Gatsby Cloud Migration Guide

V1.0 | August 2023

Table of Contents

Gatsby Cloud to Netlify Migration Guide

Audience

This guide will support all Gatsby Cloud migrations to Netlify for the following customers:

1. All Gatsby Cloud Free, Agency and Professional Plans
2. Gatsby Cloud Enterprise Plans except those using the following features:
   a. Enterprise SSO
   b. Static IP
   c. Log Drains

If you are a Gatsby Cloud Enterprise Plan customer using a, b, or c, please contact your Customer Success Manager for a private consultation.

Helpful links to get started

- [Gatsby on Netlify | Netlify Docs](#)
- [Install the Essential Gatsby Build Plug-in | Netlify Docs](#)
- [Troubleshoot Github Repository linking | Netlify Docs](#)
- [Blog] [Gatsby Adapters: Realize the Full Potential of Gatsby on Your Platform](#)
- [Blog] [Gatsby plugins supported by Netlify](#)
- [Blog] [Migrating DNS for a Production Site. Site Migration Checklist](#)

Migrating your site from Gatsby Cloud to Netlify

Create Netlify account (skip for existing Netlify account holders)

1. Navigate to the signup page: https://app.netlify.com/signup

2. Once you confirm your email, you'll go through Netlify onboarding. The first page you see will ask how you intend to use Netlify and to name your team. The second page will allow you to link your Git provider (or skip linking a Git provider to be brought to your Team dashboard).

ℹ️ Netlify will ask for similar permissions from your Git provider to what Gatsby Cloud required when first signing up. If you were required to involve your IT admin to link your site to Gatsby Cloud upon signup, you will probably need to request their assistance again for the Netlify migration.

Install Netlify and Gatsby plugins

ℹ️ Additional step-by-step plugin installation instructions at this link.

1. New Gatsby sites on Netlify automatically install the Essential Gatsby build plugin. You can confirm this in the build logs. If you want to install it manually, you can use file-based plugin installation and add the plugin as @netlify/plugin-gatsby in your netlify.toml file.

```toml
[[plugins]]
package = "@netlify/plugin-gatsby"
```

2. In your Gatsby site's repo, run npm i gatsby-plugin-netlify and add gatsby-plugin-netlify to the plugins array in your site's gatsby-config file.

Create new site + link your repository (if not done in step 1)

1. From your team homepage's 'Sites' section, click 'Add new site,' then click 'Import an existing project.'

2. Select your Git provider. From there you'll be asked to login to that platform. Log in.

3. Once logged into your Git provider, you'll be brought back to select your repository. If you don't see your organization listed on this page, you may need to coordinate with your IT admin to get Netlify the permissions it needs. Click here for troubleshooting Github repository linking.

4. The final page in adding a new site will ask for configuration details. Netlify will attempt to scan your repository to determine what framework you're using and apply Gatsby-specific settings. If for any reason Netlify can't determine that your site uses Gatsby, the typical build settings are:

   - *Base directory: *leave blank**
   - *Build command:* npm run build
   - Publish directory: public
   - *Functions directory: *leave blank**

5. Click the 'Add environment variables' button. Here you are only able to add variables one by one, but if you skip adding env variables here you can add them in bulk after clicking the 'Deploy' button and navigating to your Site Configuration menu.

Swap DNS from Gatsby to Netlify

Note: Instructions below also outlined in this site migration checklist for Netlify's best practices for migrating DNS with no downtime.

Migrating DNS for a site that's already in production doesn't have to be terrifying. The key is managing the period of time between when your DNS records point to your old host and when they point to Netlify. Below is a complete walkthrough of the steps to do that, so you can ensure a smooth migration with no downtime. The headings of this article make up your new site migration checklist, and we'll elaborate on each step below.

We recommend fully reading through before getting started. Ideally, you will start assembling the pieces several weeks before your go-live date.

