

PLEASE HAVE IN MIND THAT THIS FILE WILL BE UPDATED.
THIS IS JUST QUICK RECAP WHAT THE APP DOES AND HOW IT IS WORKING FOR NOW.

PART I - scrap songs

POC

As a POC, I created a Python script that constantly runs on the machine and saves songs to the file.

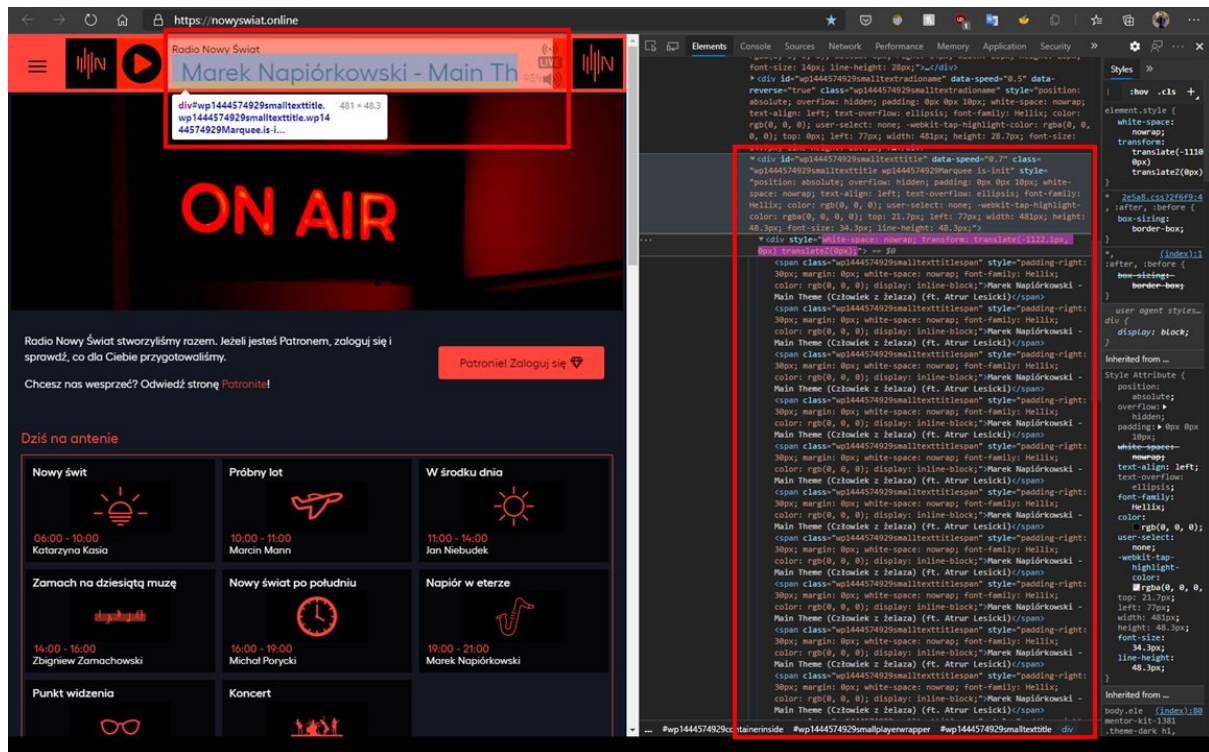
Having investigated the website (as there is any API), I thought about 3 ways to find titles of songs and make the POC done:

1. scrap main page using requests,
2. scrap main page using selenium,
3. get a title found on subpage (.txt file hosted on the server).

