This guide is no longer being updated. For current information and instructions, see the new Amazon S3 User Guide (https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html) .

Configuring a static website using a custom domain registered with Route 53

PDF (s3-dg.pdf#website-hosting-custom-domain-walkthrough)
Kindle (https://www.amazon.com/dp/B0763Z4YVV)
RSS (s3-dev-guide-rss-updates.rss)

Suppose that you want to host your static website on Amazon S3. You've registered a domain with Amazon Route 53 (for example, `example.com`), and you want requests for `http://www.example.com` and `http://example.com` to be served from your Amazon S3 content. You can use this walkthrough to learn how to host a static website and create redirects on Amazon S3 for a website with a custom domain name that is registered with Route 53. You can work with an existing website that you want to host on Amazon S3, or you use this walkthrough to start from scratch.

Once you complete this walkthrough, you can optionally use Amazon CloudFront to improve the performance of your website. For more information, see Speeding up your website with Amazon CloudFront (./website-hosting-cloudfront-walkthrough.html) .

Note

Amazon S3 does not support HTTPS access to the website. If you want to use HTTPS, you can use Amazon CloudFront to serve a static website hosted on Amazon S3.

For more information, see How do I use CloudFront to serve a static website hosted on Amazon S3? (http://aws.amazon.com/premiumsupport/knowledge-center/cloudfront-serve-static-website/) and Requiring HTTPS for communication between viewers and CloudFront (https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/using-https-viewers-to-cloudfront.html) .

Automating static website set up with an AWS CloudFormation template

You can use an AWS CloudFormation template to automate your static website setup. The AWS CloudFormation template sets up the components that you need to host a secure static website so that you can focus more on your website's content and less on configuring components.

The AWS CloudFormation template includes the following components:

- Amazon S3-Creates an Amazon S3 bucket to host your static website.
- CloudFront- Creates a CloudFront distribution to speed up your static website.
- Lambda@Edge-Uses Lambda@Edge (https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/lambda-at-the-edge.html) to add security headers to every server response. Security headers are a group of headers in the web server response that tell web browsers to take extra security precautions. For more information, refer to this blog post: Adding HTTP security headers using Lambda@Edge and Amazon CloudFront (http://aws.amazon.com/blogs/networking-and-content-delivery/adding-http-security-headers-using-lambdaedge-and-amazon-cloudfront/) .

This AWS CloudFormation template is available for you to download and use. For information and instructions, see Getting started with a secure static website (https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/getting-started-secure-static-website-cloudformation-template.html) in the *Amazon CloudFront Developer Guide*.

Topics

Before you begin

As you follow the steps in this example, you work with the following services:

Amazon Route 53 – You use Route 53 to register domains and to define where you want to route internet traffic for your domain. The example shows how to create Route 53 alias records that route traffic for your domain (`example.com`) and subdomain (`www.example.com`) to an Amazon S3 bucket that contains an HTML file.

Amazon S3 – You use Amazon S3 to create buckets, upload a sample website page, configure permissions so that everyone can see the content, and then configure the buckets for website hosting.

Step 1: Register a custom domain with Route 53

If you don't already have a registered domain name, such as `example.com`, register one with Route 53. For more information, see Registering a New Domain (https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/domain-register.html) in the *Amazon Route 53 Developer Guide*. After you register your domain name, you can create and configure your Amazon S3 buckets for website hosting.

Step 2: Create two buckets

To support requests from both the root domain and subdomain, you create two buckets.

- **Domain bucket** - `example.com`
- **Subdomain bucket** - `www.example.com`

These bucket names must exactly match your domain name. In this example, the domain name is `example.com`. You host your content out of the root domain bucket (`example.com`). You create a redirect request for the subdomain bucket (`www.example.com`). If someone enters `www.example.com` in their browser, they are redirected to `example.com` and see the content that is hosted in the Amazon S3 bucket with that name.

To create your buckets for website hosting

The following instructions provide an overview of how to create your buckets for website hosting. For detailed, step-by-step instructions on creating a bucket, see How Do I Create an S3 Bucket? (https://docs.aws.amazon.com/AmazonS3/latest/user-guide/create-bucket.html) in the *Amazon Simple Storage Service Console User Guide*.

1. Sign in to the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/s3/ (https://console.aws.amazon.com/s3/) .
2. Create your root domain bucket:
    1. Choose **Create bucket**.
    2. Enter the **Bucket name** (for example, `example.com`).
    3. Choose the Region where you want to create the bucket.