Before go-live day:

1. Check which hostnames and which sites will be migrated
2. Check that nothing will block Let's Encrypt from issuing an SSL certificate for your hostname
3. Check TTL of all hostnames
4. Get a custom certificate for no downtime
5. Test the custom certificate before switching DNS records

On go-live day:

1. Update DNS records to point to Netlify
2. Once the migration is complete, get a Let's Encrypt SSL certificate from us

Check which hostnames and which sites will be migrated

Make a note of which hostnames you'll be migrating, including the bare domain—like example.com—and any subdomains—like api.example.com and www.example.com.

Further down the checklist, you'll have to get a custom SSL certificate, and you'll want to make sure that your certificate will cover every hostname you'll be migrating.

You can assign your domain to your Netlify site at this point or wait until later in the process. Don't worry, this doesn't mean that we'll start serving traffic to your site. Your DNS still doesn't point to Netlify, so the hostname is more of a placeholder and will also let you test your custom certificate later on.

If you want to be extra thorough, we'd recommend doing a run-through of this checklist where you set a test hostname that's covered by your custom certificate, like test.example.com. That way, you won't encounter any surprises when you do the process again with your production hostnames.

Check that nothing will block Let's Encrypt from issuing an SSL certificate for your hostname

Some companies, organizations, and universities set CAA or SOA records on their hostnames that specify which SSL certificate authorities can issue certificates. You'll need to check that your hostname either doesn't have these records at all, or that Let's Encrypt is listed as an acceptable certificate issuing authority. Otherwise, we won't be able to issue you an SSL certificate.

```
host -t caa example.com
```

```
host -t soa example.com
```

Check TTL of all hostnames

For each hostname you're migrating, you'll want to check the time to live (TTL) of that hostname's DNS records. You can find this value either using your terminal or Google's online dig tool.

```
dig example.com
```

```
dig www.example.com
```

We're interested in the ANSWER SECTION of the dig output. The number after the domain name is the record's TTL in seconds.

```
; <<>> DiG 9.10.6 <<>> example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16098
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;example.com.                    IN      A

;; ANSWER SECTION:
example.com.            55795   IN      A       93.184.216.34

;; Query time: 16 msec
;; SERVER: 192.168.0.1#53(192.168.0.1)
;; WHEN: Tue Mar 16 14:00:26 PDT 2021
;; MSG SIZE  rcvd: 45
```

The TTL is the amount of time that a given DNS record is kept in DNS cache before it is re-requested. A TTL of 3600 means that the record will not be re-requested for 3600 seconds, or 1 hour. Ahead of the migration, you want to set a low TTL so that DNS will request your new DNS records—the ones pointing to Netlify—in as short a time as possible, and, at the same time, purge the old DNS records—the ones pointing to your old site host—from the DNS cache. Our step-by-step support guide has more information on this piece of the process.

Get a custom certificate for no downtime

We can only get you a Let's Encrypt SSL certificate once your DNS is correctly configured and pointing to Netlify. Once you configure the DNS records, the best case is that your old records expire according to their TTL, and your new records then propagate to replace them. Because there are multiple caches (like the browser's DNS cache) along the way, however, DNS propagation can sometimes take several hours. During this interim period, we will not yet be able to automatically get you an SSL certificate.

That's why we recommend bringing your own custom certificate for site migrations: you can set up the certificate to cover the site before DNS is fully propagated, or before you've made any DNS changes at all,

so that your site will remain accessible during the propagation period. Without a custom certificate in place, visitors to your site will encounter a security warning in the browser until we are able to provision the Let's Encrypt certificate.

