Amazon Route 53

Getting Started Guide

API Version 2010-10-01

amazon
webservices™

Amazon Route 53: Getting Started Guide

Table of Contents

Get Started with Amazon Route 53

Whether you are creating a new DNS domain or migrating an existing domain, getting started with Amazon Route 53 is easy: You sign up for the service, make a few simple API calls using the Amazon Route 53 dnscurl.pl utility, and then update your domain registrar to use the Amazon Route 53 name servers. That's all there is to it.

To get started:

1	Sign up for Amazon Route 53.
2	Use dnscurl.pl to create a hosted zone.
3	Add DNS records (such as CNAME) to your Amazon S3 bucket and Amazon CloudFront distribution.
4	Call `GetHostedZone` to retrieve the name server NS records.
5	Use your registrar's web site to update the nameservers for your domain.

Sign Up for Amazon Route 53

To use Amazon Route 53, you need an AWS account. If you don't already have one, you'll be prompted to create one when you sign up for Amazon Route 53. You're not charged for any services unless you use them.

To sign up for Amazon Route 53

1. Go to http://aws.amazon.com/route53/ and click **Sign Up for Amazon Route 53**.
2. Follow the on-screen instructions.

You now have an AWS account and you are signed up for Amazon Route 53.

Create a Hosted Zone

Creating a hosted zone involves creating an XML request and using the Amazon Route 53 dnscurl.pl tool to send your request, as described in the following steps. Text shown in red indicates where you need to modify the values.

To create a hosted zone

1. Download `dnscurl.pl` from Amazon Route 53 Developer Tools.

2. Create an XML file containing the `CreateHostedZoneRequest` request. In this tutorial, we will use the file MyCreateRequest.xml as an example. Your request should look similar to the following example.

   In the example request, you enclose the registered domain name that you're creating the hosted zone for in the `Name` tags and use the `CallerReference` tag to specify a unique description of your request. CallerReference can be any unique string; you might choose to use a string that identifies your project, such as `MyDNSMigration_01`. Each request you send must include a unique CallerReference. The `HostedZoneConfig` value is not required but you can use it to record notes, such as the purpose for creating the hosted zone.

```
<CreateHostedZoneRequest xmlns="https://route53.amazonaws.com/doc/2010-10-
01/">
    <Name>example.com.</Name>
    <CallerReference>a_unique_reference</CallerReference>
    <HostedZoneConfig>
        <Comment>Migrate an existing domain to Route 53</Comment>
    </HostedZoneConfig>
</CreateHostedZoneRequest>
```

   For more information about the `CreateHostedZone` API, go to the CreateHostedZone in the Amazon Route 53 API Reference.

3. Prepare an access key file:

   a. Create a new file. In our example, we call it .aws-secrets.

   b. Go to the AWS Security Credentials page and look up your Access Credentials.

   c. Copy your Access Key ID and Secret Access Key into your new file, as shown in the example.

   d. Save the file as read-only.

   Your credentials file should look similar to the following example.

```
%awsSecretAccessKeys = (
    "my-aws-account" => {
        id => "0F1V9UKMN23IEXAMPLE2",
        key => "lkTwZRalREYSU7lbSS5VoUhdYREcwsasEXAMPLEQ",
    },
);
```

4. Enter the following at the command line to set the permissions on your .aws-secrets file:

```
chmod 600 .aws-secrets
```

5. Send the create request:

a. From a new shell, navigate to the directory where you downloaded dnscurl.pl.

b. At the prompt, enter:

```
dnscurl.pl --keyname my-aws-account -- -H "Content-Type: text/xml; char
set=UTF-8" -X POST
--upload-file ./MyCreateRequest.xml https://route53.amazonaws.com/2010-10-
01/hostedzone
```

The response you receive includes information similar to the following.

```
<?xml version="1.0"?>
<CreateHostedZoneResponse xmlns="https://route53.amazonaws.com/doc/2010-10-
01/">
  <HostedZone>
    <Id>/hostedzone/Z21DW1QVGID6NG</Id>
    <Name>example.com.</Name>
    <CallerReference>a_unique_reference</CallerReference>
    <Config>
      <Comment>Migrate an existing domain to Route 53</Comment>
    </Config>
  </HostedZone>
  <ChangeInfo>
    <Id>/change/C24LD0DUV5VOVE</Id>
    <Status>PENDING</Status>
    <SubmittedAt>2010-12-02T01:34:20.633Z</SubmittedAt>
  </ChangeInfo>
  <DelegationSet>
    <NameServers>
      <NameServer>ns-1638.awsdns-12.co.uk</NameServer>
      <NameServer>ns-144.awsdns-18.com</NameServer>
      <NameServer>ns-781.awsdns-33.net</NameServer>
      <NameServer>ns-1478.awsdns-56.org</NameServer>
    </NameServers>
  </DelegationSet>
</CreateHostedZoneResponse>
```

This response shows you the following:

- Your request was received
- The hosted zone now has an assigned ID (enclosed in the `Id` tag)
- A description of the hosted zone and name servers

> **Tip**
>
> Take note of the hosted zone ID, you will need it later to update your records.

