Every basic SEO guide mentions [site speed](https://trinity.one/insights/web-development/site-speed/). You need to have a fast website, your users are impatient, your rankings will tumble, etc. While the importance of site speed is addressed almost constantly, few guides take the extra step of going into what improving site speed involves – or even what factors go into play when making your site faster.

If you need to improve your site speed to boost your SEO, start with looking at your **time to first byte (TTFB)**. This metric significantly affects your rankings and user experience. It can also be one of the hardest to change if you don’t know what you’re doing. Fortunately, we’re here to help.

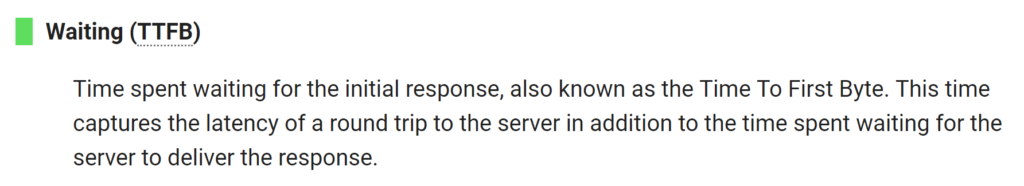
**What is Time to First Byte (TTFB)?**

TTFB is the time it takes to request information from the server and send the information that was requested. (In layman’s terms, it is the time between when you navigate to a webpage and when it starts to render.) This time period includes:

* The server request, which can vary based on location and internet connection
* The time it takes to process a request, or generate a response
* The time it takes to send information based on the question.

Return time accounts for 40% of total TTFB. The slower the TTFB, the longer it takes for your users to see any content on your site.

Here is [Google’s official definition](https://developers.google.com/web/tools/chrome-devtools/network/understanding-resource-timing) of TTFB:



Google heavily weighs TTFB in search rankings, and this metric is considerably different from your page load speed. In fact, many SEO professionals have seen how Google places significantly more value on TTFB than on page speed.

Like most elements of SEO, TTFB ties closely to the user experience. Users that have to wait for your page to load (with no actual indication that the site is live and intending to load in the next few seconds) and are more likely to bounce because of a bad experience. The more you frustrate your users, the more Google will devalue your site.

**What is a Good TTFB?**

According to SearchEnginePeople and Google, your TTFB needs to be less than 200 milliseconds (ms). This number also differs by the type of content on your page. Static content should load at 100ms while dynamic content should load at a speed of 200 – 500ms.

The 500ms mark is the maximum amount for both Google and your users to tolerate – especially because the rest of the page still needs to load after the first byte hits.

**How to Check Your TTFB**

There are several resources at your disposal to check your time to first byte and monitor your speed for various pages and content types:

* [Bytecheck](https://www.bytecheck.com/) is one of the most straightforward tools. It gives you a clear TTFB report and also shows other elements and data points that you might be interested in. Bytecheck will also rate your site out of five stars.
* [WebPageTest](https://www.webpagetest.org/) offers more options. You can choose to run the test by location (like Chicago, Sydney, or Berlin) and by device. You can also run the test on different browsers to see if your TTFB changes dramatically between Chrome, Mozilla, etc.
* [KeyCDN](https://tools.keycdn.com/performance) is an overall performance tester that checks several different elements related to your site health. You can sign up for a free trial or look at their various plans if you’re testing multiple sites over time.

These are just a few of the options that are available to you to monitor your TTFB. You may find one on this list that you love, or need to keep looking to find a tool that meets your needs.

**How to Improve Your TTFB**

As we said earlier, understanding and tracking your time to first byte is the easy part. Improving your TTFB tends to be more complicated, especially as there are multiple factors involved – and many of them are out of your control. Within WordPress, factors that contribute to TTFB include:

* **Network latency:** communication delays within the network.
* **High web traffic:** the demand on the servers to pull information.
* **Server configuration:** the type of servers pulling information and their performance.
* **DNS response time:** the time it takes for the server to recognize your domain name and translate it to an IP address.
* **Dynamic content:** blog posts, videos, and updates to your website that are added frequently.

As you can see, elements like high web traffic are mostly out of your control. Additionally, you’re not going to stop creating dynamic content just because it has lower TTFB metrics. However, a few of these elements are in your control, and you can take steps to improve your TTFB even if you don’t consider yourself very tech-savvy.

A few best practices to keep in mind to improve your TTFB include:

* **Choose a fast web host:** choosing a fast host for your site takes the burden off of you to have a fast TTFB. This is a great option for small businesses that just want a basic website.
* **Keep your plug-ins and themes updated:** old plug-ins and themes are clunky and slow. Many developers include performance improvements to their updates, so your TTFB can keep improving over time.
* **Use a CDN:** a CDN (content delivery network) uses global servers to deliver static content faster. This reduces network latency because users are getting the content from a server that is closer to them. A CDN is particularly useful for [eCommerce sites](https://trinity.one/insights/user-experience/best-practices-ecommerce-homepage-ux/) or brands that receive traffic from large geographic areas.
* **Find a Premium DNS services:** when you choose your hosting service, see if you can upgrade to a premium DNS service if you need to. This is typically an add-on for many hosting options. If your other TTFB improvements aren’t driving the results you need, you can upgrade to a premium DNS service for your pages.

There are many elements that factor into TTFB – and that’s a good thing. It means that there are multiple strings that you can pull to improve it. You can test a few of these options separately to see which ones have the biggest impact and then compound them onto each other.

**Improve Your Site Speed and Other SEO Factors**

Your time to first byte isn’t a silver bullet solution to fix your rankings and user experience. This is one element out of several that Google takes into consideration when interacting with your brand. If you want to see how your site stacks up, check out our [free SEO analysis](https://trinity.one/seo/). You can input your website and receive a report on your site strengths, opportunities, and weaknesses. One of our team members is also happy to walk through your report with you.

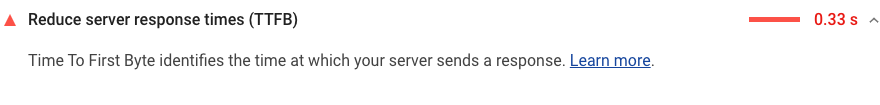
SEO is a process. You can’t expect to have high rankings overnight and to keep them there forever. Work on your TTFB and other SEO factors to stay on top and drive new customers to your brand.

# Reduce server response times (TTFB)

May 2, 2019 • Updated Oct 4, 2019

Appears in: [Performance audits](https://web.dev/lighthouse-performance)

The Opportunities section of your Lighthouse report reports Time to First Byte, the time that it takes for a user's browser to receive the first byte of page content:



## Slow server response times affect performance [#](https://web.dev/time-to-first-byte/#slow-server-response-times-affect-performance)

This audit fails when the browser waits more than 600 ms for the server to respond to the main document request. Users dislike when pages take a long time to load. Slow server response times are one possible cause for long page loads.

When users navigate to a URL in their web browser, the browser makes a network request to fetch that content. Your server receives the request and returns the page content.

The server may need to do a lot of work in order to return a page with all of the content that users want. For example, if users are looking at their order history, the server needs to fetch each user's history from a database, and then insert that content into the page.

Optimizing the server to do work like this as quickly as possible is one way to reduce the time that users spend waiting for pages to load.

## How to improve server response times [#](https://web.dev/time-to-first-byte/#how-to-improve-server-response-times)

The first step to improving server response times is to identify the core conceptual tasks that your server must complete in order to return page content, and then measure how long each of these tasks takes. Once you've identified the longest tasks, search for ways to speed them up.

There are many possible causes of slow server responses, and therefore many possible ways to improve:

* Optimize the server's application logic to prepare pages faster. If you use a server framework, the framework may have recommendations on how to do this.
* Optimize how your server queries databases, or migrate to faster database systems.
* Upgrade your server hardware to have more memory or CPU.

## Stack-specific guidance [#](https://web.dev/time-to-first-byte/#stack-specific-guidance)

### Drupal [#](https://web.dev/time-to-first-byte/#drupal)

Themes, modules, and server specifications all contribute to server response time. Consider finding a more optimized theme, carefully selecting an optimization module, or upgrading your server. Your hosting servers should make use of PHP opcode caching, memory caching systems like memcached or Redis to reduce database query times, as well as optimized application logic to prepare pages faster.

### Magento [#](https://web.dev/time-to-first-byte/#magento)

Use Magento's [Varnish integration](https://devdocs.magento.com/guides/v2.3/config-guide/varnish/config-varnish.html).

### React [#](https://web.dev/time-to-first-byte/#react)

If you are server-side rendering any React components, consider using [renderToNodeStream()](https://reactjs.org/docs/react-dom-server.html#rendertonodestream) or renderToStaticNodeStream() to allow the client to receive and hydrate different parts of the markup instead of all at once.

### WordPress [#](https://web.dev/time-to-first-byte/#wordpress)

Themes, plugins, and server specifications all contribute to server response time. Consider finding a more optimized theme, carefully selecting an optimization plugin, and/or upgrading your server

## What affects time to first byte?

**TTFB is impacted by three key actions:** 1) sending a request from a client machine to the server, 2) processing that request on the server and generating a response, and 3) sending the response from the server to the client.