Where does one get a custom certificate, you ask? There are many vendors to choose from including [DigiCert](#) and [Comodo](#). Regardless of which vendor you use, you will need to generate a certificate signing request (CSR) for your custom certificate. DigiCert has this helpful [resource](#) for generating a CSR locally, or there are several online tools like https://decoder.link/csr_generator. [Our docs](#) explain the specific files you'll need to copy/paste on the Netlify side: a PEM formatted certificate, a private key, and a CA certificate chain.
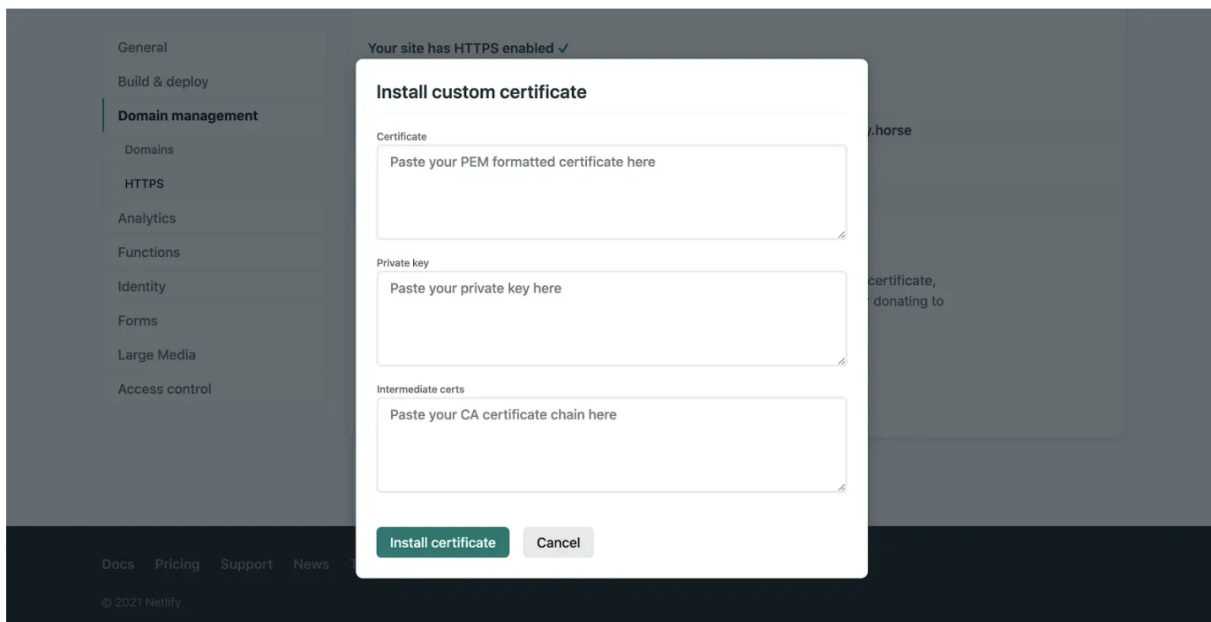
Test the custom certificate before switching DNS records

Once you've uploaded your custom certificate, but before you've updated DNS records, you'll want to test that the certificate works. You can do this by using curl to force web requests to resolve to your custom domain as if it were already served by Netlify—so, as if you had updated DNS records, even though you haven't actually updated them yet:

```
curl https://www.example.com --connect-to www.example.com:443:104.198.14.
curl https://example.com --connect-to example.com:443:104.198.14.52
```

Or for older versions of curl:

```
curl https://www.example.com --resolve www.example.com:443:104.198.14.52
curl https://example.com --resolve example.com:443:104.198.14.52
```

You're looking for a 200 response, the body of the page you requested, and no SSL errors. If you can only curl by ignoring SSL errors with curl -k https://… after uploading your certificate, something is wrong and you'll want to repair it before moving to the next step.

Update DNS records to point to Netlify

When you've confirmed that your custom SSL certificate is correctly installed, it's time to update your DNS records to point to Netlify. Remember the TTLs you looked at earlier in this process? You'll need those again now. Make a note of them, then adjust the TTLs down to 10, and wait for the original TTL to pass. Once the original TTL has passed, your DNS records have the new TTL of 10, so it's time to update the A record and CNAME values:

1. A record pointing your bare domain to our load balancer, 75.2.60.5
2. CNAME record pointing www to your *.netlify.app Netlify URL, i.e. brave-curie-12345.netlify.app

Note: If you are an [enterprise customer](#) hosting on the High-Performance Edge, you will have different A and CNAME records than the ones described above. Your accounts team will share those details with you.
Your DNS migration should be complete once 10 seconds pass. You can confirm with host:

```
host -t a example.com

example.com has address 104.198.14.52
```

```
host -t cname www.example.com

www.example.com is an alias for brave-curie-12345.netlify.app
```

Once you've confirmed, it's time to crank the TTLs on your records back up to their original values, so visitors to your site can benefit from getting your DNS destination from cache instead of requesting it again every 10 seconds.

