Cloudflare Fundamentals

What is Cloudflare?

1 min read

Cloudflare is a global network of servers

. When you add your application to Cloudflare, we use this network to sit in between requests and your origin server.

This position allows us to do several things — speeding up content delivery and user experience (CDN), protecting your website from malicious activity (DDoS, Web Application Firewall), routing traffic (Load balancing, Waiting Room), and more.

How Cloudflare works

3 min read

Fundamentally, Cloudflare is a large network of servers that can improve the security, performance, and reliability of anything connected to the Internet.

Cloudflare does this by serving as a reverse proxy

for your web traffic. All requests to and from your origin flow through Cloudflare and — as these requests pass through our network — we can apply various rules and optimizations to improve security, performance, and reliability.

Life of a request

Even though it feels pretty instantaneous, there's a lot happening when you type `www.example.com` into your browser.

A website's content does not technically live at a URL like `www.example.com`, but rather at an IP address like `192.0.2.1`. It's similar to how we say that Cloudflare's headquarters is 101 Townsend St., San Francisco, CA 94107, but really that address is just a placeholder for latitude and longitude coordinates (37.780259, -122.390519). URLs and street addresses are much easier for humans to remember.

The process of converting a human-readable URL (`www.example.com`) into a machine-friendly address (`192.0.2.1`) is known as a DNS lookup

.

Without Cloudflare

Without Cloudflare, DNS lookups for your application's URL return the IP address of your origin server

.

URL	Returned IP address
example.com	192.0.2.1

When using Cloudflare with unproxied DNS records, DNS lookups for unproxied domains or subdomains also return your origin's IP address.

Another way of thinking about this concept is that visitors directly connect with your origin server.

ConnectionVisitor
Origin server

With Cloudflare

With Cloudflare — meaning your domain or subdomain is using proxied DNS records — DNS lookups for your application's URL will resolve to Cloudflare Anycast IPs

instead of their original DNS target.

URL	Returned IP address
example.com	104.16.77.250

This means that all requests intended for proxied hostnames will go to Cloudflare first and then be forwarded to your origin server.

Visitor ← Connection → Cloudflare global network ←Connection→Origin Server

Cloudflare assigns specific Anycast IPs to your domain dynamically and these IPs may change at any time. This is an expected part of the operation of our Anycast network and does not affect the proxy behavior described above.

Benefits

When your traffic is proxied through Cloudflare before reaching your origin server, your application gets additional security, performance, and reliability benefits.

Security

Beyond hiding your origin's IP address from potential attackers, Cloudflare also stops malicious traffic before it reaches your origin web server.

Cloudflare automatically mitigates security risks using our WAF and DDoS protection.

For additional details on security, refer to our guide on how to Secure your website.

Performance

For proxied traffic, Cloudflare also serves as a Content Delivery Network (CDN)

, caching static resources and otherwise optimizing asset delivery.

For additional details on performance, refer to our guides on Optimizing Site Speed and Caching.

Reliability

Cloudflare's globally distributed Anycast network

routes visitor requests to the nearest Cloudflare data center.

Combined together with our CDN

and DDoS protection, our network helps keep your application online.

Cloudflare IPs

2 min read

Cloudflare has several IP address ranges

which are shared by all proxied hostnames.

Together, these IP addresses form the backbone of our Anycast network

, helping distribute traffic amongst various edge network servers.

Cloudflare uses other IP ranges for various products and services, but these addresses will not make connections to your origin.

Allow Cloudflare IP addresses

Because of how Cloudflare works, all traffic to proxied DNS records pass through Cloudflare before reaching your origin server. This means that your origin server will stop receiving traffic from individual visitor IP addresses and instead receive traffic from Cloudflare IP addresses

, which are shared by all proxied hostnames.

This setup can cause issues if your origin server blocks or rate limits connections from Cloudflare IP addresses. Because all visitor traffic will appear to come from Cloudflare IP addresses, blocking these IPs — even accidentally — will prevent visitor traffic from reaching your application.

To avoid rate limiting or blocking these requests, you will need to allow Cloudflare IPs at your origin server.

For Magic Transit customers, Cloudflare routes the traffic instead of proxying it. Once Cloudflare starts advertising your IP prefixes, it will accept IP packets destined for your network, process them, and then output these packets to your origin infrastructure.

Customize Cloudflare IP addresses

If they do not want to use Cloudflare IP addresses — which are shared by all proxied hostnames — Enterprise customers have two potential alternatives:

- **Bring Your Own IP (BYOIP)**: Cloudflare announces your IPs in all our locations.
- **Static IP addresses**: Cloudflare sets static IP addresses for your domain. For more details, contact your account team.

Business and Enterprise customers can also reduce the number of Cloudflare IPs that their domain shares with other Cloudflare customer domains by uploading a Custom SSL certificate.

Reference architectures

1 min read

Reference architecture documents and diagrams are designed to provide a foundational knowledge of Cloudflare solutioning for a variety of products. Building on the information in these documents, you can architect software solutions based on your specific context and needs.

- Content Delivery Network
- Magic Transit
- Multi-vender Application Security and Performance

Account setup

To create a Cloudflare account:

1. Go to the Sign up page
1. .
2. Enter your **Email** and **Password**.
3. Click **Create Account**.

Once you create your account, Cloudflare will automatically send an email to your address to verify that email address.

