AWS re:Start

LAB

Working with Commands

WEEK 2

aws re/start

Overview

Mastering essential Unix commands like splitting, filtering, sorting, and transforming data directly from the command line is crucial for streamlining workflows and improving productivity. Chaining commands using pipes further enhances efficiency by ensuring seamless data flow between operations, making command-line tasks more effective.

Additionally, combining these tools with advanced text manipulation techniques enables users to perform complex data transformations directly within their command-line environment. This integration supports task automation, efficient handling of data processing, and optimization of workflows, ultimately enhancing productivity and enabling smoother data management in Unix-based systems.

Note: This lab was made using Windows Subsystem for Linux.

Topics covered

- Use the tee command to direct output to a file
- Use the sort command to reorganize the contents of a .csv file
- Use the cut command to edit the contents of a file
- Use the sed command
- Use the pipe operator

Task 1

Use SSH to connect to an Amazon Linux EC2 instance

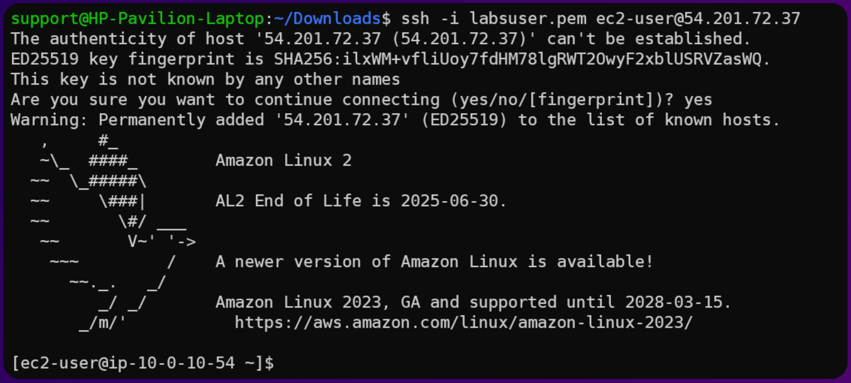
Initial Preparations

In the AWS Management Console, select the EC2 instance and make note of the **Public IPv4 address**.

Download the **private key file** labsuser.pem. Change to the Downloads directory and modify the permissions on the key to be read-only (r--------).

Connect to the instance using SSH

Establish a connection to the EC2 instance using the ssh command, the key and the instance's public IPv4 address.



```
support@HP-Pavilion-Laptop:~/Downloads$ ssh -i labsuser.pem ec2-user@54.201.72.37
The authenticity of host '54.201.72.37 (54.201.72.37)' can't be established.
ED25519 key fingerprint is SHA256:ilxWM+vfliUoy7fdHM78lgRWT2OwyF2xblUSRVZasWQ.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.201.72.37' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-0-10-54 ~]$
```

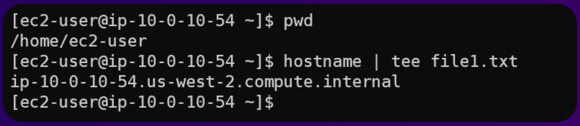
aws re/start

Task 2

Use the tee command

The tee comand

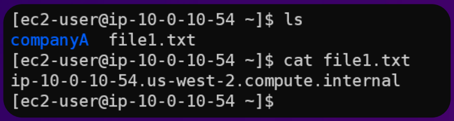
Use the tee command to display the output to the screen and to a file. Enter the command hostname | tee file1.txt. From the following output, you can see that the tee command wrote the hostname to the screen.



```
[ec2-user@ip-10-0-10-54 ~]$ pwd
/home/ec2-user
[ec2-user@ip-10-0-10-54 ~]$ hostname | tee file1.txt
ip-10-0-10-54.us-west-2.compute.internal
[ec2-user@ip-10-0-10-54 ~]$
```

Verify the file

Confirm that the file **file1.txt** has been created and review its content. From the following output, you can see that the tee command also wrote the hostname to the file **file1.txt** as expected.



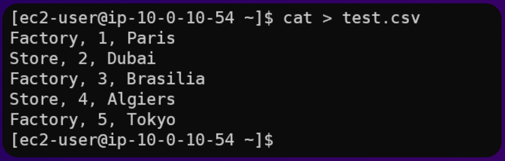
```
[ec2-user@ip-10-0-10-54 ~]$ ls
companyA  file1.txt
[ec2-user@ip-10-0-10-54 ~]$ cat file1.txt
ip-10-0-10-54.us-west-2.compute.internal
[ec2-user@ip-10-0-10-54 ~]$
```

Task 3

Use the sort command and pipe operator

Add content to a test file

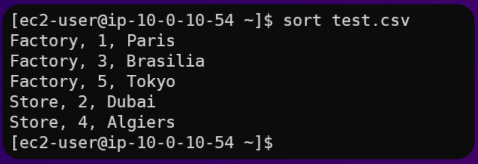
Enter a list with a few items within the file **test.csv** using the command cat > test.csv.



```
[ec2-user@ip-10-0-10-54 ~]$ cat > test.csv
Factory, 1, Paris
Store, 2, Dubai
Factory, 3, Brasilia
Store, 4, Algiers
Factory, 5, Tokyo
[ec2-user@ip-10-0-10-54 ~]$
```

