| Section Metadata | |
| --- | --- |
| style | content |



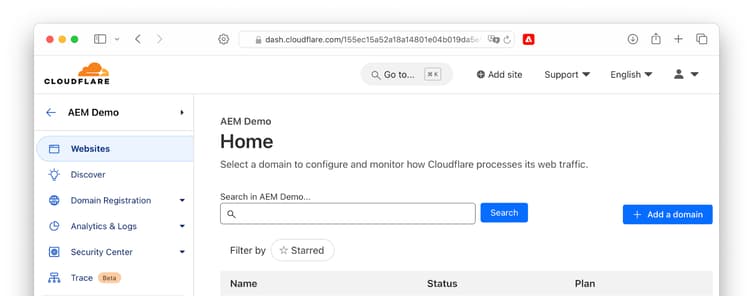
# Cloudflare Setup

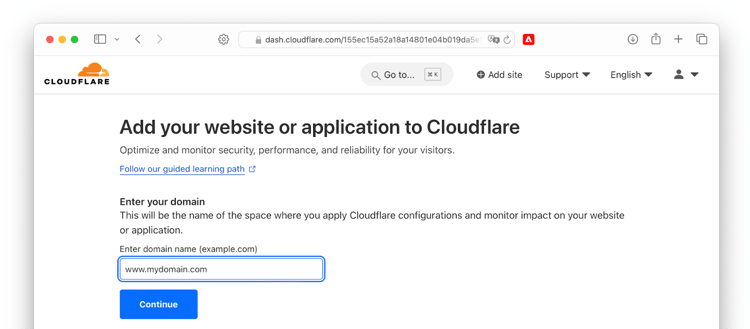
The following screenshots illustrate how to configure [Cloudflare](https://www.cloudflare.com/) to deliver content. Essential settings are marked with a red circle.

This setup can be completely done in the browser by using the Cloudflare Dashboard only. If you are already familiar with [Cloudflare Workers](https://workers.cloudflare.com/), [Wrangler](https://developers.cloudflare.com/workers/wrangler/) & GitHub and not afraid of entering commands in a terminal window you might want to follow the [instructions for Cloudflare with](https://www.hlx.live/docs/byo-cdn-cloudflare-worker-wrangler-setup) wrangler instead.

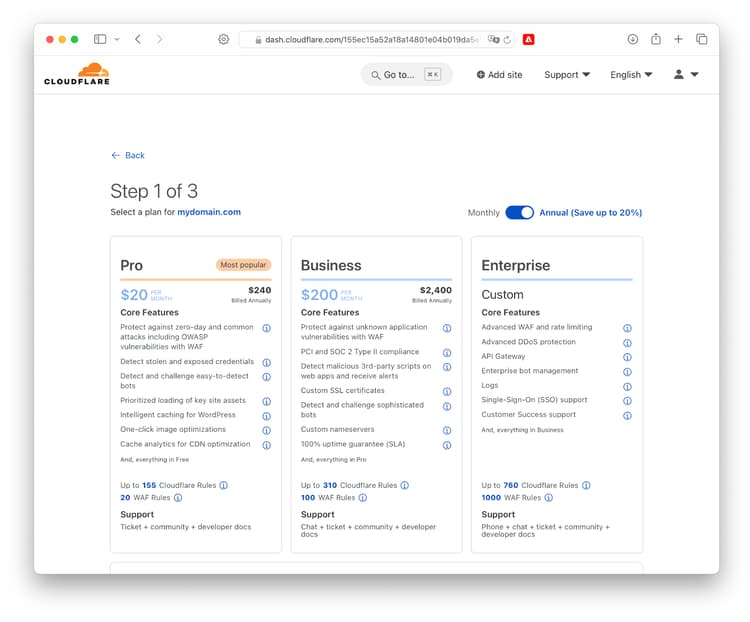
## Create a Cloudflare site

If you already have a Cloudflare site and DNS set up you can skip forward to the [Setup push invalidation](#setup-push-invalidation) section.



