Configuration Manual

MSc Research Project
Cloud Computing

Anjali Sandeep Wagaskar
Student ID: x23340363

School of Computing
National College of Ireland

Supervisor: Dr. Punit Gupta

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Anjali Sandeep Wagaskar
Student ID:	x23340363
Programme:	Cloud Computing
Year:	2025
Module:	MSc Research Project
Supervisor:	Dr. Punit Gupta
Submission Due Date:	11/08/2025
Project Title:	Configuration Manual
Word Count:	844
Page Count:	9

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Anjali Sandeep Wagaskar
Date:	14th September 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

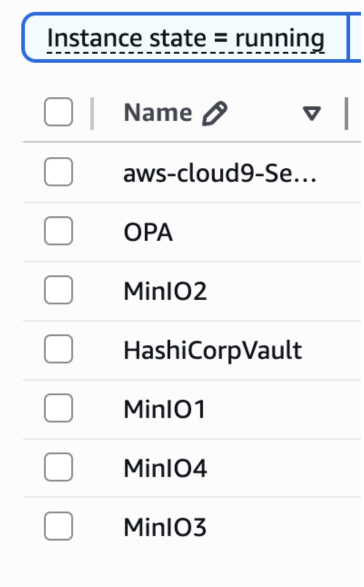
Configuration Manual

Anjali Sandeep Wagaskar
x23340363

1 Prerequisites on EC2

EC2 instances running Ubuntu 20.04 or Amazon Linux 2 At least 4 vCPU, 8GB RAM recommended Security Groups configured to allow:

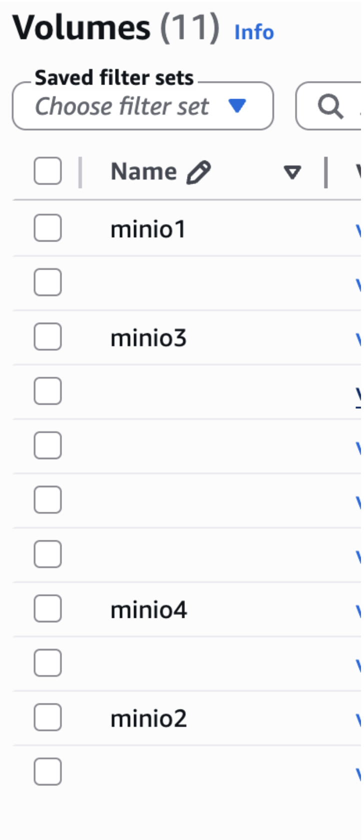
- SSH (port 22) from your IP
- Vault port (8200) between app and Vault instances
- OPA port (8181) open internally
- MinIO port (9000) open internally or externally based on use
- FastAPI port (8000) open to your client IP or public
- Attached EBS volumes or instance store for MinIO data



Instance state = running

□	Name
□	aws-cloud9-Se...
□	OPA
□	MinIO2
□	HashiCorpVault
□	MinIO1
□	MinIO4
□	MinIO3

Figure 1: configured ec2



□	Name
□	minio1
□	
□	minio3
□	
□	
□	
□	
□	minio4
□	
□	minio2
□	

Figure 2: set and attach EBS volume on all 4 instances

2 Environment Setup on EC2 Instances

2.1 Update and install dependencies

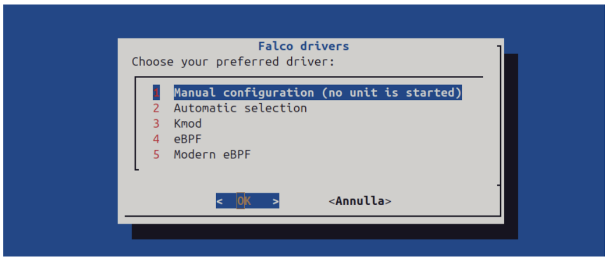
```
sudo apt update && sudo apt upgrade -y

sudo apt install -y wget curl unzip python3 python3-pip
```

2.2 Falco Installation and Configuration on EC2

set up according to installation guide of falco on there official site(Falco Security Project; 2025).