        Choose a Region close to you to minimize latency and costs, or to address regulatory requirements. The Region that you choose determines your Amazon S3 website endpoint. For more information, see Website endpoints (./WebsiteEndpoints.html) .
    4. To accept the default settings and create the bucket, choose **Create**.
3. Create your subdomain bucket:
    1. Choose **Create bucket**.
    2. Enter the **Bucket name** (for example, `www.example.com`).
    3. Choose the Region where you want to create the bucket.

        Choose a Region close to you to minimize latency and costs, or to address regulatory requirements. The Region that you choose determines your Amazon S3 website endpoint. For more information, see Website endpoints (./WebsiteEndpoints.html) .
    4. To accept the default settings and create the bucket, choose **Create**.

In the next step, you configure `example.com` for website hosting.

Step 3: Configure your root domain bucket for website hosting

In this step, you configure your root domain bucket (`example.com`) as a website. This bucket will contain your website content. When you configure a bucket for website hosting, you can access the website using the Website endpoints (./WebsiteEndpoints.html) .

To enable static website hosting

1. Sign in to the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/s3/ (https://console.aws.amazon.com/s3/) .
2. In the **Buckets** list, choose the name of the bucket that you want to enable static website hosting for.
3. Choose **Properties**.
4. Under **Static website hosting**, choose **Edit**.
5. Choose **Use this bucket to host a website**.
6. Under **Static website hosting**, choose **Enable**.
7. In **Index document**, enter the file name of the index document, typically `index.html`.

   The index document name is case sensitive and must exactly match the file name of the HTML index document that you plan to upload to your S3 bucket. When you configure a bucket for website hosting, you must specify an index document. Amazon S3 returns this index document when requests are made to the root domain or any of the subfolders. For more information, see Configuring an index document (https://docs.aws.amazon.com/AmazonS3/latest/dev/IndexDocumentSupport.html) .
8. To provide your own custom error document for 4XX class errors, in **Error document**, enter the custom error document file name.

   The error document name is case sensitive and must exactly match the file name of the HTML error document that you plan to upload to your S3 bucket. If you don't specify a custom error document and an error occurs, Amazon S3 returns a default HTML error document. For more information, see Configuring a custom error document (https://docs.aws.amazon.com/AmazonS3/latest/dev/CustomErrorDocSupport.html) .
9. (Optional) If you want to specify advanced redirection rules, in **Redirection rules**, enter XML to describe the rules.

   For example, you can conditionally route requests according to specific object key names or prefixes in the request. For more information, see Configuring advanced conditional redirects (https://docs.aws.amazon.com/AmazonS3/latest/dev/how-to-page-redirect.html#advanced-conditional-redirects) .
10. Choose **Save changes**.

    Amazon S3 enables static website hosting for your bucket. At the bottom of the page, under **Static website hosting**, you see the website endpoint for your bucket.
11. Under **Static website hosting**, note the **Endpoint**.

    The **Endpoint** is the Amazon S3 website endpoint for your bucket. After you finish configuring your bucket as a static website, you can use this endpoint to test your website.

In the next step, you configure your subdomain (`www.example.com`) to redirect requests to your domain (`example.com`).

Step 4: Configure your subdomain bucket for website redirect

After you configure your root domain bucket for website hosting, you can configure your subdomain bucket to redirect all requests to the domain. In this example, all requests for `www.example.com` are redirected to `example.com`.

To configure a redirect request

1. On the Amazon S3 console, in the **Buckets** list, choose your subdomain bucket name (`www.example.com` in this example).
2. Choose **Properties**.
3. Under **Static website hosting**, choose **Edit**.
4. Choose **Redirect requests for an object**.
5. In the **Target bucket** box, enter your root domain, for example, `example.com`.
6. For **Protocol**, choose **http**.
7. Choose **Save changes**.

Step 5: Configure logging for website traffic

If you want to track the number of visitors accessing your website, you can optionally enable logging for your root domain bucket. For more information, see Amazon S3 Server Access Logging (https://docs.aws.amazon.com/AmazonS3/latest/dev/ServerLogs.html) . If you plan to use Amazon CloudFront to speed up your website, you can also use CloudFront logging.

