OBJECT STORAGE QUICK START

To use this guide, you must have executed the procedures in the Ceph Deploy Quick Start guide first. Ensure your Ceph Storage Cluster is in an `active + clean` state before working with the Ceph Object Storage.

Note: Ceph Object Storage is also referred to as RADOS Gateway.

INSTALL APACHE AND FASTCGI

Ceph Object Storage runs on Apache and FastCGI in conjunction with the Ceph Storage Cluster. Install Apache and FastCGI on the server node. Use the following procedure:

1. Install Apache and FastCGI on the server machine.

```
sudo apt-get update && sudo apt-get install apache2 libapache2-mod-fastcgi
```

2. Enable the URL rewrite modules for Apache and FastCGI.

```
sudo a2enmod rewrite
sudo a2enmod fastcgi
```

3. Add a line for the `ServerName` in the Apache configuration file (e.g., `/etc/apache2/httpd.conf` or `/etc/apache2/apache2.conf`). Provide the fully qualified domain name of the server machine (e.g., `hostname -f`).

```
ServerName {fqdn}
```

4. Restart Apache so that the foregoing changes take effect.

```
sudo service apache2 restart
```

INSTALL CEPH OBJECT STORAGE

Once you have installed and configured Apache and FastCGI, you may install the Ceph Object Storage daemon (`radosgw`).

```
sudo apt-get install radosgw
```

For details on the preceding steps, see Ceph Object Storage Manual Install.

CREATE A DATA DIRECTORY

Create a data directory on the server node for the instance of `radosgw`.

```
sudo mkdir -p /var/lib/ceph/radosgw/ceph-radosgw.gateway
```

MODIFY THE CEPH CONFIGURATION FILE

On the admin node, perform the following steps:

1. Open the Ceph configuration file.

```
vim ceph.conf
```

2. Add the following settings to the Ceph configuration file:

```
[client.radosgw.gateway]
host = {host-name}
keyring = /etc/ceph/keyring.radosgw.gateway
rgw socket path = /tmp/radosgw.sock
log file = /var/log/ceph/radosgw.log

#Add DNS hostname to enable S3 subdomain calls
rgw dns name = {hostname}
```

3. Use `ceph-deploy` to push a copy the configuration file from the admin node to the server node.

```
ceph-deploy --overwrite-conf config push {hostname}
```

CREATE A GATEWAY CONFIGURATION FILE

The example configuration file will configure the gateway on the server node to operate with the Apache FastCGI module, a rewrite rule for OpenStack Swift, and paths for the log files. To add a configuration file for Ceph Object Storage, we suggest copying the contents of the example file below to an editor. Then, follow the steps below to modify it (on your server node).

```
FastCgiExternalServer /var/www/s3gw.fcgi -socket /tmp/radosgw.sock


<VirtualHost *:80>

        ServerName {fqdn}
        <!--Remove the comment. Add a server alias with *.{fqdn} for S3 subdomains-->
        <!--ServerAlias *.{fqdn}-->
        ServerAdmin {email.address}
        DocumentRoot /var/www
        RewriteEngine On
        RewriteRule ^/([a-zA-Z0-9-_.]*)([/]?.*) /s3gw.fcgi?page=$1&params=$2&%{QUERY_STRING}

        <IfModule mod_fastcgi.c>
        <Directory /var/www>
                        Options +ExecCGI
                        AllowOverride All
                        SetHandler fastcgi-script
                        Order allow,deny
                        Allow from all
                        AuthBasicAuthoritative Off
                </Directory>
        </IfModule>

        AllowEncodedSlashes On
        ErrorLog /var/log/apache2/error.log
        CustomLog /var/log/apache2/access.log combined
        ServerSignature Off

</VirtualHost>
```

1. Replace the `{fqdn}` entry with the fully-qualified domain name of the server server.
2. Replace the `{email.address}` entry with the email address for the server administrator.
3. Add a `ServerAlias` if you wish to use S3-style subdomains.
4. Save the contents to the `/etc/apache2/sites-available` directory on the server machine.
5. Enable the site for `rgw.conf`.

```
sudo a2ensite rgw.conf
```

6. Disable the default site.

```
sudo a2dissite default
```

See Create rgw.conf for additional details.

ADD A FASTCGI SCRIPT

FastCGI requires a script for the S3-compatible interface. To create the script, execute the following procedures on the server node.

1. Go to the `/var/www` directory.

```
cd /var/www
```

2. Open an editor with the file name `s3gw.fcgi`.

```
sudo vim s3gw.fcgi
```

3. Copy the following into the editor.

```
#!/bin/sh
exec /usr/bin/radosgw -c /etc/ceph/ceph.conf -n client.radosgw.gateway
```

4. Save the file.
5. Change the permissions on the file so that it is executable.

```
sudo chmod +x s3gw.fcgi
```

GENERATE A KEYRING AND KEY

Perform the following steps on the server machine.

