

AppSentinels API Security Platform

Backup & restore

Contents

1. Introduction.....	4
2. Method 1: Platform directory restore to New Instance:.....	4
3. Method 2: Snapshot and Restore to New Instance	9

Revision	Date Modified	Author	Comments
1.0	28-Oct-24		Manjunatha

1. Introduction

The document outlines two main methods:

1. Platform directory restore to New Instance:
 - Takes a backup of the as_platform directory
 - Involves creating a new user with sudo access (optional)
 - Requires checking and configuring proxy settings
 - Installation of Docker and docker-compose
 - Pulling AppSentinels images using a login token
 - Setting up necessary user groups and permissions
 - Configuring the hostname and IP addresses
 - Bringing up AppSentinels services using docker-compose
2. Snapshot and Restore to New Instance (AWS EC2 focused):
 - Creating a volume snapshot from the existing instance
 - Creating a new volume from the snapshot
 - Setting up a new EC2 instance
 - Detaching the original volume and attaching the snapshot-created volume
 - Updating IP configurations in docker-compose
 - Starting the services on the new instance

Please note that

- Both the above methods preserve existing data and configurations
- Users can log in with the same credentials after restoration
- Requires careful attention to IP address updates in configurations
- Involves Docker container management
- Includes specific steps for different operating systems (Ubuntu vs RHEL)

2. Method 1: Platform directory restore to New Instance:

1. On Running Platform, please take a back up the as_platform directory as a

On a New Instance follow the below steps

2. Create user with sudo access (OPTIONAL, recommended if allowed to create separate user or use existing user with sudo privileges)

Ignore this step if the user is already created.

- sudo useradd -m -s /bin/bash appsentinels
- sudo passwd appsentinels
- sudo usermod -aG sudo appsentinels (if Ubuntu)

`sudo usermod -aG wheel appsentinels` (if REDHAT)

- `su - appsentinels` (change to appsentinels user)

```
[ec2-user@ip-10-101-3-250 ~]$ sudo useradd -m -s /bin/bash appsentinels
[ec2-user@ip-10-101-3-250 ~]$ sudo passwd appsentinels
Changing password for user appsentinels.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-10-101-3-250 ~]$ sudo usermod -aG wheel appsentinels
[ec2-user@ip-10-101-3-250 ~]$ su - appsentinels
Password:
[appsentinels@ip-10-101-3-250 ~]$
```

3. Check the proxy settings

- Check the proxy configuration
- if proxy enabled, it should be disabled or excluded for localhost & HOST IP
- Docker environment should not be using proxy

4. Download the AppSentinels Platform installation files to home directory of appsentinels

`wget appsentinels-platform-installation-XX.XX.RX.tar.gz`

5. Install docker & docker-compose

- `cd pre-req`
- `bash docker-install.sh`
- logout & login. Change user back to appsentinels

6. Pull appsentinels images

- `cd pre-req`
- Please run below command to pull the docker images. Please Ask Appsentinels Team for the logintoken

`bash appsentinelsimages.sh <logintoken>`

```
[appsentinels@ip-10-101-3-250 ~]$ cd installation/
[appsentinels@ip-10-101-3-250 installation]$ ls
certs check_list.txt config_files deployment_src install_saas pre-req readme.txt utils
[appsentinels@ip-10-101-3-250 installation]$ cd pre-req/
[appsentinels@ip-10-101-3-250 pre-req]$ ls
appsentinelsimages.sh docker-install-rhel.sh docker-install-ubuntu.sh other-utils-install-rhel.sh
configure_docker_to_use_proxy.txt docker-install.sh docker-rpm-download.txt other-utils-install-ubuntu.sh
[appsentinels@ip-10-101-3-250 pre-req]$ ls -lrt
total 36
-rw-rw-r-- 1 appsentinels appsentinels 170 Oct 25 11:04 other-utils-install-ubuntu.sh
-rw-rw-r-- 1 appsentinels appsentinels 151 Oct 25 11:04 other-utils-install-rhel.sh
-rw-rw-r-- 1 appsentinels appsentinels 1301 Oct 25 11:04 docker-rpm-download.txt
-rw-rw-r-- 1 appsentinels appsentinels 2192 Oct 25 11:04 docker-install-ubuntu.sh
-rw-rw-r-- 1 appsentinels appsentinels 5164 Oct 25 11:04 docker-install.sh
-rw-rw-r-- 1 appsentinels appsentinels 708 Oct 25 11:04 docker-install-rhel.sh
-rw-rw-r-- 1 appsentinels appsentinels 631 Oct 25 11:04 configure_docker_to_use_proxy.txt
-rw-rw-r-- 1 appsentinels appsentinels 2444 Oct 25 11:04 appsentinelsimages.sh
[appsentinels@ip-10-101-3-250 pre-req]$ bash docker-install.sh
Docker not found. Installing Docker...

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for appsentinels:

[appsentinels@ip-10-101-3-250 installation]$ cd pre-req/
[appsentinels@ip-10-101-3-250 pre-req]$ bash appsentinelsimages.sh
```

