Primeiro trabalho de shell
29/09/2021

1 Requisitos obrigatórios

1. Seu script deve se chamar "tshell1.sh".
2. Este script deve estar com a hashbang e ter permissão 700.
3. Ele deverá ser executado em uma destas máquinas: cpu1, cpu2 ou orval.
4. Use boas práticas de programação, como indentação, bons nomes para variáveis, comentários no código, ...
5. Seu script deve fazer uso de pelo menos uma função.

2 O problema

O C3SL mantém uma cópia das ocorrências da prefeitura de Curitiba registradas no telefone 156. Estes dados são públicos. Nós queremos fazer algumas análises automáticas sobre as reclamações feitas e para isso iremos baixar automaticamente os dados do serviço. A página de onde os dados podem ser baixados é esta aqui: `http://dadosabertos.c3sl.ufpr.br/curitiba/156/`. Como são muitos dados não queremos baixar todos os arquivos manualmente e queremos aprender a fazer *raspagem web* (do inglês *Web scraping*), que é um processo de coleta automatizada de dados estruturados a partir da web (Internet).

Caso você faça corretamente este trabalho, ao final dele você poderá observar quais são os temas mais frequêntes presentes no 156. Qual será a coisa que mais incomoda os curitibanos? Quais temas você acha que, na sua intuição, deveriam ou mereciam estar antes? Pense a respeito, reflita.

Embora seja possível baixar todos os arquivos do 156, nós temos limitação de quota nos computadores do dinf, então baixaremos somente uma parte dos arquivos referentes ao ano 2021.

O processo de raspagem combina muito bem com scripts escritos em shell. Ele tem início, normalmente, baixando-se o código fonte de um site, no nosso caso esta URL acima. Em geral se observa atentamente os detalhes deste código fonte, se identificam padrões que permitem construir progressivamente scripts. Em geral não se constroi o script inteiro de uma vez, mas ele vai sendo construído pela observação destes padrões e usando-se as diversas ferramentas que estudamos, tais como `grep, cut, sort, sed, awk`

. . . , além dos conceitos de redirecionamento de entrada e saída e uso de pipes e variáveis.

3 A tarefa

1. O script que você vai entregar como solução para este exercício deve ser feito para rodar em uma das servidoras do dinf, logo você deve saber se conectar por SSH em uma destas três máquinas: cpu1, cpu2 ou orval. Estando em casa, você deverá se conectar na macalan primeiro.

2. Para não prejudicar a sua quota de discos você deve baixar todos os dados em um diretório sob seu `/nobackup`. Aprenda onde está este teu diretório e crie um subdiretório nele chamado `156`.

3. Baixe o código fonte da página em questão em um arquivo de nome `index.html`:
   `wget http://dadosabertos.c3sl.ufpr.br/curitiba/156/`

4. Observe no entanto que, como você usa boas práticas de programação, seu script deve usar variáveis.

5. Este arquivo `index.html` deve ser observado e dele devem ser extraídas as URL's das páginas que contém os arquivos em formato CSV que serão analisados posteriormente. Observe que as URL's estão entre aspas após a string `href=`. Você deve criar uma variável que contém todas elas, e somente elas (as URL's que interessam).

6. Extraia (baixe) todos os arquivos referentes ao ano 2021 que não tenham nem a palavra "Histórico" e nem a string `_201` em seu nome. Todos estes arquivos estão em formato CSV e devem ficar em um subdiretório do diretório 156 que você criou acima.

7. Se você fez o script corretamente até aqui, você deve ter em seu subdiretório cerca de 9 arquivos CSV. O problema é que estes arquivos foram gerados em um sistema operacional `windows` e lamentavelmente contém caracteres de controle deste sistema que atrapalharão o seu script. Elimine os caracteres indesejados rodando um `dos2unix` neles todos.

8. Use o comando `file` para observar a diferença de formatação destes arquivos antes e depois do comando `dos2unix`. Você verá que, antes, o `file` mostrava que eles eram do tipo `data` e que depois eles continuam

sendo deste tipo. Observe os arquivos usando o comando `less` e tente ver que ainda há caracteres indesejados no arquivo.

9. Infelizmente, o `dos2unix` não consegue se livrar de todos os caracteres indesejados, vai sobrar um que é um `^@` que você não quer. Existem várias maneiras de se livrar deste caractere, que é um caractere "não imprimível" (non printable caractere). Comandos como `sed` ou `iconv` podem te ajudar. Quando você conseguir obter arquivos CSV que estejam em formatação UTF8 e que estejam livres de carateres indesejados então você poderá continuar a fazer este trabalho. O comando `file` vai te confirmar que tudo está em UTF8 (e não em formato `data`).

10. Caso o script seja executado mais de uma vez, você não deve baixar novamente os arquivos já baixados. Seu script deve ter este cuidado.

11. Observe com atenção o padrão dos arquivos CSV e o significado de cada coluna. As colunas de interesse neste trabalho são aquelas relativas às strings `ASSUNTO` e `SUBDIVISAO`. Esta observação é feita na primeira linha de cada arquivo.

12. Filtre por estas colunas e obtenha duas saídas como resposta para este trabalho:

    - Uma listagem de todas as strings `ASSUNTO` na qual as strings aparecem uma única vez e com a frequência de contagem de cada uma em todos os arquivos.
    - Uma listagem de todas as strings `SUBDIVISAO` na qual as strings aparecem uma única vez e com a frequência de contagem de cada uma em todos os arquivos.

13. Finalmente, submeta como resposta ao moodle o seu script `tshell1.sh`.

4 Comandos úteis

Aqui tem uma lista não exaustiva de comandos que talvez sejam úteis. Não se limite a eles, pode ser que você encontre soluções alternativas e, quem sabe, melhores.

```
awk
cat
cut
for
```

```
head
popd
pushd
sed
tr
for
if
[ ]
```

5 Conceitos úteis

Você provavelmente vai precisar usar os conceitos apresentados, tais como (lista não exaustiva):

- Substituição de comandos
- Variáveis
- Símbolos especiais
- Entrada, saída e pipes
- Os comandos: for, if, [ ], funções

Um comando interessante para ajudar a depurar scripts shell é você colocar no início do script o seguinte:

```
set -x
```

Depois comente esta linha para a entrega do trabalho.

6 Conceitos novos

Faz parte do trabalho você descobrir algumas coisas novas, para isto consulte as man pages dos comandos indicados, ou de outros que você encontre. Vários sites na internet são muito bons para encontrar soluções. Um bom exemplo é o GeekForGeeks ou o StackOverflow. Aprender a buscar informações faz parte da atividade de programação de computadores em geral. Você deve estar preparado para isso.