Best practices

If you are creating an account for your team or a business, we recommend choosing an email alias or distribution list for your **Email**, such as `cloudflare@example.com`.

This email address is the main point of contact for your Cloudflare billing, usage notifications, and account recovery.

Set-up 2FA

2 min read

Two-factor authentication (2FA) allows user account owners to add an additional layer of login security to Cloudflare accounts. This additional authentication step requires you to provide both something you know, such as a Cloudflare password, and something you have, such as an authentication code from a mobile device.

Cloudflare user accounts configured to use single sign-on (SSO) cannot configure 2FA.

Cloudflare offers the option to use either a phishing-resistant security key, like a YubiKey, or a Time-Based One-Time password (TOTP) mobile app for authentication, like Google Authenticator, or both. If you add both of these authentication methods to your account, you are initially prompted to log in with the security key, but can opt-out and use TOTP instead.

To ensure that you can securely access your account even without your mobile device or security keys, Cloudflare also provides backup codes for download.

Tip

After downloading your backup codes, we recommend saving them in a secure location.

As the user account owner, you are automatically assigned the Super Administrator role. Once 2FA is enabled, all Cloudflare account members are required to configure 2FA on their mobile devices.

Enable 2FA

We recommend that all Cloudflare user account holders enable two-factor authentication (2FA) to keep your accounts secure.

2FA can only be enabled successfully on an account with a verified email address. If you do not verify your email address first, you may lock yourself out of your account.

Super Administrators can turn on **2FA Enforcement** to require all members to enable 2FA. If you are not a Super Administrator, you will be forced to turn on 2FA prior to accepting the invitation to join a Cloudflare account as a member.

To enable two-factor authentication for your Cloudflare login:

1. Log in to the Cloudflare dashboard
1. .
2. Under the **My Profile** dropdown, select **My Profile**.
3. Select **Authentication**.
4. Select **Manage** in the Two-Factor Authentication card.
5. Configure either a TOTP mobile app or a security key to enable 2FA on your account.

Additional configurations

Cloudflare also supports 2FA with device built-in authenticators (Apple Touch ID, Android fingerprint, or Windows Hello), Yubikeys and TOTP mobile applications.

Customize your account

2 min read

After creating an account, here are a handful of configurations you can customize:

Account name

Your account name defaults to `<<YOUR_EMAIL_ADDRESS>>'s Account`.

You may want to customize the name of this account, either to help specify its purpose or to help associated with multiple accounts.

To change your account name:

1. Log into the Cloudflare dashboard
1. .

2. Go to **Manage Account** > **Configurations**.
3. For **Account Name**, select **Change Name**.
4. Enter a new account name.
5. Select **Save**.

Appearance

If you want to adjust how the Cloudflare dashboard appears on your device, you can adjust relevant settings in your account **Profile**.

To update appearance preferences:

1. Log into the Cloudflare dashboard
1. .
2. Go to **My Profile**
3. For **Appearance**, choose a value:
    - **Dark**: Defaults to darker colors.
    - **Light**: Defaults to lighter colors.
    - **Use system setting**: Defaults to whatever is used on your device.
4. Your dashboard display will update to the new appearance setting automatically.

Communication preferences

When you create an account, Cloudflare automatically chooses your **Communication Preferences**, or when Cloudflare might occasionally send you emails.

To update the communication preferences for your profile (which requires a verified email address):

1. Log into the Cloudflare dashboard
1. .
2. Go to **My Profile**
3. For **Communication Preferences**, select **Edit**.
4. If you want a specific category of emails, make sure its associated box is checked.
5. Select **Save**.

Language preferences

After you create your account, you may want to update your language preference.

To update the language preference for your profile:

1. Log into the Cloudflare dashboard

.

Go to **My Profile**

For **Language Preference**, select a value.

Your dashboard display will update to the new language automatically.

Add and manage other members

3 min read

Learn how to add new account members, edit or revoke their permissions and access, and resend verifications emails.

To manage account members, you must have a role of **Super Administrator** and have a verified email address.

View account members

To manage account members, you must have a role of **Super Administrator** and have a verified email address.

Dashboard mode:

To view members using the dashboard:

1. Log in to the Cloudflare dashboard

and select your account.

Go to **Manage Account** > **Members**.

API mode:

To view members using the API, send a GET request.

Baseurl:

GET https://api.cloudflare.com/client/v4

An API key is a token that you provide when making API calls. Include the token in a header parameter called `X-Auth-Email`.

Example: `X-Auth-Email: 123`

An API key is a token that you provide when making API calls. Include the token in a header parameter called `X-Auth-Key`.

Example: `X-Auth-Key: 123`

An API key is a token that you provide when making API calls. Include the token in a header parameter called `X-Auth-User-Service-Key`.

Example: `X-Auth-User-Service-Key: 123`

Provide your bearer token in the Authorization header when making requests to protected resources.

Example: `Authorization: Bearer 123`

Interact with Cloudflare's products and services via the Cloudflare API.

Using the Cloudflare API requires authentication so that Cloudflare knows who is making requests and what permissions you have. Create an API token to grant access to the API to perform actions.

To create an API token, from the Cloudflare dashboard, go to My Profile > API Tokens and select Create Token.

Add account members

To manage account members, you must have a role of **Super Administrator** and have a verified email address.