7. if RHEL or AL2 then run below commands

```
sudo groupadd docker-sentinels
```

```
sudo groupadd data-sentinels
```

```
sudo groupadd data-analysis
```

```
sudo usermod -aG docker-sentinels $USER
```

```
sudo usermod -aG data-sentinels $USER
```

```
sudo usermod -aG data-analysis $USER
```

```
sudo usermod -aG adm $USER
```

Other OS run below commands

```
sudo addgroup docker-sentinels
```

```
sudo addgroup data-sentinels
```

```
sudo addgroup data-analysis
```

```
sudo adduser $USER docker-sentinels
```

```
sudo adduser $USER data-sentinels
```

```
sudo adduser $USER data-analysis
```

```
sudo adduser $USER adm
```

```
[appsentinels@ip-10-101-3-250 ~]$ sudo groupadd docker-sentinels
[sudo] password for appsentinels:
[appsentinels@ip-10-101-3-250 ~]$ sudo groupadd data-sentinels
[appsentinels@ip-10-101-3-250 ~]$ sudo groupadd data-analysis
[appsentinels@ip-10-101-3-250 ~]$ sudo usermod -aG docker-sentinels $USER
[appsentinels@ip-10-101-3-250 ~]$ sudo usermod -aG data-sentinels $USER
[appsentinels@ip-10-101-3-250 ~]$ sudo usermod -aG data-analysis $USER
[appsentinels@ip-10-101-3-250 ~]$ sudo usermod -aG adm $USER
[appsentinels@ip-10-101-3-250 ~]$
```

8. Logout & login

9. untar the as_platform back up taken from previous instance and move it to BASE_DIR path

```
[appsentinels@ip-10-101-3-250 ~]$ sudo tar -xvf as_platform.tar.gz
```

10. Navigate to /as_platform/deployment folder

11. The hostname should be DNS resolvable, if not add entry to /etc/hosts in the server

12. Edit the docker-compose.yaml file and change the IP which is referenced under extra_hosts in entire file to new IP

```
environment:
  - 'DATABASE_URI=postgresql://kong:kong@postgres:5432/policy'
  - policy_component_url=https://al2.appsentinels.ai/
expose:
  - 5802
restart: on-failure
networks:
  ingestion:
    aliases:
      - scheduler
extra_hosts:
  - "al2.appsentinels.ai:10.101.3.250"
secrets:
  - devops_connection
logging:
  driver: syslog
  options:
    tag: docker-scheduler
userinterface:
  image: appsentinelsai/appsentinels-ui:24.09.R1
  expose:
    - 80
  restart: on-failure
  networks:
    ingestion:
      aliases:
        - userinterface
  environment:
    - AUTH_KEY=keycloak
    - DOMAIN=https://al2.appsentinels.ai/auth
    - CLIENTID=frontend
    - REALM=Appsentinels
    - SERVERURL=https://al2.appsentinels.ai
  extra_hosts:
    - "al2.appsentinels.ai:10.101.3.54"
deploy:
  restart_policy:
    condition: on-failure
  resources:
    limits:
```

13. if sensor is deployed in the same host, then change the REMOTE_CONTROLLER_SERVER_NAME to new IP, otherwise skip this step

```
# docker-compose -f docker-compose-sniffer-sensor.yaml pull && docker-co
#
#installing docker-compose sudo curl -L "https://github.com/docker/compos
er-compose
#https://docs.docker.com/compose/environment-variables/#substitute-enviro

version: "3.3"
services:
  sniffer:
    container_name: sniffer-sensor
    restart: on-failure:5
    image: appsentinels/ng-controller:latest
    hostname: appsentinels-sniffer-sensor
    environment:
      - REMOTE_CONTROLLER_SERVER_NAME=10.101.3.250
      - REMOTE_CONTROLLER_SERVER_PORT=9006
      - ENVIRONMENT=scale
      - TAP_INTERFACE=default
      - TAP_FILTER=port-8000-and-tcp
      - RELAY_PROTOCOL=http
    network_mode: "host"
    logging:
      driver: local
      options:
        max-size: 10m
    volumes:
      - /var/crash:/var/crash
      - /var/log/appsentinels:/var/log/appsentinels
  http_service:
```