> **Note**
>
> If you don't receive a response to your request, you can resubmit the request with the same caller reference. You can use the `ListHostedZones` action to find the caller reference for your request. For information about the `ListHostedZones` action, see ListHostedZones.

Congratulations, you have created a hosted zone!

Create New DNS Records

Now that you have a hosted zone, you need to populate it with your existing DNS data. For example, if your domain contains CNAMEs to other AWS resources such as an Amazon S3 bucket and a CloudFront distribution, you will want to create two new resource record sets. For more information about the types of DNS resource records that Amazon Route 53 supports, see Appendix A: Domain Name and Resource Record Formats in the Amazon Route 53 Developer Guide.

To create new DNS records:

1. Create a new MyRecordsRequest.xml file to set up a CNAME from mybucket.example.com to mybucket.example.com.s3.amazonaws.com, and a CNAME from cdn.example.com to d1234abc.cloudfront.net. The request must include:

   a. A `ChangeResourceRecordSetsRequest` element

   b. Two CNAME records

   Your request might look like the following. Changes to each resource record are defined in separate `Change` elements. In this example, we have one `Change` element. It is a `CREATE` action that adds two CNAME records.

```
<?xml version="1.0" encoding="UTF-8"?>
<ChangeResourceRecordSetsRequest xmlns="https://route53.amazon
aws.com/doc/2010-10-01/">
    <ChangeBatch>
        <Comment>
        Migrate existing records to Amazon Route 53
        </Comment>
        <Changes>
           <Change>
               <Action>CREATE</Action>
               <ResourceRecordSet>
                   <Name>mybucket.example.com.</Name>
                   <Type>CNAME</Type>
                   <TTL>14400</TTL>
                   <ResourceRecords>
                      <ResourceRecord>
                          <Value>mybucket.example.com.s3.amazonaws.com</Value>
                      </ResourceRecord>
                   </ResourceRecords>
               </ResourceRecordSet>
           </Change>
           <Change>
               <Action>CREATE</Action>
               <ResourceRecordSet>
                   <Name>cdn.example.com.</Name>
                   <Type>CNAME</Type>
                   <TTL>14400</TTL>
                   <ResourceRecords>
                      <ResourceRecord>
                          <Value>d1234abc.cloudfront.net</Value>
                      </ResourceRecord>
```

```
                </ResourceRecords>
            </ResourceRecordSet>
        </Change>
    </Changes>
    </ChangeBatch>
</ChangeResourceRecordSetsRequest>
```

2. Enter the following command at the prompt to send your request using the dnscurl.pl tool that you used when you created your hosted zone.

```
dnscurl.pl --keyname my-aws-account -- -H "Content-Type: text/xml; char
set=UTF-8" -X POST
--upload-file ./MyRecordsRequest.xml https://route53.amazonaws.com/2010-10-
01/hostedzone/hosted zone ID/rrset
```

Your response will look similar to the following and shows that the status of the requested change, identified as `/change/CVSYPFWLOWW74`, is `PENDING`.

```
<ChangeResourceRecordSetsResponse xmlns="https://awsdns.amazon
aws.com/doc/2010-10-01/">
    <ChangeInfo>
        <Id>/change/CVSYPFWLOWW74</Id>
            <Status>PENDING</Status>
            <SubmittedAt>2010-11-27T03:14:40.707Z</SubmittedAt>
    </ChangeInfo>
</ChangeResourceRecordSetsResponse>
```

You have added CNAME records to your hosted zone.

For more information on making changes to your resource records, see Making Changes to Your Resource Record Sets in the *Amazon Route 53 Developer Guide*.

Confirm Your Requests are Complete

After you send your change request, you should check to ensure your change was applied to all Amazon Route 53 DNS servers. To check the status of your change, you poll for the status of the change. When your change is complete, its status becomes `INSYNC`.

To check the status of your requests:

*   Enter the following dnscurl command using the `ChangeInfo Id` value from the `ChangeResourceRecordSetsResponse` returned when you created the hosted zone.

In the previous example, this value was `CVSYPFWLOWW74`. The change ID is a unique value that references a single change request.

```
dnscurl.pl --keyname my-aws-account -- -H "Content-Type:
    text/xml; charset=UTF-8" https://route53.amazonaws.com/2010-10-
01/change/CVSYPFWLOWW74
```

The response to the confirmation request will look similar to the following.

```
<?xml version="1.0"?>
<GetChangeResponse xmlns="https://route53.amazonaws.com/doc/2010-10-01/">
    <ChangeInfo>
        <Id>/change/CVSYPFWLOWW74</Id>
        <Status>INSYNC</Status>
        <SubmittedAt>2010-11-27T03:14:40.707Z</SubmittedAt>
    </ChangeInfo>
</GetChangeResponse>
```

The example response indicates that the status of the change request, identified by the change ID, is `INSYNC`. If the status is `PENDING`, it indicates your changes have not yet fully propagated to all of the Amazon Route 53 DNS servers. In that case, you should check again after waiting 30 to 60 seconds. After the status changes to `INSYNC`, you know Amazon Route 53 has applied your change to all of the Amazon Route 53 DNS servers.