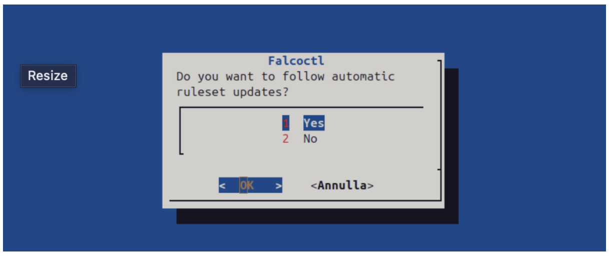
2.2.1 Set up the package repository

```
curl -fsSL https://falco.org/repo/falcosecurity-packages.asc | \
sudo gpg --dearmor -o /usr/share/keyrings/falco-archive-keyring.gpg
sudo bash -c 'cat << EOF > /etc/apt/sources.list.d/falcosecurity.list
deb [signed-by=/usr/share/keyrings/falco-archive-keyring.gpg] https://download.falco.
EOF'
sudo apt-get update -y
```



```
Falco drivers
Choose your preferred driver:

1  Manual configuration (no unit is started)
2  Automatic selection
3  Kmod
4  eBPF
5  Modern eBPF

<  OK  >          <Annulla>
```

Figure 3: Choose the Modern eBPF option



```
Falcoctl
Do you want to follow automatic
ruleset updates?

1  Yes
2  No

<  OK  >     <Annulla>
```

Figure 4: Choose YES

2.2.2 Install dialog

```
sudo apt-get install -y dialog
```

2.2.3 Install Falco

```
sudo apt-get install -y falco
```

Then select "Modern eBPF" option on the prompt,doyou want to follow autometic rules select "yes", and this enable the usage of the modern eBPF-based driver34.

make sure now if the falco server is running :

```
sudo systemctl status falco-modern-bpf.service
```

once everything is set configure falco rules in 5

```
/etc/falco/falco_rules.local.yaml
```

add this:

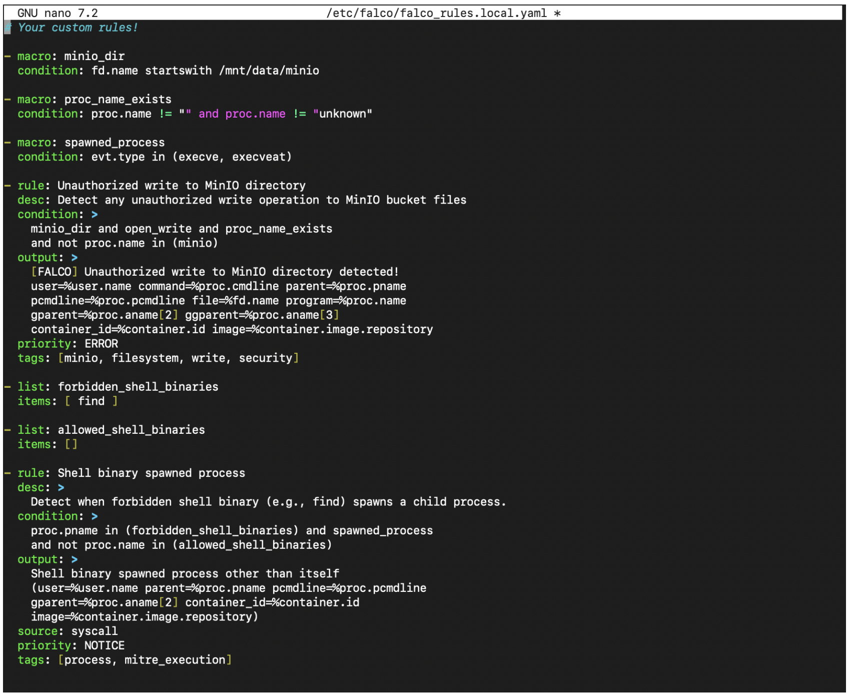
```
 macro: minio_dir
condition: fd.name startswith /mnt/data/minio
```

- `macro: proc_name_exists`
  `condition: proc.name != "" and proc.name != "unknown"`