Dashboard mode:

To add a member to your account:

1. Log in to the Cloudflare dashboard
1. and select your account.
2. Go to **Manage Account** > **Members**.
3. Select **Invite**.
4. Fill out the following information:
    - **Invite members**: Enter one or more email addresses (if multiple, separate addresses with commas).
    - **Scope**: Use a variety of fields to adjust the scope of your roles.
    - **Roles**: Choose one or more roles to assign your members.
5. Select **Continue to summary**.
6. Review the information, then select **Invite**.

If a user already has an account with Cloudflare and you have an Enterprise account, you can also select **Direct Add** to add them to your account without sending an email invitation.

API mode:

POST https://api.cloudflare.com/client/v4/accounts/{account_identifier}/members

Request Sample

curl --request POST \

  --url https://api.cloudflare.com/client/v4/accounts/account_identifier/members \

  --header 'Content-Type: application/json' \

  --header 'X-Auth-Email: ' \

```
  --data '{
  "email": "user@example.com",
  "roles": [
    "3536bcfad5faccb999b47003c79917fb"
  ],
  "status": "pending"
}'
```

Response Example


```
{
  "errors": [],
  "messages": [],
  "result": {
    "id": "4536bcfad5faccb111b47003c79917fa",
    "roles": [
      {
        "description": "Administrative access to the entire Account",
        "id": "3536bcfad5faccb999b47003c79917fb",
        "name": "Account Administrator",
        "permissions": {
          "analytics": {
            "read": true,
            "write": false
          },
          "zones": {
            "read": true,
            "write": true
```

```
        }
      }
    }
  ],
  "status": null,
  "user": {
    "email": "user@example.com",
    "first_name": "John",
    "id": "023e105f4ecef8ad9ca31a8372d0c353",
    "last_name": "Appleseed",
    "two_factor_authentication_enabled": false
  },
  "code": "05dd05cce12bbed97c0d87cd78e89bc2fd41a6cee72f27f6fc84af2e45c0fac0"
 },
 "success": true
}
```


Resend an invitation

If you invited a member to your account but they cannot find the invitation or the invitation expires, you can resend the invitation through the Cloudflare dashboard:

1. Log in to the Cloudflare dashboard and select your account[^1].
2. Go to **Manage Account** > **Members**.
3. Select a member record where their **Status** is **Invite Pending**.
4. Select **Resend invite**

Create an API token

2 min read

Prerequisite

Before you begin, find your zone and account IDs.

1. From the Cloudflare dashboard, go to **My Profile** > **API Tokens**.
2. Select **Create Token**.
3. Select a template from the available API token templates or create a custom token. We use the **Edit zone DNS** template in the following examples.
4. Add or edit the token name to describe why or how the token is used. Templates are prefilled with a token name and permissions.
5. Modify the token's permissions. After selecting a permissions group (*Account*, *User*, or *Zone*), choose what level of access to grant the token. Most groups offer `Edit` or `Read` options. `Edit` is full CRUDL (create, read, update, delete, list) access, while `Read` is the read permission and list where appropriate. Refer to the available token permissions for more information.
6. Select which resources the token is authorized to access. For example, granting `Zone DNS Read` access to a zone `example.com` will allow the token to read DNS records only for that specific zone. Any other zone will return an error for DNS record reads operations. Any other operation on that zone will also return an error.
7. (Optional) Restrict how a token is used in the **Client IP Address Filtering** and **TTL (time to live)** fields.
8. Select **Continue to summary**.
9. Review the token summary. Select **Edit token** to make adjustments. You can also edit a token after creation.
10. Select **Create Token** to generate the token's secret.
11. Copy the secret to a secure place.

Warning

The token secret is **only shown once**. Do not store the secret in plaintext where others can access it. Anyone with this token can perform the authorized actions against the resources that the token has access to.

The token secret page also includes an example command to test the token. Use the `/user/tokens/verify` endpoint to fetch the current status of the given token.

```
$ curl "https://api.cloudflare.com/client/v4/user/tokens/verify" \
    -H "Authorization: Bearer <API_TOKEN>"
```

The result:


```
{
 "result": {
   "id": "100bf38cc8393103870917dd535e0628",
   "status": "active"
```

```
},
"success": true,
"errors": [],
"messages": [
  {
    "code": 10000,
    "message": "This API Token is valid and active",
    "type": null
  }
]

}
```


With this you have successfully created an API token and can start working with the Cloudflare API. After creating your first API token, you can create additional API tokens via the API.

Add your domain to Cloudflare

Minimize downtime

2 min read

When making any change to the routing of an Internet application, there is always a possibility of downtime due to certificate issuance, misconfigured settings, or limitations at your origin server. To avoid downtime when going live, it's important to review the most common configurations.

Update and review DNS records.

Before activating your domain on Cloudflare (exact steps depend on your DNS setup), review the DNS records in your Cloudflare account.

Start with unproxied records

With a new domain, make sure all your DNS records have a proxy status of **DNS-only**.

This setting prevents Cloudflare from proxying your traffic before you have an active edge certificate or before you have allowed Cloudflare IP addresses.

Confirm record accuracy

Take extra time to confirm the accuracy of your DNS records before activating your domain, paying special attention to:

- Zone apex records (`example.com`)
- Subdomain records (`www.example.com` or `blog.example.com`)
- Email records

If you add DNS records to your authoritative DNS provider between onboarding your domain and activating your domain, you may need to also add these records within Cloudflare.