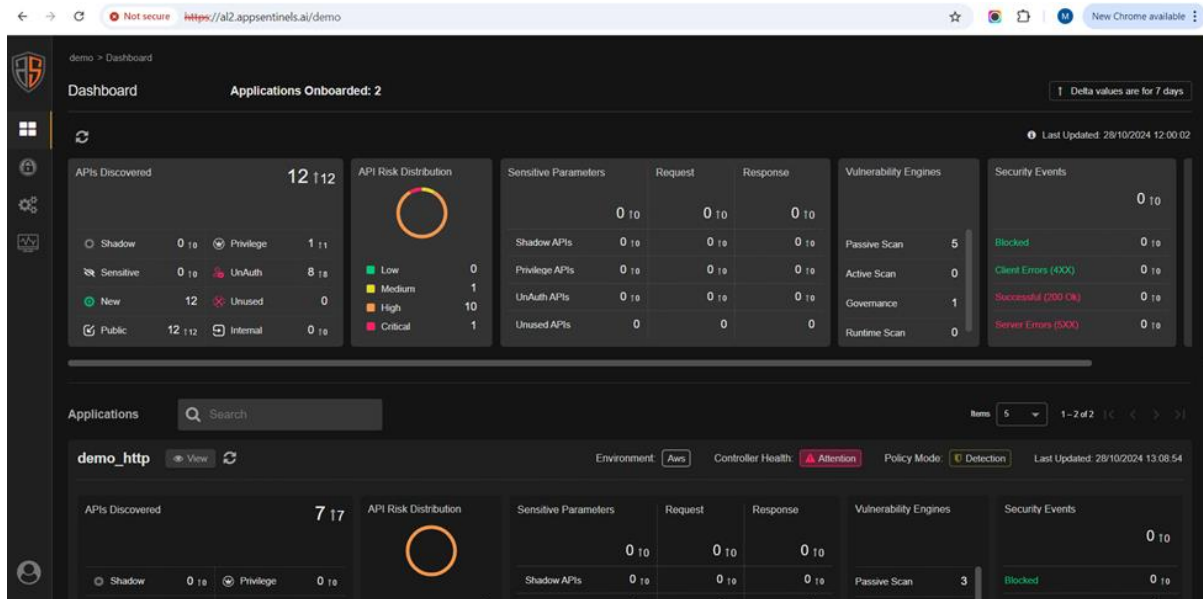
14. bring up the appsentinels services by running below command

`docker-compose up -d`

15. bring up the sensor if is deployed in the same instance otherwise skip this step

`docker-compose -f <filename> up -d`

16. Login to Appsentinels Platform with same credentials



3. Method 2: Snapshot and Restore to New Instance

1. Select the Instance, Click on Storage, Click on the Volume

Instances (1/4) Info Last updated less than a minute ago Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive) All states

Name	Instance ID	Instance state	Instance type	Status check	Ala
AL2-ARM-Build-Sachin	i-02f6270dab492af39	Stopped	t4g.micro	-	Vie
Manju-Lambda-controller	i-0924441bd9d91f8ca	Stopped	t2.medium	-	Vie
☒ DND-AL2-SaaS-Platform	i-068f529533972b5ac	Running	t3.xlarge	3/3 checks passec	Vie
☐	i-0aaaeefc57399f423	Stopped	t3.medium	-	Vie

i-068f529533972b5ac (DND-AL2-SaaS-Platform)

Filter block devices

Volume ID	Device name	Volume size (GiB)	Attachment status	Attachment time	Encrypte
☒ vol-0e1e2e14ed026187c	/dev/xvda	50	Attached	2024/10/25 16:26 GMT+5:30	No

Volume monitoring (1)

Alarm recommendations 2h 1d 1w 1h UTC timezone Add to dashboard

2. Volume is displayed

Volumes (1) Info Actions Create volume

Search Volume ID = vol-0e1e2e14ed026187c Clear filters

Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID	Created
-	vol-0e1e2e14ed026187c	gp3	50 GiB	3000	125	snap-062546a...	2024/10/2

Fault tolerance for all volumes in this Region

Snapshot summary Last updated on Fri, Oct 25, 2024, 06:50:33 PM (GMT+05:30)

Recently backed up volumes / Total # volumes

0 / 10

Data Lifecycle Manager default policy for EBS Snapshots status
No default policy set up | [Create policy](#)

3. Select the Volume & Select Create Snapshot in Actions Dropdown

Volumes (1/1) [Info](#)

Search

Volume ID = vol-0e1e2e14ed026187c

☒	Name	Volume ID	Type	Size	IOPS	Throughput
☒	-	vol-0e1e2e14ed026187c	gp3	50 GiB	3000	125

Volume ID: vol-0e1e2e14ed026187c

[Details](#) | [Status checks](#) | [Monitoring](#) | [Tags](#)

Volume ID vol-0e1e2e14ed026187c	Size 50 GiB	Type gp3	Volume status Okay
AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more	Volume state In-use	IOPS 3000	Throughput 125
Fast snapshot restored No	Availability Zone ap-south-1a	Created Fri Oct 25 2024 16:26:39	Multi-Attach enabled No

Actions

- Modify volume
- Create snapshot
- Create snapshot lifecycle policy
- Delete volume
- Attach volume
- Detach volume
- Force detach volume
- Manage auto-enabled I/O
- Manage tags
- Fault injection

4. Provide the Details & click on Create Snapshot

Source volume

Volume ID: vol-0e1e2e14ed026187c

Availability Zone: ap-south-1a

Snapshot details

Description
 Add a description for your snapshot:

 255 characters maximum.