To enable server access logging for your root domain bucket

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/ (https://console.aws.amazon.com/s3/) .
2. In the same Region where you created the bucket that is configured as a static website, create a bucket for logging, for example `logs.example.com`.
3. Create a folder for the server access logging log files (for example, `logs`).
4. (Optional) If you want to use CloudFront to improve your website performance, create a folder for the CloudFront log files (for example, `cdn`).
5. In the **Buckets** list, choose your root domain bucket.
6. Choose **Properties**.
7. Under **Server access logging**, choose **Edit**.
8. Choose **Enable**.
9. Under the **Target bucket**, choose the bucket and folder destination for the server access logs:
   - Browse to the folder and bucket location:
     1. Choose **Browse S3**.
     2. Choose the bucket name, and then choose the logs folder.
     3. Choose **Choose path**.
   - Enter the S3 bucket path, for example, `s3://logs.example.com/logs/`.

10. Choose **Save changes**.

    In your log bucket, you can now access your logs. Amazon S3 writes website access logs to your log bucket every 2 hours.

Step 6: Upload index and website content

In this step, you upload your index document and optional website content to your root domain bucket.

When you enable static website hosting for your bucket, you enter the name of the index document (for example, `index.html`). After you enable static website hosting for the bucket, you upload an HTML file with this index document name to your bucket.

To configure the index document

1. Create an `index.html` file.

   If you don't have an `index.html` file, you can use the following HTML to create one:

```
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
    <title>My Website Home Page</title>
</head>
<body>
  <h1>Welcome to my website</h1>
  <p>Now hosted on Amazon S3!</p>
</body>
</html>
```

2. Save the index file locally with the *exact* index document name that you entered when you enabled static website hosting for your bucket (for example, `index.html`).

   The index document file name must exactly match the index document name that you enter in the **Static website hosting** dialog box. The index document name is case sensitive. For example, if you enter `index.html` for the **Index document** name in the **Static website hosting** dialog box, your index document file name must also be `index.html` and not `Index.html`.

3. Sign in to the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/s3/ (https://console.aws.amazon.com/s3/) .

4. In the **Buckets** list, choose the name of the bucket that you want to use to host a static website.

5. Enable static website hosting for your bucket, and enter the exact name of your index document (for example, `index.html`). For more information, see Enabling website hosting (./EnableWebsiteHosting.html) .

   After enabling static website hosting, proceed to step 6.

6. To upload the index document to your bucket, do one of the following:

   - Drag and drop the index file into the console bucket listing.
   - Choose **Upload**, and follow the prompts to choose and upload the index file.

For step-by-step instructions, see How Do I Upload Files and Folders to an Amazon S3 Bucket? (https://docs.aws.amazon.com/AmazonS3/latest/user-guide/upload-objects.html) in the *Amazon Simple Storage Service Console User Guide*.

7. (Optional) Upload other website content to your bucket.

Step 7: Edit S3 Block Public Access settings

In this example, you edit block public access settings for the domain bucket (`example.com`) to allow public access.

By default, Amazon S3 blocks public access to your account and buckets. If you want to use a bucket to host a static website, you can use these steps to edit your block public access settings.

