

KornShell 93 auditing

New features

Skill Level: Intermediate

[Dana L. French \(dfrench@mtxia.com\)](mailto:dfrench@mtxia.com)

President
Mt Xia Inc.

02 Sep 2008

New features of the Korn Shell provide system administrators and management with the ability to monitor, track, record, and audit every command executed by any user of a system. This is different from the normal shell history, and provides detailed information that includes date, time, tty, user, and the command. This information can be stored locally or transmitted in real time to a remote logging system.

Introduction

In today's computing environment, users and system administrators may be accessing systems located across the globe. The skill sets of these users and administrators vary widely between locations, and even within a location. The variation in skill sets of users induce risk into the management of these systems and must be managed and mitigated to protect the enterprise. It is the responsibility of management to protect the enterprise they manage, which includes the customers, employees, investments, information, data, and business functions, as well as adhering to regulatory and audit requirements. In order for management to fulfill their duties, the activities of users accessing these systems must be monitored, tracked, recorded, and audited. New features of the Korn Shell provide auditing capabilities that can help to fulfill these needs and requirements.

Korn Shell 93 auditing

The latest version of Korn Shell 93 as of the time of this writing is Release "ksh93s+" dated November 5, 2007, and provides a new compile option named "SHOPT_AUDIT." This option enables keyboard logging of any user and can be configured to store the audit information locally or remotely using Korn Shell networking. The information recorded by this feature includes each and every command executed, the date and time each command is executed, who executed the command, and on what tty it was executed. This option can be configured on a user-by-user basis and can include the root user. The audit information can be stored locally on each system in a file designated by the system administrator, or can be sent to a remote system for centralized logging.

Multiple Korn Shell binaries are provided with IBM® AIX®, depending upon the filesets installed:

- /usr/bin/ksh – Korn Shell 88
- /usr/bin/ksh93 – Korn Shell 93
- /usr/dt/bin/dtksh – CDE Desktop Korn Shell 93

The default shell when creating a new user is /usr/bin/ksh, which is Korn Shell 88, and is also the default shell for the root user.

Unfortunately, the version of Korn Shell 93 (/usr/bin/ksh93) provided with IBM AIX does not provide the "SHOPT_AUDIT" feature. Implementing Korn Shell 93 auditing features on an AIX system requires the system administrator to download and compile the latest version of Korn Shell 93 from AT&T. The remainder of this article contains instructions for downloading, compiling, and implementing the Korn Shell 93 "SHOPT_AUDIT" feature on an AIX system.

If you do not want to compile it yourself, a precompiled IBM AIX binary of Korn Shell 93 with the "SHOPT_AUDIT" feature is provided at <http://www.mtxia.com/css/Downloads/Scripts/Korn/ksh93.att.audit.bin>.

If the ksh93.att.audit.bin binary is downloaded, I recommend that it be copied to the file name "/usr/bin/ksh93.att" on the AIX system with the permissions set by the following commands:

```
cp ksh93.att.audit.bin /usr/bin/ksh93.att
chown bin:bin /usr/bin/ksh93.att
chmod 0555 /usr/bin/ksh93.att
```

Do not overwrite the existing Korn Shell 93 binary at /usr/bin/ksh93, as this may be modified by the next AIX system update.

Download and compile procedures for Korn Shell 93

Open the [The KornShell Command And Programming Language](http://www.kornshell.com) site at <http://www.kornshell.com> and click on **Software**:

Figure 1. Click on "Software"



Click on the link **The Official AT&T Release of KornShell 93**:

Figure 2. Click on "Official AT&T Release"

[The Official AT&T Release of KornShell 93](#)

AT&T has released KornShell as open source. Distrib

On the Software Download Packages page, click on the link **this copy of the License**:

Figure 3. Click on "this copy of the License"

[Public License Version 1.0 \(CPL-1.0\)](#). Before you download any packages you :
 ou to download. If you do not signify agreement then download access is denied.
 re and password, listed at the bottom of [this copy of the License](#). When you click
 a cancel button -- click that and the license page, with username and password, v
 id then repeat the download process. Packages covered by licenses other than C

Read the license and copy the line at the bottom of the license that reads: I
 accept www.opensource.org/licenses/cpl.

