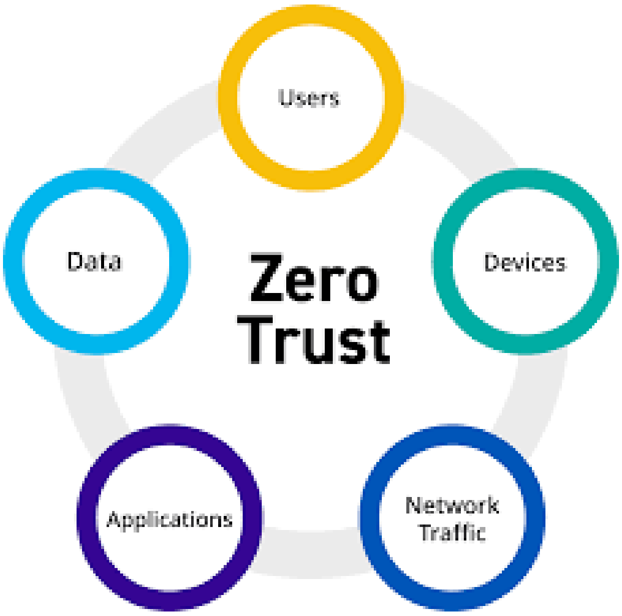
ZERO TRUST

NAVEEN KUMAR N



Zero Trust?

- Zero Trust is a security model that assumes that all network traffic and access requests are potentially malicious and should not be trusted by default. Instead, it requires authentication and authorization for all users and devices before granting access to resources, regardless of their location or whether they are inside or outside the organization's network perimeter.

- The Zero Trust model is based on the principle of "never trust, always verify". It requires continuous authentication, authorization, and monitoring of all devices, users, and traffic, regardless of whether they are within the organization's network or outside it. This approach assumes that any device or user can be compromised and that all access requests should be treated with caution.

- In a Zero Trust environment, access controls are implemented at various levels, including the network, application, and data levels. This ensures that only authorized users and devices can access the resources they need to do their jobs, and that access is continuously monitored and evaluated to detect and respond to any potential security threats.

Some key components of a Zero Trust model include:

- **Multi-factor authentication:** This requires users to provide multiple forms of authentication, such as a password and a biometric scan, before being granted access to a resource.

- **Micro-segmentation:** This involves dividing the network into smaller segments and implementing access controls and monitoring at each segment to limit the potential impact of any security breaches.

- **Least privilege access**: This restricts user access to only the resources and applications that are necessary for their job functions, reducing the potential damage that can be caused by a compromised account.

- **Continuous monitoring and analysis:** This involves monitoring all network traffic and access requests, as well as continuously analyzing and evaluating user behavior to detect and respond to potential security threats.

Benefits:

The Zero Trust model offers several benefits for organizations looking to improve their security posture and protect their critical assets from potential security threats. Some of the key benefits of Zero Trust include:

- **Reduced risk of data breaches:** The Zero Trust model assumes that all access requests are potentially malicious, so it requires continuous authentication and authorization of all users and devices. This reduces the risk of data breaches by limiting access to sensitive data and ensuring that only authorized users can access it.

- **Improved visibility and control:** The Zero Trust model provides organizations with greater visibility and control over their network traffic and access requests. This allows them to identify and respond to potential security threats more quickly and effectively.

- **Simplified compliance:** The Zero Trust model can help organizations simplify their compliance efforts by providing a framework for implementing and enforcing security policies and access controls. This can help organizations meet regulatory requirements and avoid potential penalties.

- **Enhanced user experience:** The Zero Trust model can improve the user experience by providing users with seamless access to the resources they need, regardless of their location or device. This can increase productivity and collaboration while maintaining a high level of security.

- **Protection against insider threats:** The Zero Trust model can help organizations protect against insider threats by limiting access to sensitive data and ensuring that users only have access to the resources they need to do their jobs.

- **Better defense against advanced threats:** The Zero Trust model is designed to protect against advanced threats such as phishing, malware, and ransomware by continuously monitoring and analyzing network traffic and user behavior.

PAM(Pluggable Authentication Modules)

- PAM stands for Pluggable Authentication Modules. It is a framework used on Unix-like operating systems to provide a flexible mechanism for authentication. PAM allows system administrators to configure various authentication methods, such as passwords, biometrics, smart cards, or two-factor authentication, to be used for login, access control, and other authentication-related tasks.

- PAM provides a set of APIs that allow authentication-related tasks to be delegated to dynamically loaded modules. These modules can implement different authentication methods, such as password verification, token authentication, or biometric authentication. The PAM framework includes a set of pre-defined modules that can be used out of the box, but system administrators can also develop their own custom modules.

- When a user attempts to authenticate, the PAM framework loads and executes the configured authentication modules in a defined order. Each module returns a result indicating whether authentication succeeded, failed, or if further processing is required. Based on the results, PAM determines whether the user is allowed to proceed with the requested action.

- PAM is widely used on Unix and Linux systems, and it provides a powerful and flexible mechanism for implementing authentication and access control. Its pluggable architecture allows system administrators to easily add or remove authentication methods without modifying the underlying system code. This makes it an essential component of many security-conscious systems.

- PAM is used in various system components and applications that require authentication, including login, SSH, su, sudo, FTP, and many others. It is widely used on Unix and Linux systems and is an essential component of many security-conscious systems.

Some examples of how PAM is used include:

- **Login:** When a user logs in to a Unix system, PAM is used to authenticate the user and verify their credentials. This includes checking their password, biometric data, or other authentication factors.

- **SSH:** PAM is used to authenticate remote users who access the system via SSH. This includes checking their credentials and ensuring that they have the necessary privileges to perform the requested actions.

- **Sudo:** PAM is used to authenticate users who use the sudo command to perform privileged actions. This ensures that only authorized users can perform administrative tasks on the system.

- **FTP:** PAM is used to authenticate users who access the system via FTP. This includes verifying their credentials and ensuring that they have the necessary permissions to access the requested files.

Task:

Let’s have an hands-on experience on the zero-trust.

