

Implementation Final Report - “Command Line Literacy”

Abstract

This paper describes the development and implementation of a learning tool intended to facilitate *command line literacy*. Command line literacy is defined as proficiency in utilizing a command line interface (CLI) to complete tasks within a computer software environment. The learning tool combines instruction with interactive Web-based tools, and asks users to complete activities related to CLIs. These activities evaluate user ability to gather and sort information, synthesizing Internet resources with questions related to online reading comprehension (ORC). The structure of this paper will follow that of the model used in the design of the learning tool, the ADDIE instructional design model, and concludes with a summary and interpretation of results from sample students that completed the activities.

Introduction

Many operating systems rely upon a command line interface (CLI). CLIs allow users to interact with computer software by typing commands, often combining a series of commands to achieve a goal. Commands may interact in intricate ways, with the output of each command processed by another. For this reason, CLIs are utilized for complex tasks by system administrators, computer programmers, and researchers in a variety of fields.

The CLI is the predecessor of the graphical user interface (GUI), but is still central to many operating systems (especially Unix or Unix-like systems). Command line interfaces have several advantages when contrasted with GUIs, most notably in tasks requiring batch processing or the sorting of large amounts of data. CLIs are also ideal as a problem-solving medium, as they produce real-time output for the user, including detailed error messages that would be cumbersome in a GUI environment.

In modern operating systems, GUIs hide many complex actions and configuration options. CLIs allow users to access these actions and options, configure/customize their systems to a greater degree, secure their system and investigate security vulnerabilities, and debug and potentially fix problems that arise. In this way, CLIs provide an ideal Constructivist learning environment, one where users are empowered and actively constructing knowledge of the software interface.

However, most computer users today prefer to use GUIs to perform common operations and have little or no knowledge of CLIs. This is an unfortunate consequence of GUI popularity, and relegates many users to the role of simply a consumer, unable to take full advantage of the software they have become increasingly reliant upon. Also, they are unable to reap the full benefits of automating monotonous tasks in a CLI. Rather than recognizing the strengths of GUI tools (e.g. graphic manipulation and Web browsing) along with their weaknesses (e.g. batch processing of files, text manipulation), many users incorrectly assume the GUI is the only available software environment.

The goal of the learning tool outlined in this paper is to encourage *command line literacy* in students. Command line literacy is defined as proficiency in utilizing a command line interface (CLI) to complete tasks within a computer software environment. Such an understanding will allow students to

recognize the strengths and weaknesses of CLIs and utilize the optimal tools for specific software tasks (e.g. GUI tools for editing photos, CLI tools for automating data backups).

Even a basic understanding of CLIs is an important skill to possess, one that may potentially open new doors for a future profession. The Internet relies heavily upon CLIs for its operation, and most Web servers do not have GUIs installed. Given the prominence of CLIs and their important role in computer systems, command line literacy is a valuable tool for students to possess.

Analysis

During the analysis phase of this project, the following information was determined. Discussions with peers (university students) were very valuable during this phase and influenced the target audience of the learning activities.

Audience

Users of this learning tool will be university students, both on the undergraduate and graduate level. These students have varying levels of proficiency with software, but most own a personal computer.

Rationale

Students currently enrolled in college spend a large amount of time on the computer, and most own a desktop or laptop computer. Software is essential to their academic lives, and their coursework becomes more closely integrated with the Internet each semester. Although these students are proficient in GUI environments, a CLI might be utilized to empower them as users, as well as better prepare them for interaction with operating systems in their future professions.

Goal

The goal of the learning tool is to encourage command line literacy in students, allowing them to recognize the strengths and weaknesses of CLIs as well as the GUI tools they are already familiar with.

Objectives

After completing the learning module and activities that are central to this project, students should be able to:

- 1. Locate information
- 2. Critically evaluate information
- 3. Synthesize information from multiple sources
- 4. Utilize that information for basic CLI tasks
- 5. Communicate the results efficiently to others

Delivery Options

The learning module and activities will be available in the form of a wiki website, hosted at <http://seandiggity.dreamhosters.com/epsy381wiki>. Portions of the module and activities may be better utilized in some classrooms if printed and handed out as worksheets.