Activate your domain.

Finish the DNS setup for your domain, moving the domain status to **Active**:

- Full setups: Update the authoritative nameservers at your registrar and wait for that change to be authenticated.
- Partial setups: Add the verification TXT record to your authoritative DNS and wait for that change to be authenticated.

Verify SSL/TLS edge certificates.

Before proxying your traffic through Cloudflare, verify that Cloudflare has an active **Edge Certificate** for your domain.

For more details about timing and certificate recommendations, refer to Certificate issuance.

Optional - Test configuration.

You may want to test your configuration using your local machine or proxying traffic from a development domain or subdomain.

If you experience issues, you should make sure that you have allowed Cloudflare IP addresses at your origin server.

Update proxy status.

Once you have verified that your SSL/TLS edge certificate is active and you have allowed Cloudflare IP addresses, change the proxy status of appropriate DNS records to **Proxied**.

Allow Cloudflare IP addresses

2 min read

Because of how Cloudflare works, all traffic to proxied DNS records pass through Cloudflare before reaching your origin server. This means that your origin server will stop receiving traffic from individual visitor IP addresses and instead receive traffic from Cloudflare IP addresses

, which are shared by all proxied hostnames.

This setup can cause issues if your origin server blocks or rate limits connections from Cloudflare IP addresses. Because all visitor traffic will appear to come from Cloudflare IP addresses, blocking these IPs — even accidentally — will prevent visitor traffic from reaching your application.

To avoid rate limiting or blocking these requests, you will need to allow Cloudflare IPs at your origin server.

For Magic Transit customers, Cloudflare routes the traffic instead of proxying it. Once Cloudflare starts advertising your IP prefixes, it will accept IP packets destined for your network, process them, and then output these packets to your origin infrastructure.

Review external tools

To avoid blocking Cloudflare IP addresses unintentionally, review your external tools to check that:

- Any security plugins — such as those for WordPress — allow Cloudflare IP addresses.
- The mod_security
- plugin is up to date.

Configure origin server

Allowlist Cloudflare IP addresses

To avoid blocking Cloudflare IP addresses unintentionally, you also want to allow Cloudflare IP addresses at your origin web server.

You can explicitly allow these IP addresses with a .htaccess file or by using iptables.

The following example demonstrates how your could use an iptables rule to allow a Cloudflare IP address range. Replace `$ip` below with one of the Cloudflare IP address ranges

```
# For IPv4 addresses
iptables -I INPUT -p tcp -m multiport --dports http,https -s $ip -j ACCEPT
# For IPv6 addresses
ip6tables -I INPUT -p tcp -m multiport --dports http,https -s $ip -j ACCEPT
```

Block other IP addresses (recommended)

As a best practice, we also recommend that you explicitly block all traffic that does not come from Cloudflare IP addresses or the IP addresses of your trusted partners, vendors, or applications.

For example, you might update your iptables

with the following commands:
#for IPv4
iptables -A INPUT -p tcp -m multiport --dports http,https -j DROP
#for IPv6
ip6tables -A INPUT -p tcp -m multiport --dports http,https -j DROP

Disable DNSSEC

2 min read

DNS Security Extensions (DNSSEC) adds an extra layer of authentication to DNS, ensuring requests are not routed to a spoofed domain.

Disable DNSSEC

If you are onboarding an existing domain to Cloudflare, make sure DNSSEC **is disabled** at your registrar (where you purchased your domain name). Otherwise, your domain will experience connectivity errors when you change your nameservers.

Why do I have to disable DNSSEC?

When your domain has DNSSEC enabled, your DNS provider digitally signs all your DNS records. This action prevents anyone else from issuing false DNS records on your behalf and redirecting traffic intended for your domain.

However, having a single set of signed records also prevents Cloudflare from issuing new DNS records on your behalf (which is part of using Cloudflare for your authoritative nameservers). So if you change your nameservers without disabling DNSSEC, DNSSEC will prevent Cloudflare's DNS records from resolving properly.

Add a site

2 min read

1. Log in to the Cloudflare dashboard.
2. In the top navigation bar, click **Add site**.

3. Enter your website's apex domain (`example.com`) and then click **Add Site**.

If Cloudflare is unable to identify your domain as a registered domain, make sure you are using an existing top-level domain

(`.com`, `.net`, `.biz`, or others).

Additionally, Cloudflare requires your `apex domain` to be one level below a valid TLD defined in the Public Suffix List (PSL).

1. Select your plan level. For more details on features and pricing, refer to our Plans page
   .
2. Review your DNS records.
   When you add a new site to Cloudflare, Cloudflare automatically scans for common records and adds them to the DNS zone. The records show up under the respective zone **DNS** > **Records** page.

3. Since this scan is not guaranteed to find all existing DNS records, you need to review your records, paying special attention to the following record types:
   a. Zone apex records (`example.com`)
   b. Subdomain records (`www.example.com` or `blog.example.com`)
   c. Email records
4. If you activate your domain on Cloudflare *without* setting up the correct DNS records for your domain and subdomain, your visitors may experience DNS_PROBE_FINISHED_NXDOMAIN errors.
5. If you find any missing records, manually add those records.
6. Depending on your site setup, you may want to adjust the proxy status for certain `A`, `AAAA`, or `CNAME` records.
7. Click **Continue**.
8. Go through the **Quick Start Guide** and when you have finished, click **Finish**.