Figure 4. Copy line at the bottom of the license

PLEASE READ THE FOREGOING LICENSE AGREEMENT CAREFULLY. By using the following entire agreement and will be bound to use the downloaded software only in accordance with the license agreement by successfully entering the User Name and Password that is included, but is not limited to: clicking the "OK" button or typing the ENTER key on the browser or other HTTP or FTP interfaces.

When prompted for license agreement authorization use this User Name:

I accept www.opensource.org/licenses/cpl

and this Password:

.

(one period.)

Go back one page to [Software Download Packages](#) and click on the software package titled **INIT**.

Figure 5. Click on the software package "INIT"

| PACKAGE NAME | DESCRIPTION |
|--|--|
| INIT | the package command with support scripts and utilities |
| ast-ast | the ast library, period |
| { ast-make ast-ksh ksh ast-ast } | ast-base - ksh, pax, nmake, tw, sfio, and ast libraries |
| { ast-ksh-locale } | ast-base-locale - ast-base C locale messages with a few machine translations |
| | ast-gpl - ast GPL open source commands and libraries |
| { ksh } | ast-ksh - ksh and support libraries |

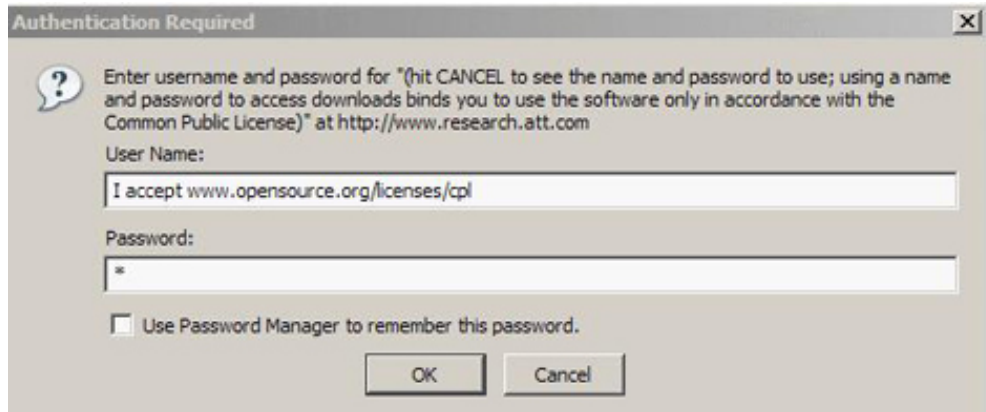
Download the latest **BASE SOURCE** file for **INIT**.

Figure 6. BASE SOURCE for INIT

| RELEASE | TYPE | SIZE | MD5 |
|-----------------------------------|--------|--------|-----------------------------------|
| 2008-02-02 BASE * | SOURCE | 347291 | 740c6fc775bf2f7b6bfff463bdbad1c31 |

An authorization window will appear. For the user field, enter the line of text that you saved previously (in [Figure 4](#)), I accept www.opensource.org/licenses/cpl. For the password field, enter a single dot ..

Figure 7. Authentication page



Go back one page to [Software Download Packages](#) and click on the software package titled **ast-ksh**.

Figure 8. Click on the software package "ast-ksh"

| PACKAGE NAME | DESCRIPTION |
|--|--|
| INIT | the package command with support scripts and utilities |
| ast-ast | the ast library, period |
| { ast-make ast-ksh ksh ast-ast } | ast-base - ksh, pax, nmake, tw, sfio, and ast libraries |
| { ast-ksh-locale } | ast-base-locale - ast-base C locale messages with a few machine translations |
| | ast-gpl - ast GPL open source commands and libraries |
| { ksh } | ast-ksh - ksh and support libraries |