### Action 1: Sending a request to the server

Measuring TTFB begins with the request. The time it takes for a server to receive a request can vary based on the time it takes to perform a DNS lookup, the speed of the user’s network, the distance to the server, and any interruptions in the connection. Enterprises have no control over the link between the user and the Internet, but any delays will still impact their TTFB.

### Action 2: Processing and generating the response

Once a server receives a request, it has to generate a response. This involves starting processes, making database calls, running web scripts and communicating with other systems on the network. Common strategies used by enterprises to reduce TTFB include [caching web pages](https://blog.stackpath.com/glossary-content-caching/), optimizing server-side code, and improving hardware resources.

### Action 3: Sending the response back to the client

Once a server generates a response, it needs to transmit it back to the user. This step is dependent on both the enterprise’s connection speed and the user’s connection speed. The TTFB is determined the moment the client begins receiving the response, literally when the client receives the first byte. Transmitting a request and a response over a network can account for [almost 40% of the TTFB](http://www.webpagetest.org/forums/showthread.php?tid=557&pid=2671#pid2671).

## Example of time to first byte

As part of their website optimization project, spy camera store My-Spycam sought to reduce their TTFB from over 4 seconds to 2. Their strategy involved separating the areas of a page that changed on a per-user basis, while caching everything else on a CDN. My-Spycam modified Magento, their underlying eCommerce platform, to cache everything except for specific pages and page contents (such as a user’s shopping cart).

By adding dynamic caching, My-Spycam reduced their TTFB from 1162ms to 152ms. This strategy of “hole-punching” content on a cached page resulted in dynamic pages that load faster than static pages, without impacting the user experience.

## Benefits of optimizing time to first byte

Optimizing TTFB benefits both users and content providers.

* **Users see an improved browsing experience** since they have to spend less time waiting for a web service to generate a response.
* **Enterprises see higher customer engagement and retention** as users are less likely to leave due to delays or slow loading times.

## Conclusion

The time it takes to load a web page has an enormous impact on customer retention. When [40% of users](https://blog.kissmetrics.com/loading-time/) abandon a website that takes more than 3 seconds to load, having a low TTFB becomes essential. It not only reduces the chance of losing a customer, but ensures users have a fast and engaging online experience.

**What Is Considered A Good Time To First Byte?**

According to [Google](https://developers.google.com/speed/docs/insights/Server#overview), your website’s TTFB should be 200 milliseconds or less.

If your site is much slower than that, such as 500 milliseconds or more, it needs some work.

[Average Time to First Byte is 100-500ms. Google says to aim for 200ms](https://twitter.com/share?text=Average+Time+to+First+Byte+is+100-500ms.+Google+says+to+aim+for+200ms&via=senginepeople&related=senginepeople&url=https://www.searchenginepeople.com/blog/16081-time-to-first-byte-seo.html)

[Click To Tweet](https://twitter.com/share?text=Average+Time+to+First+Byte+is+100-500ms.+Google+says+to+aim+for+200ms&via=senginepeople&related=senginepeople&url=https://www.searchenginepeople.com/blog/16081-time-to-first-byte-seo.html)

[Typical good TTFB speeds](https://www.rackaid.com/blog/time-to-first-byte/) are:

* 100ms for static content (content that already exists on the sever as files)
* 200-500ms for dynamic content (content that is put together from a database and templates, like WordPress does, for example)

**Testing Your Website’s TTFB**

While not the only way to test TTFB, WebPageTest is one of the most reliable and trusted tools.

1. Enter your website URL at [WebPageTest](http://www.webpagetest.org/).
2. If you’re in the US, choose a “Test Location” in North America.
3. In the “Browser” dropdown menu, choose the web browser you’re using.
4. Select the “Start Test” button.

WebPageTest is   
one   
**of** the   
**best tools** to   
test TTFBWebPageTest will give your website a letter grade for TTFB, among other things. It also provides detailed analytics.

WebPageTest "uses the socket connect time for the base page as an estimate for the round-trip time to the server. It then takes the DNS time, socket connect time, SSL time and adds on one additional round trip as it's estimate for the fastest possible TTFB and then allows for 100ms on top of that" [before grading you down](http://www.webpagetest.org/forums/showthread.php?tid=11441&pid=18155#pid18155).

**What Causes Slower TTFB?**

Four causes of slow TTFB are:

1. High web traffic
2. Network problems
3. Dynamic content:
   * disk usage
   * disk speed
   * RAM usage
   * database setup
   * database speed
4. Server configuration:
   * PHP/ASP settings
   * database settings
   * shared server?
   * etc.

There isn’t much you can do to solve high amounts of website traffic or network issues. There are, however, ways to improve slow TTFB caused by dynamic content and server configuration.

**What Is Dynamic Content, And How Does It Slow TTFB?**

Dynamic content (which is commonly used by WordPress) is one of the biggest causes of slower TTFB. Most pages on the web as we know it are dynamically served.



Unlike static webpages that are sent to a web browser instantly upon request, dynamic web pages need to be assembled each time a user requests the webpage. This assembling process is what slows TTFB.

Dynamic webpages are usually put together from content stored in a database and template files stored on the webserver's hard disk. Usually the images and other additional resources like CSS and JavaScript also comes from files stored on the hard disk. Some systems store *everything* in the database.

The time to look find and get the relevant content from the database, get the files, and put everything together can easily form a bottleneck, especially if the server has to do so for multiple visitors at the same time.

**Improving Your Website’s TTFB By Caching**

Significant improvements to TTFB occur by caching your dynamic pages. Caching your pages make the pages “pre-built,” with HTML ready to go as soon as a browser requests the pages. Here are a few ways you can cache:

* **WordPress Plugins.** Plugins such as [WP Super Cache](https://wordpress.org/plugins/wp-super-cache/) will help you cache your dynamic pages. The plugin keeps tracks of when a "saved" dynamic page should be renewed so that the freshest copy possible is served. (See also: [A Complete Guide To Using W3 Total Cache](http://www.searchenginepeople.com/blog/15103-w3-total-cache.html))
* **WordPress hosting that caches automatically.** WordPress hosting can give you caching capabilities that improve your site’s TTFB.
* **Server Configuration.** A common way to cache files -- but not dynamic content -- is to configure the .htaccess web file. (.htaccess is short for hypertext access.) Beware that minor errors in this file can lead to major problems; hiring a professional is advised.

**Conclusion**

When improving TTFB, there are many details to consider. Those who follow through and take measures to improve their TTFB enjoy competitive advantages. Just like SEO and other labor intensive website initiatives that give consumers a better experience, it’s all worth it at the end of the day.

## 4 Ways to Reduce TTFB on Your WordPress Site

Now let’s dive into some ways on how to reduce the TTFB on your WordPress site.