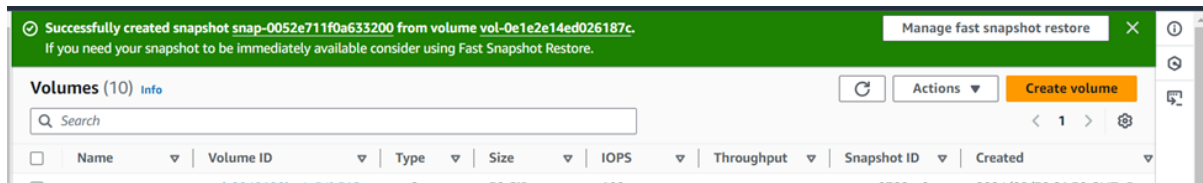
Encryption [Info](#)
 Not encrypted

Tags [Info](#)
 A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

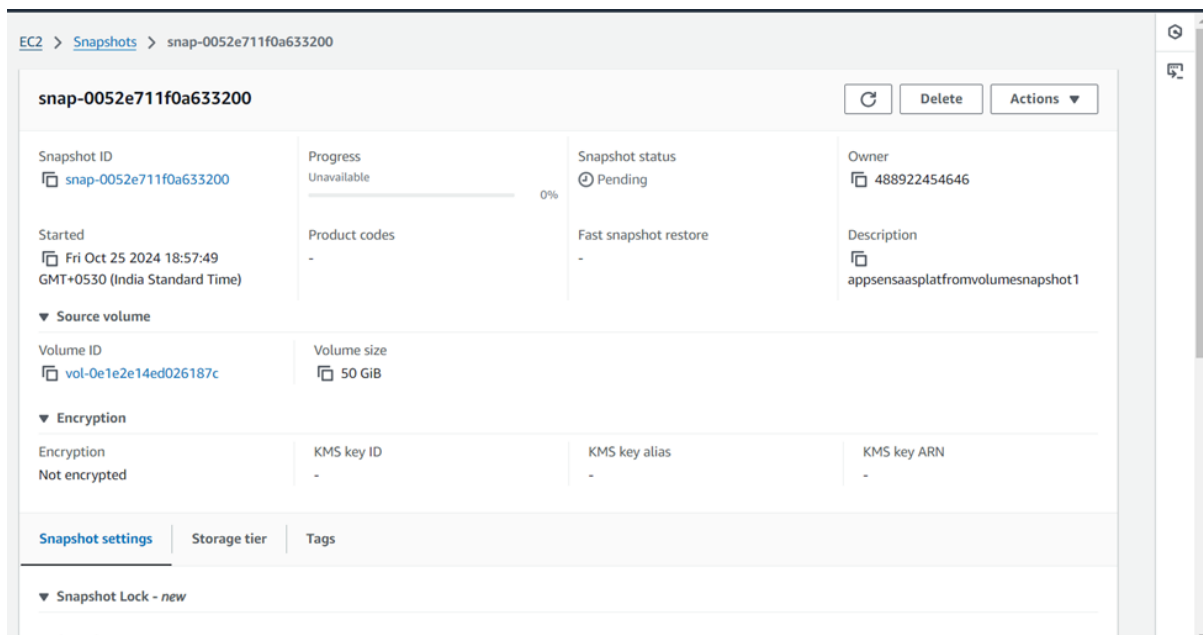
No tags associated with the resource.

You can add 50 more tags.

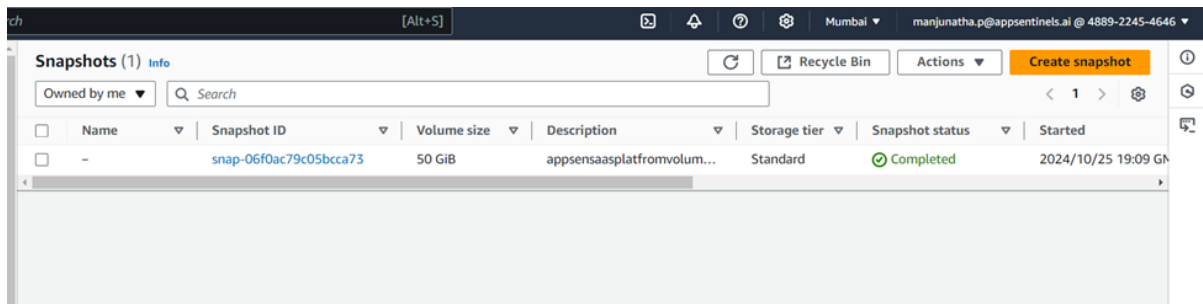
5. Snapshot will be created and click on the snapshot click