1. Ensure the server node is set up with administrator privileges. From the admin node, execute the following:

```
ceph-deploy admin {hostname}
```

2. Create a keyring for Ceph Object Storage.

```
sudo ceph-authtool --create-keyring /etc/ceph/keyring.radosgw.gateway
sudo chmod +r /etc/ceph/keyring.radosgw.gateway
```

3. Create a key for Ceph Object Storage to authenticate with the Ceph Storage Cluster.

```
sudo ceph-authtool /etc/ceph/keyring.radosgw.gateway -n client.radosgw.gateway --gen-key
sudo ceph-authtool -n client.radosgw.gateway --cap osd 'allow rwx' --cap mon 'allow rw' /
```

4. Add the key to the Ceph keyring.

```
sudo ceph -k /etc/ceph/ceph.client.admin.keyring auth add client.radosgw.gateway -i /etc/
```

ENABLE SSL

Some REST clients use HTTPS by default. So you should consider enabling SSL for Apache on the server machine.

```
sudo a2enmod ssl
```

Once you enable SSL, you should use a trusted SSL certificate. You can generate a non-trusted SSL certificate using the following:

```
sudo mkdir /etc/apache2/ssl
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/apache.key
```

Then, restart Apache.

```
service apache2 restart
```

ADD WILDCARD TO DNS

To use Ceph with S3-style subdomains (e.g., `bucket-name.domain-name.com`), you need to add a wildcard to the DNS record of the DNS server you use with the `radosgw` daemon.

Tip: The address of the DNS must also be specified in the Ceph configuration file with the `rgw dns name = {hostname}` setting.

For `dnsmasq`, consider addding the following `address` setting with a dot (.) prepended to the host name:

```
address=/.{hostname-or-fqdn}/{host-ip-address}
address=/.ceph-node/192.168.0.1
```

For `bind`, consider adding the a wildcard to the DNS record:

```
$TTL    604800
@       IN      SOA     ceph-node. root.ceph-node. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ceph-node.
@       IN      A       192.168.122.113
*       IN      CNAME   @
```

Restart your DNS server and ping your server with a subdomain to ensure that your Ceph Object Store `radosgw` daemon can process the subdomain requests.

```
ping mybucket.{fqdn}
ping mybucket.ceph-node
```

RESTART SERVICES

To ensure that all components have reloaded their configurations, we recommend restarting your ceph and apaches services. Then, start up the `radosgw` service. For example:

```
sudo service ceph restart
sudo service apache2 restart
sudo /etc/init.d/radosgw start
```

CREATE A USER

To use the Gateway, you must create a Gateway user. First, create a gateway user for the S3-compatible interface; then, create a subuser for the Swift-compatible interface.

GATEWAY (S3) USER

First, create a Gateway user for the S3-compatible interface.

```
sudo radosgw-admin user create --uid="{username}" --display-name="{Display Name}"
```

For example:

```
radosgw-admin user create --uid=johndoe --display-name="John Doe" --email=john@example.com
```

```
{ "user_id": "johndoe",
  "rados_uid": 0,
  "display_name": "John Doe",
  "email": "john@example.com",
  "suspended": 0,
  "subusers": [],
  "keys": [
    { "user": "johndoe",
      "access_key": "QFAMEDSJP5DEKJO0DDXY",
      "secret_key": "iaSFLDVvDdQt6lkNzHyW4fPLZugBAI1g17LO0+87"}],
  "swift_keys": []
 }
```

Creating a user creates an access_key and secret_key entry for use with any S3 API-compatible client.

Important: Check the key output. Sometimes radosgw-admin generates a key with an escape () character, and some clients do not know how to handle escape characters. Remedies include removing the escape character (), encapsulating the string in quotes, or simply regenerating the key and ensuring that it does not have an escape character.

SUBUSER

Next, create a subuser for the Swift-compatible interface.

```
sudo radosgw-admin subuser create --uid=johndoe --subuser=johndoe:swift --access=full
```

```
{ "user_id": "johndoe",
  "rados_uid": 0,
  "display_name": "John Doe",
  "email": "john@example.com",
  "suspended": 0,
  "subusers": [
    { "id": "johndoe:swift",
      "permissions": "full-control"}],
  "keys": [
    { "user": "johndoe",
      "access_key": "QFAMEDSJP5DEKJO0DDXY",
      "secret_key": "iaSFLDVvDdQt6lkNzHyW4fPLZugBAI1g17LO0+87"}],
  "swift_keys": []}
```

```
sudo radosgw-admin key create --subuser=johndoe:swift --key-type=swift
```

```
{ "user_id": "johndoe",
  "rados_uid": 0,
  "display_name": "John Doe",
```

```
  "email": "john@example.com",
  "suspended": 0,
  "subusers": [
     { "id": "johndoe:swift",
       "permissions": "full-control"}],
  "keys": [
    { "user": "johndoe",
      "access_key": "QFAMEDSJP5DEKJO0DDXY",
      "secret_key": "iaSFLDVvDdQt6lkNzHyW4fPLZugBAI1g17LO0+87"}],
  "swift_keys": [
    { "user": "johndoe:swift",
      "secret_key": "E9T2rUZNu2gxUjcwUBO8n\/Ev4KX6\/GprEuH4qhu1"}]}
```

This step enables you to use any Swift client to connect to and use RADOS Gateway via the Swift-compatible API.

RGW's `user:subuser` tuple maps to the `tenant:user` tuple expected by Swift.

Note: RGW's Swift authentication service only supports built-in Swift authentication (`-V 1.0`) at this point. See RGW Configuration for Keystone integration details.

SUMMARY

Once you have completed this Quick Start, you may use the Ceph Object Store tutorials. See the S3-compatible and Swift-compatible APIs for details.