1.2. Create an Amazon S3 API user called `S3 Operator` with the UID of `operator`. Assign an access key of `12345` and a secret of `67890`, and grant the user full access.

```
[admin@serverc ~]$ sudo cephadm shell -- radosgw-admin user create \
  --uid="operator" --access="f ull" --display-name="S3 Operator" \
  --access_key="12345" --secret="67890"
...output omitted...
```

1.3. Create a Swift subuser called `operator:swift`. Set `opswift` as the subuser secret and grant full access.

```
[admin@serverc ~]$ sudo cephadm shell -- radosgw-admin subuser create \
  --uid="operator" --subuse r="operator:swift" --access="full" --secret="opswift"
...output omitted...
```

2. Configure the AWS CLI tool to use the `operator` user credentials. Create a bucket called `log-artifacts`. The RADOS Gateway service is running on the default port on the `serverc` node.

2.1. Configure the AWS CLI tool to use operator credentials. Enter `12345` as the access key and `67890` as the secret key.

```
[admin@serverc ~]$ aws configure --profile=ceph
AWS Access Key ID [None]: 12345
AWS Secret Access Key [None]: 67890
Default region name [None]: Enter
Default output format [None]: Enter
```

2.2. Create a bucket called `log-artifacts`.

```
[admin@serverc ~]$ aws --profile=ceph --endpoint=http://serverc:80 s3 mb \
  s3://log-artifacts
make_bucket: log-artifacts
```

2.3. Verify that the AWS bucket exists.

```
[admin@serverc ~]$ aws --profile=ceph --endpoint=http://serverc:80 s3 ls
2021-11-03 06:00:39 log-artifacts
```

3. Create a container called `backup-artifacts`. The RADOS Gateway service is on the default port on the `serverc` node.

3.1. Create a Swift container called `backup-artifacts`.

```
[admin@serverc ~]$ swift -V 1.0 -A http://serverc:80/auth/v1 -U operator:swift \
  -K opswift post backup-artifacts
```

3.2. Verify that the container exists.