- `macro: spawned_process`
  `condition: evt.type in (execve, execveat)`

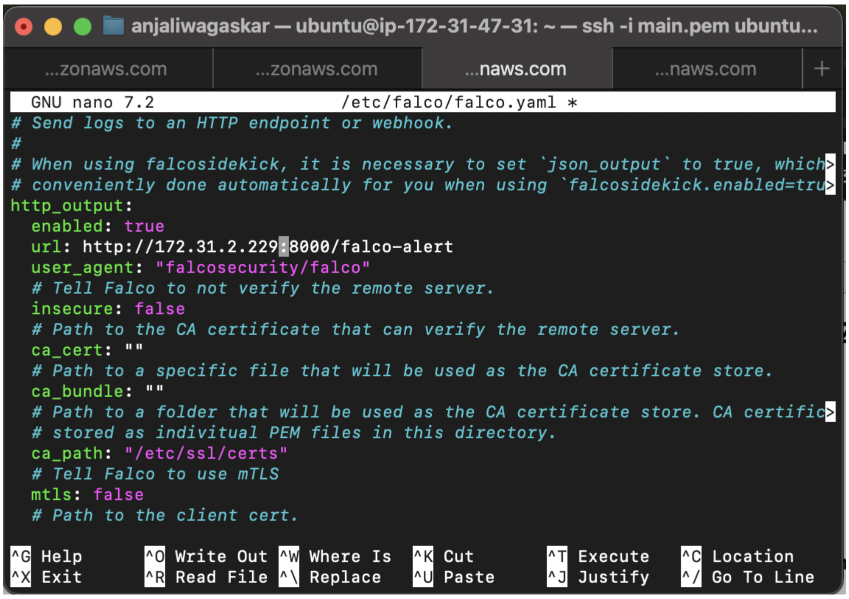
- `rule: Unauthorized write to MinIO directory`
  `desc: Detect any unauthorized write operation to MinIO bucket files`
  `condition: >`

```
    minio_dir and open_write and proc_name_exists
    and not proc.name in (minio)
  output: >
    [FALCO] Unauthorized write to MinIO directory detected!
    user=%user.name command=%proc.cmdline parent=%proc.pname
    pcmdline=%proc.pcmdline file=%fd.name program=%proc.name
    gparent=%proc.aname[2] ggparent=%proc.aname[3]
    container_id=%container.id image=%container.image.repository
  priority: ERROR
  tags: [minio, filesystem, write, security]

- list: forbidden_shell_binaries
  items: [ find ]

- list: allowed_shell_binaries
  items: []

- rule: Shell binary spawned process
  desc: >
    Detect when forbidden shell binary (e.g., find) spawns a child process.
  condition: >
    proc.pname in (forbidden_shell_binaries) and spawned_process
    and not proc.name in (allowed_shell_binaries)
  output: >
    Shell binary spawned process other than itself
    (user=%user.name parent=%proc.pname pcmdline=%proc.pcmdline
    gparent=%proc.aname[2] container_id=%container.id
    image=%container.image.repository)
  source: syscall
  priority: NOTICE
  tags: [process, mitre_execution]
```



```
GNU nano 7.2                    /etc/falco/falco_rules.local.yaml *
# Your custom rules!

- macro: minio_dir
  condition: fd.name startswith /mnt/data/minio

- macro: proc_name_exists
  condition: proc.name != "" and proc.name != "unknown"

- macro: spawned_process
  condition: evt.type in (execve, execveat)

- rule: Unauthorized write to MinIO directory
  desc: Detect any unauthorized write operation to MinIO bucket files
  condition: >
    minio_dir and open_write and proc_name_exists
    and not proc.name in (minio)
  output: >
    [FALCO] Unauthorized write to MinIO directory detected!
    user=%user.name command=%proc.cmdline parent=%proc.pname
    pcmdline=%proc.pcmdline file=%fd.name program=%proc.name
    gparent=%proc.aname[2] ggparent=%proc.aname[3]
    container_id=%container.id image=%container.image.repository
  priority: ERROR
  tags: [minio, filesystem, write, security]

- list: forbidden_shell_binaries
  items: [ find ]

- list: allowed_shell_binaries
  items: []

- rule: Shell binary spawned process
  desc: >
    Detect when forbidden shell binary (e.g., find) spawns a child process.
  condition: >
    proc.pname in (forbidden_shell_binaries) and spawned_process
    and not proc.name in (allowed_shell_binaries)
  output: >
    Shell binary spawned process other than itself
    (user=%user.name parent=%proc.pname pcmdline=%proc.pcmdline
    gparent=%proc.aname[2] container_id=%container.id
    image=%container.image.repository)
  source: syscall
  priority: NOTICE
  tags: [process, mitre_execution]
```