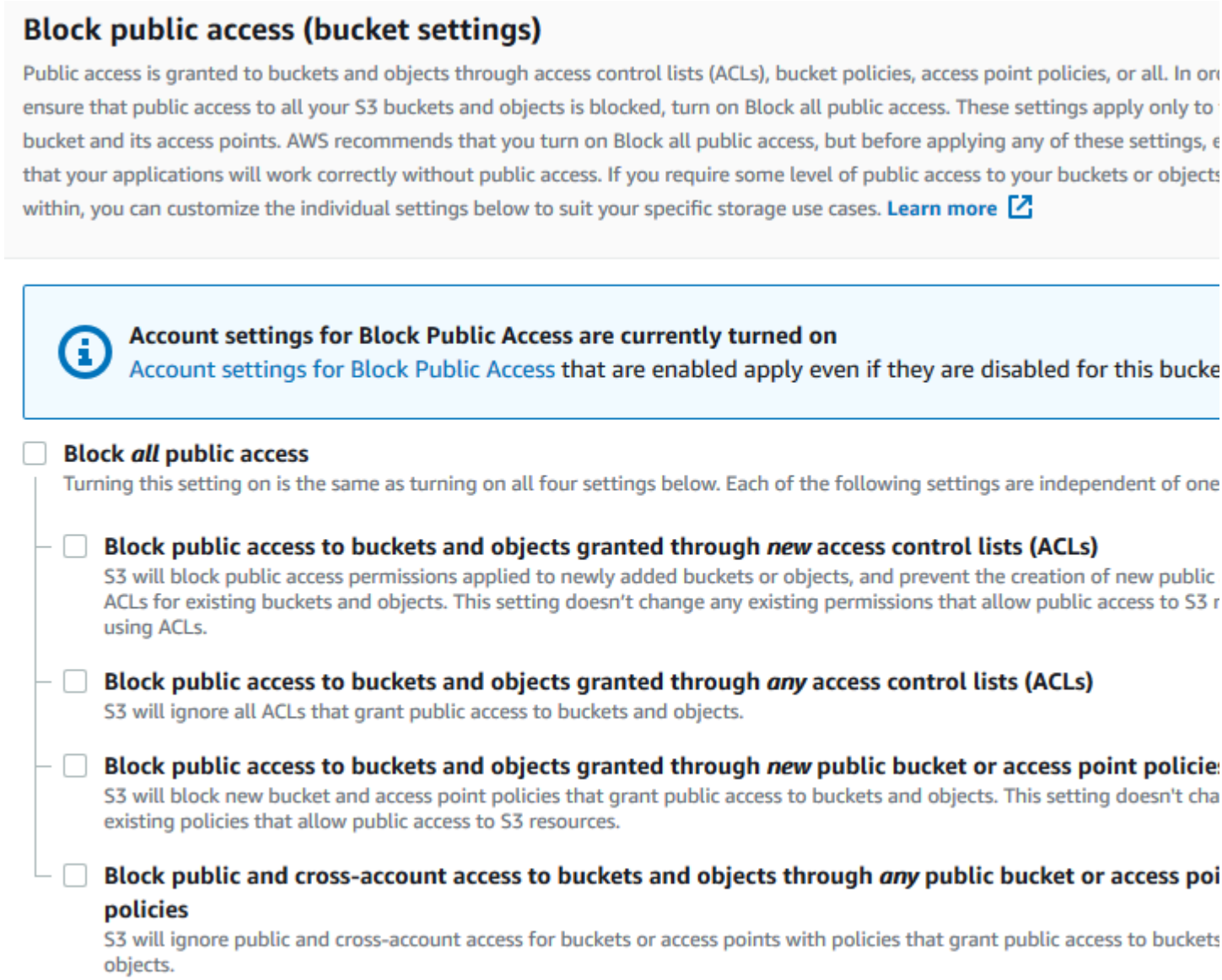
⚠Warning

Before you complete this step, review Using Amazon S3 Block Public Access (https://docs.aws.amazon.com/AmazonS3/latest/dev/access-control-block-public-access.html) to ensure that you understand and accept the risks involved with allowing public access. When you turn off block public access settings to make your bucket public, anyone on the internet can access your bucket. We recommend that you block all public access to your buckets.

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/ (https://console.aws.amazon.com/s3/) .
2. Choose the name of the bucket that you have configured as a static website.
3. Choose **Permissions**.
4. Under **Block public access (bucket settings)**, choose **Edit**.
5. Clear **Block *all* public access**, and choose **Save changes**.

   ⚠Warning

   Before you complete this step, review Using Amazon S3 Block Public Access (https://docs.aws.amazon.com/AmazonS3/latest/dev/access-control-block-public-access.html) to ensure you understand and accept the risks involved with allowing public access. When you turn off block public access settings to make your bucket public, anyone on the internet can access your bucket. We recommend that you block all public access to your buckets.



Amazon S3 turns off Block Public Access settings for your bucket. To create a public, static website, you might also have to edit the Block Public Access settings (https://docs.aws.amazon.com/AmazonS3/latest/user-guide/block-public-access-account.html) for your account before adding a bucket policy. If account settings for Block Public Access are currently turned on, you see a note under **Block public access (bucket settings)**.

Step 8: Attach a bucket policy

After you edit S3 Block Public Access settings, you can add a bucket policy to grant public read access to your bucket. When you grant public read access, anyone on the internet can access your bucket.

Important

The following policy is an example only and allows full access to the contents of your bucket. Before you proceed with this step, review How can I secure the files in my Amazon S3 bucket? (https://aws.amazon.com/premiumsupport/knowledge-center/secure-s3-resources/) to ensure that you understand the best practices for securing the files in your S3 bucket and risks involved in granting public access.

1. Under **Buckets**, choose the name of your bucket.
2. Choose **Permissions**.
3. Under **Bucket Policy**, choose **Edit**.