### 1. Utilize a Fast WordPress Host

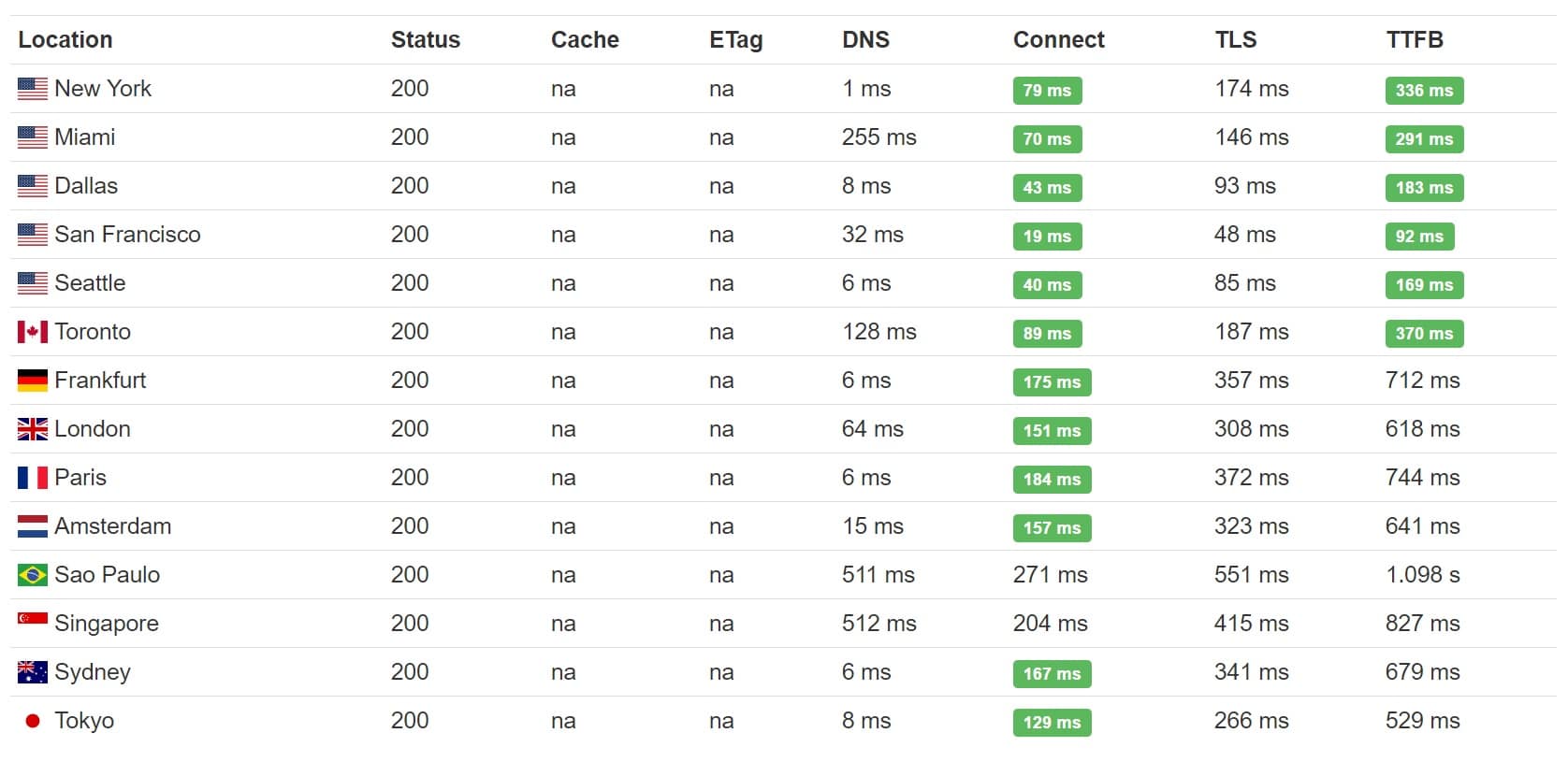
The first way to reduce TTFB is to ensure you are using a fast WordPress host. We compared a 3rd party shared host’s TTFB (located in Phoenix, AZ) and Kinsta’s TTFB (located in Council Bluffs, Iowa). We utilized the exact same setup with the default Twenty Seventeen theme running. Remember that Kinsta now has all 29 [Google Cloud Platform locations](https://kinsta.com/knowledgebase/google-cloud-data-center-locations/) available, so it is important to strategically place your WordPress site closer to your visitors.

Switching to a faster host could decrease your site’s TTFB by up to 200%. [Try Kinsta for Free](https://hubs.ly/H0pklC_0).

Kinsta also **includes Google Cloud Platform’s premium tier network** on all hosting plans. A lot of other hosting providers use Google Cloud’s standard tier network, which results in slower speeds.

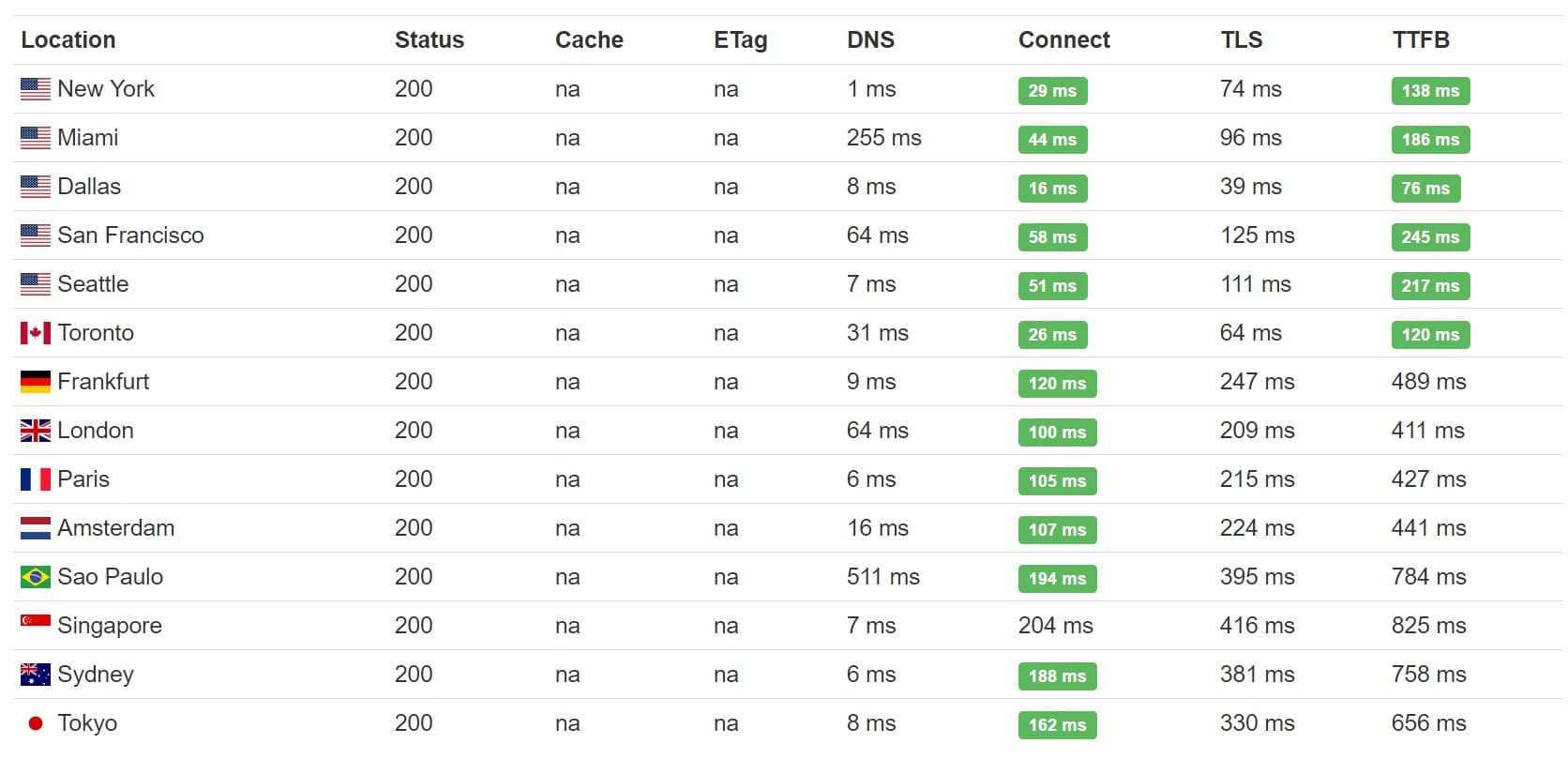
**Shared Host TTFB**

Across all regions, the **average TTFB was 520 ms**. Across the United States and Canada, the **average TTFB was 240 ms**.

Shared hosting TTFB

**Kinsta TTFB**

Across all regions, the **average TTFB was 412 ms**. Across the United States and Canada, the **average TTFB was 164 ms**. If you host with Kinsta, you can also choose to host your WordPress site in Europe or Asia. See the list of [Google Cloud Data Center Locations](https://kinsta.com/knowledgebase/google-cloud-data-center-locations/).

Managed WordPress hosting TTFB

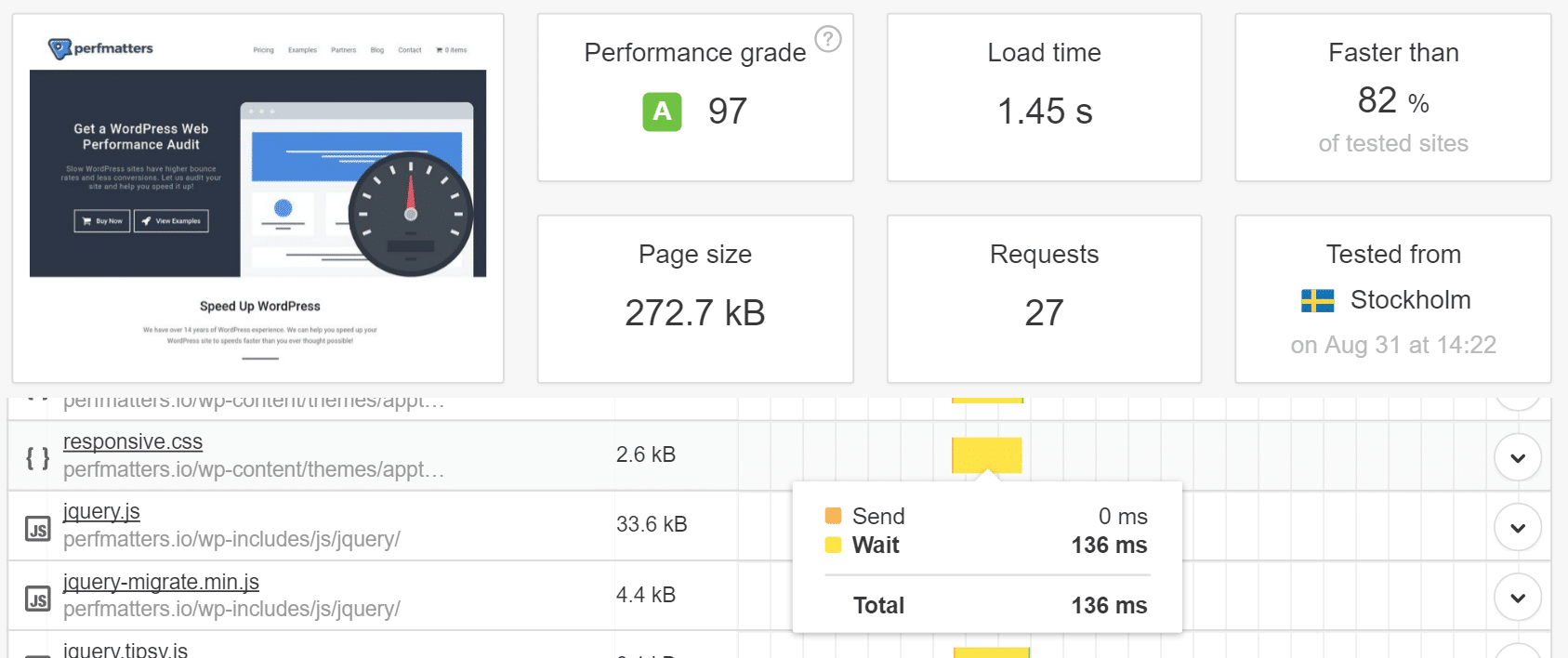
So simply by using a faster host, we saw a **20% decrease in TTFB** globally. And a **32% decrease in TTFB** across the United States and Canada. Having a good WordPress host with a carefully thought out architecture is crucial to lowering your TTFB. This also makes a good case for carefully choosing a place physically located in a region where your customers are. If most of your customers are in the United States, don’t host your server in Europe (although a CDN can help negate some of that).

