Лабораторная работа №1. Shell

Цель

Целью выполнения этого компьютерного практикума является знакомство со средой пользователя ОС семейства Unix.

В результате его выолнения будет получено общее представление про интерфейс взаимодействия с системой, изучены основные команды командной оболочки, получены навыки написания скриптов командной оболочки.

Задание

Необходимо написать shell-скрипт, который обработает текстовый файл log.txt в формате *Acess log* веб-сервера Apache и выведет в консоль информацию, которая описана в отдельном документе "Вариант задания по КП1". Этот скрипт должен использовать стандартные инструменты shell (которые включают такие утилиты, как: `cut`, `grep`, `sort`, `awk`, `date` и др.), но не использовать другие языки программирования, такие как C, Perl, Python и др.

Файл состоит из записей, каждая из которых занимает одну строку.

Пример записи:

```
host-24-225-218-245.patmedia.net - -
[01/Oct/2006:06:33:45 -0700] "GET /example/example.atom
HTTP/1.1" 304 - "-" "NetNewsWire/2.0b37 (Mac OS X; Lite;
http://ranchero.com/netnewswire/)"
```

Формат записи:

```
<хост клиента> - - [<Штамп времени с временной зоной>]
<Строка HTTP-запроса (тип, URL, версия)> <Код HTTP-
ответа> <Количество переданных байт или '-', если ответ
не имеет тела> <Строка реферера ('-' означает прямой
запрос без реферера)> <Название клиента (браузер)>
```

Расшифровка:

01/Oct/2006:06:33:45 -0700 с хоста **host-24-225-218-245.patmedia.net** по протоколу **HTTP/1.1** был выполненн запрос типа **GET** для получения ресурса, находящегося по ссылке **/example/example.atom**. Код ответа на запрос от сервера: **304**. Такой ответ не предполагет наличия тела ответа (количество

* pdf generated on 2014-10-20 from markdown sources with SBCL 1.1.14 on Linux 3.11.0-12-generic with 3bmd, cl-pdf & cl-typesetting

переданных байт - **0**). Запрос выполнялся напрямую, а не по ссылке с другого сайта (поле референер - пустое). Клиент использовал для обращения программу **NetNewsWire/2.0b37**, ОС клиента: **Mac OS X**

Пояснения

Shell-скрипт — это программа, написанная для интерпретации командной оболочкой ОС (shell). Эта программа существует в текстовом виде и не требует отдельного этапа компиляции перед выполнением. По соглашению, первой строчкой скрипта является указание конкретного интерпретатора, который должен исполнять его. Вообще говоря, в Unix-ОС скрипты не обязательно должны выполняться именно командной оболочкой, а могут быть написаны на любом языке, который поддерживает интерпретацию (например, Perl или Python).

Shell-скрипт Hello World:

```
#!/bin/sh
echo "Hello World"
```

Если этот текст сохранить в файл hello.sh в текущей директории, то выполнить его можно двумя способами:

- `$ sh hello.sh` - в этом случае мы запускаем команду `sh` (собственно, shell) и передаем ей в качестве аргумента имя файла скрипта
- `$ chmod +x hello.sh; ./hello.sh` - в этом случае выполняются 2 команды: сначала файлу скрипта дается право на выполнение, затем запускается сам файл и командная оболочка, в которой выполняется эта команда, анализирует начало файла. Если это скомпилированная программа, то она содержит в первых байтах т.н. магический номер (magic number), который идентифицирует формат исполняемого файла, в который она скомпилированна — в этом случае shell передает управление загрузчику программы, который поддерживает соответствующий формат. Если это скрипт и он содержит т.н. shebang-строку (строку, начинающуюся с символов `#!`), то все его содержимое передается программе, путь к которой указан в этой строке (в данном случае: `/bin/sh`). Иначе программа считается скриптом самой командной оболочки и выполняется ей самой.

Примечание 1: `$` - приглашение консоли ОС. Такая запись означает, что команду нужно выполнить в консоли (без самого знака $).

Примечание 2: программная оболочка по-умолчанию в среде Linux — `bash`

* pdf generated on 2014-10-20 from markdown sources with SBCL 1.1.14 on Linux 3.11.0-12-generic with 3bmd, cl-pdf & cl-typesetting

(Bourne Again Shell). В данном примере в качестве интерпретатора указан `sh` (Bourne Shell), который является более простым вариантом командной оболочки, предшественником `bash`. В случаях, когда в скрипте не используются специфические расширения `bash`, правилом хорошего тона является указывать в качестве интерпретатора именно `sh` (из соображений большей переносимости кода: не на всех системах может быть установлен `bash`).