Update your nameservers

1 min read

Once you have added a domain (also known as a *zone*) to Cloudflare, that domain will receive two assigned authoritative nameservers.

Before your domain can begin using Cloudflare for DNS resolution, you need to add these nameservers at your registrar. Make sure DNSSEC **is disabled** at this point.

Domain Resolution

Ensure all your traffic is proxying through Cloudflare successfully.

Objectives

By the end of this module, you will be able to:

- Confirm your zone is set up correctly on Cloudflare
- Recognize and troubleshoot issues with your DNS records and SSL/TLS certificates

Review DNS records

1 min read

When you add a new site to Cloudflare, Cloudflare automatically scans for common records and adds them to the DNS zone. The records show up under the respective zone **DNS** > **Records** page.

The DNS records quick scan is not automatically invoked in the following cases:

- If you choose Enterprise plan and, instead of the **Quick Scan**, choose to upload a DNS zone file or add records manually.
- If you add a zone via the API.

You can manually invoke the quick scan via API with the Scan DNS Records endpoint. Note that the quick scan is a best effort attempt based on a predefined list of commonly used record names and types. You can read more about this in the reference page.

Since this scan is not guaranteed to find all existing DNS records, you need to review your records, paying special attention to the following record types:

- Zone apex records (`example.com`)
- Subdomain records (`www.example.com` or `blog.example.com`)
- Email records

If you want more control over which DNS records are imported and how, import a zone file.

If your domain is added to Cloudflare by a hosting partner, manage your DNS records via the hosting partner.

Proxy status

3 min read

The **Proxy status** of a DNS record affects how Cloudflare treats incoming traffic to that record. Cloudflare recommends enabling our proxy for all `A`, `AAAA`, and `CNAME` records.

Proxied records

Note that if you have multiple `A/AAAA` records on the same name and at least one of them is proxied, Cloudflare will treat all `A/AAAA` records on this name as being proxied.

When you proxy specific DNS records through Cloudflare - specifically `A`, `AAAA`, or `CNAME` records — DNS queries for these will resolve to Cloudflare Anycast IPs instead of their original DNS target. This means that all requests intended for proxied hostnames will go to Cloudflare first and then be forwarded to your origin server.

This behavior allows Cloudflare to optimize, cache, and protect all requests to your application, as well as protect your origin server from DDoS attacks

Because requests to proxied hostnames go through Cloudflare before reaching your origin server, all requests will appear to be coming from Cloudflare's IP addresses (and could potentially be blocked or rate limited). If you use proxied records, you may need to adjust your server configuration to allow Cloudflare IPs.

Cloudflare Anycast IPs used to proxy traffic on your domain are assigned automatically. These IPs might change at any time for operational reasons. If you need to allowlist Cloudflare IPs on your infrastructure or hosting provider, include the full list of Cloudflare Anycast IPs

As an Enterprise customer, you have the option to get static IPs or bring your own IPs (BYOIP).

Limitations

Record types

By default, Cloudflare only supports proxied `A`, `AAAA`, and `CNAME` records. You cannot proxy other record types.

If you encounter a `CNAME` record that you cannot proxy — usually associated with another CDN provider — a proxied version of that record will cause connectivity errors. Cloudflare is purposely preventing that record from being proxied to protect you from a misconfiguration.

Ports and protocols

By default, Cloudflare only proxies HTTP and HTTPS traffic.

If you need to connect to your origin using a non-HTTP protocol (SSH, FTP, SMTP) or the traffic targets an unsupported port at the origin, either leave your records unproxied (DNS-only) or use Cloudflare Spectrum.

Pending domains

When you add a domain to Cloudflare, Cloudflare protection will be in a pending state until we can verify ownership. This could take up to 24 hours to complete.

This means that DNS records - even those set to proxy traffic through Cloudflare – will be DNS-only until your zone has been activated and any requests to your DNS records will return your origin server's IP address.

If this warning is still present after 24 hours, refer to Troubleshooting.

For enhanced security, we recommend rolling your origin IP addresses at your hosting provider after your zone has been activated. This action prevents your origin IPs from being leaked during onboarding.

Windows authentication

Because Microsoft Integrated Windows Authentication, NTLM, and Kerberos violate HTTP/1.1 specifications, they are not compatible with proxied DNS records.

Enable DNSSEC

2 min read

DNS Security Extensions (DNSSEC) adds an extra layer of authentication to DNS, ensuring requests are not routed to a spoofed domain.

For additional background on DNSSEC, visit the Cloudflare Learning Center

.

When you enable DNSSEC, Cloudflare signs your zone, publishes your public signing keys, and generates your **DS** record.

Step 1 - Activate DNSSEC in Cloudflare

1. Log in to the Cloudflare dashboard
1. and select your account and domain.
2. Go to **DNS** > **Settings**.
3. For **DNSSEC**, click **Enable DNSSEC**.
4. In the dialog, you have access to several necessary values to help you create a **DS** record at your registrar. Once you close the dialog, you can access this information by clicking **DS record** on the **DNSSEC** card.