Once the migration is complete, get a Let's Encrypt SSL certificate from us

Now you're in the home stretch! The only thing left to do is swap out your custom SSL certificate. You can switch to the Let's Encrypt certificate within your site dashboard, which we'll now be able to provision for you and then automatically renew every three months going forward. There will be no interruption of service when the one SSL certificate replaces the other.
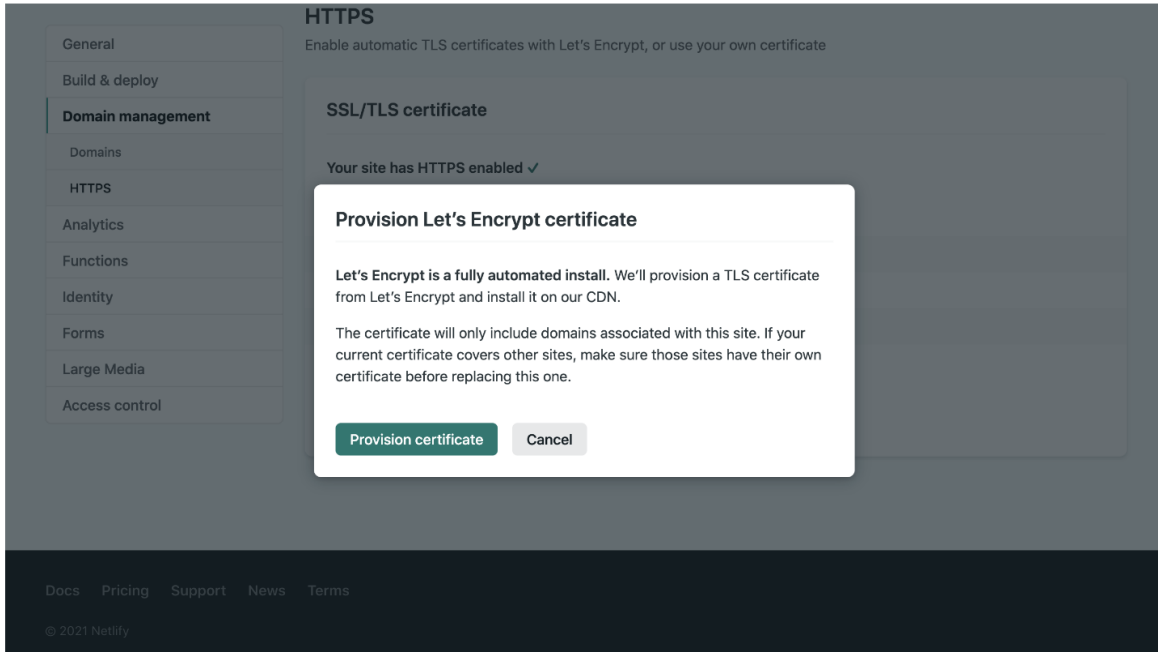
Congratulations! You did it. If you have any more questions about live site migrations, please reach out in the [Netlify Forum](#).

Migration Support

If you execute the above steps and encounter issues or troubleshooting does not work, please open a support ticket here: https://www.netlify.com/support/ so we can assist!

Migration Frequently Asked Questions (FAQ)

1. **Which Gatsby Cloud customers are affected by this discontinuation? All customers or Enterprise only?**

   a. All Gatsby Cloud customers are affected. Depending on the Gatsby Cloud plan, there are different end of service dates, contract options, terms, and migration paths.

2. **What are the end of service dates for each Gatsby Cloud Plan type?**

   a. For all **Gatsby Cloud Free plan** customers, your Gatsby Cloud instance will be discontinued on September 29, 2023. Please leverage the below Migration Guide to migrate all Gatsby Cloud properties to Netlify prior to that date otherwise you may lose all existing Gatsby Cloud data.