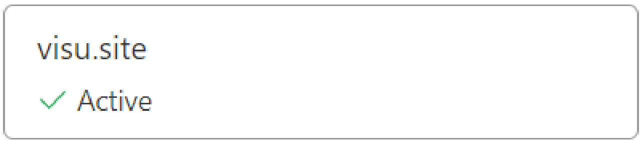
Pre-requisites:

1. Any Domain
2. Cloud flare account

Step1: Login to Cloud flare account and check the status of you site(active/inactive)

Step 2: If it brought the domain from other vendors just transfer the domain to the cloud flare.

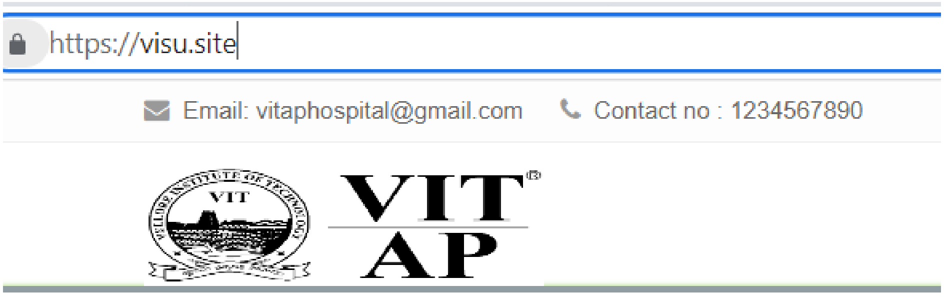
Ex: here I going to demonstrate with visu.site.



visu.site

✓ Active

Step3: Lets request the domain and check for results.



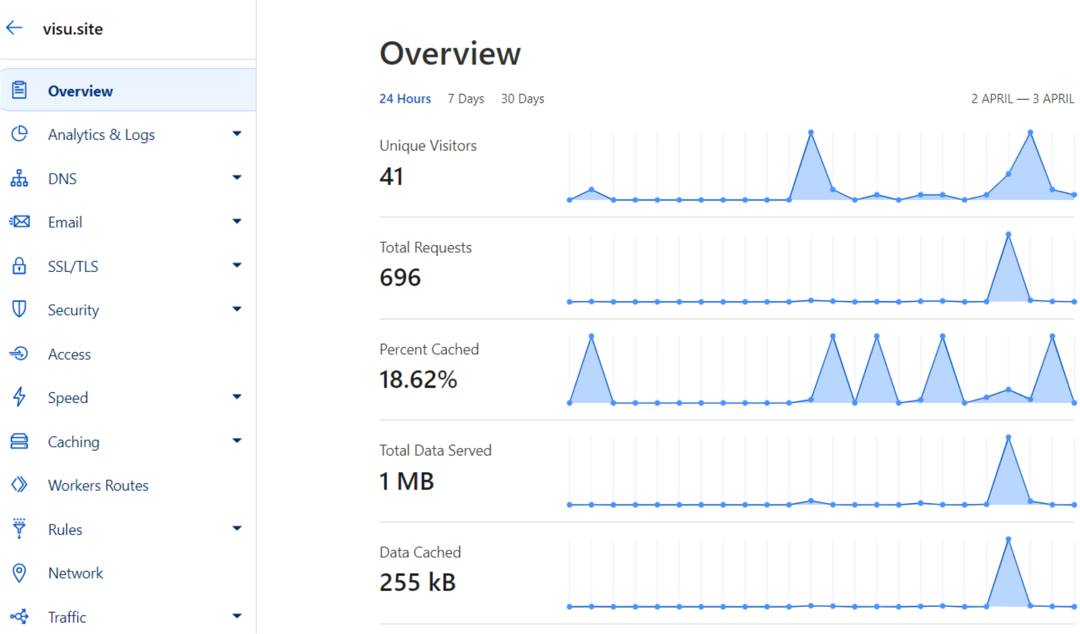
https://visu.site

Email: vitaphospital@gmail.com   Contact no : 1234567890

As we can see that when we gave the url it redirected to the web portal on visu.site.But, now I want to restrict the site to only particular email id's.

You can choose want are the restrictions you can apply to your web portal. It is explained in the upcoming steps.

Step 4: under the overview you can visuilize the total requests , Unique Visitors, Percent cached and etc.



Overview

Unique Visitors 41

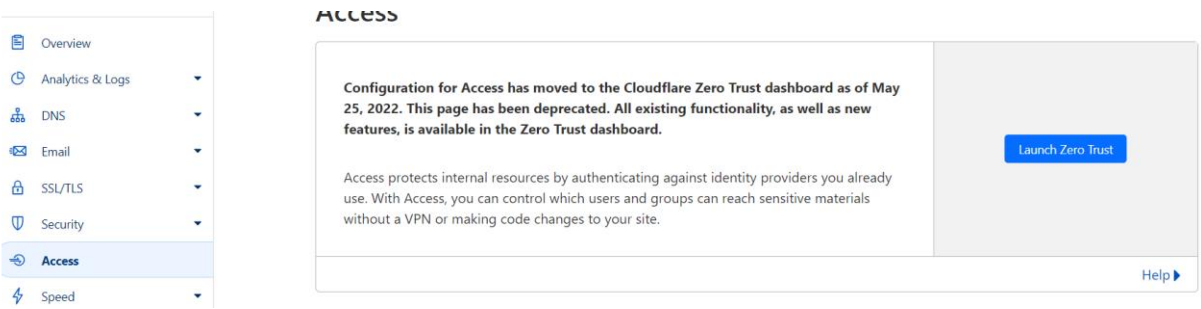
Total Requests 696

Percent Cached 18.62%

Total Data Served 1 MB

Data Cached 255 kB

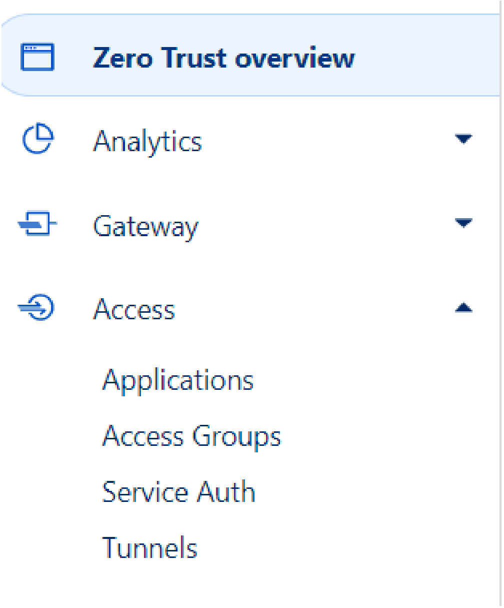
Step 5: Now select the Access on the left listed menu to launch the Zero trust.



