

Fundamentals of UNIX
Lab 15.2.1 – Customizing the Korn Shell
(Estimated time: 45 min.)

Objectives:

- Develop an understanding of Korn shell customization
- Review initialization files, variables and commands
- Review system-wide initialization files
- Review user-specific initialization files
- Modify your .profile file
- Create your .kshrc file
- Test your customized initialization files

Background:

In this lab, you will work with UNIX initialization files to customize your **Korn** shell login environment and tailor various options to your needs. In the previous lab, you worked with aliases and custom prompts that were only active for the current session. When you closed the terminal window or logged out, the custom settings were lost. These and other features can be automatically made available each time you login by modifying specific initialization files. In this lab you will work with various initialization files to make these changes take effect each time you login. You will also review system-wide and user-specific initialization files and variables.

Tools / Preparation:

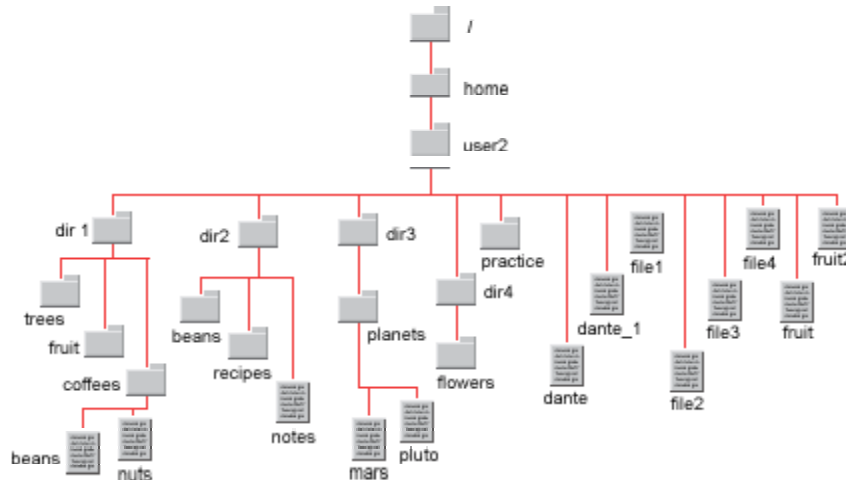
- a) Before starting this lab, review Chapter 15, Section 1 – Initialization Files and Section 2 – Shell Customization
- b) You will need the following:
 - 1. A login user ID (e.g. user2) and password assigned by your instructor.
 - 2. A computer running the UNIX operating system
 - 3. Networked computers in classroom with class file system installed

Notes:

Fundamentals of UNIX
Lab 15.2.1 – Customizing the Korn Shell
Worksheet

Use the diagram of the sample Class File System directory tree to assist with this lab.

Class File Tree Structure



Step 1. Log in to CDE

Login with the user name and password assigned by your instructor in the CDE entry box provided.

Step 2. Access the Command Line

Right click on the **workspace** backdrop and click on **Tools**. Select **Terminal** from the menu to open a terminal window.

Step 3. Review Initialization Files, Variables and Commands

Initialization files – Initialization files contain a series of commands and variable settings that are executed when a shell is started. These files are not executable but are read by the shell when the user logs in to customize their environment. There are two levels of initialization files. The first level is **system-wide**. System initialization files are maintained by a system administrator and reside in the **/etc** directory. The second level is **user-specific** initialization files that reside in a user's home directory. The Korn shell user has two files: **.profile** and **.kshrc** (Korn shell run control) to customize their environment. Note that both of these have a dot (.) as the first character which means they are hidden.

Variables - In the previous lab, you worked with variables like **PS1** (prompt string 1) to customize your Korn shell prompt. Since these were defined from the command line and were not put in an initialization file, they were only active for the current shell and when that shell was closed, they were lost. With this lab, you will edit the initialization files and add variables and commands so that they will be available every time you log in. Variables are either pre-defined and set automatically when you login, or set by the user (or system administrator for the user). They can apply to the current shell only (local) or to all shells and sub-shells (global) depending on which initialization file they are in. Users can customize many of these variables for their own environment by placing them in the initialization files in their home directory. To make the customizations available to all subsequent shells, the variable must be exported.

Commands - Unlike variables, commands (like aliases) cannot be exported (made available to all sub-shells). If commands are to be available in a subshell, they must be placed in a secondary initialization file that is read each time a subshell is opened such as the **.kshrc**.