The sort command

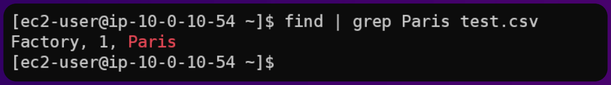
Use the sort command to reorder the list alphabetically.



```
[ec2-user@ip-10-0-10-54 ~]$ sort test.csv
Factory, 1, Paris
Factory, 3, Brasilia
Factory, 5, Tokyo
Store, 2, Dubai
Store, 4, Algiers
[ec2-user@ip-10-0-10-54 ~]$
```

The pipe | operator

Search and list the content of the **test.csv** file and redirect the results to the grep command to search for the Paris pattern.



```
[ec2-user@ip-10-0-10-54 ~]$ find | grep Paris test.csv
Factory, 1, Paris
[ec2-user@ip-10-0-10-54 ~]$
```

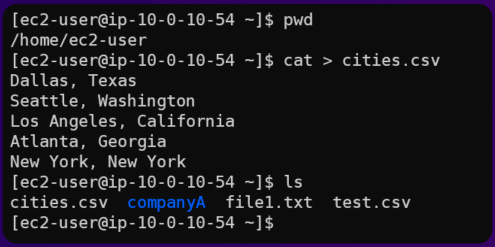
aws re/start

Task 4

Use the cut command

Add content to a test file

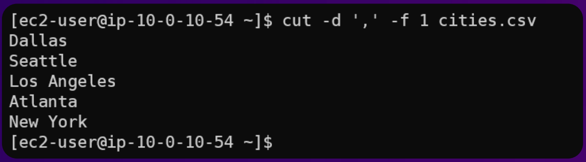
Enter a list with a few items within the **cities.csv** file using the cat > cities.csv command.



```
[ec2-user@ip-10-0-10-54 ~]$ pwd
/home/ec2-user
[ec2-user@ip-10-0-10-54 ~]$ cat > cities.csv
Dallas, Texas
Seattle, Washington
Los Angeles, California
Atlanta, Georgia
New York, New York
[ec2-user@ip-10-0-10-54 ~]$ ls
cities.csv  companyA  file1.txt  test.csv
[ec2-user@ip-10-0-10-54 ~]$
```

The cut command

Use the cut command to cut the first field from lines from the **cities.csv** file using a comma (,) delimiter.



```
[ec2-user@ip-10-0-10-54 ~]$ cut -d ',' -f 1 cities.csv
Dallas
Seattle
Los Angeles
Atlanta
New York
[ec2-user@ip-10-0-10-54 ~]$
```

aws re/start

Task 4

Use the cut command

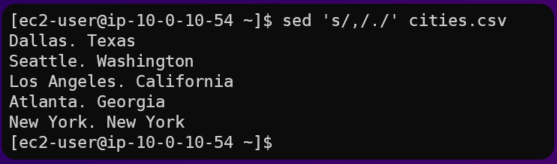
Additional challenge

Use only the sed command to make changes or do all the changes in one line.

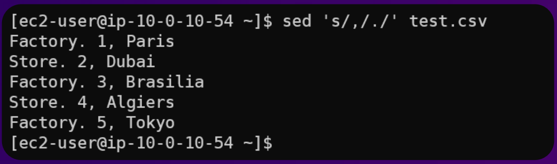
Remember, the sed command is mainly used to replace some text in a file for different text.

The sed command searches the file text for an occurrence of the first string, and will replace any matches with the second.

Use the sed command to replace the first comma (,) with periods (.) in both the **cities.csv** and **test.csv** files.



```
[ec2-user@ip-10-0-10-54 ~]$ sed 's/,/./' cities.csv
Dallas. Texas
Seattle. Washington
Los Angeles. California
Atlanta. Georgia
New York. New York
[ec2-user@ip-10-0-10-54 ~]$
```



```
[ec2-user@ip-10-0-10-54 ~]$ sed 's/,/./' test.csv
Factory. 1, Paris
Store. 2, Dubai
Factory. 3, Brasilia
Store. 4, Algiers
Factory. 5, Tokyo
[ec2-user@ip-10-0-10-54 ~]$
```

Conclusions

The tee command

Divides command output, sending it to multiple destinations like files or other commands. Useful for logging command output while still displaying it in real-time.

The sort command

Rearranges lines of text alphabetically or numerically. Options include sorting in ascending or descending order, ignoring case sensitivity, and handling numerical values correctly.

The | grep command

Searches for and filters lines of text that match specified patterns or regular expressions. Efficiently extracts relevant data from files or command output based on given criteria.

The cut command

Extracts specific fields or columns from lines of text, typically using a delimiter like a comma or space. Useful for parsing structured data such as CSV files.

The sed command

Stream editor for text manipulation. Performs tasks like search-and-replace, text substitution, and batch editing. Highly versatile and scriptable, aiding in automating text processing tasks efficiently.

aws re/start

Cristhian Becerra

cristhian-becerra-espinoza

+51 951 634 354

cristhianbecerra99@gmail.com

Lima, Peru

aws re/start