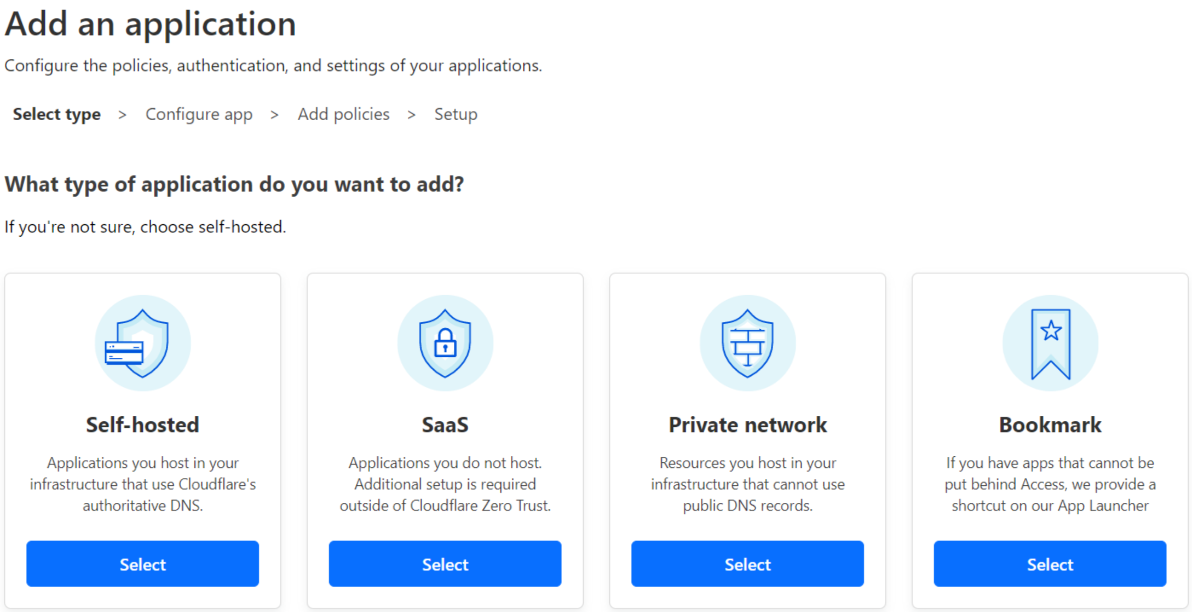
Configuration for Access has moved to the Cloudflare Zero Trust dashboard as of May 25, 2022. This page has been deprecated. All existing functionality, as well as new features, is available in the Zero Trust dashboard.

Access protects internal resources by authenticating against identity providers you already use. With Access, you can control which users and groups can reach sensitive materials without a VPN or making code changes to your site.

Step 6: Now , we redirected to the Zero Trust Overview page. As we need to restict the access the users for our application select the Application under the Access menu.



Step 7: As we are hosting our application we need to select the Self-hosted plafform which means our application use’s cloudflare’s authoritative DNS.



Add an application

Configure the policies, authentication, and settings of your applications.

What type of application do you want to add?

If you're not sure, choose self-hosted.

Self-hosted

Applications you host in your infrastructure that use Cloudflare's authoritative DNS.

SaaS

Applications you do not host. Additional setup is required outside of Cloudflare Zero Trust.

Private network

Resources you host in your infrastructure that cannot use public DNS records.

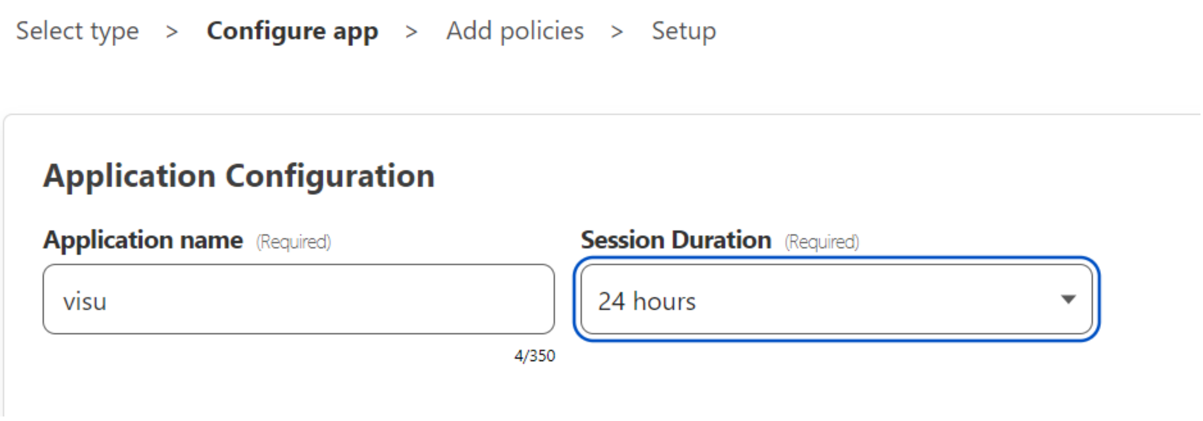
Bookmark

If you have apps that cannot be put behind Access, we provide a shortcut on our App Launcher

Step 8: Under the Application Configuration name the application to visuilize the data of the particular application.