Figure 5: setting up rules



```
GNU nano 7.2                    /etc/falco/falco.yaml *
# Send logs to an HTTP endpoint or webhook.
#
# When using falcosidekick, it is necessary to set `json_output` to true, which>
# conveniently done automatically for you when using `falcosidekick.enabled=tru>
http_output:
  enabled: true
  url: http://172.31.2.229:8000/falco-alert
  user_agent: "falcosecurity/falco"
  # Tell Falco to not verify the remote server.
  insecure: false
  # Path to the CA certificate that can verify the remote server.
  ca_cert: ""
  # Path to a specific file that will be used as the CA certificate store.
  ca_bundle: ""
  # Path to a folder that will be used as the CA certificate store. CA certific>
  # stored as indivitual PEM files in this directory.
  ca_path: "/etc/ssl/certs"
  # Tell Falco to use mTLS
  mtls: false
  # Path to the client cert.
```

Figure 6: providing url where you want to print alert

then restart falco

```
sudo systemctl restart falco
```

3 HashiCorp Vault Installation and Setup on EC2

install vault according to the hashicorp vault installation guide(HashiCorp; 2025)
Download Vault binary

```
wget https://releases.hashicorp.com/vault/1.12.0/vault_1.12.0_linux_amd64.zip
unzip vault_1.12.0_linux_amd64.zip
sudo mv vault /usr/local/bin/
```

Configure Vault server Example for dev mode

```
vault server -dev -dev-root-token-id="root" -dev-listen-address="0.0.0.0:8200" &
```

For production, create a config file /etc/vault/config.hcl:

```
storage "file" {
  path = "/mnt/data/vault"
}

listener "tcp" {
  address = "0.0.0.0:8200"
  tls_disable = 1
}

ui = true
```

then start server in production

```
vault server -config=/etc/vault/config.hcl &
```

Export environment variable on app

```
export VAULT_ADDR='http://<Vault-EC2-IP>:8200'
VAULT_TOKEN='<hvs.currently runing server token>'
vault operator init
vault operator unseal <unseal_key>
vault policy write new-policy ./policy.hcl
vault token create -policy="new-policy"
```

Run OPA server with policy loaded:

```
opa run --server --addr 0.0.0.0:8181 policy.rego
```

4 Open Policy Agent (OPA) Setup on EC2

use user guide to install OPA(Lee; 2022). Download and install OPA binary:

```
wget https://openpolicyagent.org/downloads/latest/opa_linux_amd64
chmod +x opa_linux_amd64
sudo mv opa_linux_amd64 /usr/local/bin/opa
```

Place your policy.rego on the EC2 instance (e.g., /home/ubuntu/policy.rego)

