# [DEMO] Creating our lab environment

Christophe • September 12, 2023

To get started, you can either use CloudGoat through Docker or not, as we've talked about when we first set up CloudGoat. I'll show you how to use it through Docker first but I would actually recommend not running it through Docker unless you're used to working with containers.

## If you are running CloudGoat through Docker:

You can either mount your AWS credentials when you start the container:

```
$ docker run -it -v ~/.aws:/root/.aws/ rhinosecuritylabs/cloudgoat:latest

$ ls ~/.aws/

$ ./cloudgoat.py config profile

$ ./cloudgoat.py create vulnerable_lambda
```

Or you can launch the container:

```
$ docker run -it rhinosecuritylabs/cloudgoat:latest
WARNING: The requested image's platform (linux/amd64) does not match the detected host platform
(linux/arm64/v8) and no specific platform was requested
bd7d0c15925c:/usr/src/cloudgoat# ls ~/.aws

$ ls: /root/.aws: No such file or directory

$ aws configure
AWS Access Key ID [None]:
etc...

$ ./cloudgoat.py config profile

$ ./cloudgoat.py create vulnerable_lambda
```

The first approach is faster and easier to spin up and down, but it does expose your credentials to anyone who has access to this container. The choice is yours!

## If you're not running CloudGoat through Docker:

To get started, we need to tie CloudGoat's configuration to our AWS credentials.

Let's make sure we're in our CloudGoat directory so we can run this command:

```
./cloudgoat.py config profile
```

It will ask you to provide the name of your default AWS profile. This needs to match the name that you used when you configured your AWS CLI — if you left it to default, then type in default.

```
No configuration file was found at /Users/christophe/Documents/Projects/cybr/teaching/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: default
```

After that, we are ready to create this scenario by typing in:

```
./cloudgoat.py create vulnerable_lambda
```

We then need to wait about 30 seconds to maybe a minute for all of the resources to get created via Terraform.

If successful, you should see an output that looks something like this:

```
Apply complete! Resources: 8 added, 0 changed, 0 destroyed.

Outputs:

cloudgoat_output_aws_account_id = "272281913033"
cloudgoat_output_bilbo_access_key_id = "AKIAT6ZKEI3EXH3ZPB2N"
cloudgoat_output_bilbo_secret_key = <sensitive>
profile = "default"
scenario_cg_id = "vulnerable_lambda_cgidbdmusq6y4k"

[cloudgoat] terraform apply completed with no error code.

[cloudgoat] terraform output completed with no error code.
cloudgoat_output_aws_account_id = 272281913033
cloudgoat_output_bilbo_access_key_id = AKIAT6ZKEI3EXH3ZPB2N
cloudgoat_output_bilbo_secret_key = BaJK5CUk4OFdisf5EMeT5OuG/kRuKf5PgHSMr9vP
profile = default
scenario_cg_id = vulnerable_lambda_cgidbdmusq6y4k
```

Yours will be slightly different of course but you should have 8 new resources created.

Now it's important that you save the generated access key id and secret key:

```
cloudgoat_output_aws_account_id = 272281913033
cloudgoat_output_bilbo_access_key_id = AKIAT6ZKEI3EXDRFU47W
cloudgoat_output_bilbo_secret_key = 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
profile = default
```

Because CloudGoat created a user named Bilbo for us to use in this scenario, and those access keys give us access to that user! So either keep this terminal window open, copy/paste them somewhere else, or remember this path:

```
/Users/christophe/Documents/Projects/cybr/teaching/cloudgoat/vulnerable_lambda_cgidbdmusq6y4k/start.txt
```

And then you can always use `cat` to output the credentials:

```
cat /Users/christophe/Documents/Projects/cybr/teaching/cloudgoat/vulnerable_lambda_cgidbdmusq6y4k/start.txt
```

We're now ready to get started! Complete this lesson and move on to the next!

---

## Responses

Rafael
May 14, 2024

I am getting the following error. I've uninstalled and reinstalled but for some reason I keep getting an error saying Terraform is not found.

┌──(kali㉿kali)-[~/cloudgoat]
└─$ ./cloudgoat.py config profile
Terraform not found. Please install Terraform before using CloudGoat.

┌──(kali㉿kali)-[~/cloudgoat]
└─$ terraform -version
Command 'terraform' not found, but can be installed with:
sudo apt install terraform
Do you want to install it? (N)y/y
sudo apt install terraform
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
terraform is already the newest version (1.6.3-0kali1).
The following packages were automatically installed and are no longer required:
libabsl20220623 libaio1 libatk-adaptor libboost-dev libboost1.83-dev libopenblas-dev libopenblas-pthread-dev libopenblas0 libpython3-all-dev libpython3.12 libpython3.12-dev libxsimd-dev python3-all-dev python3-anyjson python3-beniget python3-gast python3-pyatspi python3-pypdf2 python3-pyppeteer python3-pyrsistent python3-pythran python3.12-dev xtl-dev
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 12 not upgraded.

Reply

Rafael
May 14, 2024

Fixed the below, please disregard.

If you're using `vim` to edit the `/etc/apt/sources.list.d/hashicorp.list` file and it shows as read-only, you can follow these steps to edit the file:

1. Open the file with `vim` using sudo:
`sudo vim /etc/apt/sources.list.d/hashicorp.list`

2. You'll see the file contents in read-only mode. To switch to edit mode, press the following keys:
`:`
This will put you in command mode.

3. Then, type the following command and press Enter:
`:set modifiable`
This will allow you to edit the file.

4. Now, you can use the standard `vim` commands to edit the file. To replace the entire contents with the new line, first, press the following keys to delete all the existing lines:
`dG`
This will delete all lines from the current line to the end of the file.

5. Next, press the following keys to enter insert mode:
`i`

6. Paste the new line by right-clicking in the terminal or by pressing `Ctrl+Shift+V`:
`deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com jammy main`

7. After pasting the line, press the `Esc` key to exit insert mode.

8. Finally, save the changes and exit `vim` by typing the following command and pressing Enter:
`:wq`
This will save the changes (`w`) and quit (`q`) the editor.

After following these steps, the `/etc/apt/sources.list.d/hashicorp.list` file should contain only the provided line, and you should be able to proceed with updating the package index and installing Terraform.

Remember, when editing system files, it's essential to have the necessary permissions, which is why we use `sudo` to run `vim` with elevated privileges.

Reply

Christophe
May 14, 2024

Thanks for sharing your solution!

Reply