ADS Lab 04 - Scripting

Authors: Loic Brasey & Bastien Pillonel Date: March 25, 2024

Task 1: Set up web directory

1. Create the directory public_html. Create a file foo.txt in it and retrieve the file using the browser on your local machine.

Une fois connecté sur le serveur, il suffit de créer un nouveau répo public_html et d'ajouter un fichier foo.txt.

```
mkdir public_html
cd public_html
touch foo.txt
vim foo.txt
```

Ajouter le text à afficher dans le foo.txt

2. Navigate to the URL http://ads.iict.ch/~albert_einstein/foo.txt. You should see the contents of the file.

On retrouve le contenu du fichier foo.txt à l'URL (pour notre groupe):

https://ads.iict.ch/~laba/foo.txt

Task 2: Create thumbnails

1. Download a zip archive containing the picture and brochure files from this URL: http://ads.iict.ch/lab04_raw_files.zip Use the commands curl to download and unzip to unarchive. By placing the files into your web directory you can inspect them using your browser.

2. Display the dimensions of a few pictures by using ImageMagick's identify command. This command has a powerful feature where one can specify a format string (similar to the printf() format string in C) that specifies the information to print

3. Write a script called show_dimensions that loops through all the picture files and shows for each its name and its dimensions. For the loop use the for .. in .. do .. done control structure.

show_dimension script

4. Write a script called rename_pictures that produces picture files that have the dimensions in their name. For example if a picture is called building.jpg and has a width of 1024 and a height of 768 pixels the script should create a file building_1024_768.jpg . The script should not modify the original files, but create new ones. When you run the script several times how do you

prevent the dimensions from accumulating in the name, like building_1024_768_1024_768_1024_768.jpg ? The orginial files can be named anything. They could have the dimensions in the file name accidentally. Change the script and/or the organization of the files so that the dimensions don't accumulate ad infinitum. Put a comment into the script explaining how you did it. Hint: There is a very simple solution. Analyzing the filename is way too complicated.

rename_picture script

5. With a few pictures try to create a smaller thumbnail where the largest side is 300 pixels. Use ImageMagick's convert command like so:

```
convert -geometry 300 picture.jpg picture_thumb.jpg
```

6. Write a script called make_thumbnails that loops through all the picture files and creates a thumbnail for each. If the picture file is named building.jpg the corresponding thumbnail should be named building_thumb.jpg When you run the script several times how do you prevent making thumbnails from thumbnails? Add a comment to the script explaining your solution.

make_thumbnail script

7. With a few PDF files try to create a thumbnail. In contrast to pictures PDF documents can have several pages. One needs to specify to the convert command that only the first page should be processed. This is done by appending the string [0] to the filename like so (no space between the two):

```
convert -geometry 300 document.pdf[0] document_thumb.jpg
```

Since the latest versions of ImageMagick, PDF conversion is disabled by default for security reasons. Automated conversion of PDF documents provided by visitors, e.g. uploaded on a website, can cause the execution of malicious code contained in the PDF. To reactivate this feature, you must comment out the line at the end of the /etc/ImageMagick6/policy.xml file

8. Improve the script so that it generates thumbnails for both pictures and PDF documents.

make_thumbnail script

Task 3 : Generate HTML file

make_html script

make_php script

Task 4 : Use SSH Tunneling

SSH command :

```
ssh -L 9090:localhost:3306 laba@ads.iict.ch
```