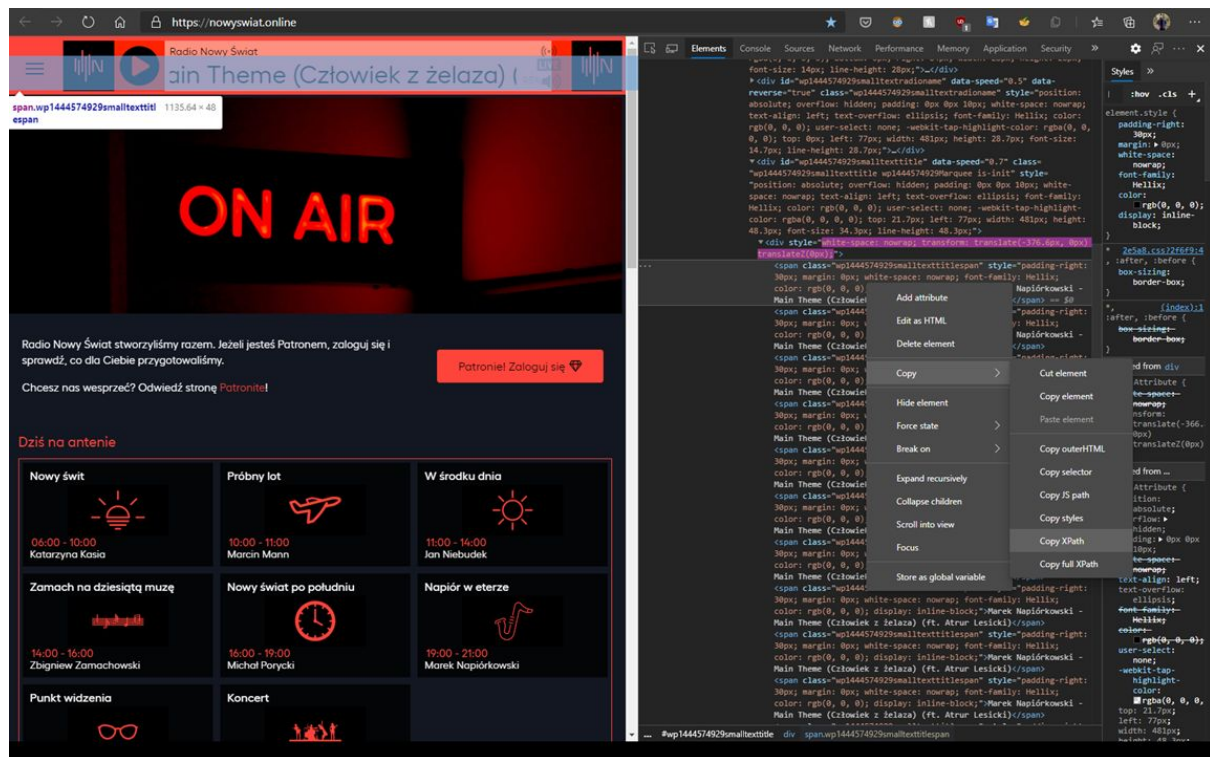
Requests

I used XPath, so I could find a title and scrap it directly.

However, it turned out that the page does not return the value of XPath. This could be caused because data is run by JavaScript (song's title is not in the website html). This is because (as I think) if a title is long and it cannot be displayed on the screen, the text flows from left to right (like text on TV news at the bottom of screen).



Copy XPath



and try to request and get the value from XPath.

```
import requests
from lxml import html

url = 'http://nowyswiat.online'
page = requests.get(url)

tree = html.fromstring(page.content)
song = tree.xpath('//*[@id="wp1444574929smalltexttitle"]/div/span[1]/text()')

print(song)
|
```

xpath_test x

C:\Users\mateusz.tomzynski\AppData\Local\Programs\Python\Python38-32\python.exe "C:\Users\mateusz.tomzynski\AppData\Local\Programs\Python\Python38-32\python.exe" C:\Users\mateusz.tomzynski\AppData\Local\Programs\Python\Python38-32\python.exe

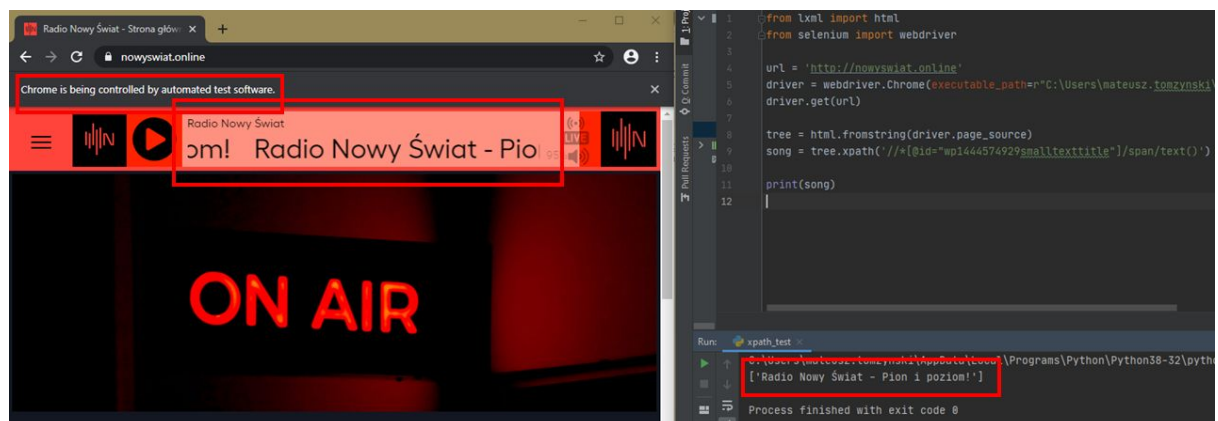
Process finished with exit code 0

As a result, it returns an empty list.