   Netlify is offering all Gatsby Cloud Self-Serve (Professional or Free) customers a simple, easy migration path to a Netlify Starter or Pro plan that fits your long-term needs. All plan details [available here](#).

   For the majority of **Enterprise Gatsby Cloud** customers, Netlify will maintain Gatsby Cloud operations until the current Gatsby contract expiration date. Custom terms are being offered to a small number of Gatsby Cloud customers to ensure a smooth migration experience.

   If the customer chooses to sign a new Netlify contract and migrate all Gatsby Cloud properties to Netlify before the completion of the current Gatsby contract, Netlify will offer a 1-year contract renewal at the same annual contract value (ACV) of the current Gatsby Cloud contract and cap the future renewal ACV increase at 10% above current Gatsby Cloud contract ACV for 1 additional year.

   Each Enterprise customer has 90 days from this announcement date to accept this offer to sign a new Netlify contract and until the current contract expiration date to migrate all Gatsby properties to Netlify.

   For all Enterprise Gatsby Cloud customers with a contract renewal date less than 90 days from today, this offer term is valid until the contract renewal date. (e.g. If contract renewal is 60 days from today, the customer has 60 days to migrate to Netlify and sign a new contract).

   Example: If current Gatsby contract expiration is 12/31/2023, the customer has 90 days (until 11/24/23) to sign a new contract with Netlify for a term lasting until 12/31/2024 at the current

Gatsby contract ACV. On 12/31/2024, the customer can choose to sign a new 1-year contract with an ACV capped at a 10% increase over the current Gatsby ACV, expiring 12/31/2025.

NOTE: Where applicable, please consider this communication as formal notice by Gatsby, Inc of its intention not to renew a customer's current subscription for a subsequent term and terminate the subscription at the end of the current term.

For all **Gatsby Cloud Professional plan** customers, Netlify will maintain Gatsby Cloud operations for a period of 30 days after your September 2023 billing date or until the end of October 2023, depending on your billing term. To ensure continued operation of all Gatsby Cloud web properties and applications, you'll need to migrate all properties to Netlify prior to the expiration of your September billing cycle date.

- For monthly billing customers, your Gatsby Cloud Pro instance will be discontinued on the Monday following your September billing cycle expiration date.

- For annual billing customers, your Gatsby Cloud instance will be discontinued on Tuesday, October 31, 2023. If your contract expires after October 31, 2023, you'll be issued a prorated refund for the remainder of the contract term.

Examples: If a current Gatsby Pro Plan monthly contract billing date is 9/15/2023, the customer will maintain Gatsby Cloud service until 10/15/2023 (30 days after September billing date). The customer's Gatsby Cloud instance will be shut down the following Monday.

If a current Gatsby Pro Plan annual contract renewal date is 12/31/2023, the customer's Gatsby Cloud instance will be shut down on 10/31/2023 and will receive a refund equal to 2 months (60 days) of the contract value.

3. **What will happen to my Gatsby Cloud instance if I do not complete the migration to Netlify or another platform by the final support date? If a Gatsby Cloud customer chooses not to migrate, what happens to all builds and data stored/cached/managed in Gatsby Cloud?**

   a. We have a migration path for you to ensure you can copy all existing sites, applications, and data that is supported by Gatsby Cloud to Netlify. Based on your current Gatsby stack, there is a custom path to make a migration as seamless and fast as possible. All migrations will need to be completed by the planned end of service date specific to your plan.

However, if a Gatsby Cloud customer chooses not to migrate existing sites and data to Netlify, all data will be deleted and sites running on Gatsby Cloud will no longer be accessible on the internet, supported, or replicated after the end of service date for your plan.

4. **When is Gatsby Cloud being discontinued for both product and support? Is there any option for current Gatsby Cloud customers to remain on Gatsby Cloud after the final support date? If not, what are all options available?**

   a. Product and service support for Gatsby Cloud will be discontinued depending on the customer's Gatsby Cloud Plan type. See Question #2 above. Netlify will continue to provide support to all existing Gatsby Cloud customers until their end of service date, but no new account signups, new product updates or feature releases are planned for Gatsby Cloud as of today's date.

