Akamai Identity Access Group Management Script

This document provides a comprehensive guide to a Python script designed to manage Akamai Identity Cloud user access groups. The script checks if a specified group already exists and creates it if it does not. The script utilizes Akamai's EdgeGrid authentication, with credentials securely managed in a `.edgerc` file.

# Authentication Using `.edgerc`

The script reads the `.edgerc` file using the `EdgeRc` class to retrieve the necessary credentials: `client\_token`, `client\_secret`, `access\_token`, and `host`. These credentials are essential for authenticating API requests to the Akamai Identity Cloud.

The `EdgeGridAuth` class is then used to securely sign these requests. This approach ensures that all API interactions are securely authenticated using the credentials stored in the `.edgerc` file.

Snippet to authenticate using `.edgerc` file:

import requests  
from akamai.edgegrid import EdgeGridAuth, EdgeRc  
  
edgerc = EdgeRc("~/.edgerc")  
session = requests.Session()  
session.auth = EdgeGridAuth.from\_edgerc(edgerc, "default")  
base\_url = f"https://{edgerc.get('default', 'host')}/"

# Session Setup

The `setup\_session` function is responsible for initializing a `requests.Session` object. This session is configured with the `EdgeGridAuth` using the credentials extracted from the `.edgerc` file. All subsequent API calls use this session to ensure they are authenticated.

Snippet for setting up the session:

def setup\_session(edgerc\_file, section\_name):  
 edgerc = EdgeRc(edgerc\_file)  
 base\_url = f"https://{edgerc.get(section\_name, 'host')}/"  
 session = requests.Session()  
 session.auth = EdgeGridAuth(  
 client\_token=edgerc.get(section\_name, 'client\_token'),  
 client\_secret=edgerc.get(section\_name, 'client\_secret'),  
 access\_token=edgerc.get(section\_name, 'access\_token')  
 )  
 return session, base\_url

# Checking Group Existence

The script checks if the specified group already exists by sending a GET request to the Akamai API. The `check\_group\_exists` function handles this task by querying the list of existing groups and verifying whether the target group is already present.

Snippet for checking group existence:

def check\_group\_exists(session, base\_url, group\_name):  
 url = urljoin(base\_url, "identity-management/v3/user-admin/groups")  
 headers = {"accept": "application/json"}  
 response = session.get(url, headers=headers)  
  
 if response.status\_code == 200:  
 groups = response.json()  
 for group in groups:  
 if group.get("name") == group\_name:  
 return True  
 return False

# Creating a Group

If the group does not exist, the `create\_group` function sends a POST request to the Akamai API to create a new group under a specified parent group ID. The function constructs the necessary payload with the group name and submits it to the API endpoint.

Snippet for creating a group:

def create\_group(session, base\_url, group\_name, parent\_group\_id):  
 url = urljoin(base\_url, f"identity-management/v3/user-admin/groups/{parent\_group\_id}")  
 headers = {  
 "accept": "application/json",  
 "content-type": "application/json"  
 }  
 payload = {  
 "groupName": group\_name  
 }  
 response = session.post(url, json=payload, headers=headers)  
 return response

# Command-Line Interface

The script is designed to be flexible and user-friendly, accepting various command-line arguments using the `argparse` library. The following arguments can be provided:

- `group\_name`: The name of the group to create.  
- `parent\_group\_id`: The ID of the parent group under which the new group will be created.  
- `--edgerc\_file`: (Optional) The path to the `.edgerc` file. Defaults to `~/.edgerc`.  
- `--section\_name`: (Optional) The section in the `.edgerc` file to use for authentication. Defaults to `default`.

# Running the Script

To execute the script, use the following command:

```bash  
python script.py "NewGroupName" "ParentGroupID" --edgerc\_file="~/.edgerc" --section\_name="default"  
```

This command checks if a group named `NewGroupName` exists under the specified `ParentGroupID`. If the group does not exist, the script creates it using the provided credentials in the `.edgerc` file under the specified section.

# Summary

This solution is robust and secure for managing access groups in Akamai Identity Cloud. By leveraging the `.edgerc` file for authentication, the script ensures that credentials are handled securely and that API interactions are properly authenticated. The script's flexibility in accepting command-line arguments makes it adaptable to various environments and use cases.