Constraints

- *Software Compatibility:* Without consistency of computer systems, it is difficult to focus learning activities upon a specific CLI tool. Although this was overcome to a large extent due to the implementation of Web-based CLI tools, differences in software environments cannot be overcome completely.
- *Personal Interaction with Peers and Instructors:* The learning module and activities do not encourage group work, nor is there an active role for the instructor to play. Students are mainly left to explore on their own, utilizing their critical literacy and online reading comprehension skills to complete the evaluations. This may be troublesome for students weak in these areas, or those disinterested with computers and software.
- *Accessibility:* Vision and motor skill impairments could hinder users completing the learning activities. CLI screens are not well-suited for users with visual impairments. Users with difficulty typing may also have a rough time exploring a CLI, due to the large strings of text it may be necessary to type.

Design

In the design phase of this project, experimentation was essential. The project's focus upon software tools introduces limitations, and it quickly became apparent that certain approaches were unfeasible. Thus, not much time was lost in exploring dead-end approaches. However, without exploring multiple designs, the project might never have reached a level of maturity.

Concept 1: Native CLI Application (shell)

Four laptops, each running an Ubuntu Linux LiveCD, were set up as the focal point of the project. Utilizing the Gnome Terminal CLI, users would follow an online tutorial, completing an evaluation at the end of the learning activities.

Pros	<ul style="list-style-type: none"> 1. Users were able to utilize a full CLI application (shell) in a Unix-like operating system, an environment very conducive to CLI interaction. 2. Users could compare and contrast different CLIs, with many shell applications readily available. 3. Many display options for the shell applications were available, perhaps overcoming limitations for the visually impaired.
Cons	<ul style="list-style-type: none"> 1. Running the Ubuntu LiveCD (completely in RAM memory) was slow, and installation of proprietary hardware drivers was cumbersome. Without these drivers, the laptop wireless Internet connections would not work. 2. The interface was perhaps too foreign for users accustomed to either the Windows or Mac OSX environments. Time was lost explaining nuances of Gnome (the default desktop environment in Ubuntu) and user exploration of GUI applications.
Concept 2: "Screencast" Videos	
"Screencast" video tutorials were recorded to coincide with the learning activities.	
Pros	<ul style="list-style-type: none"> 1. Students would have a visual stimulus to keep them interested in the material. 2. Students would be able to compare and contrast the results of their CLI commands with those in the videos.
Cons	<ul style="list-style-type: none"> 1. The interaction with the screencasts was mainly passive, not actively inviting students to interact with a CLI. 2. Students would be able to glean answers to the evaluation questions and "cheat" on the learning activities, to some degree, without exploring the CLI. 3. A video of a user entering text commands into a CLI is not very entertaining for most users, and may be difficult to follow along with.
Concept 3: Web-based CLI Tools	
Web-based command line programs were implemented, simulating specific aspects of interaction with a CLI environment. These coincided with full tutorials on a basic Unix command, <code>chmod</code> , and other Web resources.	

Pros	<ul style="list-style-type: none"> • 1. Students are able to interact with a CLI without concern for software compatibility, as the software requirements are only a modern Web browser with JavaScript enabled. • 2. The tutorials accompanying the CLI tools introduced concepts slowly to students, allowing them to explore as well as retain information. • 3. A focus on presentation as a Web project allowed for better integration of learning tools, including external Internet resources. • 4. The Web format allowed for greater integration with the concept of online reading comprehension and other new literacies.
Cons	<ul style="list-style-type: none"> • 1. Configuration of the Web-based CLI tools was time-consuming, and integration with the text and tutorials took more careful consideration than initially planned. • 2. The learning activities became less spontaneous and more constrained with the integration of the chmod tutorial and CLI tools, as a specific command and its related concepts became central to the project.

During the design phase, I focused on the following considerations:

- **Ease of use/access:** The design of the project is very simple, keeping to a basic graphical format and simple layout that can be viewed uniformly in different Web browsers and on different operating systems. The Web tools chosen for the project are all easily viewable in any modern browser, and do not require special plugins.
- **New literacies:** The concept of command line literacy is closely related to other new literacies, and the goals of this learning project are similar to those of the Online Reading Comprehension Assessment Blog (ORCA-Blog). While formulating the concept of command line literacy, I tried to keep the end-users in mind. The activities focus on independent exploration, very much rooted in the Constructivist conception of learning.