### 2. Implement a CDN

Another easy way to decrease TTFB is to utilize a [Content Delivery Network](https://kinsta.com/blog/wordpress-cdn/) (CDN). If you have a website that is serving visitors in different parts of the country, or around the globe, this can drastically decrease your TTFB. As we saw above, location is very important. We ran a little test to show the difference with KeyCDN as our CDN provider. Each test was run 5 times and the average was taken.

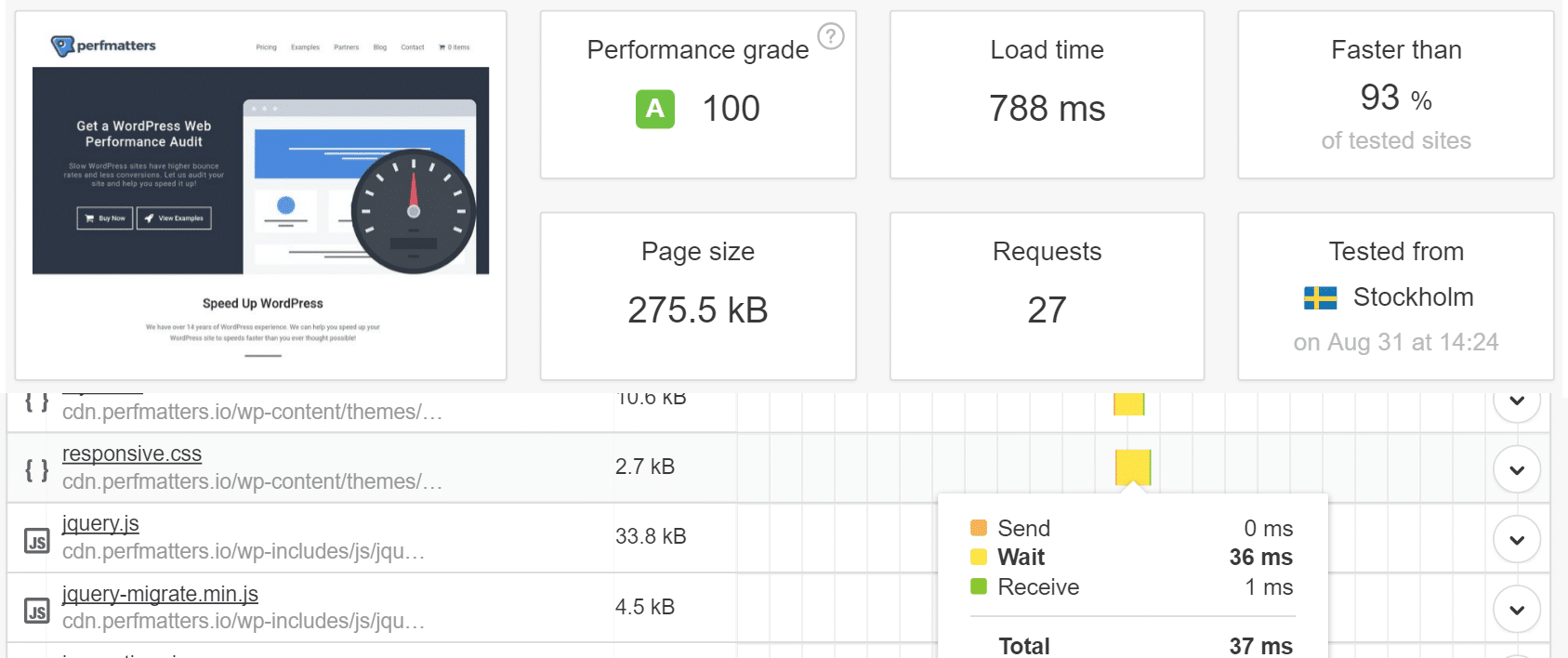
**TTFB Without CDN**

We first [ran a test](https://tools.pingdom.com/#!/i7o38/https://perfmatters.io) with our CDN disabled and as you can see our total load time was 1.45 s and our average TTFB on an asset was around 136 ms.

TTFB before adding a CDN

**TTFB With CDN**

We then enabled our CDN and [ran the test again](https://tools.pingdom.com/#!/8V7RO/https://perfmatters.io). As you can see our total load times dropped down to 788 ms and our average TTFB is now 37 ms! What a difference a CDN can make. Another important thing to note is that we chose the Stockholm location to perform this test. Why? Because we wanted to show you the real improvement that can be had by decreasing the physical distance. There is a CDN POP located in Stockholm, so our content is being served from Stockholm.

TTFB after adding a CDN

Note: If you are utilizing Cloudflare, you might have a [slightly higher TTFB](https://woorkup.com/cloudflare-alternative/). This is most likely due to the additional overhead and complexity of having the fully proxy service running. Remember that Cloudflare has additional firewalls and other features that some CDN providers don’t have. So you would need to make up your own mind which might benefit you more. If your entire site is not properly optimized, taking the hit on the slightly higher TTFB might be worth the trade-off.

However, you might also want to check out WP Bullet’s guide on using [Cloudflare page caching](https://guides.wp-bullet.com/cloudflare-cache-wordpress-posts-and-pages-guide/) to lower TTFB. This could require some additional setup and testing. Make sure to run your own tests as each environment is different.