Enter the domain:  
  


Select a plan:

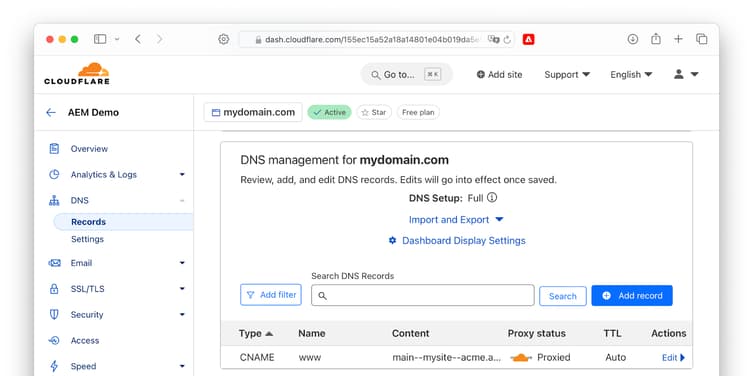


Note that surgical push invalidations are only supported on the **Enterprise** plan. On all other plans the entire cache of the site will be purged every time an author publishes a content change.  
  
For this walk-through we’ll use the [Free plan](https://www.cloudflare.com/plans/free/).

## DNS Setup

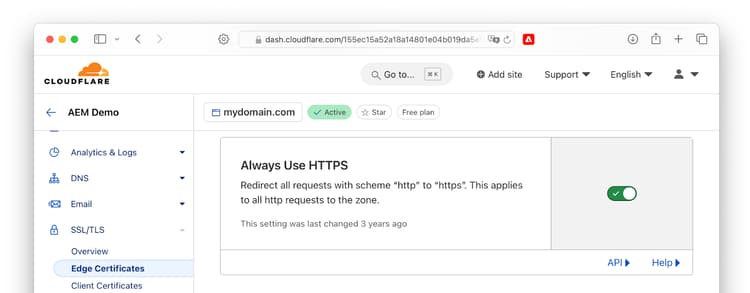
For a new site, we’ll start with a simple DNS setup.

1. Create a new CNAME record. If your zone is mysite.com and you want to serve traffic on www.mysite.com, then the name should be www
2. If you want to serve traffic on example.com (without a www) then the name should be @
3. And if you want to serve traffic on all subdomains, then the name should be \*
4. As we are using workers to serve the content, the value of the Content field does not matter. It’s easiest to use your ref--repo--owner.aem.live host name here. This is a hostname, not a URL, so leave out the leading https://

Make sure the CNAME record is [Proxied](https://developers.cloudflare.com/dns/manage-dns-records/reference/proxied-dns-records/):  
  


## SSL/TLS Setup

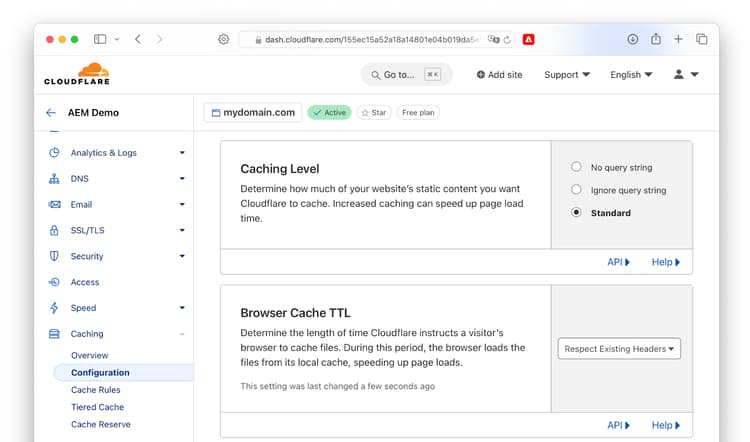
Select **SSL/TLS** from the left pane and **Edge Certificates** in the dropdown list. On the right side, scroll down to **Always Use HTTPS** and enable the setting:



## Configure Caching

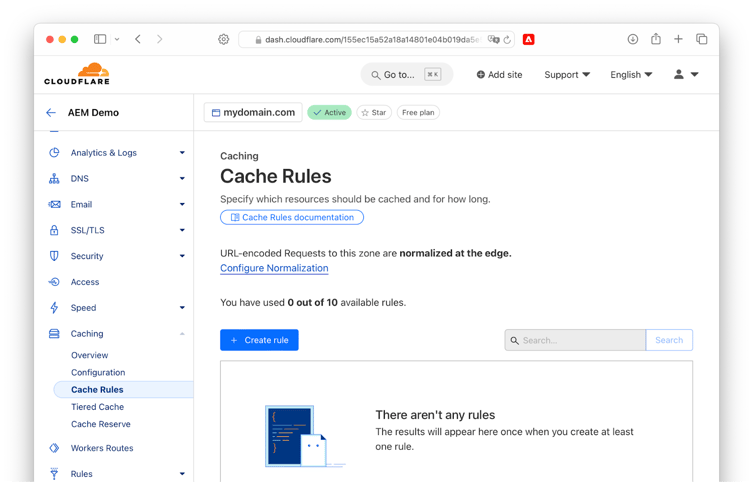
Navigate to **Caching → Configuration** and adjust following settings:

* Caching Level: Standard
* Browser Cache TTL: Respect Existing Headers

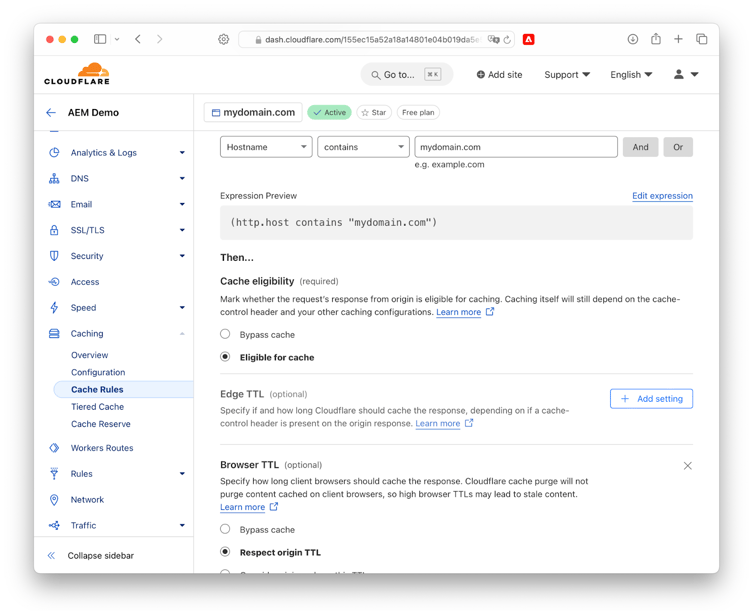


### Create Cache Rules

Next browse to **Caching → Cache Rules** and create a new cache rule



**Note: Enable Origin Cache Control option is only available for enterprise accounts**. Free, Pro, and Business customers have this option enabled by default and cannot disable it.



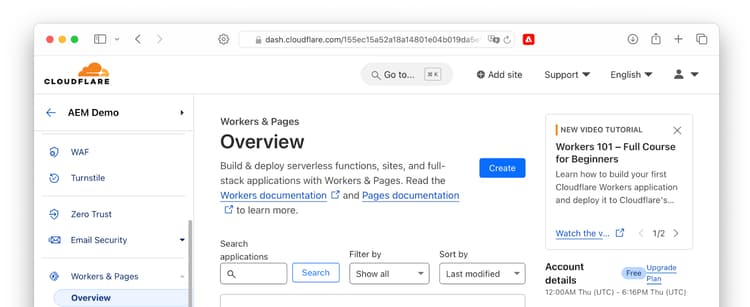
Following settings should apply:

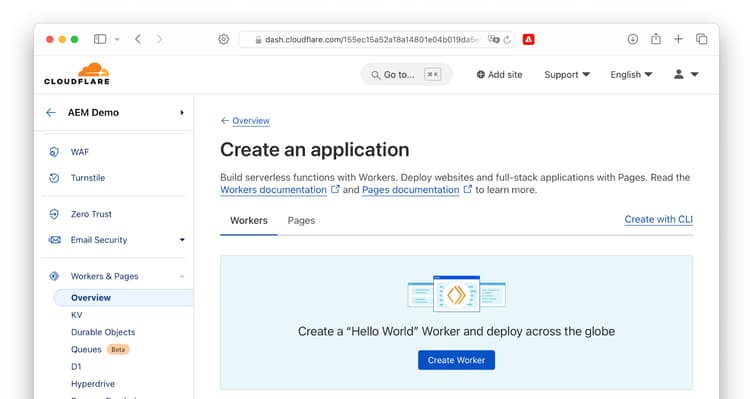
* When incoming requests match: hostname contains mydomain.com
* Then: Eligible for cache