Shell-скрипты для Boune Shell и ее вариантов могут использовать те же самые команды, которые можно вводить и с консоли операционной системы. Команда `man` - позволяет получить справку по любой команде. `$ man sh` позволит вам изучить синтаксис самого Shell. Важными операторами Shell является перенаправление вывода (>) и ввода (<), а также конвеер pipe (|), который позволяет перенаправлять вывод одной программы на ввод другой.

На равне со встроенными командами Shell может быть запущена любая программа, если она найдена в поисковом пути `PATH`, или же к ней указан полный путь. `PATH` — это одна из переменных окружения Shell, которая доступна всем процессам. Получить ее значение можно так: `$PATH`, а установить (в sh/bash): `export PATH = ...`. Узнать значение всех переменных окружения можно командой `env`. Среди переменных окружения есть несколько специальных переменных, которые устанавливаются индивидуально для каждого запущенного процесса. Например, это переменная `$*`, в которой содержатся все аргументы, с которыми запущена данная программа, или `$1`, `$2`, ..., которые содержат первый, второй и т.д. аргумент.

`PATH` не включает текущую директорию, поэтому запуск программы из текущей директории, как правило, выполняется с помощью синтаксиса `./` (т.е. `$ ./program`). Точка в Shell — это псевдоним для текущей директории. Другими псевдонимами являются `..` — директория на уровень выше, и `~` — домашняя директория. Команды и программы могут принимать строковые аргументы, которые каждая из них может интерпретировать по-своему. Как правило, эти аргументы бывают 3-х типов:

- просто значения (числа, строки), например в `$ echo "Hello World"` `"Hello World"` — это просто строка
- пути, например в `$ cat hello.txt` `hello.txt` — это путь к файлу в текущей директории. Полный путь мог бы выглядеть так: `/home/user/hello.txt`
- аргументы-ключи: начинаются с - или --, например в `$ wc -l file.txt` команда `wc` считает количество только строк (ключ `-l`) в файле `file.txt`. Ключ `--help` позволяет получить краткую справку по подавляющему большинству команд

* pdf generated on 2014-10-20 from markdown sources with SBCL 1.1.14 on Linux 3.11.0-12-generic with 3bmd, cl-pdf & cl-typesetting

* pdf generated on 2014-10-20 from markdown sources with SBCL 1.1.14 on Linux 3.11.0-12-generic with 3bmd, cl-pdf & cl-typesetting

Литература

Основы работы с Linux

- http://matt.might.net/articles/basic-unix/
- http://www.funtoo.org/Linux_Fundamentals,_Part_1

Основы работы с Bash

- http://www.linux.org.ru/books/bash-conspect.html
- http://www.tldp.org/LDP/abs/html/
- http://www.davidpashley.com/articles/writing-robust-shell-scripts.html
- http://mywiki.wooledge.org/BashFAQ
- Bash by Example
- Google Shell Style Guide

Работа с текстовыми данными

- http://www.ibm.com/developerworks/aix/library/au-unixtext/index.html
- http://radar.oreilly.com/2011/04/data-hand-tools.html
- http://www.pement.org/awk/awk1line.txt
- http://sed.sourceforge.net/sed1line.txt
- http://www.drbunsen.org/explorations-in-unix.html

Регулярные выражения

- http://www.regular-expressions.info/
- http://www.weitz.de/regex-coach/

Продвинутые инструменты командной строки

- http://www.commandlinefu.com/commands/browse/sort-by-votes
- http://www.quora.com/Linux/What-are-some-time-saving-tips-that-every-Linux-user-should-know

* pdf generated on 2014-10-20 from markdown sources with SBCL 1.1.14 on Linux 3.11.0-12-generic with 3bmd, cl-pdf & cl-typesetting

- http://www.reddit.com/r/linux/comments/mi80x/give_me_that_one_command_you_wish_you_knew_years/
- http://kkovacs.eu/cool-but-obscure-unix-tools
- http://offbytwo.com/2011/06/26/things-you-didnt-know-about-xargs.html
- http://www.tldp.org/HOWTO/Bash-Prompt-HOWTO/

* pdf generated on 2014-10-20 from markdown sources with SBCL 1.1.14 on Linux 3.11.0-12-generic with 3bmd, cl-pdf & cl-typesetting