



What does Cloudflare do?

The biggest internet company you've never heard of

 Justin
Jul 20, 2021

1 4 2 3

The TL;DR

Cloudflare provides **networking tools** that help companies distribute their apps globally in a **secure way**.

- To make sure your apps and sites are fast no matter where your users are, developers put copies of their site in data centers across the globe, called a **CDN**
- Cloudflare provides a giant, super fast CDN with **built in security features** to prevent attacks on your site
- Over time, they've added **adjacent products** like serverless workers, load balancing, video streaming, and serverless storage

Cloudflare IPO'd back in 2019, and (at the time of writing) has a market cap of \$30B - so they're doing something right. Let's figure out what it is!

The basics: a Content Delivery Network

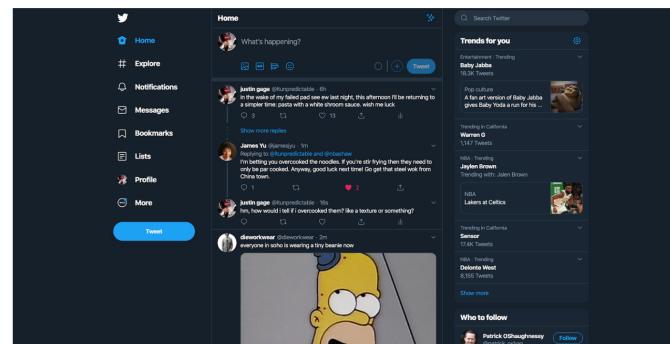
Every website or app can basically be broken down into two pieces:

1. **Static assets** - things that don't change
2. **Dynamic assets** - things that do change

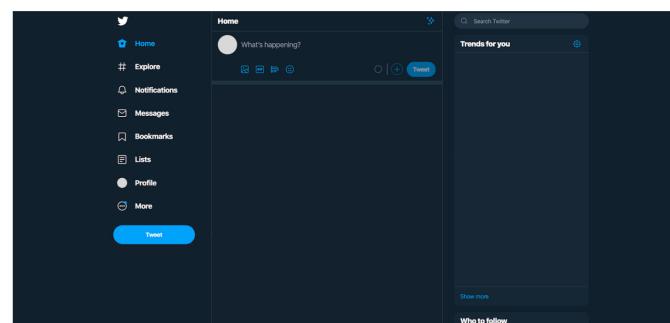
Static assets tend to include things like images (logos, product images, etc.), fonts, and the skeletons for most pages on the site - these things change very infrequently, hence the "static" denomination.

Dynamic assets, on the other hand, are things that are constantly changing - and usually differ from user to user. To quote from the [What's a Web App](#) post:

Think about your Twitter homepage:



A lot of things here don't change much. The order and the text of the tabs on the left, the logo, the search bar – all of this is the same for everyone. But what shows up in the feed, the little picture of me next to "profile" on the left, and the trending tweets on the right – those are all dynamic. If I had to guess, this is what the Twitter page looks like before the dynamic content gets put in:



Once you log in, Twitter adds your profile photo (they stored it in their database) to the two little gray circles, and populates your feed with the right tweets and trending topics based on what they think you'll want.

The separation between static assets and dynamic assets is *very important* - because it dictates **how fast** you can make your site or app. Static assets are cheap and easy to distribute - you can put copies on servers across the world, and when a user visits the site, send them the assets from whatever server is closest to them - be it in Singapore or Virginia. Dynamic assets, on the other hand, are *much harder* to distribute geographically, and tend to sit on one or a few centralized servers. Dynamic content is backed by an API, which probably sits on top of a database, i.e. it's much more complex than a bunch of HTML or CSS.

🔍 Deeper Look 🔎

The fact that content can be dynamic doesn't necessarily mean it gets *generated* away from the server - different types of apps can use *server side rendering* or *client side rendering*. The debate goes back and forth, but server side rendering is coming back into vogue.

