LISTING GITHUB REPOSITORY USERS BY USING SHELL SCRIPTING

BY SAIDINESH CHIRUMAVILLA

About: In this project we are listing the users those who can access the repository in the organization. For the integration of github API we used shell scripting. In each and every step have been written by using the github API documentation.

Step1: Login to AWS account and create EC2 instance

Step2: Connect to your EC2 instance using following commands in your terminal

Ssh ubuntu@ipaddress
Ssh -i/ (address of pem file) ubuntu@ipaddress

Step3: Now we need to write the script that list the github users using shell script. For that we need to use Vim command

Vim filename.sh
Now click esc + I to get insert mode

Step4: Shell script is as follow:

```
#!/bin/bash



# GitHub API URL

API_URL="https://api.github.com"



# GitHub username and personal access token

USERNAME=$username

TOKEN=$token



# User and Repository information

REPO_OWNER=$1

REPO_NAME=$2



# Function to make a GET request to the GitHub API

function github_api_get {

    local endpoint="$1"
```

```bash
    local url="${API_URL}/${endpoint}"


    # Send a GET request to the GitHub API with authentication
    curl -s -u "${USERNAME}:${TOKEN}" "$url"
}
# Function to list users
function list_users {
    local endpoint="repos/${REPO_OWNER}/${REPO_NAME}/collaborators"


    # Fetch the list of collaborators on the repository
    collaborators="$(github_api_get "$endpoint" |jq -r '.[]| .login')"


    # Display the list of collaborators
    if [[ -z "$collaborators" ]]; then
        echo "No users with full access found for ${REPO_OWNER}/${REPO_NAME}."
    else
        echo "Users with full access to ${REPO_OWNER}/${REPO_NAME}:"
        echo "$collaborators"
    fi
}
# Main script
echo "Listing users with full access to ${REPO_OWNER}/${REPO_NAME}..."
list_users
```

step5: Before executing the script we need to export the username and token of the github

```
export username="        "
export token = "       "
```

step6: Now we use the execution command followed by repository-owner and repository-name

```
./filename.sh repo-owner repo-name
```

Output:

Listing users with full access to repo-owner/repo-name....

Users with full access to repo-owner/repo-name:

User1

User2

User3