# **aws-metadata-json**

## What it does

* Query the metadata of an ec2 instance within AWS and provide a json formatted output.
* Retrieve the value of a particular data key.

**get\_key.py**

|  |
| --- |
|  |
|  | from get\_metadata  import get\_metadata |
|  | def gen\_dict\_extract(key, var): |
|  | if hasattr(var, 'items'): |
|  | for k, v in var.items(): |
|  | if k == key: |
|  | yield v |
|  | if isinstance(v, dict): |
|  | for result in gen\_dict\_extract(key, v): |
|  | yield result |
|  | elif isinstance(v, list): |
|  | for d in v: |
|  | for result in gen\_dict\_extract(key, d): |
|  | yield result |
|  |  |
|  |  |
|  | def find\_key(key): |
|  | metadata = get\_metadata() |
|  | return list(gen\_dict\_extract(key, metadata)) |
|  |  |
|  |  |
|  | if \_\_name\_\_ == '\_\_main\_\_': |
|  | key = input("What key would you like to find?\n") |
|  | print(find\_key(key)) |

**get\_metadata.py**

import requests

|  |
| --- |
|  |
|  | import json |
|  |  |
|  | metadata\_url = 'http://169.254.169.254/latest/' |
|  |  |
|  |  |
|  | def expand\_tree(url, arr): |
|  | output = {} |
|  | for item in arr: |
|  | new\_url = url + item |
|  | r = requests.get(new\_url) |
|  | text = r.text |
|  | if item[-1] == "/": |
|  | list\_of\_values = r.text.splitlines() |
|  | output[item[:-1]] = expand\_tree(new\_url, list\_of\_values) |
|  | elif is\_json(text): |
|  | output[item] = json.loads(text) |
|  | else: |
|  | output[item] = text |
|  | return output |
|  |  |
|  |  |
|  | def get\_metadata(): |
|  | initial = ["meta-data/"] |
|  | result = expand\_tree(metadata\_url, initial) |
|  | return result |
|  |  |
|  |  |
|  | def get\_metadata\_json(): |
|  | metadata = get\_metadata() |
|  | metadata\_json = json.dumps(metadata, indent=4, sort\_keys=True) |
|  | return metadata\_json |
|  |  |
|  |  |
|  | def is\_json(myjson): |
|  | try: |
|  | json.loads(myjson) |
|  | except ValueError: |
|  | return False |
|  | return True |
|  |  |
|  |  |
|  | if \_\_name\_\_ == '\_\_main\_\_': |
|  | print(get\_metadata\_json()) How to run Run whichever script you need:   * + python3 get\_metadata.py   + python3 get\_key.py |