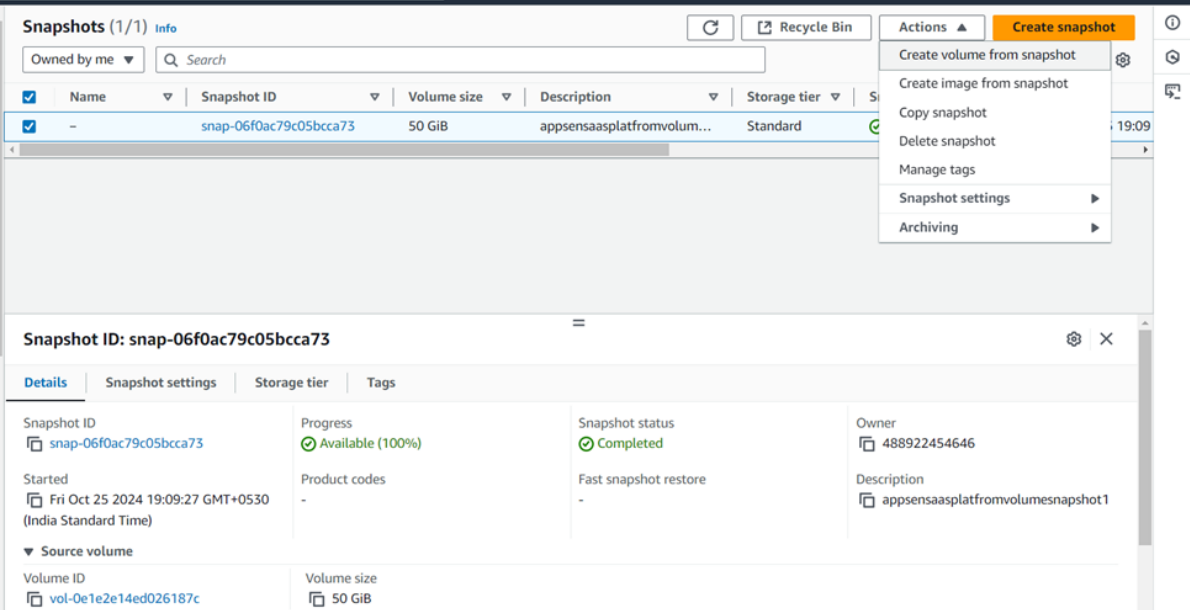
6. It will take some time to create snapshot



7. Once Snapshot is ready it will show status as Completed



8. Select the snpshot & Click on Create Volume from snapshot in Actions dropdown



Snapshots (1/1) Info

Owned by me Search

Name	Snapshot ID	Volume size	Description	Storage tier	Status	Created
-	snap-06f0ac79c05bcca73	50 GiB	appsensaasplatformfromvolum...	Standard	Completed	19:09

Actions Create snapshot

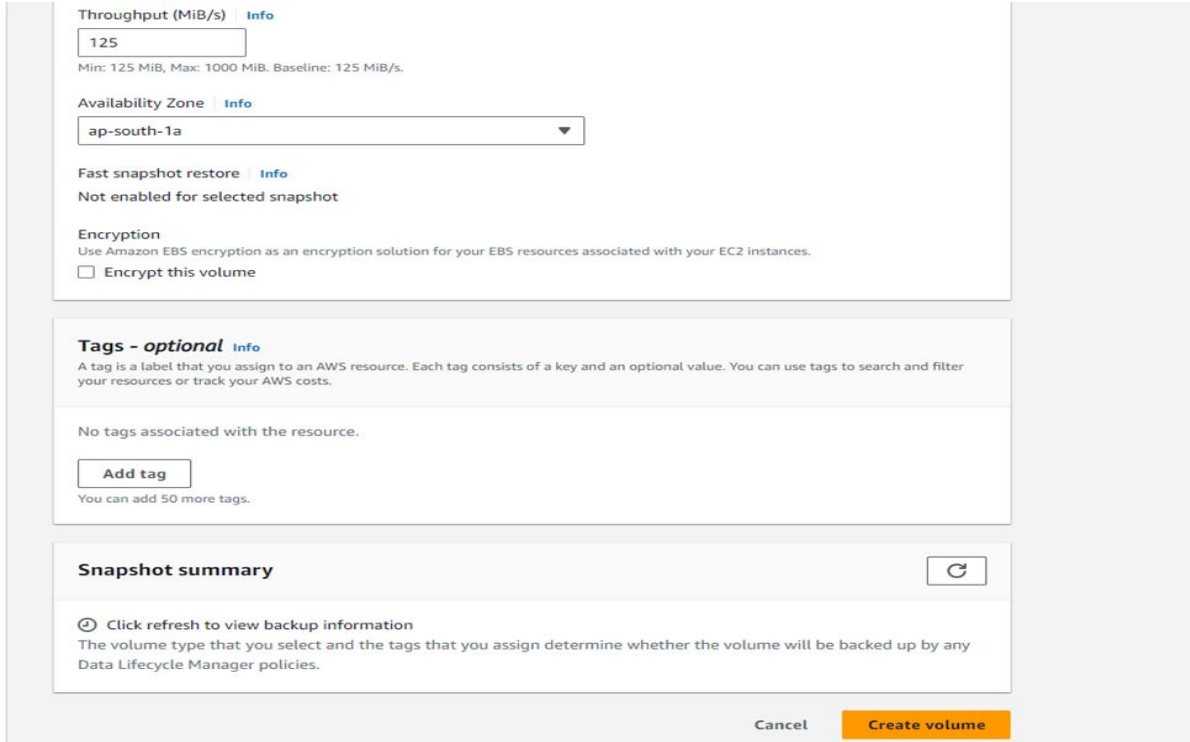
- Create volume from snapshot
- Create image from snapshot
- Copy snapshot
- Delete snapshot
- Manage tags
- Snapshot settings
- Archiving

Snapshot ID: snap-06f0ac79c05bcca73

Details Snapshot settings Storage tier Tags

Snapshot ID snap-06f0ac79c05bcca73	Progress Available (100%)	Snapshot status Completed	Owner 488922454646
Started Fri Oct 25 2024 19:09:27 GMT+0530 (India Standard Time)	Product codes -	Fast snapshot restore -	Description appsensaasplatformfromvolumesnapshot1
Source volume			
Volume ID vol-0e1e2e14ed026187c	Volume size 50 GiB		

9. Provide the Details & click on Create Volume



Throughput (MiB/s) Info

125

Min: 125 MiB, Max: 1000 MiB, Baseline: 125 MiB/s.