Step 2 — Add DS record to your registrar

Add the **DS** record to your registrar. If Algorithm 13 - Cloudflare's preferred cipher choice - is not listed by your registrar, it may also be called *ECDSA Curve P-256 with SHA-256*.

Provider-specific instructions
Note:

Cloudflare automatically adds **DS** records for domains using Cloudflare Registrar or those using `.ch` and `.cz` top-level domains.

Create a subdomain

1 min read

Most subdomains serve a specific purpose within the overall context of your website. For example, `blog.example.com` might be your blog, `support.example.com` could be your customer help portal, and `store.example.com` would be your e-commerce site.

Subdomain records

To create a new subdomain, you would first add the subdomain content at your host.

Then, you would create a corresponding A, AAAA, or CNAME record for that subdomain (`blog`, `store`).

Type	Name	IPv4 address	Proxy status
A	`www`	`192.0.2.1`	Proxied

Set up email records

1 min read

Receive email

If you only need to **receive** emails, Cloudflare offers Email Routing for free email forwarding to custom email addresses.

Send and receive email

To **send and receive** emails from your domain, you need:

- An SMTP provider.
- To create two DNS records within Cloudflare.

To route emails through Cloudflare and to your mail server:

1. Get the IP address and MX record details from your SMTP provider (vendor-specific guidelines).

2. Add an A or AAAA record for your mail subdomain that points to the IP address of your mail server.

Type	Name	IPv4 address	Proxy status
A	`mail`	`192.0.2.1`	DNS only

3. API example

4. Add an MX record that points to that subdomain.

Type	Name	Mail server	TTL
MX	`@`	`mail.example.com`	Auto

API Example:

Request:

```
curl -sX POST "https://api.cloudflare.com/client/v4/zones/<ZONE_ID>/dns_records" \

-H 'x-auth-email: <EMAIL>' \

-H 'x-auth-key: <API_KEY>' \

-H "Content-Type: application/json" \

--data '{
 "type":"MX",
 "name":"example.com",
 "content":"mail.example.com",
 "ttl":3600
}'
```

Response:

```
{
 "result": {
   "id": "<ID>",
   "zone_id": "<ZONE_ID>",
   "zone_name": "example.com",
   "name": "example.com",
   "type": "MX",
   "content": "mail.example.com",
   "priority": 10,
   "proxiable": false,
   "proxied": false,
   "ttl": 3600,
   "locked": false,
```

```
  "meta": {
    "auto_added": false,
    "managed_by_apps": false,
    "managed_by_argo_tunnel": false,
    "source": "primary"
  },
  "comment": null,
  "tags": [],
  "created_on": "2023-01-17T20:54:23.660869Z",
  "modified_on": "2023-01-17T20:54:23.660869Z"
},
"success": true,
"errors": [],
"messages": []

}
```


Default improvements

1 min read

When your DNS records are proxied through Cloudflare, Cloudflare provides free and unmetered DDoS protection and other protection measures through the Web Application Firewall (WAF).

DDoS protection

A distributed denial-of-service (DDoS) attack is where a large number of computers or devices, usually controlled by a single attacker, attempt to access a website or online service all at once. This flood of traffic can overwhelm the website's origin servers, causing the site to slow down or even crash.

For more information about DDoS attacks and Cloudflare DDoS protection, refer to Prevent DDoS attacks.

Managed rulesets

All customers have access to the Cloudflare Free Managed Ruleset, which provides mitigations against high and wide-impacting vulnerabilities.

For more details, refer to the WAF documentation.

SSL/TLS settings

2 min read

Once you make sure that your Cloudflare SSL/TLS is working correctly, you will likely want to customize your SSL/TLS setup.

Encryption mode

Your zone's **SSL/TLS Encryption Mode** controls how Cloudflare manages two connections: one between your visitors and Cloudflare, and the other between Cloudflare and your origin server.

Basic setup

The simplest way to choose your encryption mode is to enable the **SSL/TLS Recommender**, which scans your domain and recommends the appropriate setting.

To make sure you do not inadvertently block the **SSL/TLS Recommender**, review your settings to make sure your domain:

- Is accessible.
- Is not blocking requests from our bot (which uses a user agent of `Cloudflare-SSLDetector`).
- Does not have any active, SSL-specific Page Rules or Configuration rules.

Then, you can enable SSL/TLS recommendations in the dashboard:

1. Log in to the Cloudflare dashboard
1. and select your account and application.
2. Go to **SSL/TLS**.
3. For **SSL/TLS Recommender**, switch the toggle to **On**.

Once enabled, the SSL/TLS Recommender runs an origin scan using the user agent `Cloudflare-SSLDetector` and ignores your `robots.txt` file (except for rules explicitly targeting the user agent).

Based on this initial scan, the Recommender may decide that you could use a stronger SSL encryption mode. It will never recommend a weaker option than what is currently configured.

If so, it will send the application owner an email with the recommended option and add a *Recommended by Cloudflare* tag to that option on the **SSL/TLS** page. You are not required to use this recommendation.

If you do not receive an email, keep your current **SSL encryption mode**.

Secure setup

If possible, Cloudflare recommends using **Full** or **Full (strict)** modes to prevent malicious connections to your origin.

These modes usually require additional setup and can be more technically challenging.

Enforce HTTPS connections

Even if your application has an active edge certificate, visitors can still access resources over unsecured HTTP connections.