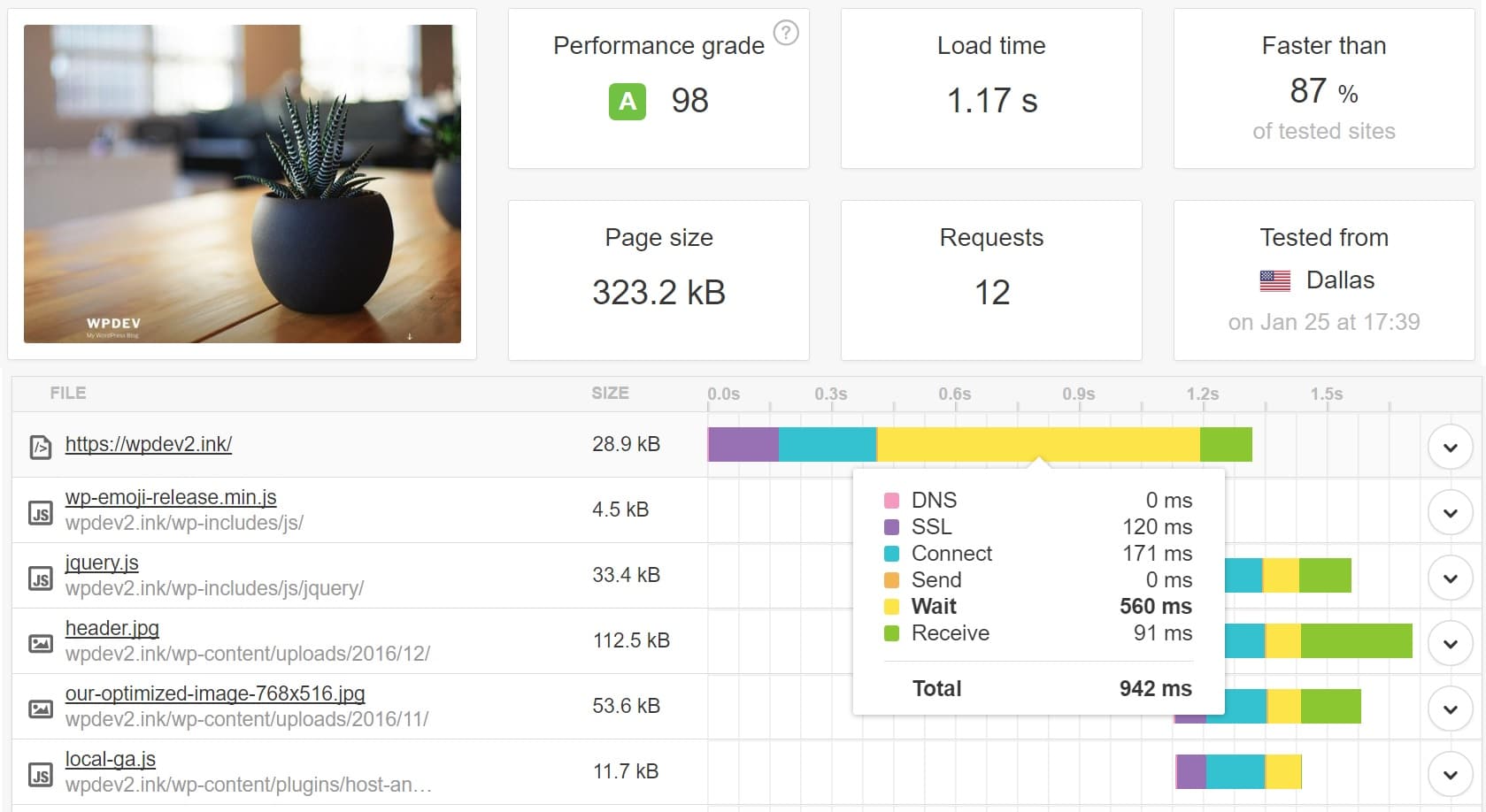
Suggested reading: [How to Set up Cloudflare APO for WordPress](https://kinsta.com/blog/cloudflare-apo-wordpress/).

### 3. WordPress Caching

A third way and probably one of the easiest ways to decrease your TTFB is to utilize caching on your WordPress site. Many only think that caching can help decrease your load times, but in fact, it also helps decrease TTFB as it helps reduce the server processing time. We ran some tests again with and without caching running. Each test was run 5 times and the average was taken.

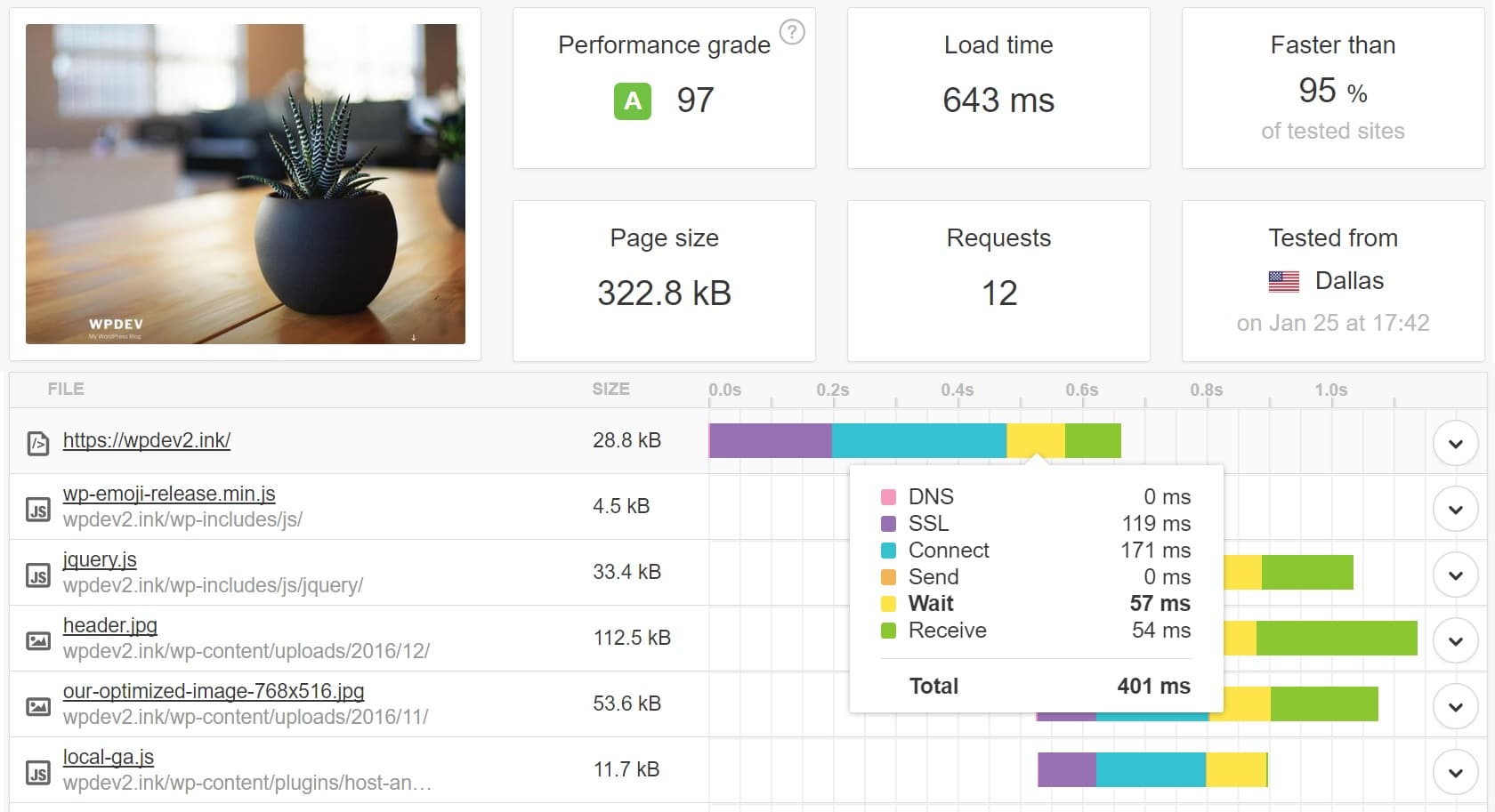
**Without Cache Running**

We ran the site through [Pingdom](https://tools.pingdom.com/#!/dhrdhX/https://wpdev2.ink/), and without cache running, our site scored a 1.17 s load time and a 560 ms TTFB.

Non-cached TTFB

**With Caching Enabled**

We then enabled caching and ran the site through [Pingdom](https://tools.pingdom.com/#!/blws5g/https://wpdev2.ink/) again. This time our site scored a 643 ms load time and a 57 ms TTFB.