Selenium

I did the same thing for selenium. The code for selenium opened a web browser (I used Google Chrome driver) and got value from XPath.

In the result, the current value was returned as an element of a list.



The image shows a Selenium WebDriver script in a code editor and its execution results. The script imports 'requests' and 'lxml' (though Selenium is used for the browser). It sets the URL to 'http://nowyswiat.online', gets the page, and uses XPath to find the song title. The output shows the song title 'Radio Nowy Świat - Pion 1 poziom!'.

```
from lxml import html
from selenium import webdriver

url = 'http://nowyswiat.online'
driver = webdriver.Chrome(executable_path=r"C:\Users\mateusz.tomzynski\AppData\Local\Programs\Google\Chrome\Application\chrome.exe")
driver.get(url)

tree = html.fromstring(driver.page_source)
song = tree.xpath('//*[@id="wp1444574929smalltexttitle"]/span/text()')

print(song)
```

Run: xpath_test x

C:\Users\mateusz.tomzynski\AppData\Local\Programs\Python\Python38-32\python.exe "C:\Users\mateusz.tomzynski\AppData\Local\Programs\Python\Python38-32\python.exe" C:\Users\mateusz.tomzynski\AppData\Local\Programs\Python\Python38-32\python.exe

Process finished with exit code 0

Radio Nowy Świat - Pion 1 poziom!

This solution works but I was not sure whether using selenium is the most convenient way to find a song's title.

Read file being hosted on subpage

Finally, I started digging into the page and found the subpage which in fact is the text file with currently playing song or - if no music is playing, the name of the radio.


This was my first solution which worked!

However... one day I loaded the page and saw that there is a subpage "Playlist". Then basically I could forget about my previous challenges.

Ready to take playlist

As I saw the new subpage "Playlist", I thought that scrapping songs' title will be as easy as I could not even imagine.

The playlist for the current day.



The screenshot shows the website <https://nowywiat.online/playlista/>. The header features the "Radio Nowy Świat" logo and the text "Radio Nowy Świat" and "Klan - Automaty". The main content area is titled "Co dzisiaj zagraliśmy?" and displays a list of songs with their durations and titles.

Time	Song
17:24	Klan - Automaty
17:20	Metronomy - Night Owl
17:11	Bruce Springsteen - Letter to You
17:04	Phil Collins - Dance into the Light
16:55	Edyta Bartosiewicz - Ten moment
16:49	Bonnie Raitt - Runaway (Remaster)
16:42	Midge Ure - If I Was
16:37	XXANAXX - Ciepło
16:33	Green Day - Oh Yeah!
16:26	Colourbox - The Moon Is Blue

The only thing I had to do was to read songs and save them.

Run the script

This is how I run the script of POC for my first working solution.

However, this will not work like described below! As a final solution I would like to use AWS Lambda

1. Script is running on AWS E2C micro machine

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

<<

>>

1 to 1 of 1

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DN	IPv4 P	IPv6 IP	Key Name	Monitoring	Launch Time	Security Groups	Owner
	i-044b02aa9c94...	t2.micro	us-east-1e	running	2/2 checks ...	None	ec2-54-1...	54.16	-	microInstance	disabled	September 6, 20...	The New World Radio	914531

2. Security Group

The Security Group for the instance was modified. Inbound Rule was set up to allow connecting to SSH only using “My IP”, so this prevents connection to my machine from other networks.