Using various Cloudflare settings, however, you can force all or most visitor connections to use HTTPS.

Evaluate additional features

After you have chosen your encryption mode and enforced HTTPS connections, evaluate the following settings:

- Edge certificates: Customize different aspects of your edge certificates, from enabling **Opportunistic Encryption** to specifying a **Minimum TLS Version**.
- Authenticated origin pull: Ensure all requests to your origin server originate from the Cloudflare network.
- Notifications: Set up alerts related to certificate validation status, issuance, deployment, renewal, and expiration.

Bot Fight Mode

1 min read

Bot Fight Mode is a simple, free product that helps detect and mitigate bot traffic on your domain. When enabled, the product:

- Identifies traffic matching patterns of known bots
- Issues computationally expensive challenges in response to these bots
- Notifies Bandwidth Alliance
- partners (if applicable) to disable bots

Considerations

Bot Fight Mode has a few limitations, including that it:

- Protects entire domains without endpoint restrictions.

- Cannot be customized, adjusted, or reconfigured via WAF custom rules.

If these limitations could cause issues with your application, do not enable this feature.

For more granular control - including the ability to use the `Skip` action for bot mitigation - consider using Super Bot Fight Mode.

Setup

To start using Bot Fight Mode:

1. Log in to the Cloudflare dashboard

and select your account and domain.
Go to **Security** > **Bots**.
For **Bot Fight Mode**, select **On**.

Secure your origin

4 min read

Your origin server

is a physical or virtual machine that is not owned by Cloudflare and hosts your application content (data, webpages, etc.).

Receiving too many requests can be bad for your origin. These requests might increase latency for visitors, incur higher costs — particularly for cloud-based machines — and could knock your application offline.

Secure origin connections

When you secure origin connections, it prevents attackers from discovering and overloading your origin server with requests.

- **DNS**:
    1. **Proxy records** (when possible): Set up proxied (orange-clouded) DNS records to hide your origin IP addresses and provide DDoS protection. As part of this, you should allow Cloudflare IP addresses at your origin to prevent requests from being blocked.
    2. **Review DNS-only records**: Audit existing **DNS-only** records (`SPF`, `TXT`, and more) to make sure they do not contain origin IP information.
    3. **Evaluate mail infrastructure**: If possible, do not host a mail service on the same server as the web resource you want to protect, since emails sent to

non-existent addresses get bounced back to the attacker and reveal the mail server IP.
4. **Rotate origin IPs**: Once onboarded, rotate your origin IPs, as DNS records are in the public domain. Historical records are kept and would contain IP addresses prior to joining Cloudflare

Application layer

1. Cloudflare Tunnel (HTTP/WebSockets)

Cloudflare Tunnel connects your resources to Cloudflare without a publicly routable IP address, by creating an outbound-only connections to Cloudflare's global network.

- **Security**: Very secure.
- **Availability**: All customers.
- **Challenges**: Requires installing the `cloudflared` daemon on origin server or virtual machine.

2. HTTP Header Validation

Only allow traffic with specific (and secret) HTTP headers.

- **Security**: Moderately secure.
- **Availability**: All customers.
- **Challenges**:
    1. Requires more configuration efforts on application- and server-side to accept those headers.
    2. Basic authentication is vulnerable to replay attacks. Because basic authentication does not encrypt user credentials, it is important that traffic always be sent over an encrypted SSL session.
    3. There might be valid use cases for a mismatch in SNI / Host headers such as through Page Rules, Load Balancing, or Workers, which all offer HTTP Host Header overrides.
- **Process**:
    1. Use Transform rules or Workers to add an HTTP Auth Header.
    2. Configure your origin server to restrict access based on the HTTP Auth Header (or perform HTTP Basic Authentication).
    3. Configure your origin server to restrict access based on the HTTP Host Header. Specifically, only allow requests which contain expected HTTP Host Header values, and reject all other requests.

3. JSON Web Tokens (JWT) Validation

Only allow traffic with the appropriate JWT.

- **Security**: Very secure.

- **Availability**: Some customers.
- **Challenges**:
  - Requires either installing incremental software or modifying application code.
  - Lots of manual work.
- **Resources**:
  - Validate JWTs for an Access application
  - Validate JWTs for an API

Transport Layer

Authenticated Origin Pulls

Authenticated Origin Pulls helps ensure requests to your origin server come from the Cloudflare network.

- **Security**: Very secure.
- **Availability**: All customers.
- **Challenges**:
  - Requires Full or Full (strict) encryption modes.
  - Requires more configuration efforts for application and server, such as uploading a certificate and configuring the server to use it.
  - For more strict security, you should upload your own certificate. Although Cloudflare provides you a certificate for easy configuration, this certificate only guarantees that a request is coming from the Cloudflare network.
  - Not scalable for large numbers of origin servers.

Cloudflare Tunnel (SSH / RDP)

Cloudflare Tunnel connects your resources to Cloudflare without a publicly routable IP address, by creating an outbound-only connections to Cloudflare's global network.

- **Security**: Very secure.
- **Availability**: All customers.
- **Challenges**: Requires installing the `cloudflared` daemon on origin server or virtual machine.

Network Layer

Allowlist Cloudflare IP addresses

Explicitly block all traffic that does not come from Cloudflare IP addresses (or the IP addresses of your trusted partners, vendors, or applications).