   At this time, there is not an option to keep data or operate websites and/or applications on Gatsby Cloud beyond the end of service date. All future product engineering and support services will be allocated to support the future product development and innovation of Netlify.

5. **What are the future plans for the Gatsby Framework? Will this receive continued support & updates after the Gatsby Cloud EOL?**

   a. The Gatsby open source framework is an important part of the modern front-end ecosystem and important to many of our customers. We're committed to maintaining it as an enterprise grade framework with focus on stability and support for large scale projects. To do this it's important to free up our Gatsby specific resources that's been spent on the closed source Gatsby Cloud product and keep the framework truly open.

   New Announcement (8/24/23): Gatsby Adapters Now Available!

   Netlify Tweet from 8/22/23: "Updates to React 18 and Gatsby Adapters are the most imminent updates, but we're also busy investing in platform primitives for a stronger Gatsby.js (and other frameworks too)."

   Additional context: With recent restructure to Netlify, we're still working through plans and resource allocation to support all projects. We are committed to supporting Gatsby and all frameworks, but our teams do need time to create long-term roadmaps given the recent org and team changes.

6. Can I get a trial period to test the Netlify platform and performance before migrating?

a. Netlify offers free Self-Serve accounts signups at https://app.netlify.com/signup.

However, if you have a **Gatsby Cloud Enterprise account** that would require access to Netlify Enterprise plans/features, it will require migration to a comparable Netlify Enterprise plan. Please contact your Gatsby or Netlify Account Manager to discuss.

7. What is the long-term roadmap and vision for Netlify's platform?

a. Netlify is the digital platform for enterprises to build highly-performant and dynamic websites, e-commerce stores, and web apps. If you and your team would like to see the Netlify Product Roadmap, please reach out to your Netlify Account Manager or Customer Success Manager.

8. What new features will I have access to once I migrate to Netlify?

a. **Unique Netlify Features**

1. **Unlimited branch deploys per-site**

In Gatsby Cloud if you wanted to have long-lived branches at a deployed URL, you needed 1 site per branch. In Netlify, a 'site' is more analogous to a repository, where you select the 'main' branch to show as your 'production' site, but can select from any number of branches in your repository to each get their own URL as well. TLDR; developer workflows are massively simplified in Netlify when compared to Gatsby Cloud or DIY builds + hosting.

Example Use Case: A website with "main", "staging", and "dev" branches that correspond to their environments. In Gatsby Cloud, a user would create 3 sites and need to copy + paste all env variables and site settings between the 3. With Netlify, you simply select each branch and they all build with the same settings + environment variables unless the user specifies differences between them.

2. **Collaborative Deploy Previews**

Use Netlify's Collaborative Deploy Previews across QA, Marketing, Product, Design, and Development teams to review updates, changes, and new features. This feature gives the entire team the ability to collaborate before pushing changes to production.

Example Use Case: A new promotions page is launched on an e-commerce site. The Marketing and Design teams are able to suggest updates to the layout. The QA team can test the user journey and spot bugs that need to be fixed before pushing the new page into production.

3. **Compatible with All Frameworks**

Gatsby Cloud was designed to build and host Gatsby.js sites. With Netlify, all types of sites and stacks can live on one platform, simplifying your workflow and mental model.

Example Use Case: Customer who has a Gatsby.js site for their product info, but the company's web application is built with Next.js. Netlify supports both frameworks and all web properties can live on a single platform.

4. **Netlify Enterprise Platform**

Netlify offers many enterprise-grade features for businesses of all sizes. Netlify features not only fast build times and a robust CDN for web hosting, but also the following features and products:

- [Fully featured CLI](#)
- [Platform integrations](#)
- [A/B testing](#)
- [Form captures](#)
- [Built-in serverless functions (Edge and Lambda)](#)
- [Enterprise Team Management](#)
- [Netlify Connect](#)
- [Stackbit](#)