Fundamentals of UNIX
Lab 15.2.1 – Customizing the Korn Shell
Worksheet – Cont.

Using the information above fill in the blanks in the following sentences.

- a. The two levels of initialization files are _____ and _____.
- b. For the Korn shell user, the _____ file and the _____ file reside in the user's home directory and can be used to customize their login environment.
- c. _____ are used to customize your shell and are either global or local.
- d. In order for a variable to be available to sub-shells, it must be _____.
- e. If commands are to be available in a subshell, they must be placed in a _____ file that is read each time a subshell is opened such as the **.kshrc**.

Step 4. Review System-wide Initialization Files

The primary system-wide initialization file is the **profile** file which is kept in the **/etc** directory. The **/etc/profile** file is created by default when the operating system is installed and can be edited and customized by a **system administrator**. The **/etc/profile** applies to all users. When a Korn shell user logs in, the system reads the **/etc/profile** file first, and then it reads the user's **.profile** file and then the **.kshrc** file. This means that the user's preferences for variable settings can override the default settings set and maintained by a system administrator in the **/etc/profile** file. The **/etc/profile** file performs several functions, some of which are listed below:

Exports Environment Variables - Makes environment variables available to sub-shells such as LOGNAME for login name.

Exports the PATH variable for Default Command Path - The default path is a list of directories where the shell will look when a command is executed. Exporting it makes the list of directories available to all shells and sub-shells.

Sets the TERM Variable Default Terminal Type - This defines the screen and keyboard characteristics of your workstation.

Displays Contents of /etc/motd File - The 'message of the day' file can be customized to display greetings or provide system information.

Sets Default File Creation Permissions - Sets umask values that determine the default permissions when a new file or directory is created.

Using the information above fill in the blanks in the following sentences.

- a. The primary system-wide initialization file is the _____ file which is kept in the _____ directory.
- b. The **/etc/profile** file is created by default when the operating system is installed and can be edited and customized by a **system administrator**.
- c. When a Korn shell user logs in, the system reads the **/etc/profile** file first, and then it reads the user's **.profile** file and then the **.kshrc** file.

Lab 15.2.1 – Customizing the Korn Shell
Worksheet – Cont.

- d. The **/etc/profile** file exports the _____ variable which defines a list of directories where the shell will look when a command is executed.
- e. The **/etc/profile** file sets the _____ variable that defines the screen and keyboard characteristics of your workstation.

Step 5. Review User-specific Initialization Files

After the system-wide initialization file is read that sets system-wide defaults, the user initialization files for the shell are read. The user initialization files provide great flexibility to the user for customizing their environment. Login initialization and customization can be accomplished using the system-wide file **/etc/profile** by itself or in combination with the user-specific initialization files.

The user initialization files can be set up as templates by the system administrator, and then modified by the user. The user-specific initialization file(s) **.profile** and **.kshrc** are stored in the home directory of the user. The **.kshrc** file is the primary file that is used to customize the Korn shell user's environment. Depending on network administration policy, user-specific initialization files can perform all or part of the following:

Set Default Prompt - As in the previous lab

Define Default Printer - Provides access to a printer

Set Default Permissions - Set umask for new files and directories

Set Default Terminal Type - Used by vi and other tools

Set noclobber - Prevent overwriting of files during redirection

Set Command Path - Defines directories to look in for executable files

Define Custom Commands - Defines aliases as described as in previous lab

Using the information above fill in the blanks in the following sentences.

- a. After the _____ is read for system-wide defaults, the _____ for the shell are read.
- b. Login initialization and customization can be accomplished using the system-wide file _____ by itself or in combination with the user-specific initialization files: **.profile** and _____
- c. The _____ file is the primary file that is used to customize the Korn shell user's environment.
- d. Among other things, the _____ user initialization file can be used to define _____ as described as in previous lab as substitutes for commands.

Step 6. Modify your .profile File

The system administrator, as part of creating a user account, assigns the login shell. The login shell determines which initialization files are read during login. The Korn shell uses two user-specific environment files to set the user's environment: **~/.profile** and **~/.kshrc**. The **~/.profile** typically contains one-time-only commands and variable definitions.

Fundamentals of UNIX
Lab 15.2.1 – Customizing the Korn Shell
Worksheet – Cont.