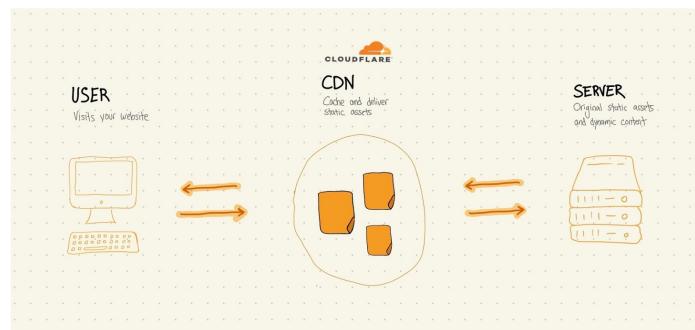
🔍 Deeper Look 🔎

This is where the **Content Delivery Network (CDN)** comes in. It's a network of geographically diverse servers across the world that you can store your static assets to instead of building data centers yourself. If that sounds like public cloud, that's because it is - AWS has their own CDN product ([Cloudfront](#)), as well as GCP and Azure. You just put your static assets there, and they take care of serving users from the data center that's closest to them. Which means faster apps.

The core Cloudflare product: CDN

Cloudflare's core product is just that - a big, fast CDN. The kicker, though, is that they implement it in a kind of funky way - via a middleman tactic.

Instead of giving Cloudflare direct *access* to your static assets - like you might do with Netlify or Vercel - you use them as a DNS, or a Domain Name System. That means that you trust them to route your *domain* - something like technically.dev - to your actual servers. The first time someone makes a request from their browser to your site, Cloudflare acts as a middleman, and *forwards* that request to your servers. They then *cache* the response (i.e. the static files) in their network, replicate it across their hundreds of data centers, and deliver that the next time a user asks for it.



In layman's terms (as we are all laypeople), it's a bit like a franchise. The head office for McDonalds (the server) decides what the branding and product suite looks like, and all of the franchises (the servers on the CDN) copy that. When the head office wants to make a change, like adding the McRib (please god), all of the individual stores need to take down their menus and update them. That's what it means to **invalidate the cache** - it tells Cloudflare's servers that the static assets have changed, and they need to re-request them.



Checking your browser before accessing

This process is automatic. Your browser will redirect to your requested content shortly.

If you've ever seen the above screen, you've used Cloudflare. While the *core* responsibility of Cloudflare's CDN is to serve static assets to users really quickly, they've also added a few killer features on top that make the product more compelling - the basis of which is **security**. Cloudflare's CDN protects your site against **DDoS**, or Distributed Denial of Service attacks, where attackers try to take down your site by hitting it with a ton of requests. This is a notoriously high profile method of hacking, and Cloudflare even has a list of [newsworthy DDoS attacks on their site](#).

The Cloudflare product suite

Aside from a DDoS-protected CDN, Cloudflare offers a *lot* of other stuff in the networking space. There's too much to cover here, so we'll tackle a few illustrative examples:

→ Cloudflare Workers

[Cloudflare Workers](#) let you go beyond *static* assets, and write *scripts* (think: Javascript, C+, etc.) that run on Cloudflare's distributed network. This is similar to [AWS's Lambda](#), and represents a lot of promise for having developers skip infrastructure entirely and just worry about application code. As of recently, Cloudflare also supports [storing persistent data via a key-value store](#).

The screenshot shows the Cloudflare Workers dashboard. At the top, there are tabs for 'Workers', 'KV', and 'Plans'. Below the tabs, there is a search bar and a 'Create a Worker' button. A list of workers is displayed in a table format:

Worker Name	Last Update	Actions
demo-hello-world	2 days ago	...
rust-app	10 days ago	...
contacts-api	10 days ago	...
colo	15 days ago	...
[REDACTED]	2 months ago	...
web-counter	2 months ago	...
svelte	2 months ago	...

