 99, which signals the end of the menu definition. A sample .menushrc file might look like this:

```
Operator
1=Kill a process=/bin/kill -9=PID to kill: =no
2=User Management=/bin/smitty security=none=no
3=Kill a print job=/bin/smitty qcan=none=no
4=Show status of print jobs=/bin/smitty qchk=none=yes
5=Schedule Jobs=/bin/smitty at=none=no
6=Change Operator Password=/usr/bin/passwd operator=none=no
99
```

In the above example, the menuname is Operator. Options that don't require further input from the operator have prompt set to none. All lines must start in the first column. The commands portion of the option definition line should specify the complete pathname to the command or script to be executed since no PATH is set for this account. This file should be owned by root and have read and write permission for the owner only.

- Assign an initial password to the operator account.

Now the operator will only be able to access this account by using the su command. This is done so that a record exists of when a user logs in as operator (this is recorded in the /var/adm/sulog). In addition, the script will write a log file in the operator $HOME directory called menush.log that records the date, time, and the menu option selected.

Security Concerns

This script can create security problems depending on the options that are available to the operator. An example of this can be seen in the sample .menushrc file above. In the option User Management, the SMIT fastpath to the Security & Users menu is invoked. Using this option, the operator could select the SMIT option to change the characteristics of a user and change his own initial program to a regular shell, thus allowing access to all functions reserved for root.

The system administrator should therefore take care when selecting what options to give to the operator account and test for possible security holes.

Please note that calling SMIT in an option might seem a security hole due to the fact that SMIT has an option to invoke a shell (F9), however if this option in SMIT is selected, it will invoke another copy of the menush script and not a root shell.

Reader's Comments

Please fax this form to (512) 823-4009, attention "AIXServ Information". You may also e-mail comments to: elizabet@austin.ibm.com. These comments should include the same customer information requested below.

Use this form to tell us what you think about this document. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

If you need technical assistance, contact your local branch office, point of sale, or 1-800-CALL-AIX (for information about support offerings). These services may be billable. Faxes on a variety of subjects may be ordered free of charge from 1-800-IBM-4FAX. Outside the U.S. call 415-855-4329 using a fax machine phone.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Note: If you have a problem report or item number, supplying that number may help us determine why a procedure did or did not work in your specific situation.

Problem Report or Item #: Branch Office or Customer #:

Be sure to print your name and fax number below if you would like a reply:

Name: Fax Number:

END OF DOCUMENT (operator.menu.32-41.cmd)