EC2 > Security Groups > sg-0e8be57aec06d0aa9 - The New World Radio > Edit inbound rules

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Type Info

Protocol Info

Port range Info

Source Info

Description - optional Info

Delete

SSH

TCP

22

My IP

Add rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel

Preview changes

Save rules

Security Groups (1/1) Info

Filter security groups

search: sg-0e8be57aec06d0aa9 X Clear filters

<input checked="" type="checkbox"/>	Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules
<input checked="" type="checkbox"/>	-	sg-0e8be57aec06d0aa9	The New World Radio	vpc-fbf6eb81	The New World Radio created 2020-09-06...		1 Permission entry

sg-0e8be57aec06d0aa9 - The New World Radio

Details

Inbound rules

Outbound rules

Tags

Inbound rules

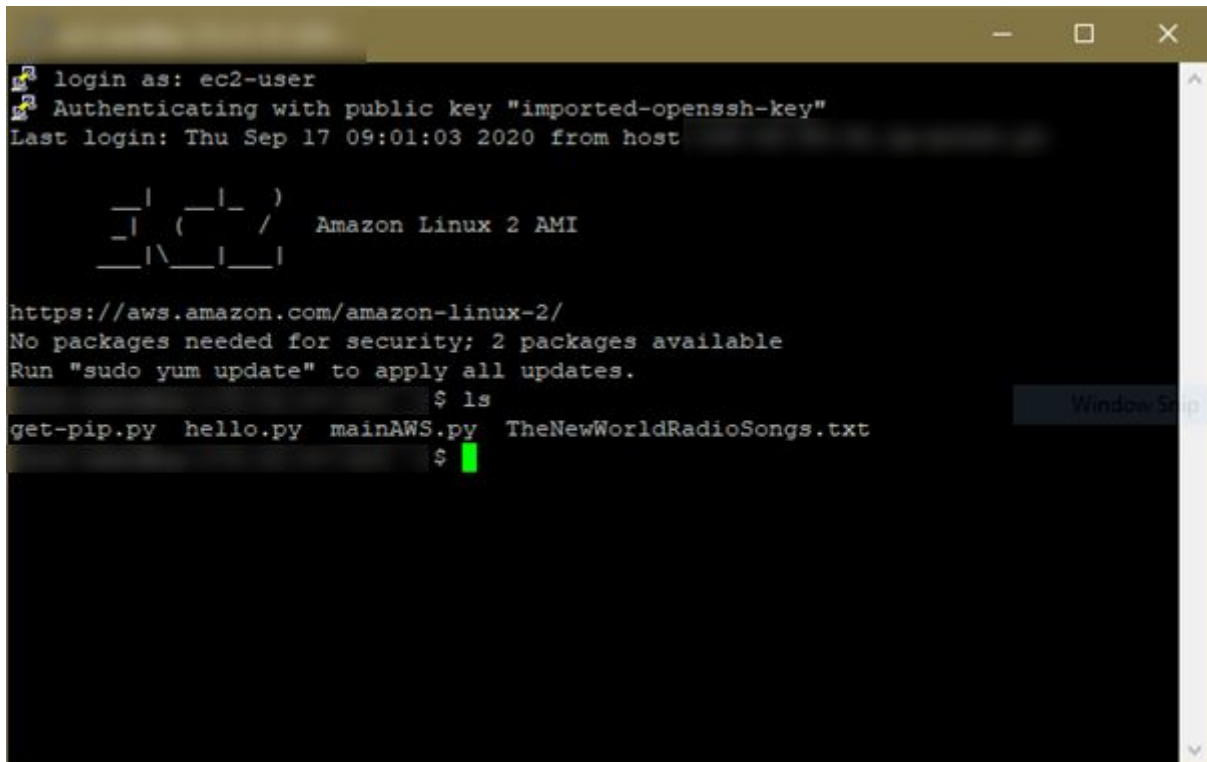
Edit inbound rules

Type	Protocol	Port range	Source	Description - optional
SSH	TCP	22		-

3. Create Key Pair for SSH connection

Key pairs (5)			
<div><div></div><div>Filter key pairs</div></div>			
<input type="checkbox"/>	Name	Fingerprint	ID
<input type="checkbox"/>	microInstance		

4. Connect to SSH via Putty



The screenshot shows a terminal window with a dark background and a gold title bar. The terminal displays the following text:

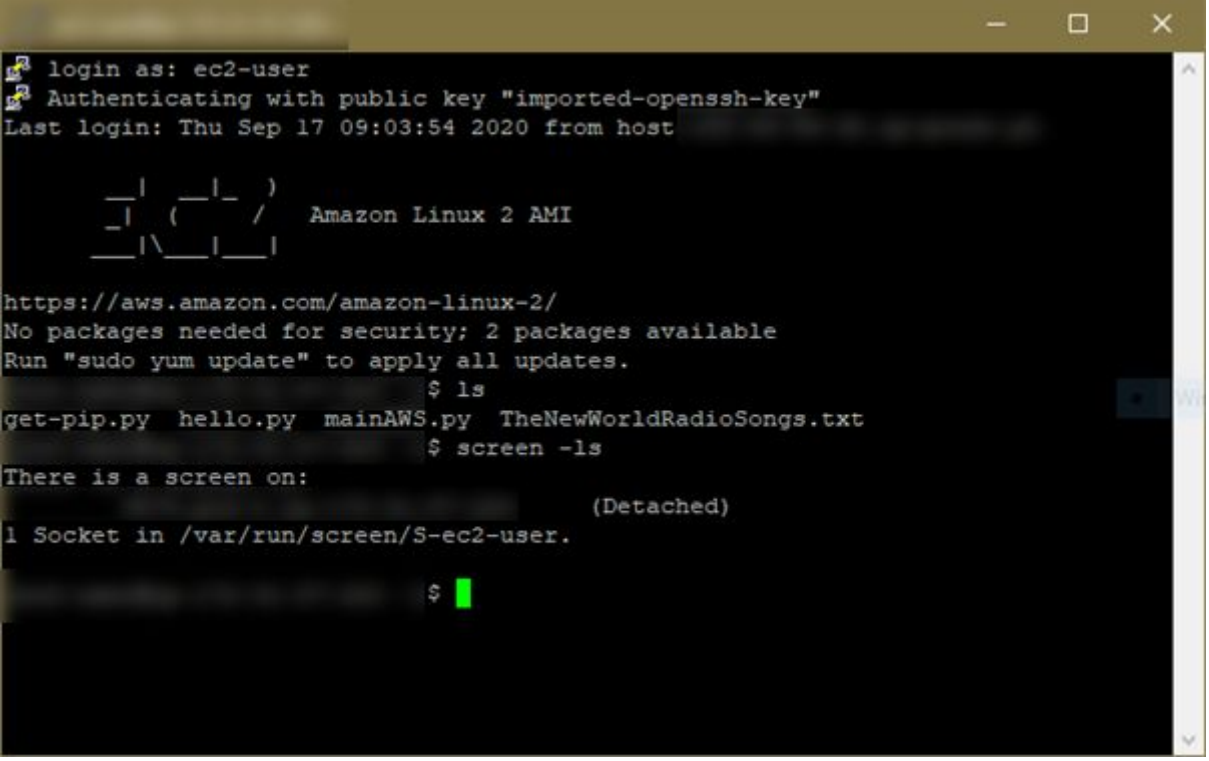
```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Thu Sep 17 09:01:03 2020 from host
    _ | _ | _ )
    _ | ( _ | /   Amazon Linux 2 AMI
    _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 2 packages available
Run "sudo yum update" to apply all updates.
$ ls
get-pip.py  hello.py  mainAWS.py  TheNewWorldRadioSongs.txt
$
```

A vertical scrollbar is visible on the right side of the terminal window.

5. See screen command

`screen -ls`

A terminal window with a dark background and a gold title bar. The window shows the output of an SSH login session. The user 'ec2-user' has logged in from a host. The terminal displays the Amazon Linux 2 AMI logo, a URL to the Amazon Linux 2 documentation, and a message about available updates. The user runs 'ls' and 'screen -ls'. The 'screen -ls' command shows a detached screen session. The prompt is a green cursor on a line with a dollar sign.