Ex. If I have more than 3 application need to monitor individualy I need to provide unique Application name:

And restrict the client session time. You can terminate the the client session with changing the Session duration.



Select type > **Configure app** > Add policies > Setup

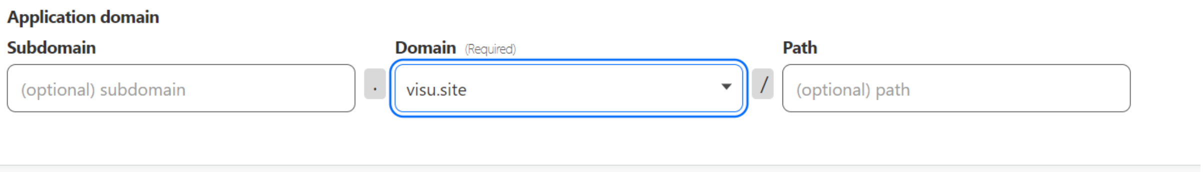
Application Configuration

Application name (Required): visu

4/350

Session Duration (Required): 24 hours

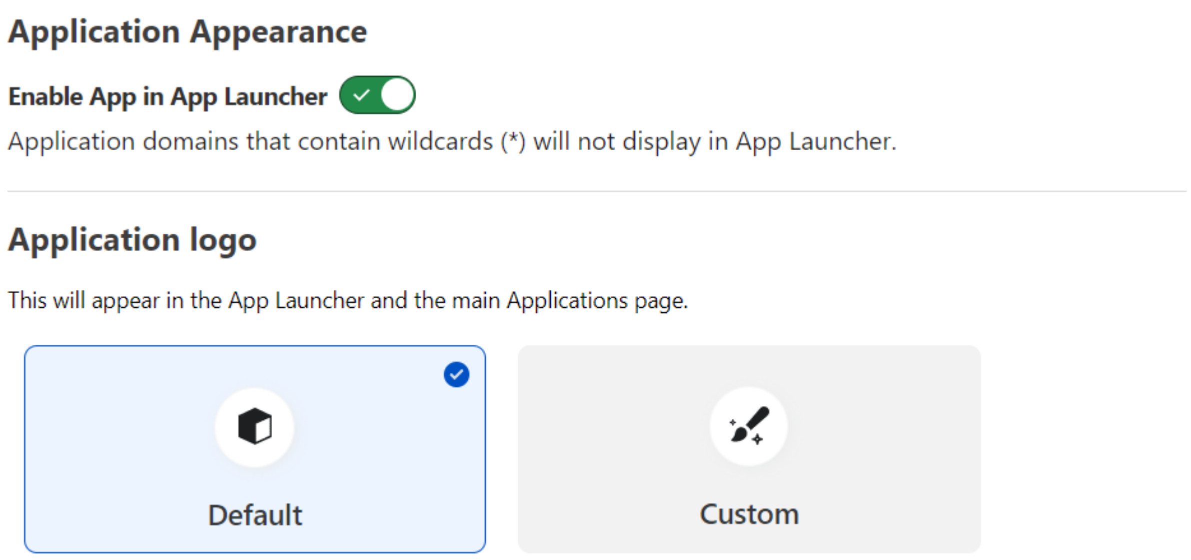
Step 9: we need to provide the application url so that the configuration will be applied to that particular application.



Application domain

Subdomain: (optional) subdomain . **Domain** (Required): visu.site / **Path**: (optional) path

Step 10: When ever the clients requests for the application you can customize the Application Log to authenticate.



Application Appearance

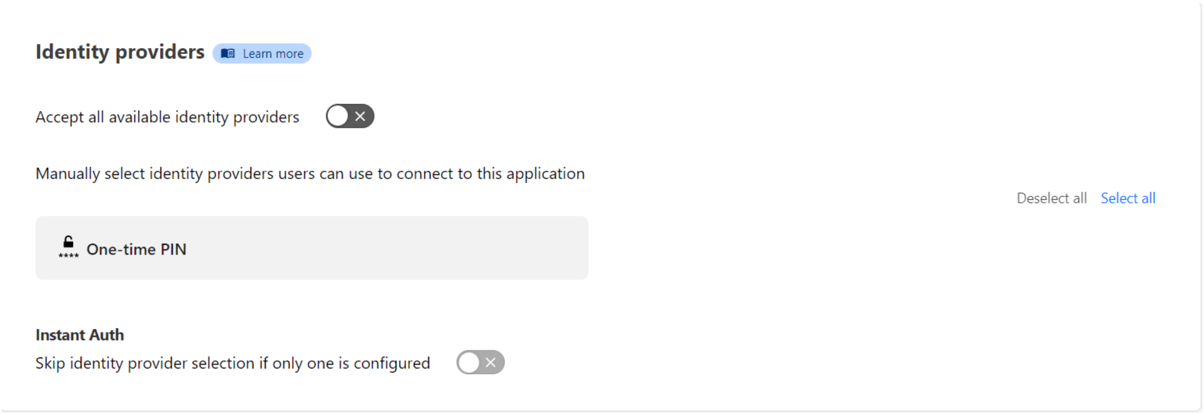
Enable App in App Launcher ✓

Application domains that contain wildcards (*) will not display in App Launcher.

Application logo

This will appear in the App Launcher and the main Applications page.

Step 11: For an Identity Providers we can customize on accept all the available identity or go with the ONE-time PIN.