Update the Registrar's Name Server Records

After your changes are propagated and return the `INSYNC` status, you can update the name servers for your hosted zone with your registrar. To change the name server records, you replace the name servers that the registrar is currently using with the Amazon Route 53 name servers associated with your Amazon Route 53 hosted zone.

Caution

Before you reset the name servers to Route 53, read Migrating Your Existing DNS . You may want to preserve some settings in the zone records, and some settings can temporarily persist after you transfer your records to Route 53. There can be an interval of forty eight hours or more after you reassign name servers to Route 53 during which default name servers, that the provider set to register the domain name, are still being used by DNS resolvers. These settings might be be cached on DNS resolvers or or client computers.

To retrieve your name server information:

• Enter the following dnscurl.pl command to get your hosted zone ID.

You will need the hosted zone ID you received when you created the hosted zone. In our example, it was Z21DW1QVGID6NG. Your request will look like the following example.

```
dnscurl.pl --keyname my-aws-account -- -H "Content-Type:
    text/xml; charset=UTF-8" https://route53.amazonaws.com/2010-10-01/hosted
zone/Z21DW1QVGID6NG
```

The response to the confirmation request will look similar to the following.

```
<?xml version="1.0" encoding="UTF-8"?>
<GetHostedZoneResponse xmlns="https://route53.amazonaws.com/2010-10-01/hosted
zone">
    <HostedZone>
        <Id>/hostedzone/<hosted zone id></Id>
        <Name>example.com.</Name>
```

```
        <CallerReference>myUniqueIdentifier</CallerReference>
        <Config>
            <Comment>This is my first hosted zone.</Comment>
        </Config>
    </HostedZone>
    <DelegationSet>
        <NameServers>
            <NameServer>ns-01.awsdns-00.com</NameServer>
            <NameServer>ns-500.awsdns-11.net</NameServer>
            <NameServer>ns-1112.awsdns-31.org</NameServer>
            <NameServer>ns-1600.awsdns-27.co.uk</NameServer>
        </NameServers>
    </DelegationSet>
</GetHostedZoneResponse>
```

To update the registrar's name server records:

• Give your registrar your new name server names returned in the `DelegationSet` element.

You can usually update this information on your registrar's website, with your account information. If you cannot locate this information, refer to the Help documentation published by your registrar or contact your registrar directly.

⚠️ **Important**

> The changes you make to update the records at your registrar might not be effective immediately. You should check to ensure that your registrar applied your changes. For example, if you created a hosted zone that has a CNAME for an Amazon S3 bucket, when propagation is complete you should be able to type your CNAME into your browser and be directed to the location for your Amazon S3 bucket.

Congratulations, you are now using Amazon Route 53 to serve DNS queries for your domain!

For more information on using Amazon Route 53, go to

Where Do I Go from Here?

Topics

Although fairly simple to use, Amazon Route 53 is rich in functionality. There are a variety of resources you can use to learn more about Amazon Route 53. In addition to the Amazon Route 53 documentation, resources include links to third-party applications that work with Amazon Route 53. The list below includes a few of these resources that you might find helpful.

Graphical User Interfaces

The following third-party tools provide graphical user interfaces (GUIs) for working with Amazon Route 53:

- DNS30
- easyRoute53 (easyDNS)
- Interstate 53
- MyR53DNS
- Ylastic

Tools and Libraries

The tools and libraries listed in the following table are available for use with Amazon Route 53.

Tool or Library	Description
AmazonRoute53AppsScript (via webos-goodies)	Google spreadsheet management of Amazon Route 53
AWS Component for .NET (via SprightlySoft)	SprightlySoft .NET Component for Amazon Web Services with support for REST operations and Route 53

Tool or Library	Description
Boto API download (via github) Boto API documentation (Boto v.2.0)	Boto Python interface to Amazon Web Services
cli53 (via github)	Command line interface for Amazon Route 53
Dasein Cloud API	Java-based API
dev.SquareCows.com (blog and script download) script download (via github)	Python script to transfer a BIND zone to Amazon Route 53
easyRoute53 (easyDNS)	GUI tools, registrar services, and zone transfer services
RIAForge	ColdFusion based components for managing DNS using Amazon Route 53
RightScripts (via RightScale)	Scripts to configure or update your RightScale server for use with Amazon Route 53
RightScale Support Tutorials	RightScale tutorial for domain setup with Amazon Route 53
route53d	DNS front-end to Amazon Route 53 API (enables incremental zone transfer (IXFR))
Route53Manager (via github)	Web-based interface
Ruby Fog (via github)	The Ruby cloud services library

Please Provide Feedback

Your input is important to help make our documentation helpful and easy to use. Please tell us about your experience getting started with Amazon Route 53 by completing our Getting Started Survey.

Thank you.