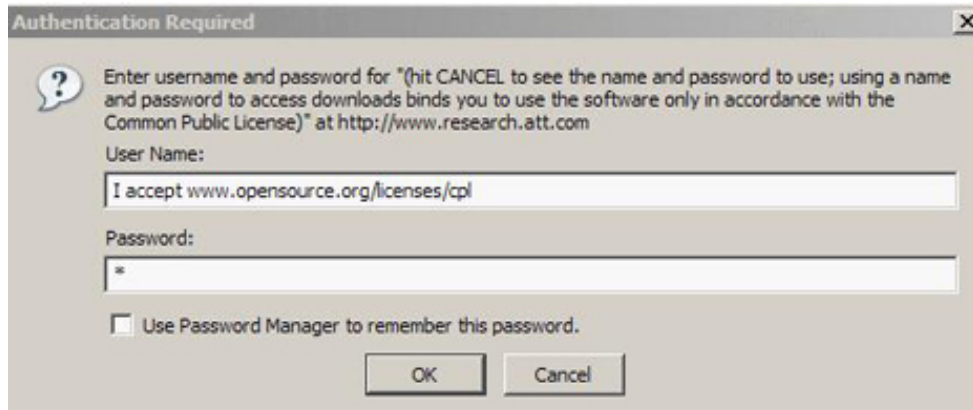
Download the latest **BASE SOURCE** file for "ast-ksh."

Figure 9. Download BASE SOURCE for ast-ksh

| RELEASE | TYPE | SIZE | MD5 | REQUIRES |
|---|--------|---------|----------------------------------|----------------------|
| 2008-02-02 BASE * | SOURCE | 1686309 | d2a71e220fbaa7a0fd950a27c7e4b099 | INIT |

An authorization window will appear. For the user field, enter the line of text that you saved previously (in [Figure 4](#)), I accept www.opensource.org/licenses/cpl. For the password field, enter a single dot ..

Figure 10. Authentication page



The following procedures assume the system on which these commands are executed contains a recent version of the GNU C compiler gcc, and the system is running AIX 5.3. These procedures are untested on anything else.

Download the files from the AT&T Website, create a directory in which to store the source code, and untar the files in that directory. Then run the following commands from the directory where the files were untarred:

- Create a shell variable called CC whose value is the full path file name to the gcc binary:

```
export CC="/usr/bin/gcc"
```

- Create a shell variable called URL whose value is the URL to the AT&T download Website:

```
URL="http://www.research.att.com/sw/download"
```

- Run the `package` command to set up the INIT base package source code to be compiled on the local system:

```
bin/package authorize "I accept  
www.opensource.org/licenses/cpl" password "." setup source ${URL} INIT
```

- Run the `package` command to set up the “ast-ksh” package source code to be compiled on the local system:

```
bin/package authorize "I accept www.opensource.org/licenses/cpl" password "." setup  
source ${URL} ast-ksh
```

- Run the `package` command to update the “INIT” package source code to be compiled on the local system:

```
bin/package authorize "I accept www.opensource.org/licenses/cpl" password "." flat setup
source ${URL} INIT
```

- Run the `package` command to update the “ast-ksh” package source code to be compiled on the local system:

```
bin/package authorize "I accept www.opensource.org/licenses/cpl" password "." flat setup
source ${URL} ast-ksh
```

This may seem redundant; however, it does seem to make a difference in the compile. If you need to recompile everything, do not simply remove *.o and *.a; remove all directories except "bin" and rerun the previous "bin/package authorize" commands.

Now modify the “SHOPT_AUDIT” option to enable this feature by editing the “Makefile” at the following location:

```
vi src/cmd/ksh93/Makefile
```

Change the value of “SHOPT_AUDIT” from “0” to “1” to enable the Korn Shell 93 auditing feature:

```
SHOPT_AUDIT == 1
```

Compile Korn Shell 93 by running the “package make” from the top-level directory where the packages were untarred:

```
bin/package make
```

Alternate download and compile procedure

As an alternative to using a graphical Web browser for downloading the Korn Shell 93 source code packages, the following procedure uses the character-based Web browser “wget” to download the packages. This procedure assumes the “wget” binary is installed on the AIX system where the packages will be downloaded:

- Create a **bin** directory in your user home directory:

```
test -d bin || mkdir bin
```

- Create a shell variable called **URL** whose value is the URL to the AT&T download Website:

```
URL="http://www.research.att.com/sw/download"
```

- Create a shell variable called **url** whose value is the URL to the AT&T download Website, with the “package” binary file name on the end:

```
url=${URL}/package
```

- Using the **wget** command, download the “package” binary file as follows:

```
(wget -O bin/package $url || curl $url || hurl $url) > bin/package
```