Identity providers Learn more

Accept all available identity providers ✕

Manually select identity providers users can use to connect to this application

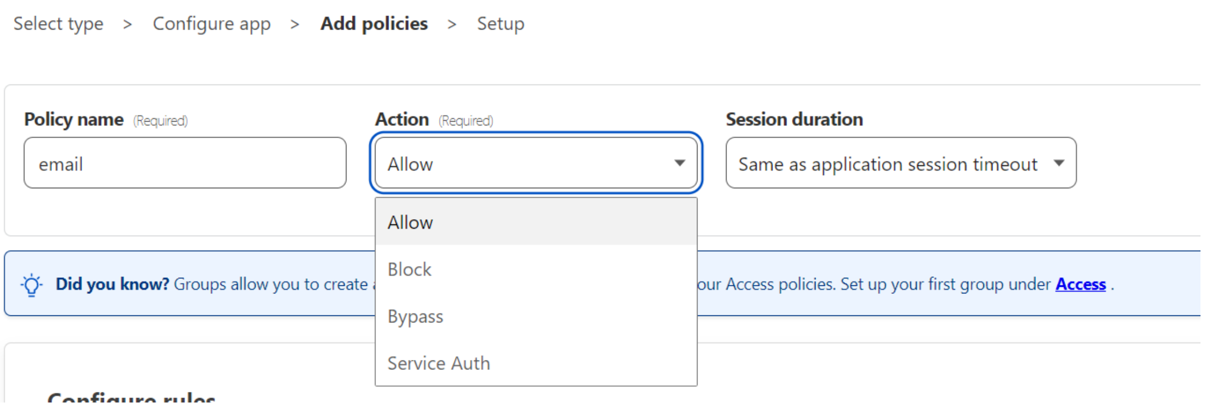
One-time PIN

Instant Auth

Skip identity provider selection if only one is configured ✕

Step 12: when coming to adding policies . Write a policy name and provide a particular action to take on the reguests to Allow ,Block,Bypass etc. And alos we can customize the Session Duration.

Here I’m writing a policy to allow only particular persons with an given Email.id



Policy name (Required): email

Action (Required): Allow

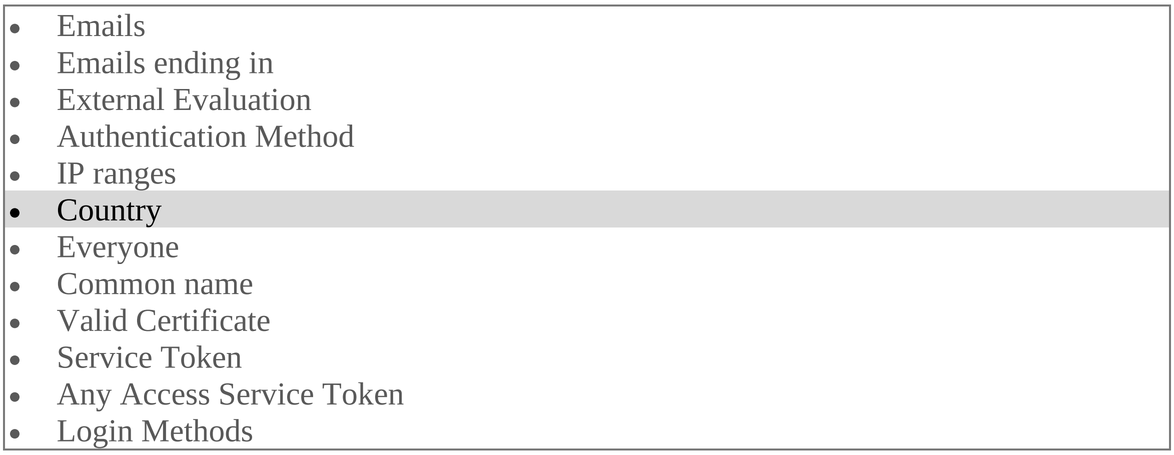
Session duration: Same as application session timeout

Did you know? Groups allow you to create our Access policies. Set up your first group under **Access** .

Configure rules

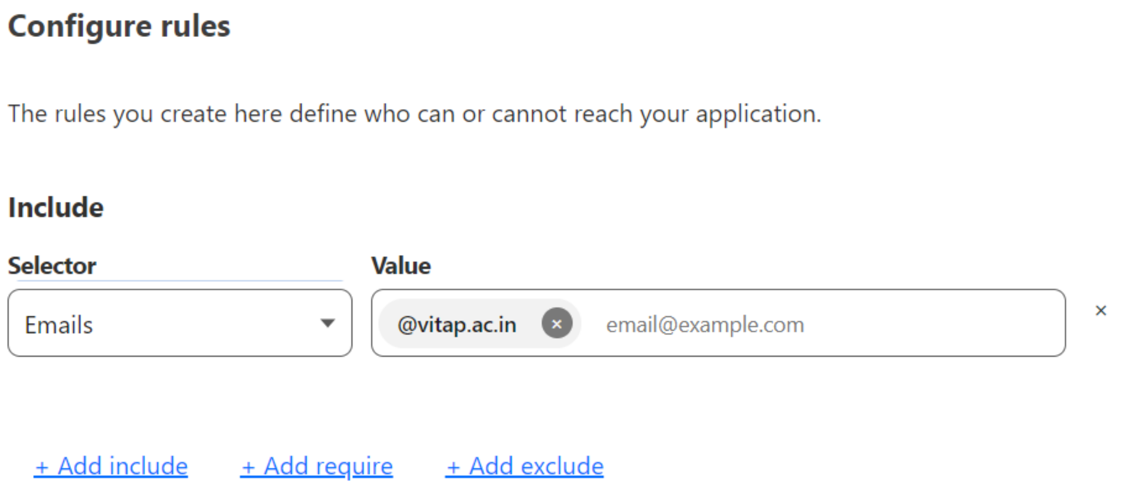
Step 13:Under the policy configuration rules we can select on what feature we can block or allow or bypass the request .

Here are the list of all Selectors



- Emails
- Emails ending in
- External Evaluation
- Authentication Method
- IP ranges
- Country
- Everyone
- Common name
- Valid Certificate
- Service Token
- Any Access Service Token
- Login Methods

I have selected Emails and customized rule on the only users who's email ends with the @vitap.ac.in can access the application.(You can add more than one feature)



Configure rules

The rules you create here define who can or cannot reach your application.