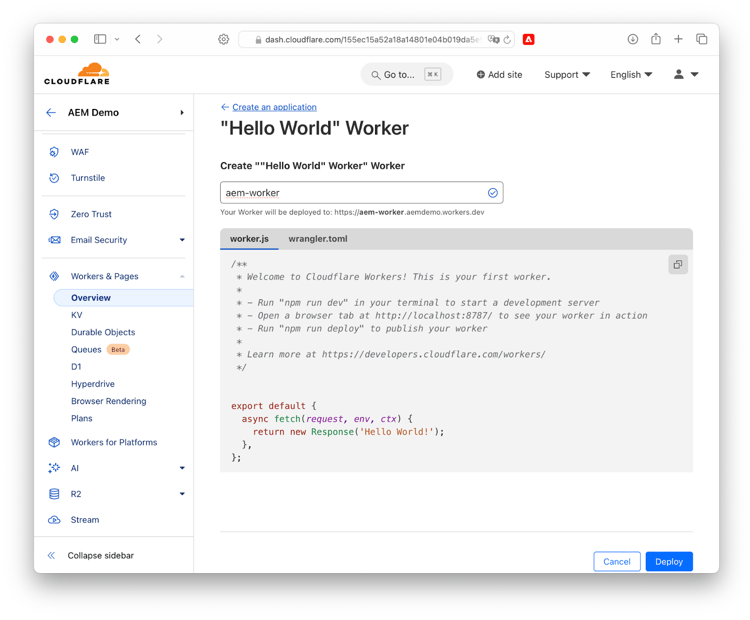
Under Browser TTL, click "add setting", then apply "Respect Origin TTL"

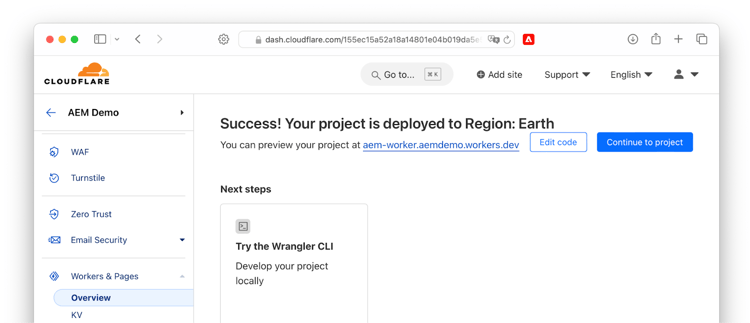
## Create Worker

Return to the Cloudflare Dashboard homepage, then navigate the sidebar: Workers & Pages → Overview. Click "create" to create a new worker.



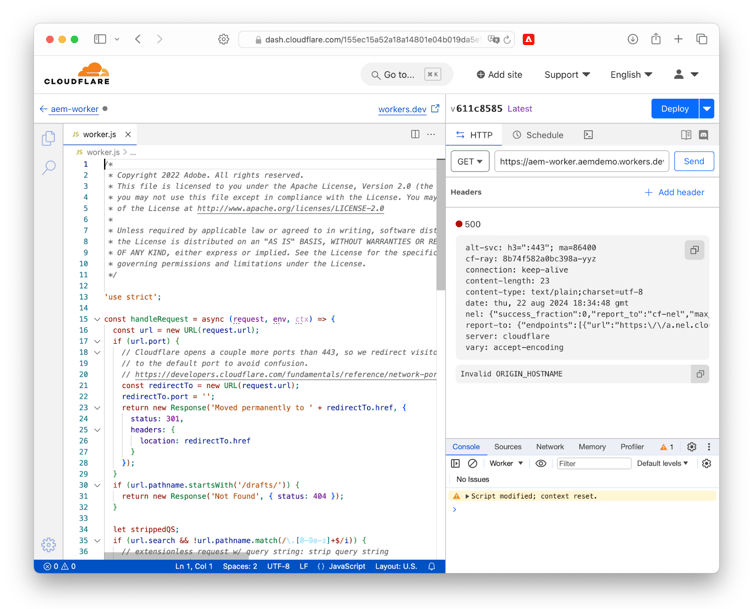
On the next screen, click "create worker", because that's what you want to achieve.  