- Make the “package” binary file executable by the current user:

```
chmod +x bin/package
```

- Create a shell variable to contain the user name required by the AT&T download site:

```
U="I accept www.opensource.org/licenses/cpl"
```

- Run the **package** command to set up the “INIT” base package source code to be compiled on the local system:

```
bin/package authorize "${U}" password "." setup source ${URL} INIT
```

- Run the **package** command to set up the “ast-ksh” package source code to be compiled on the local system:

```
bin/package authorize "${U}" password "." setup source ${URL} ast-ksh
```

- Run the **package** command to update the “INIT” package source code to be compiled on the local system:

```
bin/package authorize "${U}" password "." flat setup source ${URL} INIT
```

- Run the **package** command to update the “ast-ksh” package source code to be compiled on the local system:

```
bin/package authorize "${U}" password "." flat setup source ${URL} ast-ksh
```

- Now modify the “SHOPT_AUDIT” option to enable this feature by editing the “Makefile” at the following location:

```
vi src/cmd/ksh93/Makefile
```

- Change the value of “SHOPT_AUDIT” from “0” to “1” to enable the Korn Shell 93 auditing feature:

```
SHOPT_AUDIT == 1
```

- Compile Korn Shell 93 by running the “package make” from the top-level directory where the packages were untarred:

```
bin/package make
```

- Once the compilation is complete, copy the new “ksh” binary to the file name “/usr/bin/ksh93.att” and set the ownership and permissions:

```
cp src/ksh93/arch/aix/bin/ksh /usr/bin/ksh93.att  
chown bin:bin /usr/bin/ksh93.att  
chmod 0555 /usr/bin/ksh93.att
```

Korn Shell 93 auditing configuration procedures

The newly compiled Korn Shell 93 binary with the “SHOPT_AUDIT” option enabled requires configuration on each system and for each user to be audited. This configuration designates where the audit information will be stored, either locally on each system or remotely on another system. The default configuration file is `/etc/ksh_audit`, and can be modified by changing the `SHOPT_AUDITFILE` value in the make file `src/cmd/ksh93/Makefile`. To change this value now would require a recompile.

The configuration file `/etc/ksh_audit` should contain one record line that defines the storage location for the audit information, followed by the UID number for each user whose commands are to be stored at the designated location. The configuration file record line will contain a single storage location and may be followed by one or more UID numbers. An example configuration record line might appear as:

```
/tmp/ksh_audit.out;201;202;207;251;330
```

This configuration record line designates the file `/tmp/ksh_audit.out` as the storage location for audit information, and identifies users corresponding with UID numbers 201, 202, 207, 251, and 330 to audit. The field delimiter is a semi-colon (“;”).

Example audit records stored in the `/tmp/ksh_audit.out` file would appear as follows:

```
201;1194497953;/dev/pts/1; ls
201;1194497954;/dev/pts/1; pwd
202;1194497955;/dev/pts/1; ls -alrt
207;1194497958;/dev/pts/1; ls -alrt /tmp/*.out
251;1194497963;/dev/pts/1; cat /tmp/ksh_audit.out
251;1210561980;/dev/pts/0; ls
207;1210561983;/dev/pts/0; pwd
330;1210561986;/dev/pts/0; ls -alrt /tmp/ksh*
202;1210561994;/dev/pts/0; cat /tmp/ksh_audit.out
330;1212266429;/dev/pts/2; set -o emacs
201;1212266437;/dev/pts/2; set -o vi
```

Again, the field separator is a semi-colon (“;”). The first field is the UID number of the user who issued the command. The second field is the date and time in seconds since the epoch. The third field is the tty on which the command was issued, and the last field is the command executed by the user.

Using the Korn Shell 93 networking capabilities, audit information can be stored at any remote location. This permits audit information to be stored on a centralized server to which only auditors have access. As an example, the following configuration record line designates the “syslog” network port (514) on a remote system:

```
/dev/udp/10.1.1.25/514;278;288;289;290
```

The remote location specified by this configuration record line is identified by the IP address “10.1.1.25” and network port “514.” The users whose audit information will be sent to this remote location are associated with UID numbers 278, 288, 289, and 290.