Development

Early on in the design process, I decided that MediaWiki, the wiki software that runs Wikipedia, would be the ideal platform for this learning project. MediaWiki allows for a very clean, organized layout and easy manipulation of text and graphics. Furthermore, I already had some development experience with MediaWiki. The final version of the learning project contains many tweaks, which include:

- Fine-grained permissions management, for restricting public editing of all pages and viewing of others. This level of control was also necessary to keep student responses isolated so that they would not influence each other.
- Modified search engine, which allows for quick searching with Google, so that users may easily find external Internet resources related to the learning activities.
- Restricted toolbox and toolbar, so that the "Print Version" tool is the only one available to students. Otherwise, they would have much broader access to the wiki, access which could compromise the results of the learning activities.

- Forced "new window" links for external websites. This allows students to explore external resources while staying on the website.
- Modified comments extension, which allows students to post feedback and their responses to the learning activities.
- Flash demo extension, which is functional but currently unused (see Concept 2 above).
- Interactive quiz and quiz results extension, which was superseded by the simpler and less-strict format of the comments extension.

Two additional tools were also integrated into the project, a tutorial on the Unix `chmod` command (which includes interactive command line simulators) and a `chmod` calculator (which calculates numeric values for Unix file permissions). Both of these tools were freely available on the Internet, but required slight modification to function correctly within the scope of this project.

I utilized Wikipedia heavily for this project, linking to it frequently to define computer-related terms within the learning module and activities. I also relied upon it for graphics, paying close attention to the licensing information of all content.

My development process focused upon these concerns:

- Simplicity of Appearance
- Platform consistency
- Clear Navigation
- Printable Content
- Web Standards
- Accessibility

Overall, I feel the approach was successful. The simplification of this learning project allows users to focus on the content, without becoming sidetracked by plugins or visual overload.

The project runs upon the LAMP stack (Linux/Apache/MySQL/PHP), and it is fitting that I had to utilize a CLI often to install and configure the project.

Implementation

The learning project was initially made available to a small sample of five university students, ages 20-26, with varying backgrounds. Of these five students, three responded fully to the learning activities.

These students were given access to the project at <http://seandiggity.dreamhosters.com/epsy381wiki> and were asked to review the learning module before moving on to the activities. The learning module consisted of a broad overview of command line literacy and command line interfaces, including examples of modern CLIs in various operating systems. A series of brainstorming questions were posed to the students before moving on to information about more specific CLI usage. In this way, the value of command line literacy was explained to the students before they "got their hands dirty".

Before accessing the learning activities, students were exposed to two interactive tools that focused on a simple Unix command, `chmod`. This command provides an appropriate introduction to CLIs, since it introduces basic concepts about filesystems and allows for non-destructive interaction with files. These two tools consisted of a detailed tutorial on `chmod`/Unix file permissions, and a calculator of numeric `chmod` values.

These tools exposed the students to the instant, detailed feedback of the command line while also introducing them to multiple methods of manipulating files. Such CLI utilization can lead to creative, even playful, interaction with the software environment (albeit with more complex commands than `chmod`). This type of empowerment is in contrasts starkly with most GUI environments, where users are typically pigeon-holed into a few menu or icon choices.

After spending some time with the `chmod` tutorial, students moved on to the learning activities, which challenged them with questions based upon their exposure to the CLI tools. The learning activities consisted of three scenarios (Susan's Homework, Juan the Sysadmin, Seila's Spreadsheet), each with a consistent format.

Questions in each learning activity challenge students to:

- 1. Understand strengths and weaknesses of the CLI
- 2. Locate information from the tutorial and apply it to an example
- 3. Clearly communicate CLI concepts to others
- 4. Complete tasks with CLI tools
- 5. Search and locate Web resources with a search engine
- 6. Critically evaluate Web resources

Evaluation

Evaluation of the command line literacy learning project was built into its design. The last question of each of the three activities was always, "Please list any of the resources you may have used to answer the questions above." In addition to this, students that participated in the activities were asked to respond to separate feedback questions.

