

## Supporting Static Web Hosting with RADOS Gateway

RADOS Gateway supports static website hosting in S3 buckets, which can be more efficient than using virtual machines for website hosting. This is suitable for websites that only use static elements such as XHTML or HTML files, or CSS.

Deploying RADOS Gateway instances for static web hosting has restrictions.

- Instances cannot also be used for S3 or Swift API access.
- Instances should have domain names that are different from and do not overlap those of the standard S3 and Swift API gateway instances.
- Instances should use public-facing IP addresses that are different from the standard S3 and Swift API gateway instances.

## RADOS Gateway Deployment

The `cephadm` tool deploys RADOS Gateway services as a collection of daemons that manage a single cluster or a multisite deployment. Use the `client.rgw.*` section in the centralized configuration database to define parameters and characteristics for new RADOS Gateway daemons.

Use the Ceph Orchestrator to deploy or remove RADOS Gateway services. Use the Ceph Orchestrator with either the command-line interface or a service specification file.

```
[ceph: root@node /]# ceph orch apply rgw <service-name> [--realm=<realm>] \
[--zone=<zone>] [--port=< port>] --placement="<num-daemons> [<host1> ...]" \
[--unmanaged]
```

In this example, the Ceph Orchestrator deploys the `my_rgw_service` RADOS Gateway service with two daemons in a single cluster, and presents the service on port 80.

```
[ceph: root@node /]# ceph orch apply rgw my_rgw_service
```

If no configuration is defined in the `client.rgw.*` section or passed to the Ceph orchestrator at build time, then the deployment uses these default settings.

The following example YAML file contains common parameters defined for a RADOS Gateway deployment.

```
service_type: rgw
service_name: rgw_service_name
placement:
  count: 2
  hosts:
    - node01
    - node02
spec:
  rgw_frontend_port: 8080
  rgw_realm: realm_name
  rgw_zone: zone_name
  ssl: true
  rgw_frontend_ssl_certificate: |
    -----BEGIN PRIVATE KEY-----
```