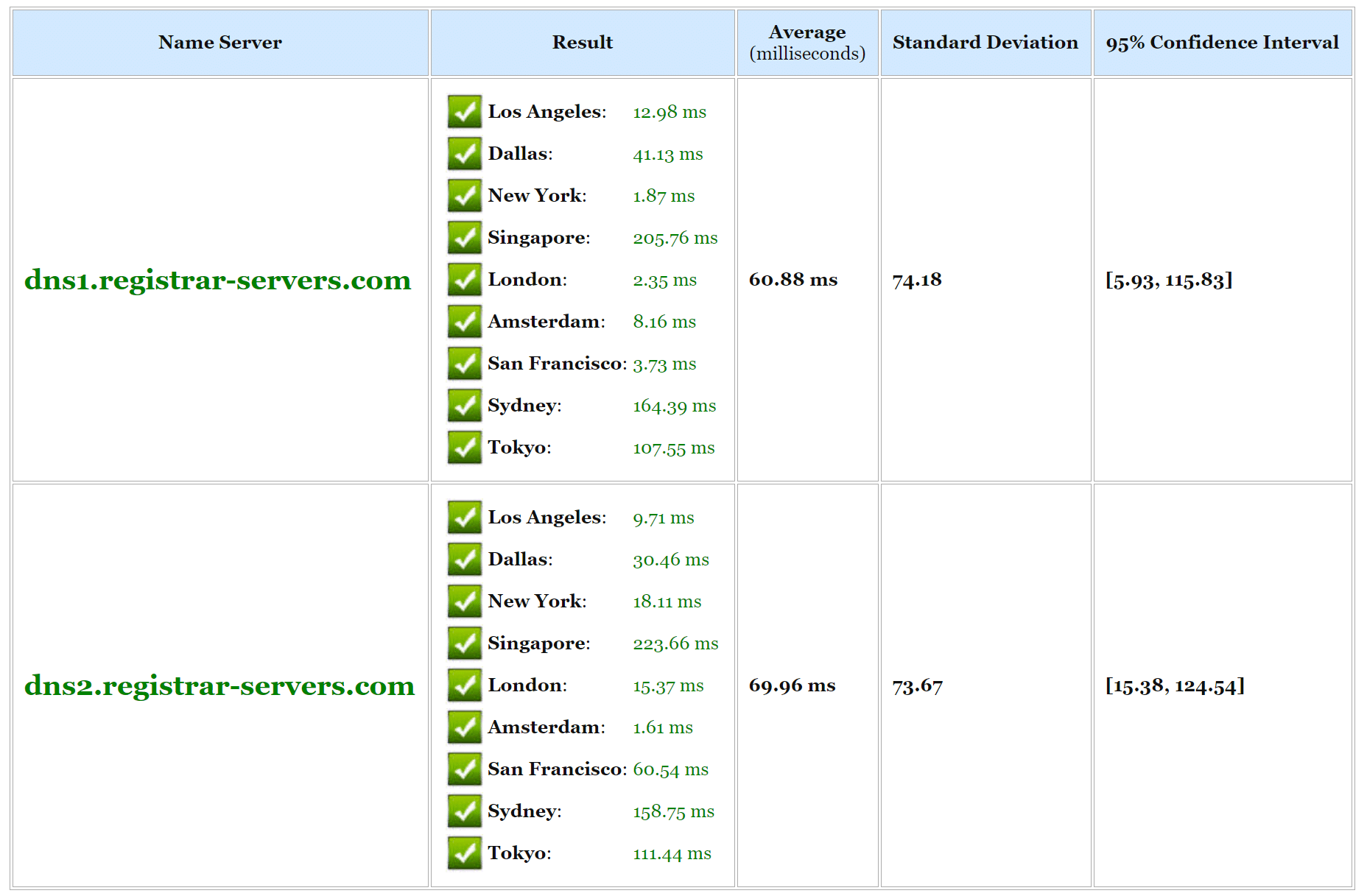
TTFB with WordPress caching enabled

So by enabling caching, we were able to **reduce our TTFB by a whopping 90%!** You can read more about [Kinsta’s caching](https://kinsta.com/blog/wordpress-cache/). We do this at a server-level which means no caching plugins are required. If you aren’t using a managed WordPress host, we recommend using a [free caching plugin like the Cache Enabler](https://kinsta.com/blog/wordpress-caching-plugins/).

### 4. Use a Premium DNS Provider

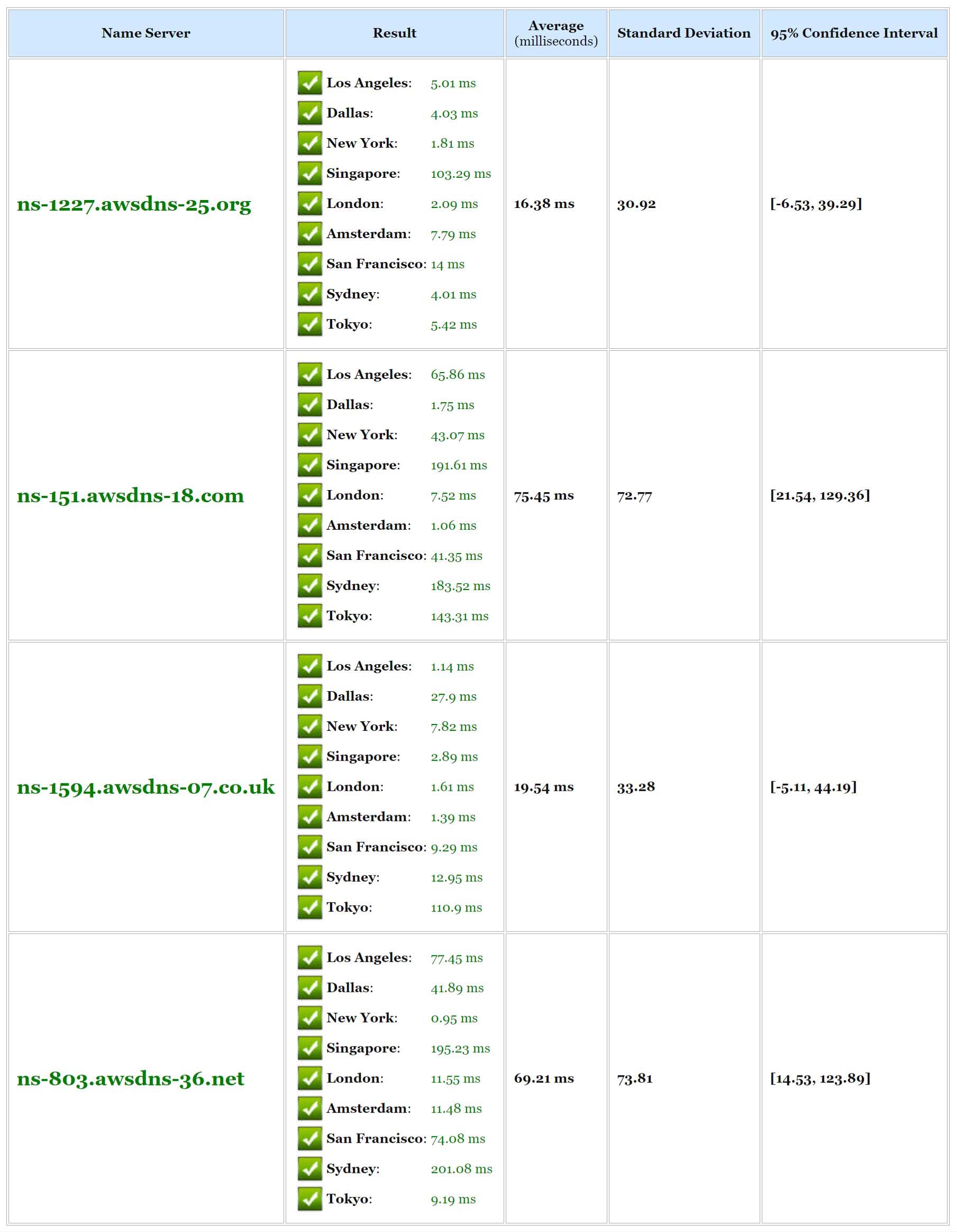
And last but not least, DNS plays a part in TTFB as well. It is hard to exactly calculate how much it is affected, but you can still see overall DNS lookup times and see that there are faster and slower providers out there. We ran a couple of tests with the [SolveDNS speed test](http://www.solvedns.com/dnsspeedtest/) tool. Here is an example of a domain using NameCheap’s free DNS and the response times.

**Free NameCheap DNS**

Free DNS speed

And below is an example using Amazon Route 53’s premium DNS. As you can see in general, DNS lookup times are much faster with Amazon. Typically premium DNS providers will have better speeds. Cloudflare is a free one that also has great performance.

**Amazon Route 53 DNS**

Amazon premium DNS speed

Make sure to check out our post on why you should be using a [premium DNS provider](https://kinsta.com/blog/premium-dns/). We partnered with [Amazon Route 53](https://kinsta.com/help/dns/#nameservers) here at Kinsta and it is available to all customers free of charge.

## Summary

There is a multitude of other things you could optimize or fix to reduce TTFB, such as database caching, Disk IO, Swap usage, RAM, PHP settings, MySQL settings, network settings, TLS overhead, etc. But the ones mentioned above are fairly easy to implement and will give you the fastest performance boost. So the next time someone asks you how to reduce your TTFB, remember that a fast WordPress host, CDN, caching, and DNS all play a huge part. Fixing or improving those bottlenecks will do the trick.

**What Time to First Byte (TTFB) Is**

TTFB measures your server’s responsiveness to a visitor’s browser request. In other words, it’s the amount of time it takes between when a user arrives on a web page and when the server recognizes it.

This is important because the longer it takes for the browser to receive the first byte of data from the server, the longer the page takes to load. And because TTFB is the very initial interaction, a long Time to First Byte means the rest of the page can’t even begin loading. Because the server has not yet responded.

Therefore, TTFB plays a pivotal role in your website’s [User Experience (UX)](https://www.elegantthemes.com/blog/design/how-to-perform-a-ux-audit-of-your-wordpress-site), as well as its [Search Engine Optimization (SEO)](https://www.elegantthemes.com/blog/marketing/what-is-seo). It is also one of the [Web Vitals](https://web.dev/vitals/) (along with [FCP](https://www.elegantthemes.com/blog/wordpress/fcp-first-contentful-paint), [LCP](https://www.elegantthemes.com/blog/wordpress/lcp), and CLS) that Google considers to determine your site’s page loading experience, and in turn its rankings.

TTFB is similar to [First Contentful Paint (FCP)](https://web.dev/fcp/), which is also used to measure the loading experience and the user’s wait time. However, the FCB measures how long it takes until the visitor can *see* the first element appear on the page. TTFB is any information, visible or not.

A website’s TTFB is made up of three main processes:

1. An HTTP request is sent from the visitor’s browser to the server.
2. The server processes the request and prepares its response.
3. The server sends a response back to the browser.

The quicker this process is, the faster and more responsive your loading times will be. However, a handful of factors and issues can delay and slow it down throughout each step. Therefore, to optimize your site’s speed and performance, you’ll likely want to focus on reducing TTFB.