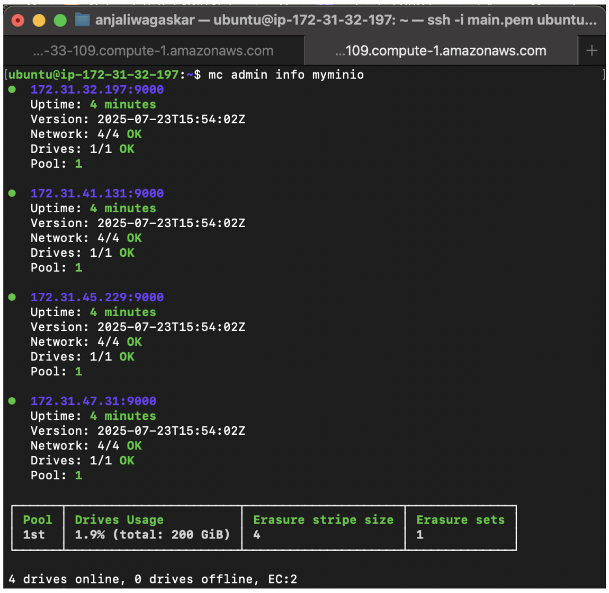
```
package main

default allow := false
```

```
roles := {
 "manager": {"allowed_endpoints": [
                {"path": "/v1/secret/myapp", "actions": ["read", "create", "updat
                {"path": "api/login/token", "actions": ["read"]},
        ]},
 "employee": {"allowed_endpoints": [
                {"path": "/v1/secret/myapp", "actions": ["read", "create"]},
                {"path": "api/login/token", "actions": ["read"]},
        ]},
 "admin": {"allowed_endpoints": []},
}

allow if {
 is_admin
}
    has_path_perms if {
 ae := roles[input.roles[_]].allowed_endpoints[_]
 not endswith(ae.path, "*")
 input.path == ae.path
 input.action in ae.actions
}

has_path_perms if {
 ae := roles[input.roles[_]].allowed_endpoints[_]
 endswith(ae.path, "*")
 startswith(input.path, trim_suffix(ae.path, "*"))
 input.action in ae.actions
}
```

5 MinIO Installation and Setup on EC2

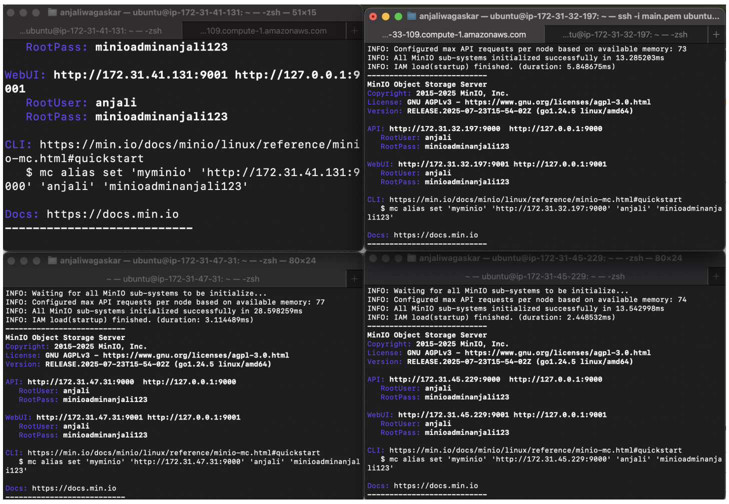
install MinIO from there official installation set up in distributed manner(Hernandez; 2022). system: ubantu 20.04 run every command on all the systems : Note: 10.0.0.1 should be the private IP of every system

```
sudo apt update && sudo apt upgrade -y

sudo apt install wget unzip -y

sudo useradd -r minio-user -s /sbin/nologin

sudo mkdir -p /mnt/data/minio
sudo chown -R minio-user:minio-user /mnt/data/minio

wget https://dl.min.io/server/minio/release/linux-amd64/minio
chmod +x minio
sudo mv minio /usr/local/bin/
sudo bash -c 'cat > /etc/systemd/system/minio.service <<EOF

[Unit]
Description=MinIO
After=network.target

[Service]
User=minio-user
Group=minio-user
Environment="MINIO_ACCESS_KEY=admin"
Environment="MINIO_SECRET_KEY=password123"
ExecStart=/usr/local/bin/minio server \\
 http://10.0.0.1/mnt/data/minio \\
 http://10.0.0.2/mnt/data/minio \\
 http://10.0.0.3/mnt/data/minio \\
 http://10.0.0.4/mnt/data/minio
Restart=always
LimitNOFILE=65536

[Install]
WantedBy=multi-user.target
EOF'

sudo systemctl daemon-reload
sudo systemctl enable minio
sudo systemctl start minio
```