4. To grant public read access for your website, copy the following bucket policy, and paste it in the **Bucket policy editor**.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": "*",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::example.com/*"
            ]
        }
    ]
}
```

5. Update the `Resource` to your bucket name.

   In the preceding example bucket policy, `example.com` is the bucket name. To use this bucket policy with your own bucket, you must update this name to match your bucket name.

6. Choose **Save changes**.

   A message appears indicating that the bucket policy has been successfully added.

   If you see an error that says `Policy has invalid resource`, confirm that the bucket name in the bucket policy matches your bucket name. For information about adding a bucket policy, see How do I add an S3 bucket policy? (https://docs.aws.amazon.com/AmazonS3/latest/user-guide/add-bucket-policy.html)

   If you get an error message and cannot save the bucket policy, check your account and bucket Block Public Access settings to confirm that you allow public access to the bucket.

In the next step, you can figure out your website endpoints and test your domain endpoint.

Step 9: Test your domain endpoint

After you configure your domain bucket to host a public website, you can test your endpoint. For more information, see Website endpoints (./WebsiteEndpoints.html) . You will only be able to test the endpoint for your domain bucket since your subdomain bucket is set up for website redirect and not static website hosting.

Note

Amazon S3 does not support HTTPS access to the website. If you want to use HTTPS, you can use Amazon CloudFront to serve a static website hosted on Amazon S3.

For more information, see How do I use CloudFront to serve a static website hosted on Amazon S3? (http://aws.amazon.com/premiumsupport/knowledge-center/cloudfront-serve-static-website/) and Requiring HTTPS for communication between viewers and CloudFront (https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/using-https-viewers-to-cloudfront.html) .

1. Under **Buckets**, choose the name of your bucket.
2. Choose **Properties**.
3. At the bottom of the page, under **Static website hosting**, choose your **Bucket website endpoint**.

   Your index document opens in a separate browser window.

In the next step, you use Amazon Route 53 to enable customers to use both of your custom URLs to navigate to your site.

Step 10: Add alias records for your domain and subdomain

In this step, you create the alias records that you add to the hosted zone for your domain maps `example.com` and `www.example.com`. Instead of using IP addresses, the alias records use the Amazon S3 website endpoints. Amazon Route 53 maintains a mapping between the alias records and the IP addresses where the Amazon S3 buckets reside. You create two alias records, one for your root domain and one for your subdomain.

▼ Add an alias record for your root domain and subdomain

To add an alias record for your root domain (`example.com`)

1. Open the Route 53 console at https://console.aws.amazon.com/route53/ (https://console.aws.amazon.com/route53/) .

   Note

   If you don't already use Route 53, see Step 1: Register a Domain (https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/getting-started.html#getting-started-find-domain-name) in the *Amazon Route 53 Developer Guide*. After completing your setup, you can resume the instructions.
2. Choose **Hosted zones**.
3. In the list of hosted zones, choose the name of the hosted zone that matches your domain name.
4. Choose **Create record**.
5. Choose **Simple routing**, and choose **Next**.
6. Choose **Define simple record**.
7. In **Record name** accept the default value, which is the name of your hosted zone and your domain.
8. In **Value/Route traffic to**, choose **Alias to S3 website endpoint**.
9. Choose the Region.
10. Choose the S3 bucket.

    The bucket name should match the name that appears in the **Name** box. In the **Alias Target** listing, the bucket name is followed by the Amazon S3 website endpoint for the Region where the bucket was created, for example `example.com (s3-website-us-west-2.amazonaws.com)`. **Alias Target** lists a bucket if:

- You configured the bucket as a static website.
- The bucket name is the same as the name of the record that you're creating.
- The current AWS account created the bucket.

If your bucket does not appear in the **Alias Target** listing, enter the Amazon S3 website endpoint for the Region where the bucket was created, for example, `s3-website-us-west-2.amazonaws.com`. For a complete list of Amazon S3 website endpoints, see Amazon S3 Website Endpoints (https://docs.aws.amazon.com/general/latest/gr/s3.html#s3_website_region_endpoints) . For more information about the alias target, see Alias Target (https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resource-record-sets-values-alias.html#rrsets-values-alias-alias-target) in the *Amazon Route 53 Developer Guide*.

11. In **Record type**, choose **A - Routes traffic to an IPv4 address and some AWS resources**.
12. For **Evaluate target health**, choose **No**.
13. Choose **Define simple record**.

To add an alias record for your subdomain (`www.example.com`)

1. In the hosted zone for your root domain (`example.com`), choose **Create record**.
2. In **Record name** for your subdomain, type `www`.
