**STEP 1: Installing**

./csewizcli setup

Check if wizcli binary exists on system

wizcli version

In case it does not return any version or gives error then

Download Latest Version

curl -o wizcli https://wizcli.app.wiz.io/wizcli

chmod +x wizcli

mv wizcli /usr/local/bin

wizcli version

**STEP 2: Upgrade**

./csewizcli upgrade

wizcli version

In case it returns any version or gives error then

Check if wizcli binary exists on system and remove it first

rm /usr/local/bin/wizcli

and then

Download Latest Version

curl -o wizcli https://wizcli.app.wiz.io/wizcli

chmod +x wizcli

mv wizcli /usr/local/bin

wizcli version

Note: In case it does return any version or gives error just install as like first time setup.

**STEP 3: Authenticate**

It then does authentication.

./csewizcli auth -🡪 wizcli auth --id <value> --secret <value>

“Hurray Wizcli is ready to scan Misconfiguration in Terraform, Docker, Kubernetes and Images”

**STEP 4: Scanning Terraform Azure**

./csewizcli terraform azure scan <name> <project\_name> ---------🡪 wizcli iac scan –policy “AZURE TFE” --path ./ --name <name> --project <project\_name>

or

./csewizcli terraform azure scan <path\_to\_file> <name> <project\_name> ---🡪 wizcli iac scan –policy “AZURE TFE” –path <path\_to\_file>/ --name <name> --project <project\_name>

When the custom script is executed, the user should get a message “Started Scanning Azure Terraform Misconfiguration”.

Note these above command should check the files its scanning, if its.tf, json it lists and shows the files before scanning. If the files are non .tf and. json then exit stating it did not found appropriate files to scan.

**STEP 5: Scanning Terraform GCP**

./csewizcli terraform gcp scan <name> <project\_name> ---------🡪 wizcli iac scan –policy “GCP TFE” --path ./ --name <name> --project <project\_name>

or

./csewizcli terraform gcp scan <path\_to\_file> <name> <project\_name> ---🡪 wizcli iac scan –policy “GCP TFE” –path <path\_to\_file>/ --name <name> --project <project\_name>

When the custom script is executed, the user should get a message “Started Scanning Google Terraform Misconfiguration”.

Note these above command should check the files its scanning, if its.tf, json it lists and shows the files before scanning. If the files are non .tf and. json then exit stating it did not find appropriate files to scan.

**STEP 6: Docker Scanning**

./csewizcli docker scan <name> <project\_name> ---------🡪 wizcli iac scan –policy “THD-Docker-IaC” --path ./ --name <name> --project <project\_name>

or

./csewizcli docker scan <path\_to\_file> <name> <project\_name> ---🡪 wizcli iac scan –policy “THD-Docker-IaC” –path <path\_to\_file>/ --name <name> --project <project\_name>

When the custom script is executed, the user should get a message “Started Scanning Dockerfile Misconfiguration”.

Note these above command should check the files its scanning, if its Dockefile it lists and shows the files before scanning. If the files are not Dockerfile then exit stating it did not found appropriate files to scan.

**STEP 7: Kubernetes Scanning**

./csewizcli kubernetes scan <name> <project\_name> ---------🡪 wizcli iac scan –policy “THD-Docker-IaC” --path ./ --name <name> --project <project\_name>

or

./csewizcli kubernetes scan <path\_to\_file> <name> <project\_name> ---🡪 wizcli iac scan –policy “THD-Docker-IaC” –path <path\_to\_file>/ --name <name> --project <project\_name>

When the custom script is executed, the user should get a message “Started Scanning Kubernetes Misconfiguration”.

Note these above command should check the files its scanning, if it’s. yaml it lists and shows the files before scanning. If the files are not .yaml then exit stating it did not find appropriate files to scan.

**STEP 8: Image Scanning**

./csewizcli image scan <name> <project\_name> <image> ----🡪 wizcli docker scan -i <image>

When the custom script is executed, the user should get a message “Started Image Scanning For Vulnerabilities”.