anjaliwagaskar — ubuntu@ip-172-31-32-197: ~ — ssh -i main.pem ubuntu...
...-33-109.compute-1.amazonaws.com
...109.compute-1.amazonaws.com

```
ubuntu@ip-172-31-32-197:~$ mc admin info myminio
●  172.31.32.197:9000
   Uptime: 4 minutes
   Version: 2025-07-23T15:54:02Z
   Network: 4/4 OK
   Drives: 1/1 OK
   Pool: 1

●  172.31.41.131:9000
   Uptime: 4 minutes
   Version: 2025-07-23T15:54:02Z
   Network: 4/4 OK
   Drives: 1/1 OK
   Pool: 1

●  172.31.45.229:9000
   Uptime: 4 minutes
   Version: 2025-07-23T15:54:02Z
   Network: 4/4 OK
   Drives: 1/1 OK
   Pool: 1

●  172.31.47.31:9000
   Uptime: 4 minutes
   Version: 2025-07-23T15:54:02Z
   Network: 4/4 OK
   Drives: 1/1 OK
   Pool: 1

┌──────┬──────────────────────────┬──────────────────────┬───────────────┐
│ Pool │ Drives Usage             │ Erasure stripe size  │ Erasure sets  │
│ 1st  │ 1.9% (total: 200 GiB)    │ 4                    │ 1             │
└──────┴──────────────────────────┴──────────────────────┴───────────────┘

4 drives online, 0 drives offline, EC:2
```

Figure 7: distributed setup showing drivers running



anjaliwagaskar — ubuntu@ip-172-31-41-131: ~ — -zsh — 51×15
...ubuntu@ip-172-31-41-131: ~ — -zsh
...109.compute-1.amazonaws.com

```
   RootPass: minioadminanjali123

WebUI: http://172.31.41.131:9001 http://127.0.0.1:9
001
   RootUser: anjali
   RootPass: minioadminanjali123

CLI: https://min.io/docs/minio/linux/reference/mini
o-mc.html#quickstart
   $ mc alias set 'myminio' 'http://172.31.41.131:9
000' 'anjali' 'minioadminanjali123'

Docs: https://docs.min.io
---------------------------
```

anjaliwagaskar — ubuntu@ip-172-31-32-197: ~ — ssh -i main.pem ubuntu...
...-33-109.compute-1.amazonaws.com
...tu@ip-172-31-32-197: ~ — -zsh

```
INFO: Configured max API requests per node based on available memory: 73
INFO: All MinIO sub-systems initialized successfully in 13.285203ms
INFO: IAM load(startup) finished. (duration: 5.848675ms)
---------------------------
MinIO Object Storage Server
Copyright: 2015-2025 MinIO, Inc.
License: GNU AGPLv3 - https://www.gnu.org/licenses/agpl-3.0.html
Version: RELEASE.2025-07-23T15-54-02Z (go1.24.5 linux/amd64)

API: http://172.31.32.197:9000  http://127.0.0.1:9000
   RootUser: anjali
   RootPass: minioadminanjali123

WebUI: http://172.31.32.197:9001 http://127.0.0.1:9001
   RootUser: anjali
   RootPass: minioadminanjali123

CLI: https://min.io/docs/minio/linux/reference/minio-mc.html#quickstart
   $ mc alias set 'myminio' 'http://172.31.32.197:9000' 'anjali' 'minioadminanja
li123'

Docs: https://docs.min.io
---------------------------
```

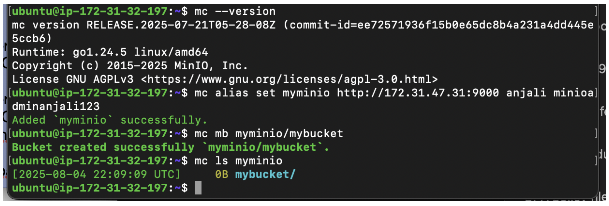
anjaliwagaskar — ubuntu@ip-172-31-47-31: ~ — -zsh — 80×24
~ — ubuntu@ip-172-31-47-31: ~ — -zsh

```
INFO: Waiting for all MinIO sub-systems to be initialize...
INFO: Configured max API requests per node based on available memory: 77
INFO: All MinIO sub-systems initialized successfully in 28.598259ms
INFO: IAM load(startup) finished. (duration: 3.114489ms)
---------------------------
MinIO Object Storage Server
Copyright: 2015-2025 MinIO, Inc.
License: GNU AGPLv3 - https://www.gnu.org/licenses/agpl-3.0.html
Version: RELEASE.2025-07-23T15-54-02Z (go1.24.5 linux/amd64)

API: http://172.31.47.31:9000  http://127.0.0.1:9000
   RootUser: anjali
   RootPass: minioadminanjali123

WebUI: http://172.31.47.31:9001 http://127.0.0.1:9001
   RootUser: anjali
   RootPass: minioadminanjali123

CLI: https://min.io/docs/minio/linux/reference/minio-mc.html#quickstart
   $ mc alias set 'myminio' 'http://172.31.47.31:9000' 'anjali' 'minioadminanjal
i123'

Docs: https://docs.min.io
---------------------------
```