When a new Korn shell user is defined, a basic version of the **.profile** file is created and placed in their home directory. To customize their environment, the user needs to modify the **.profile** file and they must define and export the **ENV** variable as shown below. The **ENV** variable defines the path to the **.kshrc** file. These commands are usually added to the end of the **.profile** file. They inform the system that the **.kshrc** file exists and is to be read each time a Korn shell is created. This command sets the environment variable to point to the **.kshrc** file in the **\$HOME** directory. **HOME** is a variable that is defined by the system to be the absolute path to the user's login directory

Command Format: `ENV=$HOME/.kshrc; export ENV`

- a. Verify that you are in your home directory and then use the **ls -la** (long list of **all** files including hidden ones – you may need to pipe this command to **more**) command to determine if the **.profile** file exists. In most cases it is created automatically when a new user is defined (If not you will need to create it). Who is the owner of the **.profile** file? _____ What are your permissions for this file? _____ Should you be able to make changes to it? _____
- b. Use the **more** command to view the contents of the **.profile** file. Note the **PATH** variable at the beginning, which defines directories where the system will look when you issue a command. What is the **PATH** variable set to? _____
- c. Is the variable exported so it can be used in sub-shells? _____
- d. The **paths** in the **PATH** variable definition are separated by **colons** (:). Is **/usr/bin** one of the paths listed? **Yes**. List the contents of the **/usr/bin** directory and **pipe** it to the **more** command. Write down at least five of the commands listed that you are familiar with.

- e. Copy the **.profile** file to create a backup of it called **.profile.bak** for safe keeping. If you make mistakes and need the backup, you can remove **.profile** and rename **.profile.bak**
- f. Use **vedit** to modify the **.profile** file. Add the ENV and export commands shown above, in the command format line, to the **end** of the file and then **save** and **quit** vedit. View the contents of the file again. Are the commands you entered present? **Yes**. Check them to verify they were type in correctly.

Step 7. Create your .kshrc File

Korn shell commands and features should be placed in the **.kshrc** (Korn shell run control) file. The contents of the **.kshrc** file typically include: A customized prompt, aliases, activation of shell features and custom variables

- a. There is not usually a basic version of the **.kshrc** in the users home directory unless an administrator places one there as a template for a starting point. You must create one or in some cases you can copy an existing one from another user.
- b. Create a new **.kshrc** file using **vedit**. What command did you enter? _____
- c. If you want to save your changes and exit the vi editor what Last-line command would you use? **:**

- d. If you want to exit the editor without saving any changes what Last-line command would you use?

Fundamentals of UNIX
Lab 15.2.1 – Customizing the Korn Shell
Worksheet – Cont.

- e. Set your prompt to the name of your machine with a dollar sign and a space after it. What did you enter? _____
- f. Set the vi editor to be activated for command line editing. What did you enter? _____
- g. Create an alias called **h** as a short cut to represent the **history** command and another alias called **c** as a short cut for the **clear** command.
- h. Create an alias called **p** as a short cut to represent the **ps -ef | sort | more** command. What command did you enter? _____
- i. Create an alias called **cp** to replace the regular cp command that would prompt you **interactively** if you were about to overwrite a file. What command did you enter? alias _____
- j. Save your changes to the **.kshrc** file and **exit** vi. You can exit without saving changes if you want to start again.
- k. Use **vi** or the **more** command to view your **.kshrc** file and verify that the commands were entered correctly.

Step 8. Test your Customized Initialization Files

Now that you have modified the **.profile** file to point to the **.kshrc** file in your home directory and also created a new **.kshrc** file, it is time to test the results.

- a. Close the terminal window and Logout (EXIT) of the system.
- b. Login again as the same user.
- c. Has your prompt changed as expected? _____
- d. Try the aliases you defined in previous steps. Do they all work? _____
- e. If none of the customization features worked, you may have a problem with the ENV variable in the **.profile** file. If some things worked and some didn't you probably have some mistakes in your **.kshrc** file.
- f. Edit the **.profile** and **.kshrc** files as necessary to get all of the customization changes to work.

Step 9 – Remove Files and Directories Created in this Lab

Remove all files and directories created in you home directory during this lab.

Step 10. Close the Terminal Window and Logout

Double click on the dash button in the upper left corner of the screen, then click the **EXIT** icon on the front panel.