Initial reactions seem to be positive, although the sample size of students that have completed the learning activities is very small. Even so, the student feedback highlights a few concerns:

- 1. The amount of information contained in the `chmod` tutorial is perhaps too much, making retention of the information less likely.
- 2. There is a perception that CLIs are old-fashioned, and that GUI tools have replaced them completely. Even though the learning module attempts to counter this perception, it is very strong amongst the target audience (and anyone younger) who have never used a terminal or mainframe before.
- 3. The supplied resources are not being utilized to complete the learning activities. Since the activities focus upon the `chmod` tutorial and the text in the learning module, users seem unlikely to view the resources listed unless out of curiosity.

I also recorded one user while browsing the project and interacting with command line tools, but he was not one of the students that responded to the learning activities. That video is [available here](#) (AVI).

Results

I created scoring rubrics for the activities (attached), maintaining a strict format for evaluation. Although the scenarios for each of the activities differed, each question related back to identical criteria in the rubrics.

Results for the three students (Miguel, Cherise, Dan):

Activity I (15 points possible)			
	Miguel	Cherise	Dan
1	3	2	1
2a	2	2	2
2b	2	2	2
3	2	1	2
4a	2	2	0
4b	2	2	1
Total	13	11	8

Activity II (15 points possible)			
	Miguel	Cherise	Dan
1	2	2	1
2a	2	2	2
2b	2	3	1
3	2	1	2
4a	2	2	2
4b	1	1	1
Total	11	11	9

Activity III (15 points possible)			
	Miguel	Cherise	Dan
1	1	2	1
2a	2	0	2
2b	2	1	2
3	2	1	2
4a	2	2	2
4b	2	2	1
Total	11	8	10

The actual student responses, as well as responses to the feedback questions, are attached.

Conclusions

Given the small sample size, results of the learning activities are difficult to interpret. Responses from each of the students varied in their depth, and at least one student (Dan) seemed to rush through the activities. However, the short or partial responses to the learning activity questions were almost always related to command line concepts, and demonstrated at least a basic proficiency with CLIs.

Apparent weaknesses of the learning module and activities:

1. The chmod tutorial is too long and explores that chmod command in too much depth for users unfamiliar with CLIs. Rather than facilitating a broad perspective for students on an unfamiliar type of user interface, the tutorial focuses on specifics tied too closely to Unix and Unix-like operating systems.
2. Users have difficulty synthesizing the resources provided, and seem inundated with information. This led to unclear or inaccurate responses, and limited reflection on the CLI concepts the activity is meant to reinforce.
3. Mixing online reading comprehension questions (4a and 4b in the lesson rubrics) with specific CLI-related questions may have detracted from the lesson objective. Respondents seemed unable to “switch gears” in some cases, just finding a quick search engine result and using it as their answer without much reflection on the Web resource itself, let alone how it related to CLIs. For example, one response given linked to a site that was unrelated to the topic, although it could be easily misinterpreted as related to the topic if not properly evaluated (or evaluated solely based upon its appearance).

Apparent strengths of the learning module and activities:

1. The introduction and CLI examples in the learning module were sufficient to facilitate understanding of CLIs and their place in modern operating systems. Respondents were generally able to compare and contrast CLIs with GUIs and did not seem to have trouble grasping the optimal scenarios for utilizing each.
2. The scenarios were clear enough for students to grasp them, even though they related to technical situations many students would not have encountered in casual computer usage (especially GUI interaction).
3. The Web-based CLI tools (those included in the chmod Tutorial as well as the chmod calculator) were accessible to respondents, and were properly utilized for technically-specific questions. Respondents had a very high success rate on questions approaching file permissions and directly challenging them to use these Web-based CLI simulators.
4. Respondents were able to easily navigate the command line literacy website, with no trouble accessing relevant information and interacting with the environment. Responses were recorded without error, and students utilized off-site Internet resources without abandoning the learning activities.

These strengths and weaknesses may require a change in the design of the learning activities, as well as the target audience for the entire project and the circumstances in which students completed the

learning activities. After reviewing the responses to the learning activities as well as the user feedback (attached), it seems apparent that a more structured approach to the learning activities would better reinforce the objectives of the project. A classroom environment, perhaps a high school classroom, would allow for more direct instructor interaction with students and allow for an in-depth discussion of the concepts behind the CLI, as well as greater reflection on the part of students in regard to their responses. I have prepared a possible lesson outline for this scenario, in which students better relate classroom activities to those of the command line literacy website. In this way, the learning project can better link interaction with the CLI to their own lives, giving them time to become familiar with the tools and concepts.