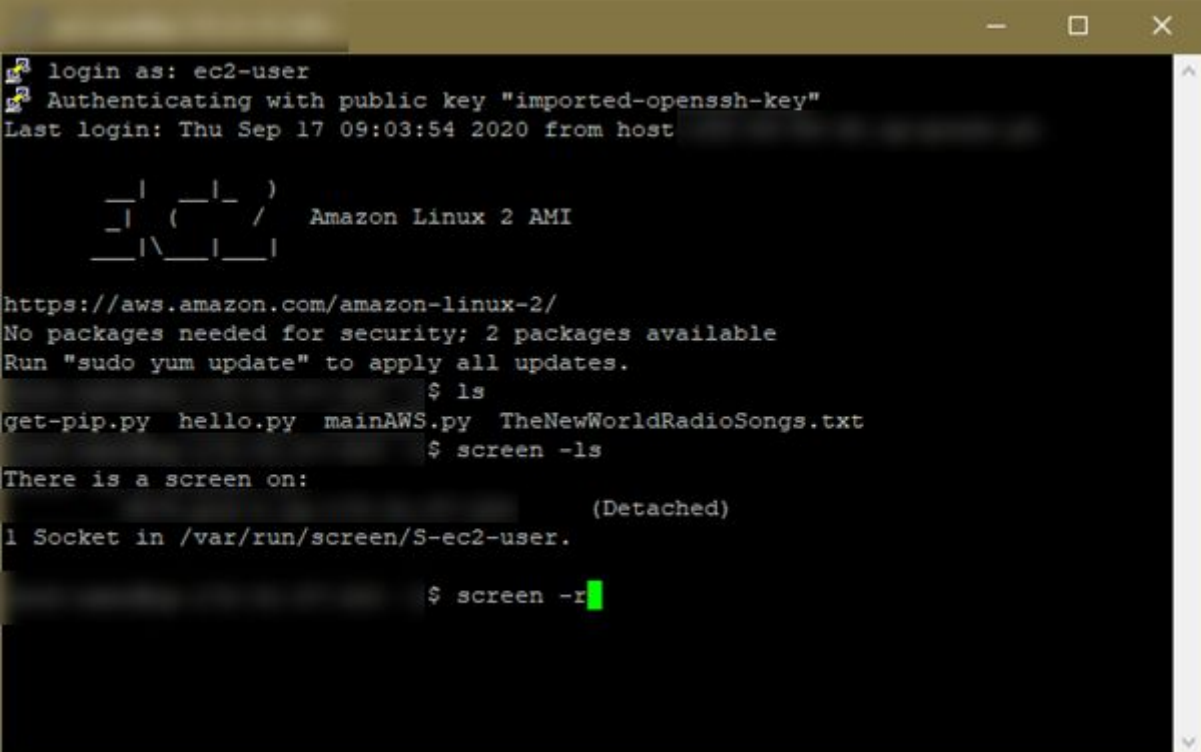
```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Thu Sep 17 09:03:54 2020 from host

  _ | _ | _ )
  _ | ( _ | /   Amazon Linux 2 AMI
  _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 2 packages available
Run "sudo yum update" to apply all updates.
$ ls
get-pip.py  hello.py  mainAWS.py  TheNewWorldRadioSongs.txt
$ screen -ls
There is a screen on:
                                (Detached)
1 Socket in /var/run/screen/S-ec2-user.
$
```