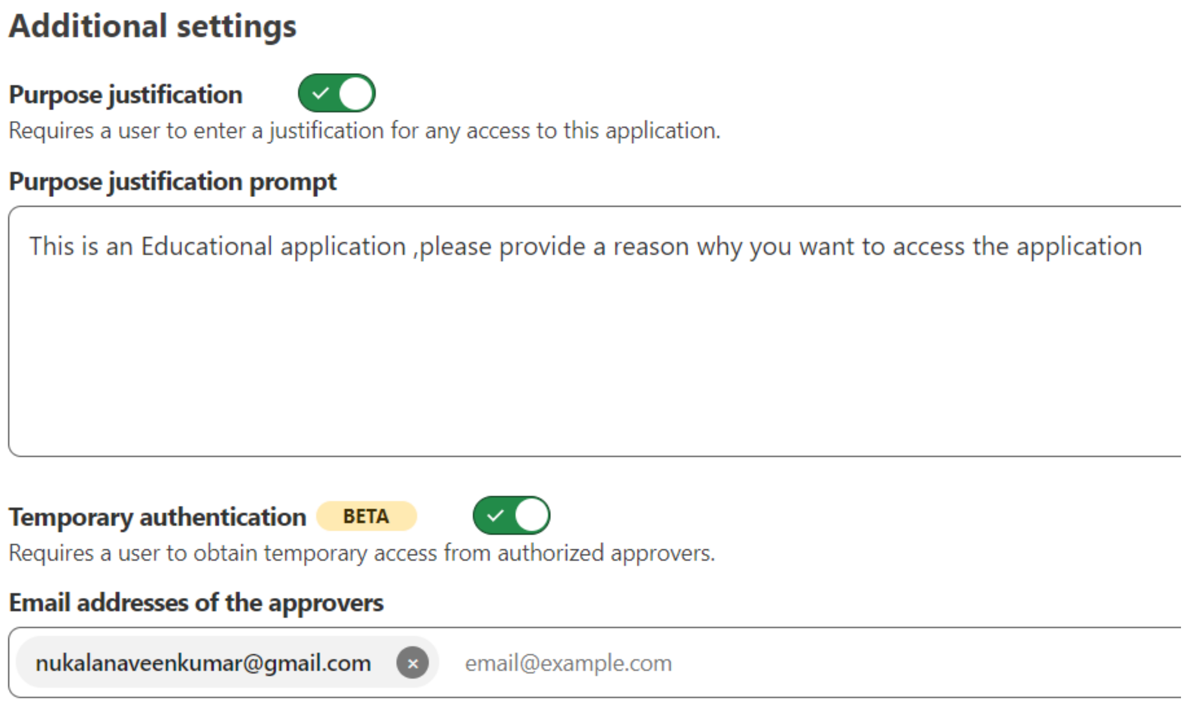
Include

Selector	Value
Emails	@vitap.ac.in

Step 14: As Zero Trust it to maintain the security on each level. Under the Additional Settings you can provide an pop-up on the application when ever any user trying to access to application.

Here I have written a messge and to accepts the requests I have set my mail.id to approve the request.

Which means for every request the customer need to provide an reasonable answer and I need to decide to approve the request to access the resource or to terminate the request.



Additional settings

Purpose justification ✓

Requires a user to enter a justification for any access to this application.

Purpose justification prompt

This is an Educational application ,please provide a reason why you want to access the application

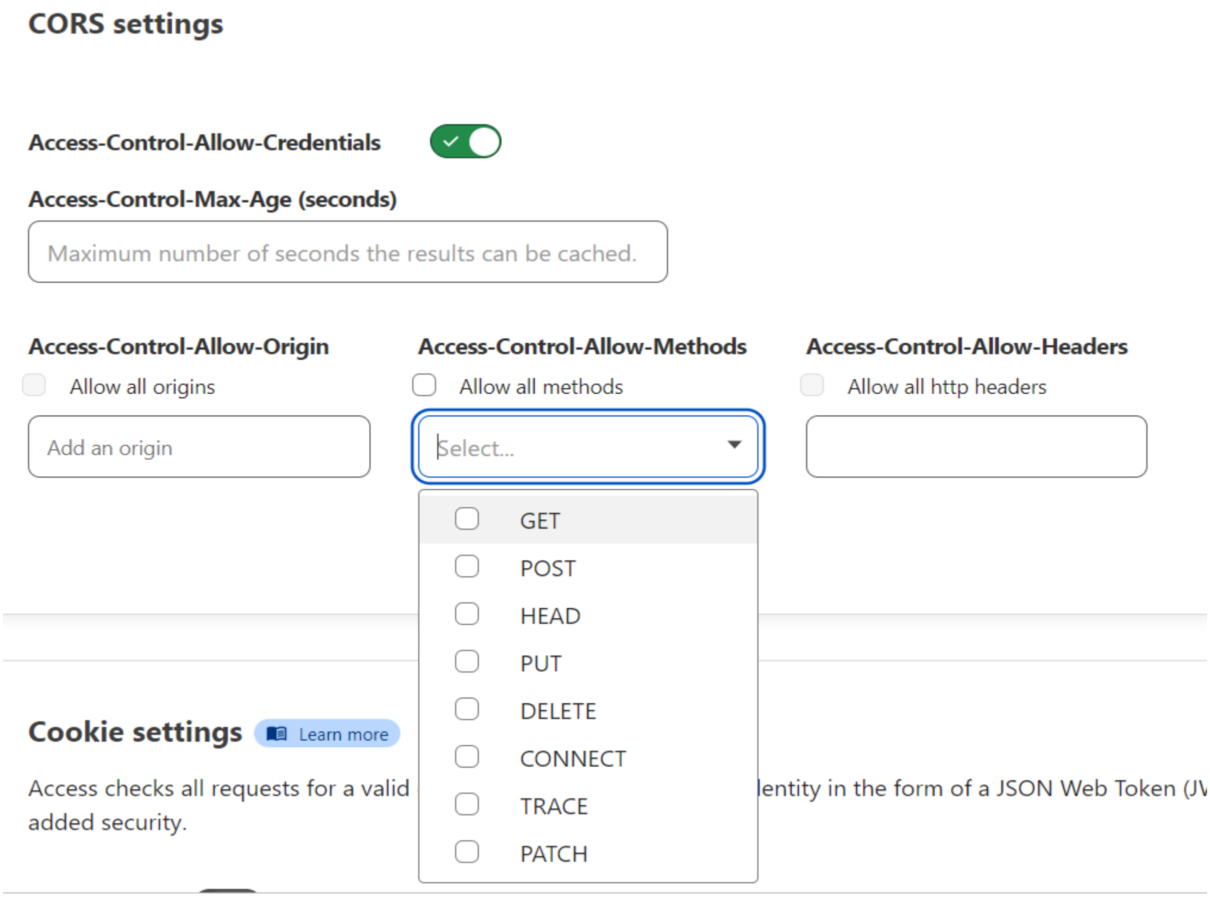
Temporary authentication BETA ✓

Requires a user to obtain temporary access from authorized approvers.

