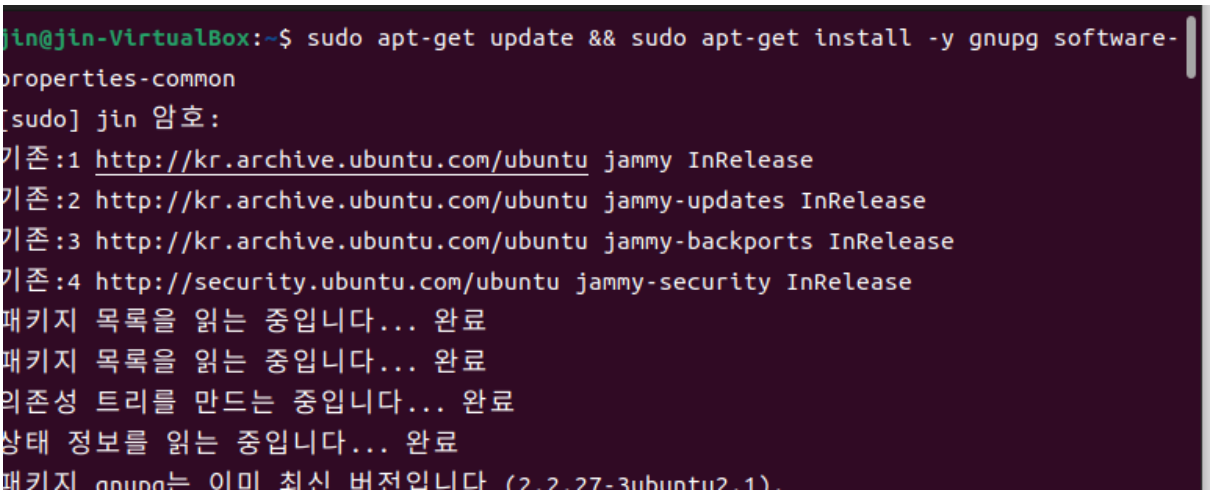
Cloudgoat_task_YUJIN CHOI(DF)



I started by configuring the AWS CLI using the `aws configure` command to set up access keys, the default region, and the output format. The profile used was `cloudgoat-admins`. Despite configuring the profile correctly, issues were encountered later in the process related to profile recognition and validation.