Availability Zone Info

ap-south-1a

Fast snapshot restore Info

Not enabled for selected snapshot

Encryption

Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.

☐ Encrypt this volume

Tags - optional Info

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

Add tag

You can add 50 more tags.

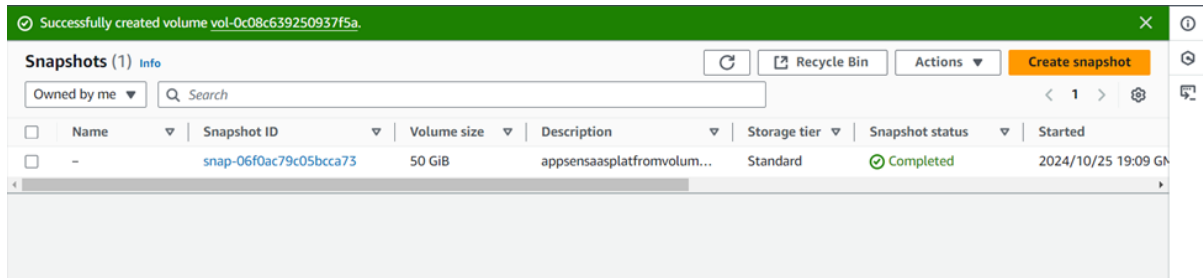
Snapshot summary

Click refresh to view backup information

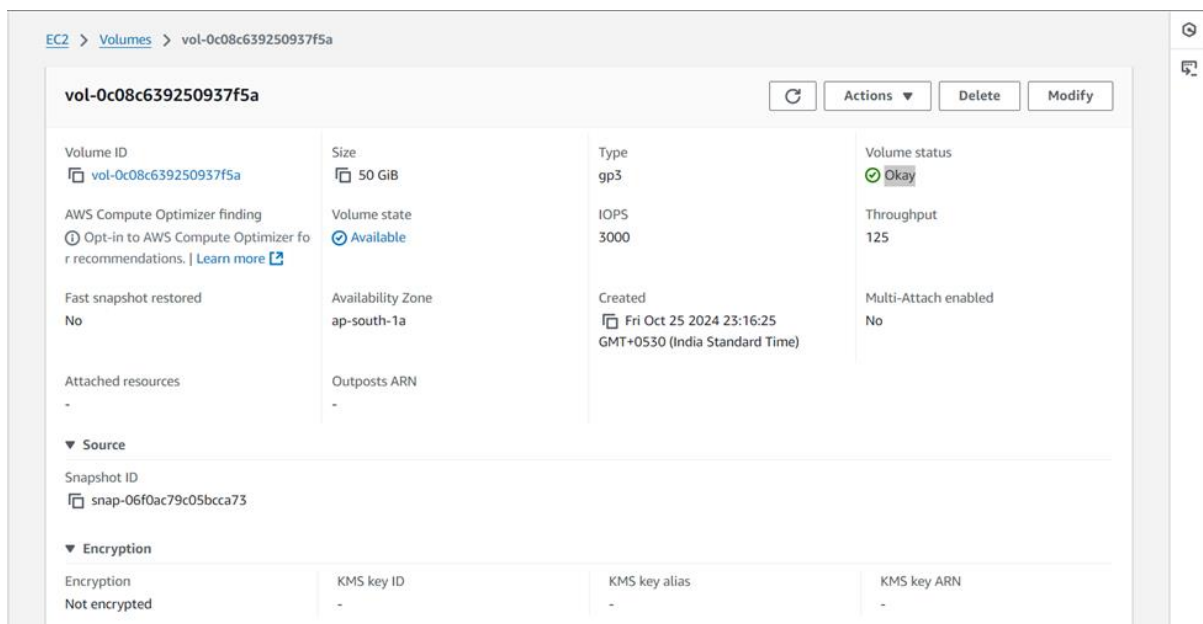
The volume type that you select and the tags that you assign determine whether the volume will be backed up by any Data Lifecycle Manager policies.

