

BIL235 Experiment V

Hacettepe University Department of Computer Science and Engineering

BIL235 Programming Laboratory Experiment 5

Subject:	Writing UNIX Shell Scripts
Advisors:	Prof. Dr. Ali SAATÇI R.A. Kerem ERZURUMLU
Submission Date:	23/11/2007
Deadline:	10/12/2007
Programming Language:	Bourne Again Shell (bash), ANSI C

AIM

The aim of this experiment is to get students familiar with both the session and the programming environment of the Unix (Linux) Operating System.

Unix and its variants have become the de-facto (i.e., not in writing but by common consent) industrial standard for multitasking server systems. In recent times, flavors of Unix (Linux) have been ported to machines with limited hardware, such as PC's (Personal Computers). A working knowledge of Unix will probably be the best advantage of a computer engineer in the future.

BACKGROUND

With Unix, users can access the system via a terminal. After the authentication phase (login) a special program called "shell" runs for each user outputting a "prompt" (normally a \$ sign) on the screen. A shell can be described as a layer between the user and Unix itself. It is a command line interpreter through which a user can execute ready made programs called shell commands such as `ls`, `ps`, `cp`, `mv`, `cat` etc. A shell has also programming capabilities like loops and conditional executions. It then gives users the possibility to write "scripts" to define a specific job as a whole. A script is a collection of commands (or executable programs) along with loops and conditions. It can be considered as a program where the statements (instructions) are the commands. A shell script is stored in a file with a given name. It is executed every time this file name is stated.

The most common shells for Unix are `sh` (Bourne Shell) and `bash` (Bourne Again Shell). In this experiment, you have to use `bash`. For its usage, please refer to the documents defined in the References.

OBJECT

In this experiment you are asked to write a shell script executing 3 different tasks as a whole every night with the help of `crontab(1,5)`¹. You are also asked to develop two small C program in order to use within your script for output creation purposes.

Task A: Create an information file for each user defined on the system including the following fields:

¹ Numbers in paranthesis indicates that the manual pages section.

BIL235 Experiment V

1. Date (Day on which the script runs)
2. User real name, User ID,
3. Group name,
4. Total login time in minutes for the day,
5. Total disk space used by user's home directory.

This file should be created following a text template including the above fields. The text part of the template should be made changeable.

Task B: Compress using `tar(1)` files in defined directories within the script and Back-up them under a given destination directory also defined in the script.

Task C: For each user Find files with a given extension under the users' home directory and Send the output list as an e-mail to the admin's e-mail address.

C Programs: You have to write two simple program in order to use it in your script to create output. The first program will get the user list of the system to work with, by using `getpwent()` system call.

The second program will get the username as parameter and it will gather the required values by "Task A" by using `getpwnam()` system call.

REFERENCES

1. Unix's on-line manual system which is accessed by command `man(1)`,
2. Using UNIX (CS Library No=122),
3. Advanced Unix Programming (CS Library No=139,151),
4. <http://www.linux.org.tr>
5. <http://www.google.com>
6. <http://www.penguin.net>

Please read also the following manuals: `cut(1)`, `sed(1)`, `awk(1)`, `grep(1)`, `mail(1)`, `bash(1)`, `for(1)`, `if(1)`, `diff(1)`, `let(1)`, `expr(1)`, `last(1)`

NOTES

1. You are asked to give your `Makefile` (a single makefile for both) with your program.
2. Backup files must include their creation date.
3. In **Task C** more than one file extension should be possible.
4. You are asked to follow announcements made to `Courses.Bil235` newsgroup which is located at `nntp://news.cs.hacettepe.edu.tr` .
5. Your report and program must be submitted at the same time.
6. Your programs will be checked with an automated Plagiarism Detector.
7. Valid Soft-Copy formats for your report are HTML and PDF.
8. E-mail submissions are not accepted.
9. Late submissions will not be accepted.
10. At the submission your experiments will be checked for common errors. Works with this kind of errors may be refused for grading. Do not leave everything to the last minute.
11. Please send any additional questions to `Courses.Bil235` newsgroup.

Good Luck