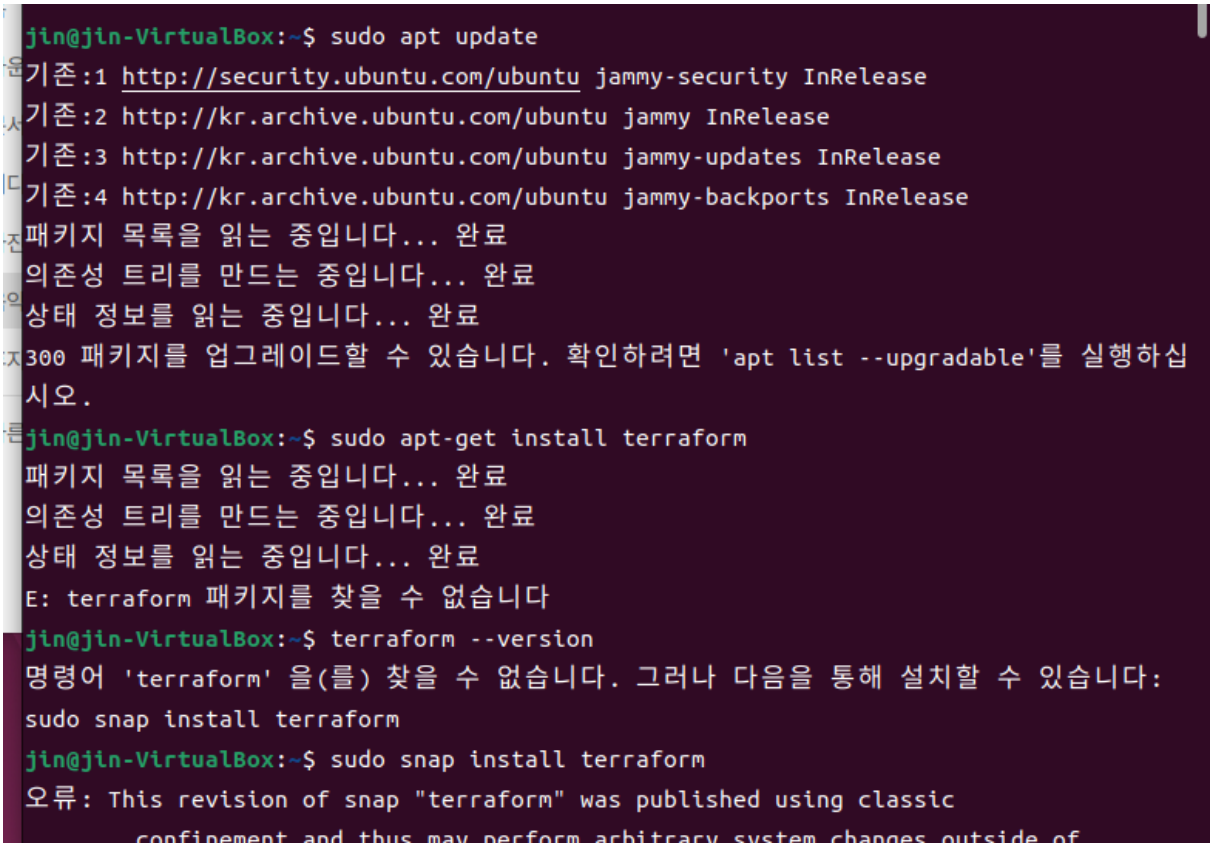
```
jin@jin-VirtualBox:~$ wget -O- https://apt.releases.hashicorp.com/gpg | \
gpg --dearmor | \
sudo tee /usr/share/keyrings/hashicorp-archive-keyring.gpg > /dev/null
--2024-07-09 14:26:15--  https://apt.releases.hashicorp.com/gpg
apt.releases.hashicorp.com (apt.releases.hashicorp.com) 해석 중... 13.225.131.99, 13
.225.131.7, 13.225.131.54, ...
다음으로 연결 중: apt.releases.hashicorp.com (apt.releases.hashicorp.com)|13.225.131
.99|:443... 연결했습니다.
HTTP 요청을 보냈습니다. 응답 기다리는 중... 200 OK
길이: 3980 (3.9K) [binary/octet-stream]
저장 위치: ‘STDOUT’

-                   100%[===================>]   3.89K  --.-KB/s    / 0s

2024-07-09 14:26:15 (1.52 GB/s) - 표준 출력에 기록: [3980/3980]

jin@jin-VirtualBox:~$ gpg --no-default-keyring \
--keyring /usr/share/keyrings/hashicorp-archive-keyring.gpg \
--fingerprint
gpg: directory '/home/jin/.gnupg' created
gpg: /home/jin/.gnupg/trustdb.gpg: trustdb created
/usr/share/keyrings/hashicorp-archive-keyring.gpg
```

Terraform was installed to manage the infrastructure, and the CloudGoat repository was cloned from GitHub. After setting up a Python virtual environment, the necessary dependencies were installed. There were some challenges with the Terraform installation, which required additional steps to configure the HashiCorp repository. These were resolved, and Terraform was successfully installed.