Enter a name for the worker (e.g. aem-worker) and click on “Deploy”:  
  
  
  
On the next screen, click "Edit code"

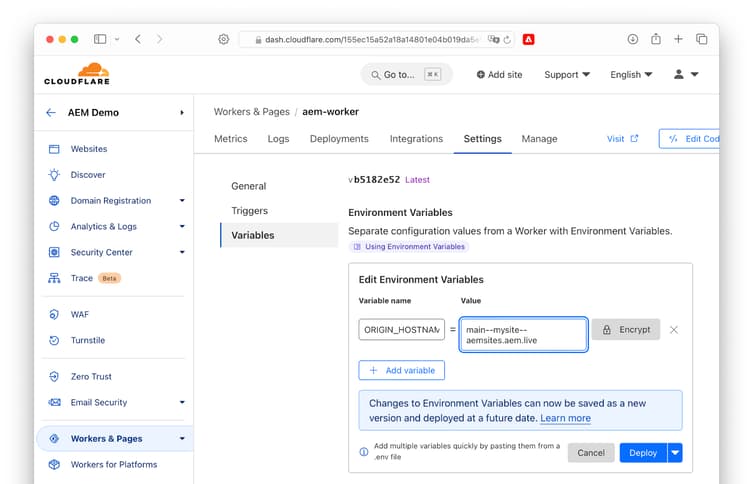


### Edit worker code

1. Copy the content of [this file](https://raw.githubusercontent.com/adobe/helix-cloudflare-prod-worker-template/main/src/index.mjs).
2. In the left pane, replace the existing content with the copied content.
3. Click on “Deploy”
4. Confirm with "Save and Deploy"

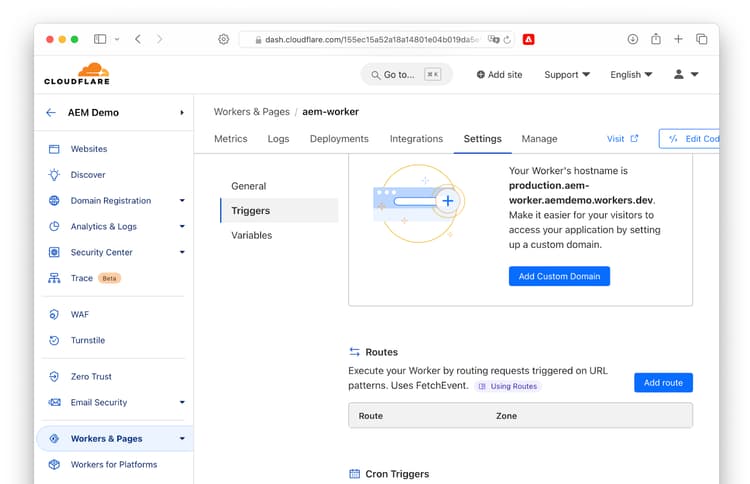
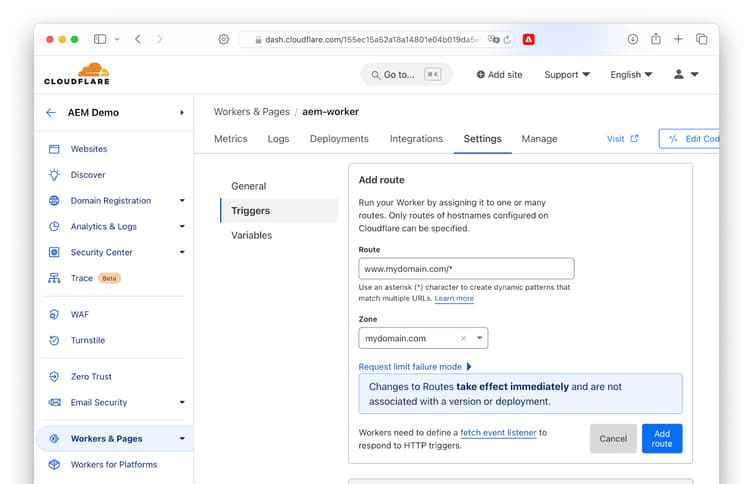