Email addresses of the approvers

nukalanaveenkumar@gmail.com × email@example.com

Step 15: we can customize all the request in the CORS settings.Which means which methods requests we need to allow etc.



CORS settings

Access-Control-Allow-Credentials ✓

Access-Control-Max-Age (seconds)

Maximum number of seconds the results can be cached.

Access-Control-Allow-Origin

□ Allow all origins

Add an origin

Access-Control-Allow-Methods

□ Allow all methods

Select...

□ GET
□ POST
□ HEAD
□ PUT
□ DELETE
□ CONNECT
□ TRACE
□ PATCH

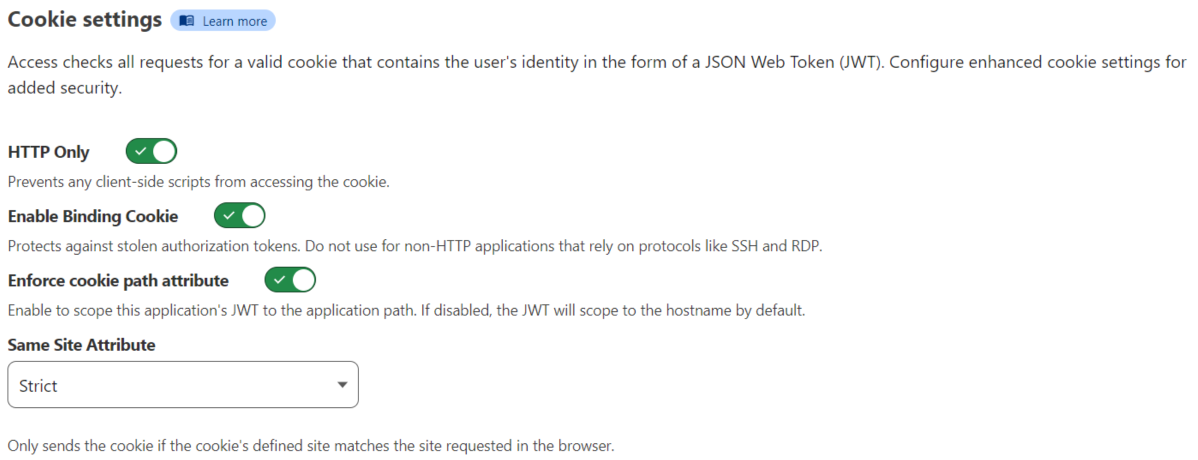
Access-Control-Allow-Headers

□ Allow all http headers

Cookie settings Learn more

Access checks all requests for a valid lentity in the form of a JSON Web Token (JV added security.

Step 16: we can set the cookie setting on the requests. And select the Add application.



Cookie settings Learn more

Access checks all requests for a valid cookie that contains the user's identity in the form of a JSON Web Token (JWT). Configure enhanced cookie settings for added security.

HTTP Only ✓

Prevents any client-side scripts from accessing the cookie.

Enable Binding Cookie ✓

Protects against stolen authorization tokens. Do not use for non-HTTP applications that rely on protocols like SSH and RDP.

Enforce cookie path attribute ✓

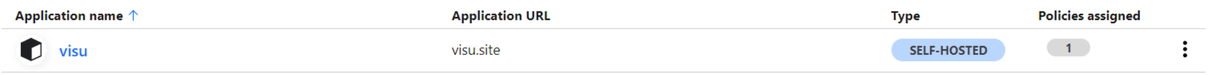
Enable to scope this application's JWT to the application path. If disabled, the JWT will scope to the hostname by default.

Same Site Attribute

Strict

Only sends the cookie if the cookie's defined site matches the site requested in the browser.

Under the application we can see our added application



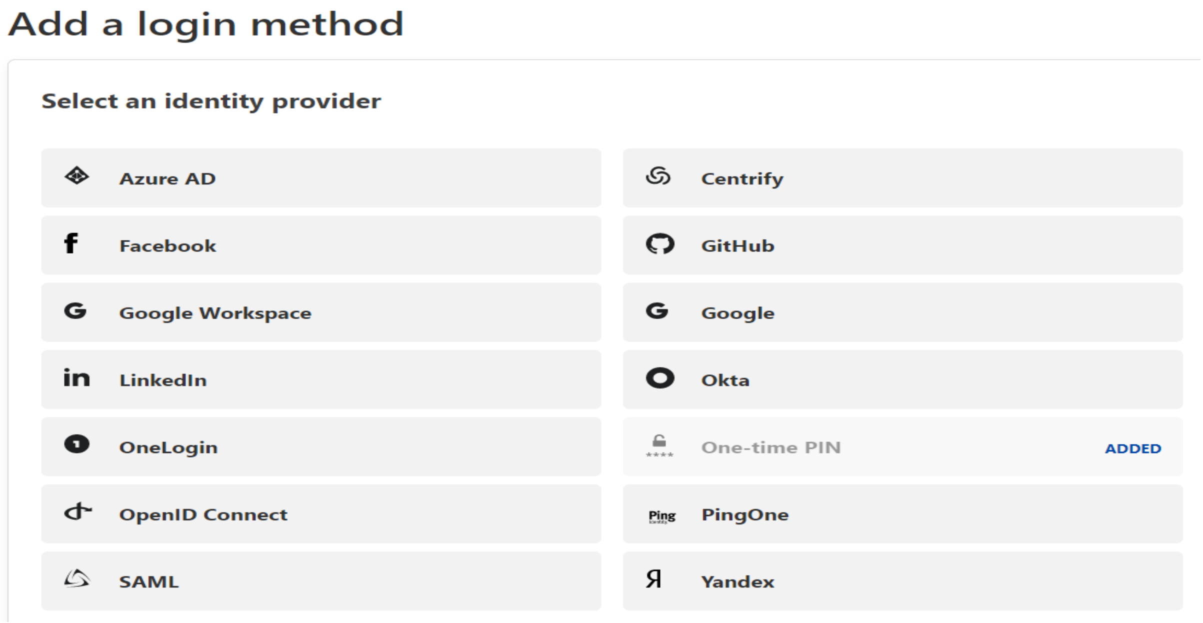
Application name ↑	Application URL	Type	Policies assigned
visu	visu.site	SELF-HOSTED	1