**How to Measure TTFB**

There are a handful of different ways you can go about measuring your TTFB, including using both lab and field tools. Meaning both simulated results (lab) and testing using real user data (field). However, before that, it is helpful to first know what a good score is.

Typically, a favorable TTFB falls somewhere between 200 milliseconds (ms) and 500 ms (.2 and .5 seconds). Of course, the lower the number, the better. Anything above 600 ms is considered a poor score. Since Google has introduced page speed as a ranking consideration, many websites shoot for a sub-100 ms time. However, in most cases anything under 200 ms is still considered “good.”

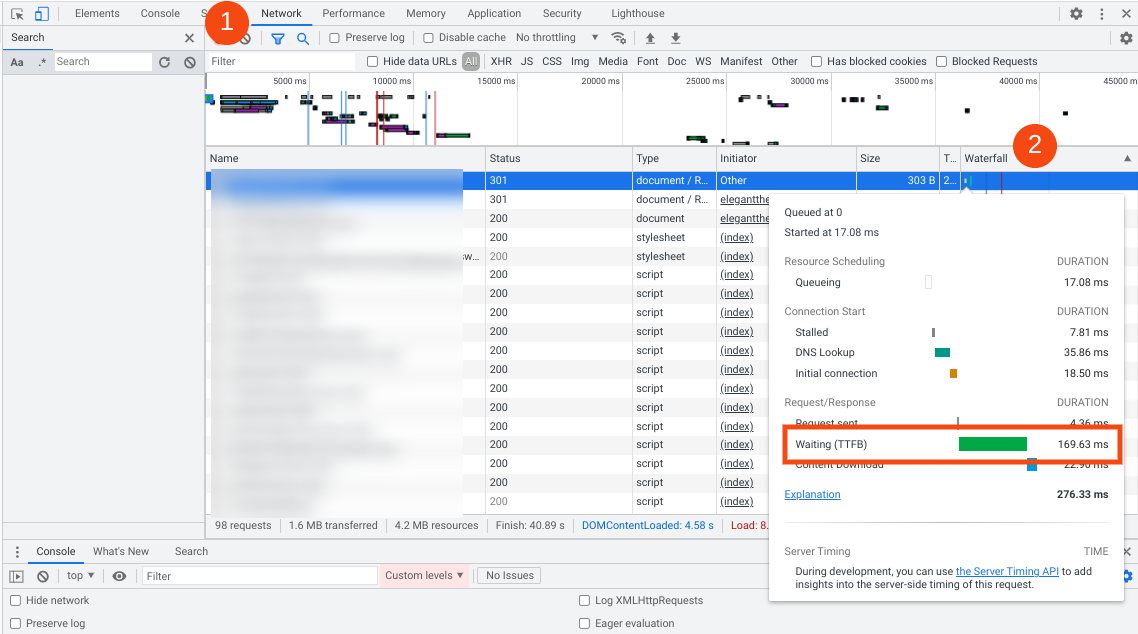
There are various tools you can use to measure TTFB.

### Google Chrome DevTools

Another way to measure TTFB is [Chrome DevTools](https://developer.chrome.com/docs/devtools/). If you’re a Chrome user, you can use this method to measure TTFB directly within your browser without using a third-party tool. Where many of the speed tests we mention above use various server locations that you can choose from, this way, it’s directly from you.

Because of that, keep in mind that your specific network conditions and network latency can influence TTFB. Therefore, the result you see when using this method may not be the same as what your site’s visitors experience.

To get started, visit your WordPress site in your browser, and then navigate to the three vertical dots in the right-hand corner of your screen. Next, navigate to More tools > Developer Tools. This will open the DevTools console.

From there, click on any element on the page, and then select the Network tab followed by the Waterfall column: 

Within the list, select the item that you want to inspect. Under the panel, you can find *Waiting (TTFB).*

## How to Reduce TTFB on Your WordPress Site (4 Key Tips)

Now that you understand more about what TTFB is and how to measure it, it’s time to see what you can do to improve it. Below are four key tips you can use to reduce TTFB on your WordPress site.

### 1. Upgrade Your WordPress Hosting

Perhaps the most influential factor in TTFB is your site’s hosting provider. You can optimize your site in every single way possible, but if your host can’t keep up, it won’t matter. The [type of hosting](https://www.elegantthemes.com/blog/wordpress/is-managed-wordpress-hosting-really-worth-it) you use influences your site’s speed and performance. For example, if you use shared hosting, you’ll likely have access to limited resources. And you’re beholden to other users on your server cluster’s bandwidth and processing usage.

Therefore, if you’re suffering from slow loading times and want to reduce your TTFB, you might want to consider upgrading to a new hosting provider or a higher-performance plan. If you’re with a shared host, upgrading your plan to a higher tier often comes with priority service and faster servers.

Additionally, [managed hosting](https://www.elegantthemes.com/blog/wordpress/is-managed-wordpress-hosting-really-worth-it) for WordPress sites is a huge benefit to TTFB. These services are designed and optimized specifically for WordPress websites, and that means a quick response time on both the user’s end and in the back-end for administration.

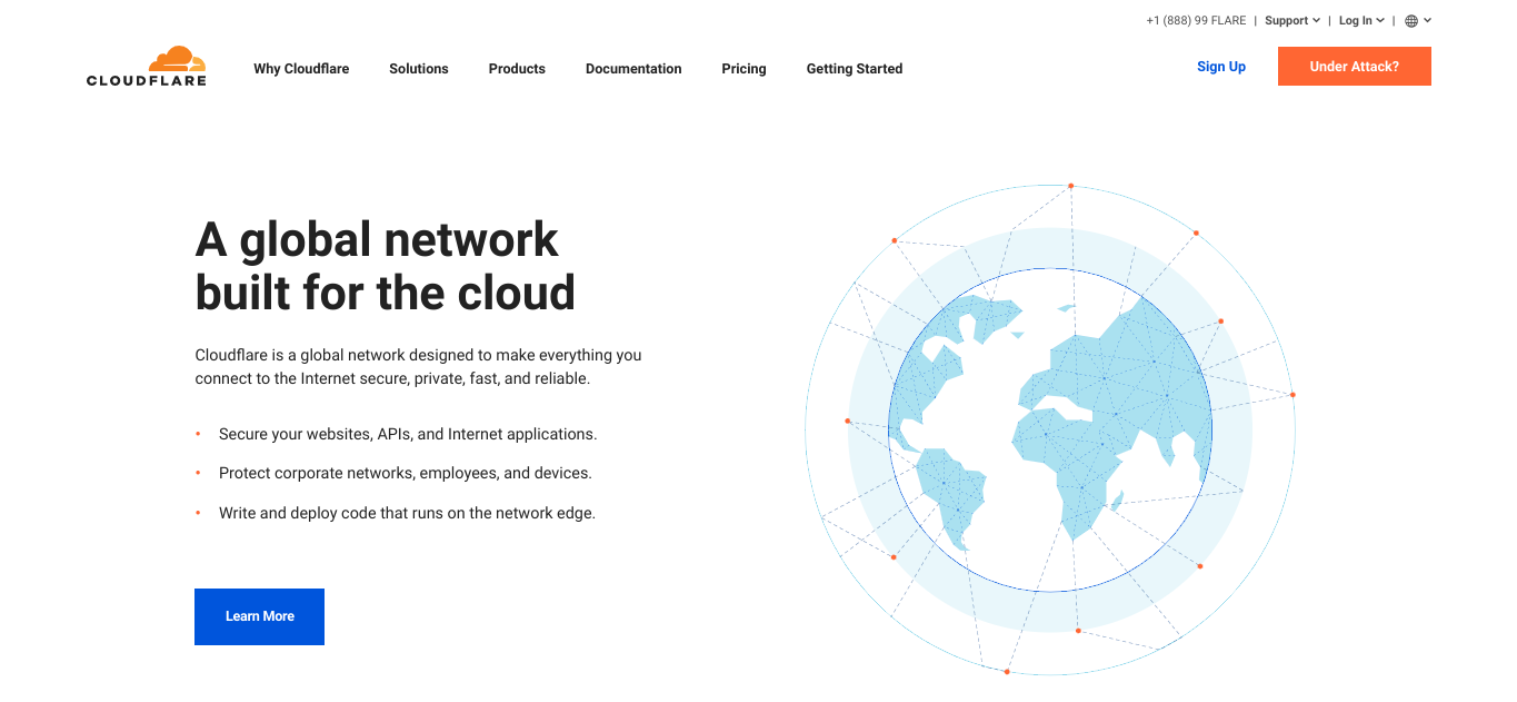
The higher tier and/or managed hosting bring a higher cost than economy hosting, but there is simply no better way to get a lower TTFB (and overall page speed) than superb hosting.

We also recommend looking for a web host that offers [premium DNS services](https://www.elegantthemes.com/blog/tips-tricks/an-introduction-to-the-domain-name-system-dns-and-how-it-works). DNS lookup times can contribute to a slow TTFB. Leveraging premium DNS tools can help reduce network latency, and in turn improve your loading times.