Possible Lesson Outline

Objective: Students will improve their command line literacy. After finishing the learning activities, students will be able to relate CLI scenarios to their own experience.

Event	Media	Prescription
1. Gaining Attention	Digital projector, Computer	Display screenshots of command line interfaces CLIs). Ask students if they know where the screenshots are from. Ask students if/when they have entered text commands to accomplish a task on a computer.
2. Inform the learner of objective	Computer	Ask students to visit Command Line Literacy website. Summarize objective of learning activities.
3. Stimulate recall of prerequisites	Computer	Guide students to learning module on CLL website. Reinforce the differences between different command line interfaces by comparing and contrasting Wikipedia articles on each (bash, PowerShell, Gnome Terminal, etc.).
4. Presenting the stimulus material and providing learning guidance	Computer	Familiarize students with the Web-based CLI tools Guide them through utilizing the chmod file permissions calculator. Relate numerical chmod values back to text values.
6. Eliciting performance	Computer	Present learning activities to students, scenarios that challenge them to synthesize the information they have gathered from chmod tutorial and related Web resources. Ask students to gather data from Web-based CLI tools (including chmod calculator) to respond to learning activity questions
7. Providing feedback	Digital projector, Computer	After students have completed the learning activities, project the correct answers in class, asking students to compare with their own learning activity responses. Ask students to leave feedback responses to the activity on the CLL website.

8. Assessing performance	Worksheet, Computer	Present a worksheet with the same scenarios as in the learning activities. Challenge students to respond to new CLI questions, writing their responses on the worksheets.
9. Enhancing retention and transfer	Home activity	Ask students to bring worksheets home, relate scenarios back to real-life situations on their own computers (or computers they have access to). Example questions: “How does security relate to my own files? How can I improve the integrity of my data (e.g. through automated backups)?”

Activity I: Susan's Homework (15 possible points)

Scenario: *Susan has asked for your help so that she can finish her homework in time for class.*

1. Understand strengths and weaknesses of the Command Line Interface (CLI)

0 = Answer does not relate to CLI or GUI *or* No answer

1 = Demonstrates basic understanding of CLI concepts

2 = Compares and contrasts CLI with GUI *or* Offers clear and detailed explanation of CLI concepts

3 = Demonstrates advanced understanding of CLI or user interface concepts

2a. Locate information from the chmod Tutorial and apply it to an example

0 = Incorrect answer *or* No answer

1 = Answer is mostly correct but technically inaccurate (unclear or partial response, flawed reasoning)

2 = Correct answer

2b. Clearly communicate CLI concepts to others

0 = Answer does not relate to the example *or* No answer

1 = Inaccurate explanation given *or* answer too short to evaluate

2 = Accurate explanation given

3 = Clear and accurate explanation given that demonstrates understanding of CLI concepts

3. Complete tasks with CLI tools

0 = Incorrect answer *or* No answer

1 = Answer is mostly correct but technically inaccurate (bad syntax, unclear or partial response)

2 = Correct answer

4a. Search and locate Web resources with a search engine

0 = Resource does not relate to topic *or* No answer

1 = Relevant resource located but either URL or title is missing *or* Inaccurate URL or title

2 = Relevant resource located and accurate URL and title are given

4b. Critically evaluate Web resources

0 = Answer does not relate to question *or* No answer

1 = Answer without explanation *or* Partial response

2 = Answer accompanied by well-reasoned explanation

3 = Answer and explanation demonstrate deep critical thinking skills

5. List Web resources utilized to answer questions

n/a, feedback for evaluation of activity

Activity II: Juan the Sysadmin (15 possible points)

Scenario: *Juan wants to train you as the future sysadmin, and has asked you to help him with his duties.*

1. Understand strengths and weaknesses of the Command Line Interface (CLI)

0 = Answer does not relate to CLI or GUI *or* No answer

1 = Demonstrates basic understanding of CLI concepts

2 = Compares and contrasts CLI with GUI *or* Offers clear and detailed explanation of CLI concepts