6. Script is running and outputting results (songs)

I have only one screen, so I can get it using
`screen -r`



```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Thu Sep 17 09:03:54 2020 from host
    _ _ _ _ _
   _| _ _ _ |
  _| ( _ _ _ /
 _| \ _ _ _ |
 _|  _ _ _ |

Amazon Linux 2 AMI

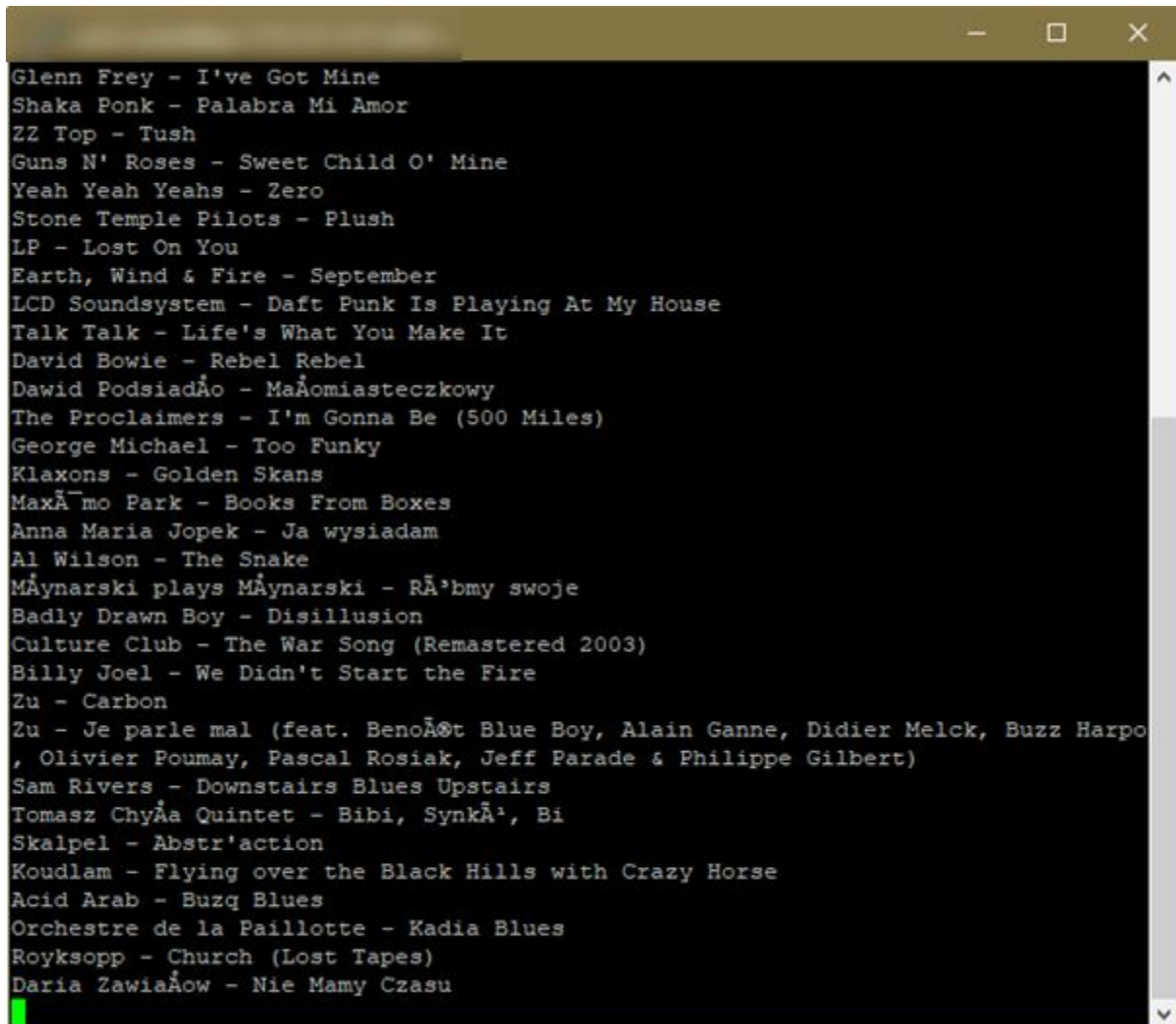
https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 2 packages available
Run "sudo yum update" to apply all updates.

$ ls
get-pip.py  hello.py  mainAWS.py  TheNewWorldRadioSongs.txt

$ screen -ls
There is a screen on:
                                (Detached)
1 Socket in /var/run/screen/S-ec2-user.

$ screen -r
```