3. In **Value/Route traffic to**, choose **Alias to S3 website endpoint**.
4. Choose the Region.
5. Choose the S3 bucket, for example `example.com (s3-website-us-west-2.amazonaws.com)`.
6. In **Record type**, choose **A - Routes traffic to an IPv4 address and some AWS resources**.
7. For **Evaluate target health**, choose **No**.
8. Choose **Define simple record**.

Note

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you can route traffic to your Amazon S3 bucket by using the names of the alias records that you created in this procedure.

▶ Add an alias record for your root domain and subdomain (old Route 53 console)

To add an alias record for your root domain (`example.com`)

The Route 53 console has been redesigned. In the Route 53 console you can temporarily use the old console. If you choose to work with the old Route 53 console, use the procedure below.

1. Open the Route 53 console at https://console.aws.amazon.com/route53/ (https://console.aws.amazon.com/route53/) .

   Note

   If you don't already use Route 53, see Step 1: Register a Domain (https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/getting-started.html#getting-started-find-domain-name) in the *Amazon Route 53 Developer Guide*. After completing your setup, you can resume the instructions.

2. Choose **Hosted Zones**.

3. In the list of hosted zones, choose the name of the hosted zone that matches your domain name.

4. Choose **Create Record Set**.

5. Specify the following values:

   Name

   Accept the default value, which is the name of your hosted zone and your domain.

   For the root domain, you don't need to enter any additional information in the **Name** field.

   Type

   Choose **A – IPv4 address**.

   Alias

   Choose **Yes**.

   Alias Target

   In the **S3 website endpoints** section of the list, choose your bucket name.

   The bucket name should match the name that appears in the **Name** box. In the **Alias Target** listing, the bucket name is followed by the Amazon S3 website endpoint for the Region where the bucket was created, for example `example.com (s3-website-us-west-2.amazonaws.com)`. **Alias Target** lists a bucket if:

   - You configured the bucket as a static website.
   - The bucket name is the same as the name of the record that you're creating.
   - The current AWS account created the bucket.

   If your bucket does not appear in the **Alias Target** listing, enter the Amazon S3 website endpoint for the Region where the bucket was created, for example, `s3-website-us-west-2.amazonaws.com`. For a complete list of Amazon S3 website endpoints, see Amazon S3 Website Endpoints (https://docs.aws.amazon.com/general/latest/gr/s3.html#s3_website_region_endpoints) . For more information about the alias target, see Alias Target (https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resource-record-sets-values-alias.html#rrsets-values-alias-alias-target) in the *Amazon Route 53 Developer Guide*.

   Routing Policy

   Accept the default value of **Simple**.

   Evaluate Target Health

Accept the default value of **No**.

6. Choose **Create**.

To add an alias record for your subdomain (`www.example.com`)

1. In the hosted zone for your root domain (`example.com`), choose **Create Record Set**.
2. Specify the following values:

   Name

   For the subdomain, enter `www` in the box.

   Type

   Choose **A – IPv4 address**.

   Alias

   Choose **Yes**.

   Alias Target

   In the **S3 website endpoints** section of the list, choose the same bucket name that appears in the **Name** field—for example, `www.example.com (s3-website-us-west-2.amazonaws.com)`.

   Routing Policy

   Accept the default value of **Simple**.

   Evaluate Target Health

   Accept the default value of **No**.

3. Choose **Create**.

Note

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you can route traffic to your Amazon S3 bucket by using the names of the alias records that you created in this procedure.

Step 11: Test the website

Verify that the website and the redirect work correctly. In your browser, enter your URLs. In this example, you can try the following URLs:

- **Domain** (`http://example.com`) – Displays the index document in the `example.com` bucket.
- **Subdomain** (`http://www.example.com`) – Redirects your request to `http://example.com`. You see the index document in the `example.com` bucket.

If your website or redirect links don't work, you can try the following:

- **Clear cache**–Clear the cache of your web browser.

- **Check name servers**–If your web page and redirect links don't work after you've cleared your cache, you can compare the name servers for your domain and the name servers for your hosted zone. If the name servers don't match, you might need to update your domain name servers to match those listed under your hosted zone. For more information, see Adding or changing name servers and glue records for a domain (https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/domain-name-servers-glue-records.html) .

Once you've successfully tested your root domain and subdomain, you can set up an Amazon CloudFront (http://aws.amazon.com/cloudfront) distribution to improve the performance of your website and provide logs that you can use to review website traffic. For more information, see Speeding up your website with Amazon CloudFront (./website-hosting-cloudfront-walkthrough.html) .