3 = Demonstrates advanced understanding of CLI or user interface concepts

2a. Locate information from the chmod Tutorial and apply it to an example

0 = Incorrect answer *or* No answer

1 = Answer is mostly correct but technically inaccurate (unclear or partial response, flawed reasoning)

2 = Correct answer

2b. Clearly communicate CLI concepts to others

0 = Answer does not relate to the example *or* No answer

1 = Inaccurate explanation given *or* answer too short to evaluate

2 = Accurate explanation given

3 = Clear and accurate explanation given that demonstrates understanding of CLI concepts

3. Complete tasks with CLI tools

0 = Incorrect answer *or* No answer

1 = Answer is mostly correct but technically inaccurate (bad syntax, unclear or partial response)

2 = Correct answer

4a. Search and locate Web resources with a search engine

0 = Resource does not relate to topic *or* No answer

1 = Relevant resource located but either URL or title is missing *or* Inaccurate URL or title

2 = Relevant resource located and accurate URL and title are given

4b. Critically evaluate Web resources

0 = Answer does not relate to question *or* No answer

1 = Answer without explanation *or* Partial response

2 = Answer accompanied by well-reasoned explanation

3 = Answer and explanation demonstrate deep critical thinking skills

5. List Web resources utilized to answer questions

n/a, feedback for evaluation of activity

Activity III: Seila's Spreadsheet (15 possible points)

Scenario: *Seila has sent you an e-mail with some questions about keeping his spreadsheet private.*

1. Understand strengths and weaknesses of the Command Line Interface (CLI)

0 = Answer does not relate to CLI or GUI *or* No answer

1 = Demonstrates basic understanding of CLI concepts

2 = Compares and contrasts CLI with GUI *or* Offers clear and detailed explanation of CLI concepts

3 = Demonstrates advanced understanding of CLI or user interface concepts

2a. Locate information from the chmod Tutorial and apply it to an example

0 = Incorrect answer *or* No answer

1 = Answer is mostly correct but technically inaccurate (unclear or partial response, flawed reasoning)

2 = Correct answer

2b. Clearly communicate CLI concepts to others

0 = Answer does not relate to the example *or* No answer

1 = Inaccurate explanation given *or* answer too short to evaluate

2 = Accurate explanation given

3 = Clear and accurate explanation given that demonstrates understanding of CLI concepts

3. Complete tasks with CLI tools

0 = Incorrect answer *or* No answer

1 = Answer is mostly correct but technically inaccurate (bad syntax, unclear or partial response)

2 = Correct answer

4a. Search and locate Web resources with a search engine

0 = Resource does not relate to topic *or* No answer

1 = Relevant resource located but either URL or title is missing *or* Inaccurate URL or title

2 = Relevant resource located and accurate URL and title are given

4b. Critically evaluate Web resources

0 = Answer does not relate to question *or* No answer

1 = Answer without explanation *or* Partial response

2 = Answer accompanied by well-reasoned explanation

3 = Answer and explanation demonstrate deep critical thinking skills

5. List Web resources utilized to answer questions

n/a, feedback for evaluation of activity

Responses: Activity I

From Command Line Literacy

Miguel said ...

1. Command lines let people perform batch actions better than a GUI. Commands can be used in combination with each other and are documented.
 - 2a. Yes she can make changes to her homework. 2b. homework.txt has read and write access, rw, for susan.
 3. 755
 - 4a. Unix file permissions tutorial, <http://www.dartmouth.edu/~rc/help/faq/permissions.html> 4b. It seems very detailed but probably not as good. The tutorial has more pages
 5. Just the module and chmod calculator
- Miguel 02:05, 10 Apr 2008 (PDT)

Cherise said ...

1. A CLI allows the user to perform tasks in a more complex way than a GUI and gives more options.
 - 2a. Yes, she can edit/save homework.txt
 - 2b. Read/write access for susan on the file
 3. chmod 755
 - 4a. <http://www.tuxfiles.org/linuxhelp/filepermissions.html> Linux file permissions
 - 4b. The site I chose is not as detailed, but I find the text easier on the eyes.
 5. I only utilized the learning module and the linked calculator
- Cherise 05:19, 26 Apr 2008 (PDT)

Dan said ...