Cancel **Create volume**

10. It will show the create volume and click on it

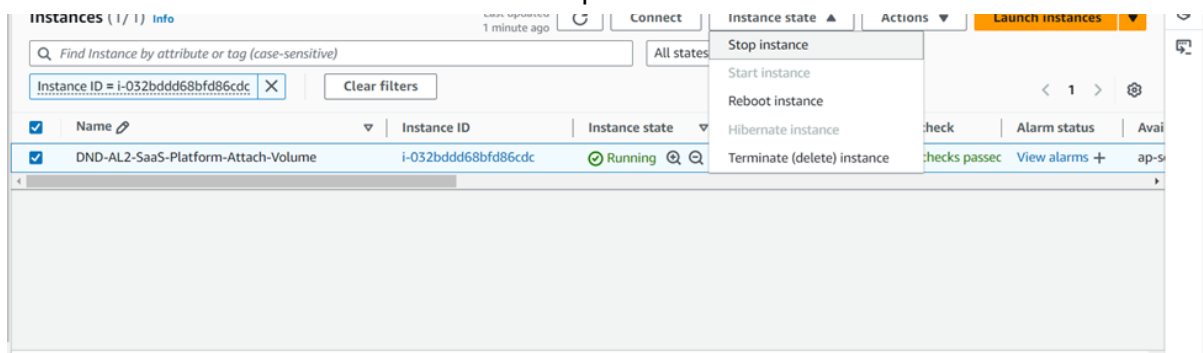


11. Wait for Volume status to be Okay

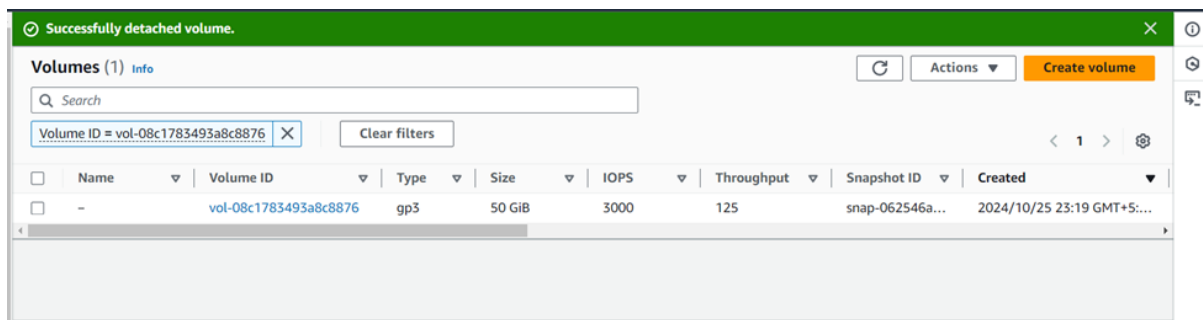


12. Stop the Previous running SaaS Platform Instance

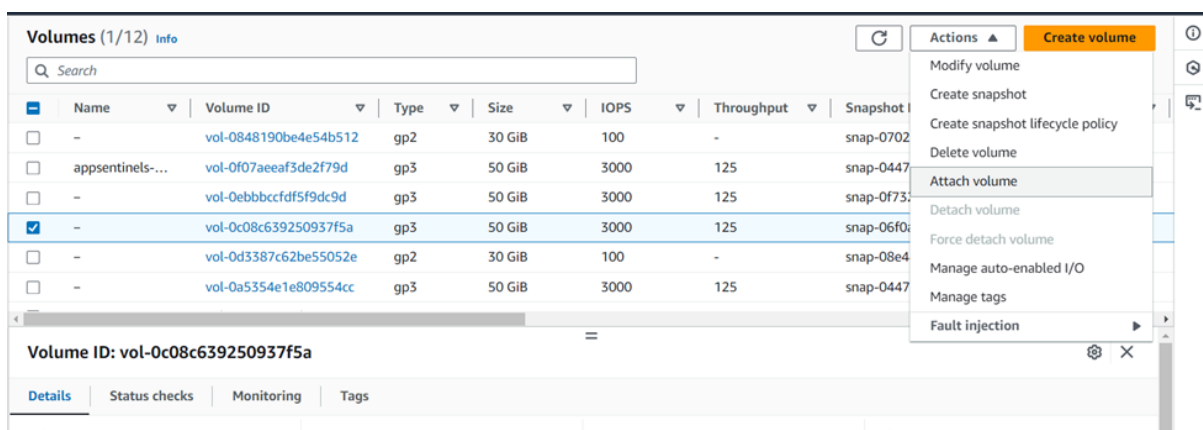
13. Create the New Ec2 Instance & Stop the instance



14. Go to the Volume of the newly created ec2 instance and Detach the Volume



15. Not select the Volume created from the snapshot and click on Attach Volume in Actions Dropdown



16. Provide the ec2 instance ID & Device Name and click on Attach Volume

Attach a volume to an instance to use it as you would a regular physical hard disk drive.

Basic details

Volume ID

 vol-0c08c639250937f5a

Availability Zone

ap-south-1a

Instance [Info](#)

i-032bddd68bfd86cdc



Only instances in the same Availability Zone as the selected volume are displayed.

Device name [Info](#)

/dev/xvda

Recommended device names for Linux: /dev/xvda for root volume. /dev/sd[f-p] for data volumes.

 Newer Linux kernels may rename your devices to **/dev/xvdf** through **/dev/xvdp** internally, even when the device name entered here (and shown in the details) is **/dev/sdf** through **/dev/sdp**.