### 2. Use a Content Delivery Network (CDN)

One of the most effective ways to enhance site speed and reduce TTFB is [implementing a CDN](https://www.cloudflare.com/learning/cdn/what-is-a-cdn/). In case you’re unfamiliar with the concept, this is a network of distributed servers located around the globe. This means your visitors can load your site from the data center location closest to them, minimizing the time it takes.

There are a lot of different [CDN services for WordPress](https://www.elegantthemes.com/blog/wordpress/best-cdn-services-for-wordpress) that you can choose from. Some of the most popular options include [Google Cloud CDN](https://cloud.google.com/cdn/), [KeyCDN](https://www.keycdn.com/), and [Cloudflare](https://www.cloudflare.com/). In fact, some hosts (such as [SiteGround](https://www.elegantthemes.com/blog/wordpress/siteground-wordpress-hosting-an-overview-and-review)) offer a free version of the Cloudflare CDN with many of their plans.



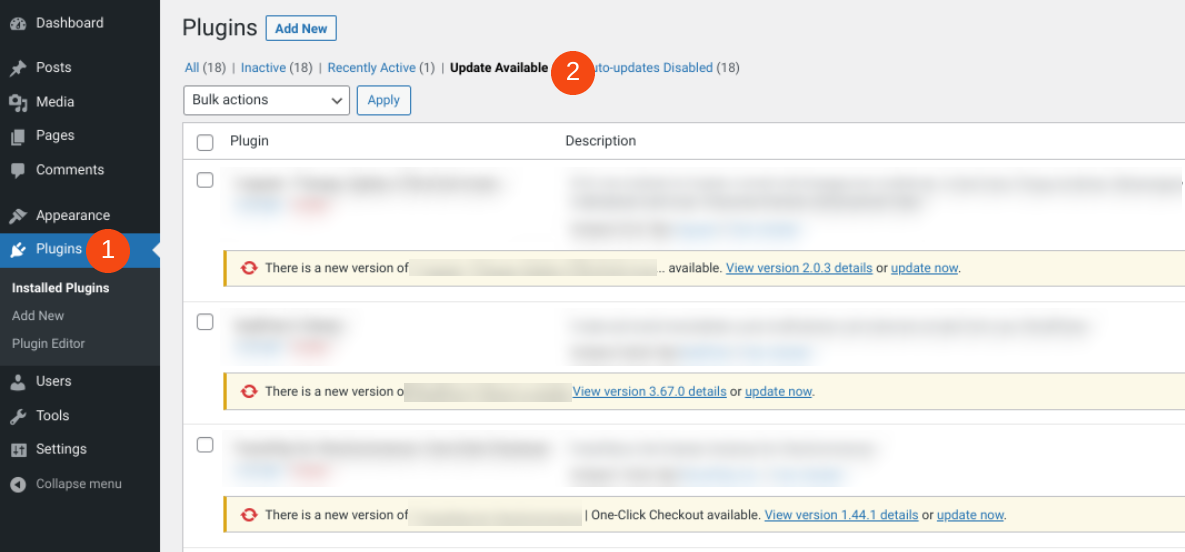
Using a CDN on your WordPress site can help you significantly reduce network latency. Beyond performance purposes, a CDN can also help strengthen your site’s security by preventing against threats such as [Distributed Denial-of-Service (DDoS) attacks](https://www.elegantthemes.com/blog/wordpress/ddos-attacks-what-wordpress-users-need-to-know).

### 3. Remove Any Unnecessary Plugins or Themes (And Keep Existing Ones Updated)

Plugins and themes can take up a lot of space and even slow your site down, especially when they’re outdated. In addition, old extensions can introduce security vulnerabilities and put unnecessary strain on your server and loading times. They might not be optimized as well as they could be, delaying your TTFB because of their code. Make sure your plugins are updated and necessary. Trimming down the number of plugins you have will definitely improve your TTFB if they’re bogging down your site’s server.

It’s easy to [remove any plugins](https://www.elegantthemes.com/blog/wordpress/how-to-uninstall-a-wordpress-plugin) or themes on your WordPress site that you are no longer using. In the same vein, it’s also important to make sure the ones you do have installed on your site stay updated. If their code isn’t optimized as well as it can be (especially with WP updates and so on), TTFB can take a hit.

From your WordPress admin dashboard, you can find the [available plugin updates](https://www.elegantthemes.com/blog/wordpress/update-wordpress-plugins) by navigating to Plugins, and then clicking on the Update Available link:



Keep in mind that plugin and theme developers regularly release performance improvements and bug fixes for their tools. Staying on top of the latest versions and eliminating any that no longer serve a purpose can help you minimize server bloat.

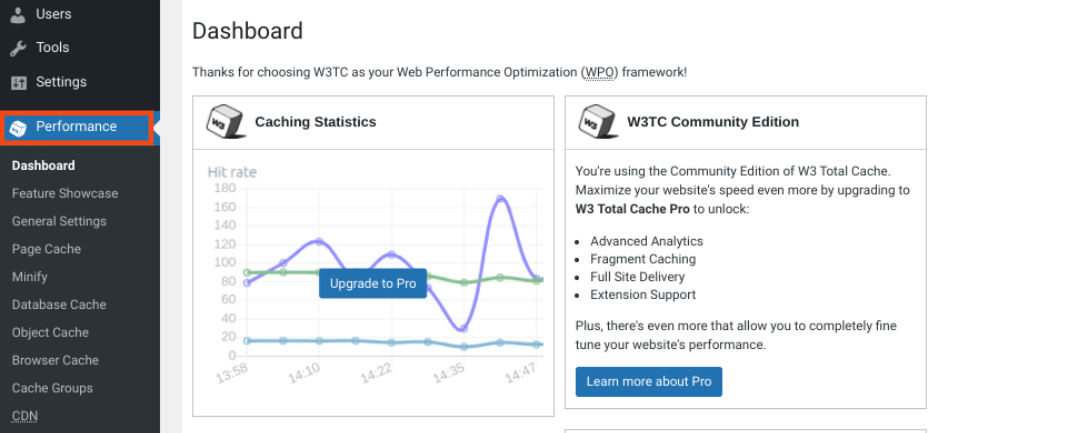
### 4. Implement Caching on Your Website

One of the most powerful ways to reduce server load is to leverage caching. Caching is a technique used to help minimize server processing time and deliver content more quickly. Therefore, it’s also a useful strategy that can help reduce TTFB.

There are a variety of [WordPress cache plugins](https://www.elegantthemes.com/blog/wordpress/best-wordpress-cache-plugins) that you can install on your website. Some of the most popular options include [WP Rocket](https://wp-rocket.me/) and [W3 Total Cache](https://wordpress.org/plugins/w3-total-cache/).



W3 Total Cache is a performance optimization tool that can help you boost your site’s speed and SEO. It can also help with CDN integration. So in this regard, you can help with TTFB in multiple ways. Once installed, you can access the settings by navigating to Performance.



This plugin is free to use. However, if you’re looking for more advanced configuration options, there is a [pro license available](https://www.w3-edge.com/solutions/w3-total-cache-pro/) for $99 per year.

## Time to First Byte (TTFB) Frequently Asked Questions

By now, you hopefully have a solid understanding of the importance of TTFB, how to measure it, and how to reduce it. Now, let’s take a look at some of the most frequently asked questions about this metric.

### How Does TTFB Fit into My Site’s Overall Performance?

TTFB refers to the time between when a visitor arrives on your site and when their browser receives the first byte of data from your server. In other words, it measures your server’s responsiveness to the user. That means without an acceptable (read: low) TTFB, the rest of your website loads slowly.

Even if your website’s content fully loads in under 1 second, if the TTFB is 4 seconds (4000 ms), then the total load time is ~5 seconds.

As such, TTFB plays a crucial role in your UX and SEO. Google considers TTFB as a key data point when considering how to rank your website. Any optimizations you make to TTFB will directly impact other page speed metrics such as [FCP](https://www.elegantthemes.com/blog/wordpress/fcp-first-contentful-paint) and [LCP](https://www.elegantthemes.com/blog/wordpress/lcp).

### What’s the Best Way to Reduce TTFB?

There’s no one single solution to reduce your TTFB. Since the primary goal is to reduce server response times, we recommend using a combination of the methods we discuss above. First and foremost, make sure that you have a reliable hosting provider with excellent performance. Then, implement a CDN and caching system on your site, keep your plugins updated, and remove any unnecessary add-ons that might be delaying your site from sending a response quickly.

TTFB isn’t always based on having a lightweight site, but having a lightweight site can still get that first byte sent back quickly.

## Conclusion

Server responsiveness is important if you want to deliver a quality experience to your website’s visitors. If your site takes too long to respond, it increases the chance that they’ll abandon your website. Even if your content loads quickly, if it takes seconds to begin that loading, users may be on their way out. To prevent this from happening, it’s crucial to pay attention to your TTFB and take steps to keep it as low as possible.