- **Security**: Moderately secure.
- **Availability**: All customers.
- **Challenges**:
  - Requires allowlisting Cloudflare IP ranges at your origin server.
  - Vulnerable to IP spoofing.

Cloudflare Network Interconnect

Cloudflare Network Interconnect allows you to connect your network infrastructure directly with Cloudflare – rather than using the public Internet – for a more reliable and secure experience.

- **Security**: Very secure.
- **Availability**: Enterprise-only.
- **Challenges**
    - Requires some networking knowledge.
    - Only applies to some customer use cases.

Cloudflare Aegis

Cloudflare Aegis

prevents external connections by providing dedicated egress IP addresses.

- **Security**: Very secure.
- **Availability**: Enterprise-only.
- **Challenges**: Requires network-level firewall policies.

Security Center

1 min read

Cloudflare Security Center brings together our suite of security products, our security expertise, and unique Internet intelligence as a unified security intelligence solution. Security Center enables you to strengthen your security posture by:

- Mapping your cyber attack surface
- Providing asset inventory and discovery
- Identifying potential security risks, misconfigurations, and vulnerabilities
- Helping you to mitigate these risks through remediation in a few clicks

For additional details and help, refer to the Security Center documentation.

Setup

To enable **Security Insights** and perform an initial security scan:

1. Log in to the Cloudflare dashboard
1. and select your account.
2. In the Account Home, go to **Security Center** > **Security Insights**.
3. Under **Enable Security Center scans**, select **Start scan**.

The initial Security Insights scan will start. The initial scan time depends on the number of IT assets in all the domains of your Cloudflare account. When the scan is complete, the status of the page will change from **Scan in Progress** to **Last scan performed on: `<DATE_TIME>`**.

Performance

Improve your application's performance by enabling and optimizing your sites settings.

Objectives

By the end of this module, you will be able to:

- Explain how - just by using Cloudflare - you can increase application performance
- Optimize caching using various Cloudflare settings
- Improve performance using different settings within Speed settings
- Set up Cloudflare Web Analytics for free, privacy-first analytics
- Evaluate other, add-on products that can improve application performance

Default improvements

1 min read

Cloudflare provides a variety of speed improvements by default.

DNS resolution

When your site is using Cloudflare, your site always benefits from Cloudflare's lightning-fast DNS resolution

.

Caching

When your DNS records are proxied through Cloudflare, Cloudflare caches certain types of resources automatically (which improves application performance).

How does caching improve performance?

Caching is the process of storing copies of files in a cache, or temporary storage location, so that they can be accessed more quickly.

When Cloudflare stores content in its cache, the request never needs to go to your application or origin server, which reduces the number of requests and gets content to the user more quickly.

Optimize caching

1 min read

Beyond default caching settings, you can further optimize your cache using different Cloudflare settings.

A few ways to optimize Cloudflare caching include:

- Creating cache rules to customize the cache properties of specific HTTP requests.
- Enabling the Tiered Cache feature, which dramatically increases cache hit ratios.
- Reviewing our other various configuration options, which may vary based on your plan and application setup.

Optimize analytics

2 min read

Web analytics let you measure user behavior - pageviews, sessions, and custom events - on your application.

Cloudflare offers two ways to improve the privacy and performance of the way you gather these analytics.

Cloudflare Web Analytics

If you want analytics without using third-party tools, check out Cloudflare Web Analytics.

Cloudflare Web Analytics provides free, privacy-first analytics for your website without changing your DNS or using Cloudflare's proxy. Cloudflare Web Analytics helps you understand the performance of your web pages as experienced by your site visitors.

All you need to enable Cloudflare Web Analytics is a Cloudflare account and a JavaScript snippet on your page to start getting information on page views and visitors. The JavaScript snippet (also known as a beacon) collects metrics using the Performance API, which is available in all major web browsers.

Setup

So long as your traffic is proxied through Cloudflare, setting up Web Analytics only involves a few steps:

1. Log in to the Cloudflare dashboard
1. , and select your account.
2. Select the **Analytics & Logs** drop-down and choose **Web Analytics**.
3. Under **Quick Actions**, select **Add a site**.
4. Select a hostname from the drop-down menu > **Done**.

Access

Once you have enabled Web Analytics, you can review analytics at any time:

1. Log in to the Cloudflare dashboard
1. , and select your account.
2. Select the **Analytics & Logs** drop-down and choose **Web Analytics**.
3. Select your zone.
4. Review the various metrics provided by Cloudflare.

Notifications

Web Analytics uses Cloudflare's Notification service. When enabled, Web Analytics sends you a weekly report with aggregate visits, page views and median page load time for all your sites, so you can monitor their performance.

To get started, add Web Analytics notification on your Cloudflare dashboard. Refer to Cloudflare Notifications to learn more.

Cloudflare Zaraz

If you already use third-party tools on your website, check out Cloudflare Zaraz.

Cloudflare Zaraz gives you complete control over third-party tools and services for your website, and allows you to offload them to Cloudflare's edge, improving the speed and security of your website. With Cloudflare Zaraz you can load tools such as analytics tools, advertising pixels and scripts, chatbots, marketing automation tools, and more, in the most optimized way.

Cloudflare Zaraz is built for speed, privacy, and security, and you can use it to load as many tools as you need, with a near-zero performance hit.