1. cli is much more powerful
2.
 - a. Yes
 - b. rw access for susan
3. 755

4.

a. Permissions for files and folders

<http://technet2.microsoft.com/windowsserver/en/library/e4be109f-5547-4df8-90f0-4d885dc302e71033.mspx?mfr=true>

b. The microsoft site links to more documents and easy to read

5. Microsoft site

--Dan 01:58, 1 May 2008 (PDT)

Retrieved from "http://seandiggity.dreamhosters.com/epsy381wiki/index.php?title=Talk:Activity_I"

Responses: Activity II

From Command Line Literacy

Miguel said ...

1. Juna uses CLI because its easy to search big text files and some servers don't have GUI.
- 2a. `chmod ug=o` 2b. It changes user and group to have same permissions as others
3. 644
- 4a. <http://www.debian.org/distrib/> Getting Debian 4b. Its the official site
5. Just module and calculator

--Miguel 02:26, 10 Apr 2008 (PDT)

Cherise said ...

1. He uses only the command line because it's better suited to the task of searching through text files.
- 2a. `chmod u=o`
- 2b. That command gives the owner of the file the same permissions as the "others".
3. `chmod 644`
- 4a. <http://www.debian.org/distrib/> Getting Debian 4b. That's the official project site of Debian GNU/Linux.
5. I only utilized the learning module and the linked calculator

--Cherise 06:20, 26 Apr 2008 (PDT)

Dan said ...

1. Server has no gui
2.
 - a. `chmod ug=o`
 - b. Changes permissions so their same as other
3. 644
- 4.

a. Debin server setup <http://www.debianhelp.co.uk/debianserver.htm>

b. Shows how to set up server

5. Debian server site

--Dan 02:10, 1 May 2008 (PDT)

Retrieved from "http://seandiggity.dreamhosters.com/epsy381/wiki/index.php?title=Talk:Activity_II"

Responses: Activity III

From Command Line Literacy

Miguel said ...

1. Seila should use CLI, since he wants to automate the task.
 - 2a. Yes, group acctg can read the file 2b. There is an "r" in the permissions for the group acctg
 3. 600
 - 4a. Protecting Personal Information, <http://www.ftc.gov/infosecurity/> 4b. Its from the government, so probably goes with security.
 5. Just the module, calculator
- Miguel 02:39, 10 Apr 2008 (PDT)

Cherise said ...

1. He should definitely use a CLI, since the task he wants to perform is better automated.
 - 2a. No.
 - 2b. No one else can write to the file.
 3. chmod 600
 - 4a. <http://searchsecurity.techtarget.com> Information Security
 - 4b. The resource seems like it's a very up to date news site on security.
 5. I only utilized the learning module and the linked calculator
- Cherise 06:41, 26 Apr 2008 (PDT)

Dan said ...

1. cli is good for that since its moving files
2.
 - a. Yes acctg can read it
 - b. acctg group has an r

3. 600

4.

a. Data security http://en.wikipedia.org/wiki/Data_security

b. Wikipedia is good resource

5. Wikipedia

--Dan 03:00, 1 May 2008 (PDT)

Retrieved from "http://seandiggity.dreamhosters.com/epsy381/wiki/index.php?title=Talk:Activity_III"

Feedback

From Command Line Literacy

Miguel said ...

1. I think the activities were effective although I didnt learn a whole lot. I never used a CLI before so it was interesting but I won't remember everything.
2. It was old school. I don't think so.
3. Yes, the calculator was really useful. The tutorial could use some color.
4. I didn't use them.

--Miguel 03:45, 10 Apr 2008 (PDT)

Cherise said ...

1. The learning activities were generally very good, and i think I learned a lot more than I expected to. I think the learning activities weren't linked strongly enough with the chmod tutorial.
2. The CLI is very interesting, and I may end up using it on my Mac.
3. I think the tutorial was almost too separate from everything. The chmod calculator was easy to use.
4. I only clicked on a few of the resources out of curiosity but I didn't utilize any of them.
5. Overall it was good.

--Cherise 07:45, 26 Apr 2008 (PDT)

Dan said ...

- 1.I learned some good stuuf
- 2.It was pretty cool but I like the GUI
- 3.The calculator was good
- 4.Didn't use them

--Dan 04:14, 1 May 2008 (PDT)

Retrieved from "<http://seandiggity.dreamhosters.com/epsy381/wiki/index.php?title=Talk:Feedback>"