I used the `cloudgoat.py` script to configure the AWS profile for CloudGoat. This involved creating a `config.yml` file and setting the default AWS profile. Although the profile was set up, subsequent attempts to deploy the scenario failed due to an "Invalid number literal" error, which suggested an issue with either the whitelist variable or the profile setup.

```
jin@jin-VirtualBox:~$ echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-key
ring.gpg] \
https://apt.releases.hashicorp.com $(lsb_release -cs) main" | \
sudo tee /etc/apt/sources.list.d/hashicorp.list
deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releas
es.hashicorp.com jammy main
jin@jin-VirtualBox:~$ sudo apt-get install terraform
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
E: terraform 패키지를 찾을 수 없습니다
jin@jin-VirtualBox:~$ git clone https://github.com/RhinoSecurityLabs/cloudgoat.git
'cloudgoat'에 복제합니다...
remote: Enumerating objects: 4978, done.
remote: Counting objects: 100% (1343/1343), done.
remote: Compressing objects: 100% (489/489), done.
remote: Total 4978 (delta 990), reused 1002 (delta 842), pack-reused 3635
오브젝트를 받는 중: 100% (4978/4978), 15.09 MiB | 2.76 MiB/s, 완료.
델타를 알아내는 중: 100% (2233/2233), 완료.
jin@jin-VirtualBox:~$ cd cloudgoat
jin@jin-VirtualBox:~/cloudgoat$ python3 -m venv .venv
jin@jin-VirtualBox:~/cloudgoat$ source .venv/bin/activate
(.venv) jin@jin-VirtualBox:~/cloudgoat$ pip3 install -r ./requirements.txt
```

```
(.venv) jin@jin-VirtualBox:~/cloudgoat$ chmod +x cloudgoat.py
(.venv) jin@jin-VirtualBox:~/cloudgoat$ ./cloudgoat.py config profile
Terraform not found. Please install Terraform before using CloudGoat.
(.venv) jin@jin-VirtualBox:~/cloudgoat$ sudo apt-get install terraform
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
E: terraform 패키지를 찾을 수 없습니다
(.venv) jin@jin-VirtualBox:~/cloudgoat$ sudo apt-get update && sudo apt-get install
-y gnupg software-properties-common
받기:1 https://apt.releases.hashicorp.com jammy InRelease [12.9 kB]
기존:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
받기:3 https://apt.releases.hashicorp.com jammy/main amd64 Packages [138 kB]
기존:4 http://kr.archive.ubuntu.com/ubuntu jammy InRelease
받기:5 https://apt.releases.hashicorp.com jammy/main i386 Packages [60.6 kB]
기존:6 http://kr.archive.ubuntu.com/ubuntu jammy-updates InRelease
기존:7 http://kr.archive.ubuntu.com/ubuntu jammy-backports InRelease
내려받기 212 kB바이트, 소요시간 2초 (125 kB바이트/초)
패키지 목록을 읽는 중입니다... 완료
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
```

```
(.venv) jin@jin-VirtualBox:~/cloudgoat$ wget -O- https://apt.releases.hashicorp.com/
gpg | \
gpg --dearmor | \
sudo tee /usr/share/keyrings/hashicorp-archive-keyring.gpg > /dev/null
--2024-07-09 14:31:55--  https://apt.releases.hashicorp.com/gpg
apt.releases.hashicorp.com (apt.releases.hashicorp.com) 해석 중... 13.225.131.99, 13
.225.131.75, 13.225.131.7, ...
다음으로 연결 중: apt.releases.hashicorp.com (apt.releases.hashicorp.com)|13.225.131
.99|:443... 연결했습니다.
HTTP 요청을 보냈습니다. 응답 기다리는 중... 200 OK
길이: 3980 (3.9K) [binary/octet-stream]
저장 위치: ‘STDOUT’

-                   100%[===================>]   3.89K  --.-KB/s    / 0s

2024-07-09 14:31:55 (3.39 GB/s) - 표준 출력에 기록: [3980/3980]

(.venv) jin@jin-VirtualBox:~/cloudgoat$ gpg --no-default-keyring \
--keyring /usr/share/keyrings/hashicorp-archive-keyring.gpg \
--fingerprint
/usr/share/keyrings/hashicorp-archive-keyring.gpg
-------------------------------------------------
pub   rsa4096 2023-01-10 [SC] [expires: 2028-01-09]
```

```
(.venv) jin@jin-VirtualBox:~/cloudgoat$ gpg --no-default-keyring \
--keyring /usr/share/keyrings/hashicorp-archive-keyring.gpg \
--fingerprint
/usr/share/keyrings/hashicorp-archive-keyring.gpg
-------------------------------------------------
pub   rsa4096 2023-01-10 [SC] [expires: 2028-01-09]
      798A EC65 4E5C 1542 8C8E  42EE AA16 FCBC A621 E701
uid           [ unknown] HashiCorp Security (HashiCorp Package Signing) <security+pa
ckaging@hashicorp.com>
sub   rsa4096 2023-01-10 [S] [expires: 2028-01-09]

(.venv) jin@jin-VirtualBox:~/cloudgoat$ sudo apt update
기존:1 https://apt.releases.hashicorp.com jammy InRelease
기존:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
기존:3 http://kr.archive.ubuntu.com/ubuntu jammy InRelease
기존:4 http://kr.archive.ubuntu.com/ubuntu jammy-updates InRelease
기존:5 http://kr.archive.ubuntu.com/ubuntu jammy-backports InRelease