As you can see, the app prints songs to the console



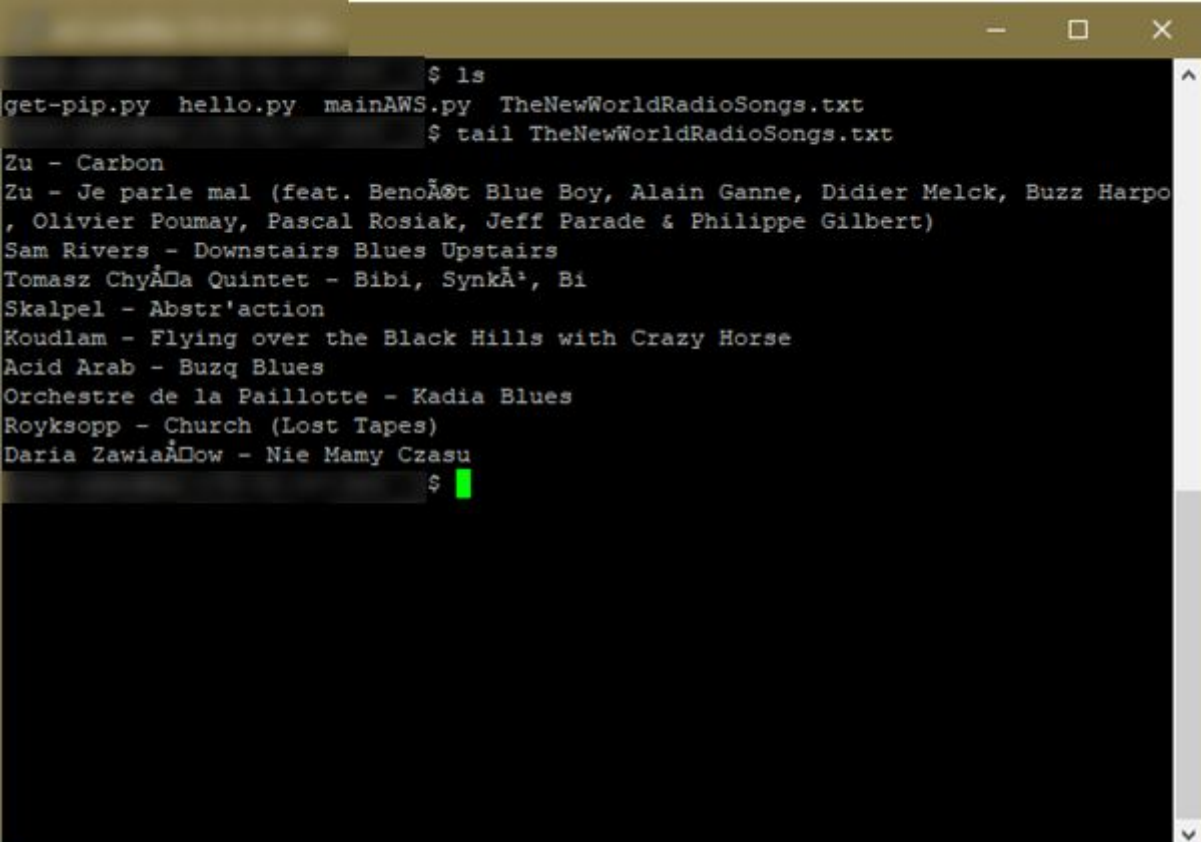
```
Glenn Frey - I've Got Mine
Shaka Ponk - Palabra Mi Amor
ZZ Top - Tush
Guns N' Roses - Sweet Child O' Mine
Yeah Yeah Yeahs - Zero
Stone Temple Pilots - Plush
LP - Lost On You
Earth, Wind & Fire - September
LCD Soundsystem - Daft Punk Is Playing At My House
Talk Talk - Life's What You Make It
David Bowie - Rebel Rebel
Dawid Podsiadłko - Małomiasteczkowy
The Proclaimers - I'm Gonna Be (500 Miles)
George Michael - Too Funky
Klaxons - Golden Skans
Maxïmo Park - Books From Boxes
Anna Maria Jopek - Ja wysiadam
Al Wilson - The Snake
Młynarski plays Młynarski - Rębnymy swoje
Badly Drawn Boy - Disillusion
Culture Club - The War Song (Remastered 2003)
Billy Joel - We Didn't Start the Fire
Zu - Carbon
Zu - Je parle mal (feat. Benoît Blue Boy, Alain Ganne, Didier Melck, Buzz Harpo
, Olivier Poumay, Pascal Rosiak, Jeff Parade & Philippe Gilbert)
Sam Rivers - Downstairs Blues Upstairs
Tomasz Chyła Quintet - Bibi, Synk', Bi
Skalpel - Abstr'aktion
Koudlam - Flying over the Black Hills with Crazy Horse
Acid Arab - Buzq Blues
Orchestre de la Paillotte - Kadia Blues
Royksopp - Church (Lost Tapes)
Daria Zawiałow - Nie Mamy Czasu
```

7. Songs are saved in a text file

Using

```
tail TheNewWorldRadioSongs.txt
```

we can see the latest saved songs and compare them to those printed on screen terminal.

A terminal window with a dark background and a gold title bar. The terminal shows the output of two commands. First, '\$ ls' lists files: 'get-pip.py', 'hello.py', 'mainAWS.py', and 'TheNewWorldRadioSongs.txt'. Then, '\$ tail TheNewWorldRadioSongs.txt' displays the contents of the text file, which lists various artists and song titles. The text is displayed in a monospaced font, and some characters appear to be garbled, likely due to encoding issues. A green cursor is visible at the end of the last line of the tail command output.

```
$ ls
get-pip.py  hello.py  mainAWS.py  TheNewWorldRadioSongs.txt
$ tail TheNewWorldRadioSongs.txt
Zu - Carbon
Zu - Je parle mal (feat. Benoît Blue Boy, Alain Ganne, Didier Melck, Buzz Harpo
, Olivier Poumay, Pascal Rosiak, Jeff Parade & Philippe Gilbert)
Sam Rivers - Downstairs Blues Upstairs
Tomasz Chyła Quintet - Bibi, Synk, Bi
Skalpel - Abstr'ation
Koudlam - Flying over the Black Hills with Crazy Horse
Acid Arab - Buzq Blues
Orchestre de la Paillotte - Kadia Blues
Royksopp - Church (Lost Tapes)
Daria Zawiałow - Nie Mamy Czasu
$
```

As you can see there is still an issue with encoding. The issue appears already on the website, so I will need to find the workaround for that.

PART II - Web Spotify API

Description to be updated