On the right side of the dashboard, there are several stats and links:

- 1 Dec - 31 Dec
- Requests this month: **1,894,726** (Unlimited)
- Your subdomain: [\[REDACTED\].workers.dev](#)
- Account ID: [\[REDACTED\]](#) (Click to copy)
- Documentation: [Workers Documentation](#), [Template Gallery](#), [Quick Start Guide](#)
- CLI and API: [CLI Quick Start Guide](#), [API Reference](#), [Get your API token](#)
- Get in touch: [\[REDACTED\]](#)

→ Video Streaming

[Cloudflare Stream Delivery](#) lets you stream video to anywhere in the world quickly, using Cloudflare's CDN. This is actually a pretty difficult problem, especially if you remember the early days of [HQ Trivia](#).

→ Rate Limiting

The same technology that Cloudflare uses on their CDN to block DDoS attacks can also be used (at a price, of course) to [prevent other types of attacks](#), and in general be specific about who can access your site.

→ Load Balancing

If your website has too much traffic for one server to handle (👉), you might have multiple. Load balancers act as an intermediary between your user and your servers - every request hits the Load Balancer, and *it* decides which server to send the request to, usually based on a round robin sort of thing. Cloudflare [provides this as a service](#), as does AWS and co.

Few of these products are super compelling in themselves, but aggregated together in the same UI and billing platform, they start to make sense as an ecosystem.

Cloudflare is in an interesting spot, strategically. I'd generally consider them a *boring* company - everything in networking is boring, of course - but they've found themselves with a product suite that's actually pretty compelling. This quote from @sperand_io sums it up well:

 **Chris Sperandio**
@sperand_io

The fact that this question is impossible to answer is the most interesting thing

 **Patrick OShaughnessy** @patrick_oshag

What is the most interesting thing about @Cloudflare (business or product)?

November 20th 2020

3 Likes

Nothing about Cloudflare is particularly exciting by itself, but they've set themselves up in a position where they own distribution (i.e. the CDN), and so adding serverless (Cloudflare Workers) on top actually worked pretty well (and the product is popular, from my experience with engineering teams). Keep an eye out.

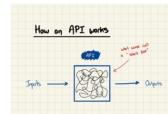
Further reading

- [The Cloudflare S-1](#) has good contextual information on how they view the business, and more financial info
- Cloudflare has an [entire learning section on their site](#) dedicated to explaining what they do!

 4  Comment  Share

 Write a comment...

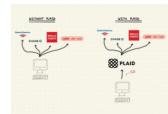
[Top](#) [New](#) [Community](#) 



What's an API?

What McDonalds and Lyft have in common

Justin Jan 9, 2020  187  4



What does Plaid do?

Technically begrudgingly tackles Fintech

Justin Jan 14, 2021  34  3 



What does New Relic do?

Keeping an eye on your servers and apps

Justin Jan 11  23  2 



What's Reverse ETL?

Getting your data OUT of your warehouse?

Justin Feb 1  19  4 



What happened to Facebook?

A basic explainer of what that outage was all about

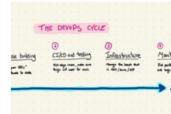
Justin Oct 5, 2021  29  4 



What does GitLab do?

The TL;DR GitLab is a somewhat contrarian take on DevOps: it's basically one giant tool for literally anything you'd want to do relating to building and...

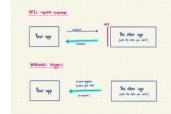
Justin Jan 4  8  4 



What's DevOps?

IT has a cool new name

Justin Jan 5, 2021 ❤ 34 ⚙ ↗



What are webhooks?

Triggered

Justin Sep 13, 2021 ❤ 28 ⚙ 5 ↗



What's Headless E-Commerce?

We may be running out of names

Justin Nov 2, 2021 ❤ 20 ⚙ 7 ↗



I built a (basic) Substack clone in a month

This was probably a waste of my time

Justin Sep 2, 2020 ❤ 50 ⚙ 4 ↗

[See all >](#)

© 2022 Justin · [Privacy](#) · [Terms](#) · [Collection notice](#)

 [Publish on Substack](#)

Our use of cookies

We use necessary cookies to make our site work. We also set performance and functionality cookies that help us make improvements by measuring traffic on our site. For more detailed information about the cookies we use, please see our [privacy policy](#).