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
300 패키지를 업그레이드할 수 있습니다. 확인하려면 'apt list --upgradable'를 실행하십
시오.
(.venv) jin@jin-VirtualBox:~/cloudgoat$ sudo apt-get install terraform
패키지 목록을 읽는 중입니다... 완료
```

```
(.venv) jin@jin-VirtualBox:~/cloudgoat$ touch ~/.bashrc
(.venv) jin@jin-VirtualBox:~/cloudgoat$ terraform -install-autocomplete
(.venv) jin@jin-VirtualBox:~/cloudgoat$ mkfit learn-terraform-docker-container
명령어 'mkfit' 을(를) 찾을 수 없습니다. 다음 명령어로 시도하시겠습니까:
  snap kfitkfit의 명령어 ' (0.1.4)'
'snap info <snapname>'에서 추가 버전을 확인하십시오.
(.venv) jin@jin-VirtualBox:~/cloudgoat$ mkdir learn-terraform-docker-container
(.venv) jin@jin-VirtualBox:~/cloudgoat$ cd learn-terraform-docker-container
(.venv) jin@jin-VirtualBox:~/cloudgoat/learn-terraform-docker-container$ terraform i
nit
Terraform initialized in an empty directory!

The directory has no Terraform configuration files. You may begin working
with Terraform immediately by creating Terraform configuration files.
(.venv) jin@jin-VirtualBox:~/cloudgoat/learn-terraform-docker-container$ cd
(.venv) jin@jin-VirtualBox:~$ ls
cloudgoat  snap  공개  다운로드  문서  바탕화면  비디오  사진  음악  템플릿
(.venv) jin@jin-VirtualBox:~$ sudo ./aws/install
sudo: ./aws/install: 명령이 없습니다
(.venv) jin@jin-VirtualBox:~$ sudo apt-get ./aws/install
E: 잘못된 작업 ./aws/install
```

```
(.venv) jin@jin-VirtualBox:~$ aws --version
명령어 'aws' 을(를) 찾을 수 없습니다. 그러나 다음을 통해 설치할 수 있습니다:
sudo snap install aws-cli  # version 1.15.58, or
sudo apt  install awscli   # version 1.22.34-1
'snap info aws-cli'에서 추가적인 버전을 확인하십시오.
(.venv) jin@jin-VirtualBox:~$ sudo apt install awscli
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
다음의 추가 패키지가 설치될 것입니다 :
  docutils-common groff gsfonts imagemagick imagemagick-6-common
  imagemagick-6.q16 libaom3 libdav1d5 libde265-0 libfftw3-double3 libheif1
  libilmbase25 libjxr-tools libjxr0 liblqr-1-0 libmagickcore-6.q16-6
  libmagickcore-6.q16-6-extra libmagickwand-6.q16-6 libnetpbm10 libopenexr25
  libx265-199 netpbm psutils python3-botocore python3-docutils
  python3-jmespath python3-pyasn1 python3-pygments python3-roman python3-rsa
  python3-s3transfer
제안하는 패키지:
  imagemagick-doc autotrace enscript ffmpeg gimp gnuplot grads graphviz hp2xx
```

```
(.venv) jin@jin-VirtualBox:~$ aws --version
aws-cli/1.22.34 Python/3.10.12 Linux/6.5.0-41-generic botocore/1.23.34
(.venv) jin@jin-VirtualBox:~$ ./cloudgoat.py config profile
bash: ./cloudgoat.py: 그런 파일이나 디렉터리가 없습니다
(.venv) jin@jin-VirtualBox:~$ cd cloudgoat/
(.venv) jin@jin-VirtualBox:~/cloudgoat$ ./cloudgoat.py config profile
No configuration file was found at /home/jin/cloudgoat/config.yml
Would you like to create this file with a default profile name now? [y/n]: n
(.venv) jin@jin-VirtualBox:~/cloudgoat$ ./cloudgoat.py config profile
No configuration file was found at /home/jin/cloudgoat/config.yml
Would you like to create this file with a default profile name now? [y/n]: y
Enter the name of your default AWS profile: BoB13CloudGoatAdminDefault
A default profile name of "BoB13CloudGoatAdminDefault" has been saved.
(.venv) jin@jin-VirtualBox:~/cloudgoat$ aws configuer --profile BoB13CloudGoatAdminD
efault
To see help text, you can run:

  aws help
```

I attempted to deploy the CloudGoat scenario using the configured profile. However, the deployment failed with errors related to AWS account validation and security tokens. Despite multiple attempts to reconfigure and retry, the deployment issues persisted.

```
The AWS profile to use

Enter a value: cloudgoat-admins-profile


Error: Invalid number literal

  on <value for var.cg_whitelist> line 1:
  (source code not available)

Failed to recognize the value of this number literal.


Error: No value for required variable

  on variables.tf line 16:
  16: variable "cg_whitelist" {

The root module input variable "cg_whitelist" is not set, and has no default
value. Use a -var or -var-file command line argument to provide a value for
this variable.
```

Due to the ongoing issues with profile configuration and deployment, I plan to create a new IAM user with the necessary permissions and try deploying the

scenario again. This will involve reconfiguring the profile and proceeding with the deployment. Once the scenario is successfully deployed, I will analyze the CloudTrail logs to identify potential detection events.