Return to your worker (by clicking the back arrow in the top right corner), then click on Settings → Variables and “Add variable”:  
  
Add a variable ORIGIN\_HOSTNAME and set the value to the hostname of your origin (e.g. main--mysite--aemsites.aem.live):



If you have enabled push invalidation, add a second environment variable PUSH\_INVALIDATION = enabled.

Apply the changes by clicking "Deploy".

Next, click on Triggers an select “Add route”:  
  
  
  
Enter your domain route (e.g. www.mydomain.com/\*), select your zone and confirm with “Add route”:  


Depending on the setup chosen in [DNS Setup](#_DNS_Setup), you would select routes www.mydomain.com/\* or mydomain.com/\*

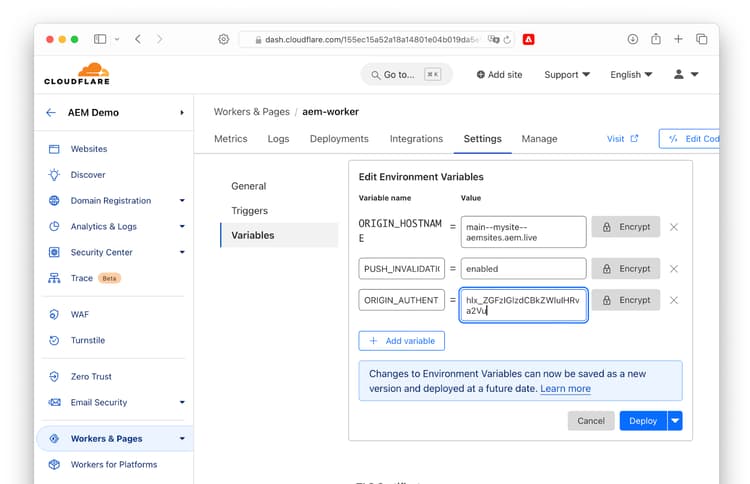
**Warning**: if you select \*.mydomain.com/\* as Cloudflare suggests in the field default, your site will be available under multiple subdomains. This will invite attackers trying to open webmail.mydomain.com and similar sites, and lead to duplicate content, potentially depressing your search engine rankings.  
  
After completing all steps you should be all set.

| Copy | |
| --- | --- |
| url | Route |

### Optional: Authenticate Origin Requests

When using [token-based Site Authentication](/docs/authentication-setup-site), add the following to enable authenticated requests from Cloudflare to AEM.

1. Return to Workers → <your worker> → Settings → Variables
2. Create a new environment variable ORIGIN\_AUTHENTICATION
3. Paste the site token value from [token-based Site Authentication](/docs/authentication-setup-site) (it starts with hlx\_)
4. Confirm by clicking "Deploy"



### Expanding the AEM footprint on your website

In case you start with having only a portion of the website being routed to your .live origin and have routed a specific folder (eg. /blog/\*) you can subsequently add more routes whenever you are ready to expose new sections of the site by simply adding more routes and repeat the last “add route” steps as needed, without changing your worker configuration.

### Watch out for duplicate content

Search engines often penalize sites for duplicate content, so it's important to make sure your content is not available on the web elsewhere. Cloudflare, unfortunately, has a default setting that will [expose your site on additional network ports](https://developers.cloudflare.com/fundamentals/reference/network-ports/#network-ports-compatible-with-cloudflares-proxy). In paid Cloudflare plans you can [block traffic on these additional ports](https://developers.cloudflare.com/fundamentals/reference/network-ports/#how-to-block-traffic-on-additional-ports). This is a recommended setting for production sites.

| Metadata | |
| --- | --- |
| name | copy |