anjaliwagaskar — ubuntu@ip-172-31-45-229: ~ — -zsh — 80×24
~ — ubuntu@ip-172-31-45-229: ~ — -zsh

```
INFO: Waiting for all MinIO sub-systems to be initialize...
INFO: Configured max API requests per node based on available memory: 74
INFO: All MinIO sub-systems initialized successfully in 13.542998ms
INFO: IAM load(startup) finished. (duration: 2.448532ms)
---------------------------
MinIO Object Storage Server
Copyright: 2015-2025 MinIO, Inc.
License: GNU AGPLv3 - https://www.gnu.org/licenses/agpl-3.0.html
Version: RELEASE.2025-07-23T15-54-02Z (go1.24.5 linux/amd64)

API: http://172.31.45.229:9000  http://127.0.0.1:9000
   RootUser: anjali
   RootPass: minioadminanjali123

WebUI: http://172.31.45.229:9001 http://127.0.0.1:9001
   RootUser: anjali
   RootPass: minioadminanjali123

CLI: https://min.io/docs/minio/linux/reference/minio-mc.html#quickstart
   $ mc alias set 'myminio' 'http://172.31.45.229:9000' 'anjali' 'minioadminanja
li123'

Docs: https://docs.min.io
---------------------------
```

Figure 8: distributed setup of minio



```
ubuntu@ip-172-31-32-197:~$ mc --version
mc version RELEASE.2025-07-21T05-28-08Z (commit-id=ee72571936f15b0e65dc8b4a231a4dd445e
5ccb6)
Runtime: go1.24.5 linux/amd64
Copyright (c) 2015-2025 MinIO, Inc.
License GNU AGPLv3 <https://www.gnu.org/licenses/agpl-3.0.html>
ubuntu@ip-172-31-32-197:~$ mc alias set myminio http://172.31.47.31:9000 anjali minioa
dminanjali123
Added `myminio` successfully.
ubuntu@ip-172-31-32-197:~$ mc mb myminio/mybucket
Bucket created successfully `myminio/mybucket`.
ubuntu@ip-172-31-32-197:~$ mc ls myminio
[2025-08-04 22:09:09 UTC]     0B mybucket/
ubuntu@ip-172-31-32-197:~$ 
```

Figure 9: checking if the files are stored and buckets are created

6 FastAPI Application Setup on EC2

create a cloud9 environment Instance type: t2.micro (1 GiB RAM + 1 vCPU)

```
python3 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
uvicorn app:app --host 0.0.0.0 --reload
```

References

Falco Security Project (2025). Try falco on linux - quickstart guide. Accessed: 2025-08-10.
URL: *https://falco.org/docs/getting-started/falco-linux-quickstart/install-falco-1*

HashiCorp (2025). Install vault. Accessed: 2025-08-10.
URL: *https://developer.hashicorp.com/vault/install*

Hernandez, C. C. (2022). How to install minio in distributed mode on aws ec2. Accessed: 2025-08-11.
URL: *https://blog.min.io/install-minio-distributed-mode-aws-ec2/*

Lee, S. (2022). *Chap 2. OPA Installation and Usage*, OPA Guidebook. Accessed: 2025-08-11.
URL: *https://sangkeon.github.io/opaguide/chap2/installandusage.html*