Example audit records stored by the remote syslog daemon would appear as follows:

```
May 31 22:26:34 10.1.1.33 278;1212287194;/dev/pts/4; pwd
May 31 22:26:35 10.1.1.33 278;1212287195;/dev/pts/4; df
May 31 22:26:36 10.1.1.33 278;1212287196;/dev/pts/4; du
May 31 22:26:38 10.1.1.33 278;1212287198;/dev/pts/4; ps -ef
May 31 22:26:51 10.1.1.33 278;1212287211;/dev/pts/4; exit
May 31 22:27:28 10.1.1.33 288;1212287248;/dev/pts/4; ls
May 31 22:27:33 10.1.1.33 290;1212287253;/dev/pts/4; ps -ef
May 31 22:27:34 10.1.1.33 288;1212287254;/dev/pts/4; pwd
May 31 22:27:37 10.1.1.33 289;1212287257;/dev/pts/4; df
May 31 22:27:37 10.1.1.33 288;1212287257;/dev/pts/4; du
```

These “syslog” records are formatted according to the configuration of the syslog daemon that is listening for incoming network connections. In this instance, each record contains the month, day of month, time, IP address of the system making the network connection, followed by the Korn Shell audit information previously described. The example records shown above represent the Korn Shell audit information as sent from a system with an IP address of 10.1.1.33.

An example script for converting the audit information to a comma-separated values file format follows. This script assumes the name of the audit information storage file is /tmp/ksh_audit.out. It reads the UID number and determines the associated user name from the /etc/passwd file.

```
#!/usr/bin/ksh93.att

PWDFILE="/etc/passwd"
AUDFILE="/tmp/ksh_audit.out"

print -- "username,uid,seconds,TTY,command"
while IFS=";" read UID SEC TTY CMD
do
  unset U
  while IFS=":" read USR PWD NBR REM
  do
    [[ "_${UID}" == "_${NBR}" ]] && U="${USR}" && break
  done < "${PWDFILE}"
  print -- "${U},${UID},${SEC},${TTY},${CMD}"
done < "${AUDFILE}"
```

Conclusion

The Korn Shell 93 audit feature provides a simple to use, configure, and maintain user-auditing mechanism. This feature can be used to monitor, track, record, and audit the activities of all users on an AIX system, including system administrators. This audit information can be stored on a remote system to which only auditors have access and can be easily processed using standard AIX tools. The need for this feature has arisen from the globalization of system administration and user access. In order to insure business continuity, the activity of all administrators and users must be auditable, and they must be aware their activities are being audited. The awareness of an organizations policy of command-by-command auditing will help to reinforce a mentality of system integrity, reliability, and business continuity.

Resources

Learn

- [The AIX and UNIX developerWorks zone](#) provides a wealth of information relating to all aspects of IBM® AIX® systems administration and expanding your UNIX skills.
- [New to AIX and UNIX?](#) Visit the New to AIX and UNIX page to learn more.
- [developerWorks technical events and webcasts](#): Stay current with developerWorks technical events and webcasts.
- [Podcasts](#): Tune in and catch up with IBM technical experts.
- [Korn Shell source code and documentation](#)
- [Pre-compiled Korn Shell 93 binaries with shell auditing enabled](#)
- [Korn Shell 93 Quick Reference Guides](#)
- [Korn Shell 93 Example Scripts](#)

Get products and technologies

- [IBM trial software](#): Build your next development project with software for download directly from developerWorks.

Discuss

- Participate in the AIX and UNIX forums:
 - [AIX Forum](#)
 - [AIX Forum for developers](#)
 - [Cluster Systems Management](#)
 - [IBM Support Assistant Forum](#)
 - [Performance Tools Forum](#)
 - [Virtualization Forum](#)
 - [More AIX and UNIX Forums](#)

About the author

Dana L. French

Mr. French's career in the IT industry has spanned three decades and numerous

industries. His work focuses primarily on the fields of business continuity, disaster recovery, and high availability, and he has designed and written numerous software packages to automate the processes of disaster recovery and high availability. He is most noted for his approach to system administration as an automated, business oriented process, rather than as a system oriented, interactive process. He is also a noted authority on the subject of Korn Shell programming.

Trademarks

IBM and AIX are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.