Cancel

Attach volume

17. Start the ec2 instance

less than a minute ago

Find Instance by attribute or tag (case-sensitive)

All states

< 1 >

☒	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Avai
☐	AL2-ARM-Build-Sachin	i-02f6270dab492af39	Stopped	t4g.micro	-	View alarms	ap-s
☐	Manju-Lambda-controller	i-0924441bd9d91f8ca	Stopped	t2.medium	-	View alarms	ap-s
☐	DND-AL2-SaaS-Platform	i-068f529533972b5ac	Stopped	t3.xlarge	-	View alarms	ap-s
☒	DND-AL2-SaaS-Platform-Attach-Volume	i-032bddd68bfd86cdc	Running	t3.xlarge	Initializing	View alarms	ap-s
☐		i-0aaaaefc57399f423	Stopped	t3.medium	-	View alarms	ap-s

18. Login to ec2 instance & switch to the user created previously (when platform installation done)


```
PS C:\Users\manju\Downloads> ssh -i .\appsen-manju.pem ec2-user@13.235.242.0
The authenticity of host '13.235.242.0 (13.235.242.0)' can't be established.
ED25519 key fingerprint is SHA256:28yRYY+wSkGzE1c5c/yz3LPKhKHxhvbwx3IDfm4Mc9A.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.235.242.0' (ED25519) to the list of known hosts.
Last login: Fri Oct 25 12:56:50 2024 from 223.185.133.167

      #_
    _/\_ #####_   Amazon Linux 2
   ~~~ \_#####\_
   ~~~  \_#####\_
   ~~~   \###|
   ~~~    \#/
   ~~~     V~'  '-->
       ~~~~~
       ~.._./
       _/_/_/_/_/_/
       _/m/'

A newer version of Amazon Linux is available!

Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-101-3-54 ~]$ su - appsentinels
Password:
Last login: Fri Oct 25 12:57:09 UTC 2024 on pts/1
[appsentinels@ip-10-101-3-54 ~]$
```

19. Navigate to deployment folder `cd /as_platform/deployment/`

20. Start the docker service `sudo systemctl start docker`

```
[appsentinels@ip-10-101-3-54 ~]$ cd /as_platform/
[appsentinels@ip-10-101-3-54 as_platform]$ sudo systemctl start docker
[appsentinels@ip-10-101-3-54 as_platform]$ ls
data data-analysis deployment log
[appsentinels@ip-10-101-3-54 as_platform]$ cd deployment/
[appsentinels@ip-10-101-3-54 deployment]$
```

21. Edit the docker-compose

```
[appsentinels@ip-10-101-3-54 deployment]$ sudo vi docker-compose.yaml
```

22. Change the IP of the previous systemIP to the new one under extra_hosts in the entire file

Ex: Previous IP was 10.101.3.222 and new IP : 10.101.2.54

```

    condition: on-failure
  resources:
    limits:
      cpus: '0.20'
      memory: 256M
  logging:
    driver: syslog
    options:
      tag: docker-hygiene
dast-controller:
  image: appsentinelsai/dast-controller:24.09.R1
  expose:
    - 3399
  restart: on-failure
  networks:
    ingestion:
      aliases:
        - dast-controller
  extra_hosts:
    - "al2.appsentinels.ai:10.101.3.54"
  deploy:
    restart_policy:
      condition: on-failure
  resources:
    limits:
      cpus: '1.00'
      memory: 4096M
  environment:
    - LOG_LEVEL=INFO
    - policy_api_timeout=60
  volumes:
    - '/as_platform/data/api_logs:/data'
    - '/as_platform/data/reports:/reports/'
  secrets:
    - devops_connection
  logging:
    driver: syslog
    options:
      tag: docker-dast-controller
  learning-controller:
"docker-compose.yaml" 1366L, 33299B

```

23. Bring up the docker services `sudo docker-compose up -d`

```

[appsentinels@ip-10-101-3-54 deployment]$ sudo docker-compose up -d
WARN[0001] /as_platform/deployment/docker-compose.yaml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
WARN[0001] Found orphan containers ([sniffer-sensor http_service]) for this project. If you removed or renamed this service in your compose file, you can run this command with the --remove-orphans flag to clean it up.
[+] Running 34/0
  ✓ Container deployment-restapi-1 Running 0.0s
  ✓ Container deployment-postgres-exporter-pgserver94-1 Running 0.0s
  ✓ Container dast_config Recreate 0.1s
  [+] Running 34/43oyment-piiserver-1 Running 0.0s
  ✓ Container deployment-restapi-1 Running 0.0s
  ✓ Container deployment-postgres-exporter-pgserver94-1 Running 0.0s
  ✓ Container dast_config Recreate 0.2s
  [+] Running 34/43oyment-piiserver-1 Running 0.0s
  ✓ Container deployment-restapi-1 Running 0.0s
  ✓ Container deployment-postgres-exporter-pgserver94-1 Running 0.0s

```

24. Now access the Dashboard and login with same credentials. All the Data is restored and available in the new ec2 instance