If you want to add the feature to application not only with the ONE-TIME password you can Customize and apply the features as per the requirments.

For that

- ➢ Go to settings
- ➢ Select authentication
- ➢ Login Methods ➔ add new

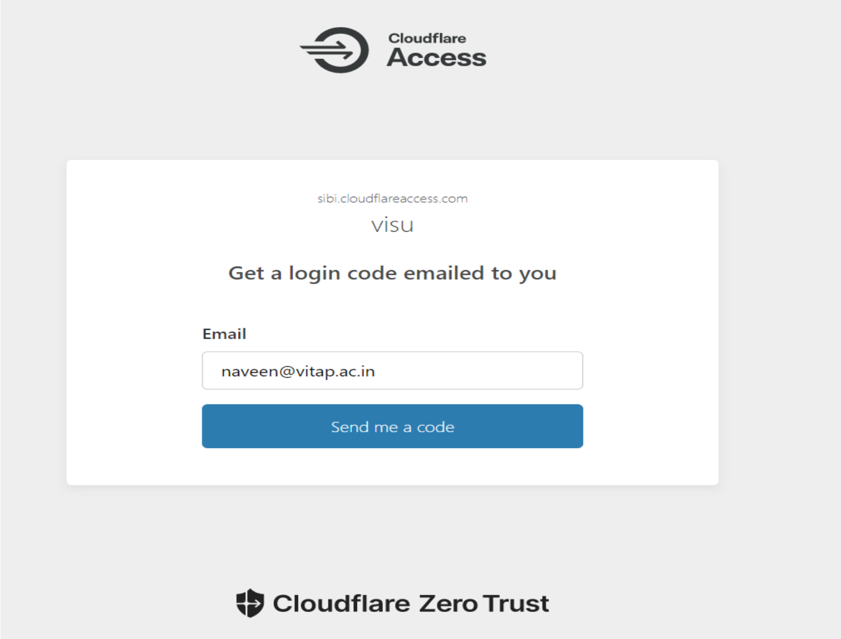
From here you can add the Security verification.



Add a login method

Select an identity provider

Step 17: Now let’s acess the visu.site.



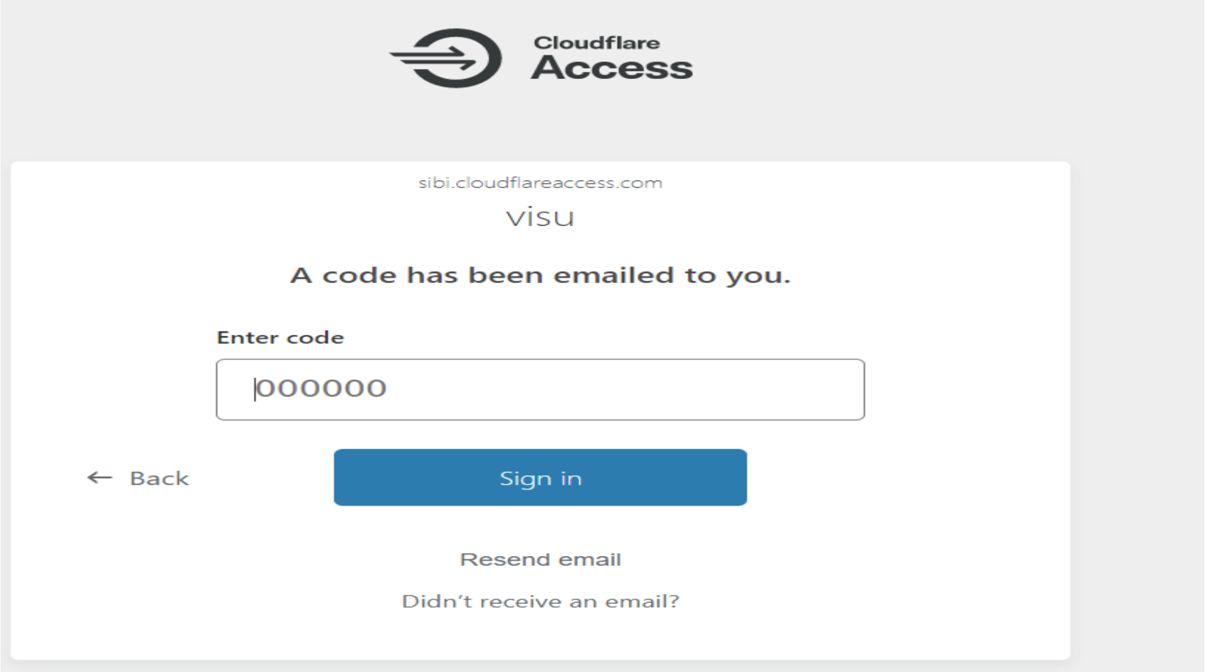
sibi.cloudflareaccess.com

visu

Get a login code emailed to you

Email: naveen@vitap.ac.in

As you can see a pop up window is displayed asking for the email id to access the resource and he need to provide the ONE-TIME password to access the resource on the Authentication



sibi.cloudflareaccess.com

visu

A code has been emailed to you.

Enter code: 000000

As there are many features with the Zero trust , we can create a tunnels and customize the Access Groups and visialze the logs.

Naveen

THANK YOU