Application Services Assignment

Purpose: This assignment is intended to evaluate your hands-on capabilities, give you a taste for the work we do, and show us how you communicate complex subjects in writing. You can refer to: Application Services, Cloudflare Docs and the Cloudflare Dashboard.

1. Project Overview

Project Name: Application Services Assignment

Created By: Micael Santos

Date: 18/02/2026

Cloud Provider: CloudFlare/AWS

2. Introduction

This document describes the implementation of a secure web application architecture using Cloudflare Application Services, as requested in the assignment.

The solution integrates multiple Cloudflare products including DNS, SSL/TLS, Cloudflare Tunnel, Zero Trust Access, Workers and R2 Storage, together with an AWS EC2 origin environment running a Flask application behind an NGINX reverse proxy.

The goal was to demonstrate the ability to design, deploy and secure an application using Cloudflare services while following best practices and documenting the process clearly.

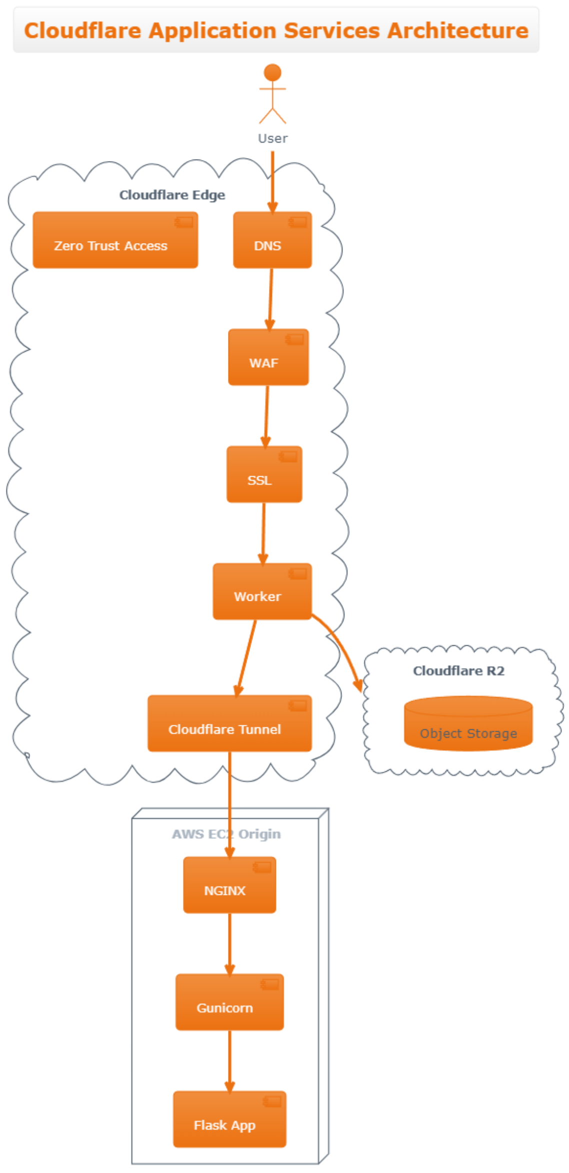


Fig. 1Architecture diagram

3. Architecture Overview

The implemented architecture consists of two main traffic flows:

1. Public Application Flow:

User → Cloudflare Edge → Cloudflare Tunnel → AWS EC2 → NGINX → Gunicorn → Flask Application

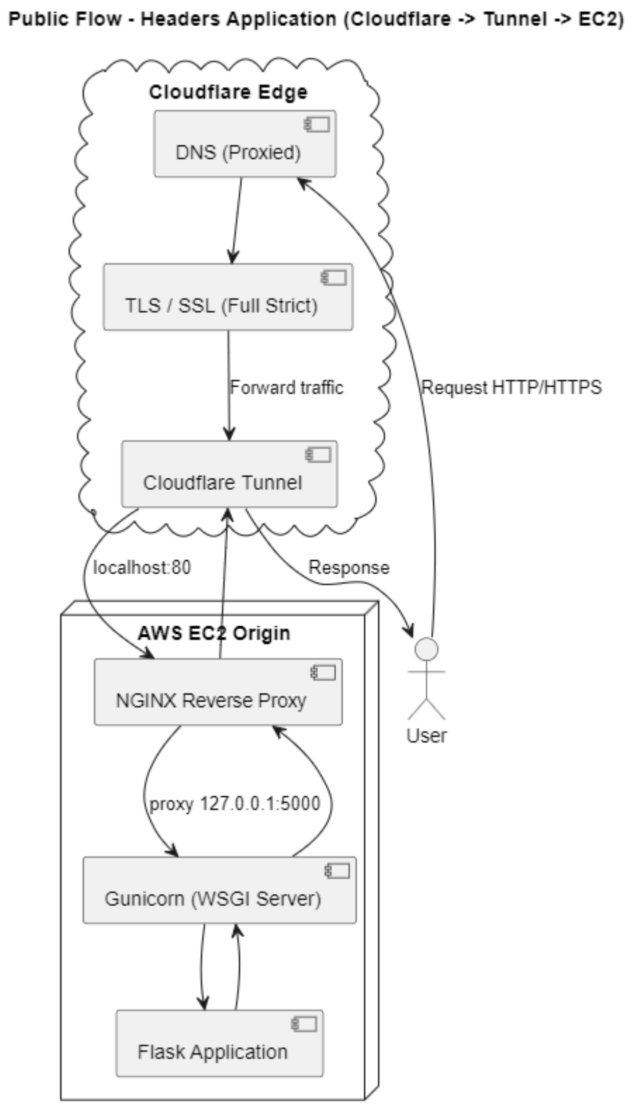


Fig. 2Public diagram

2. Secure Application Flow:

User → Cloudflare Edge → Zero Trust Access → Cloudflare Worker → Private R2 Bucket

The origin server is not directly exposed to the Internet, as all traffic is routed through Cloudflare Tunnel.

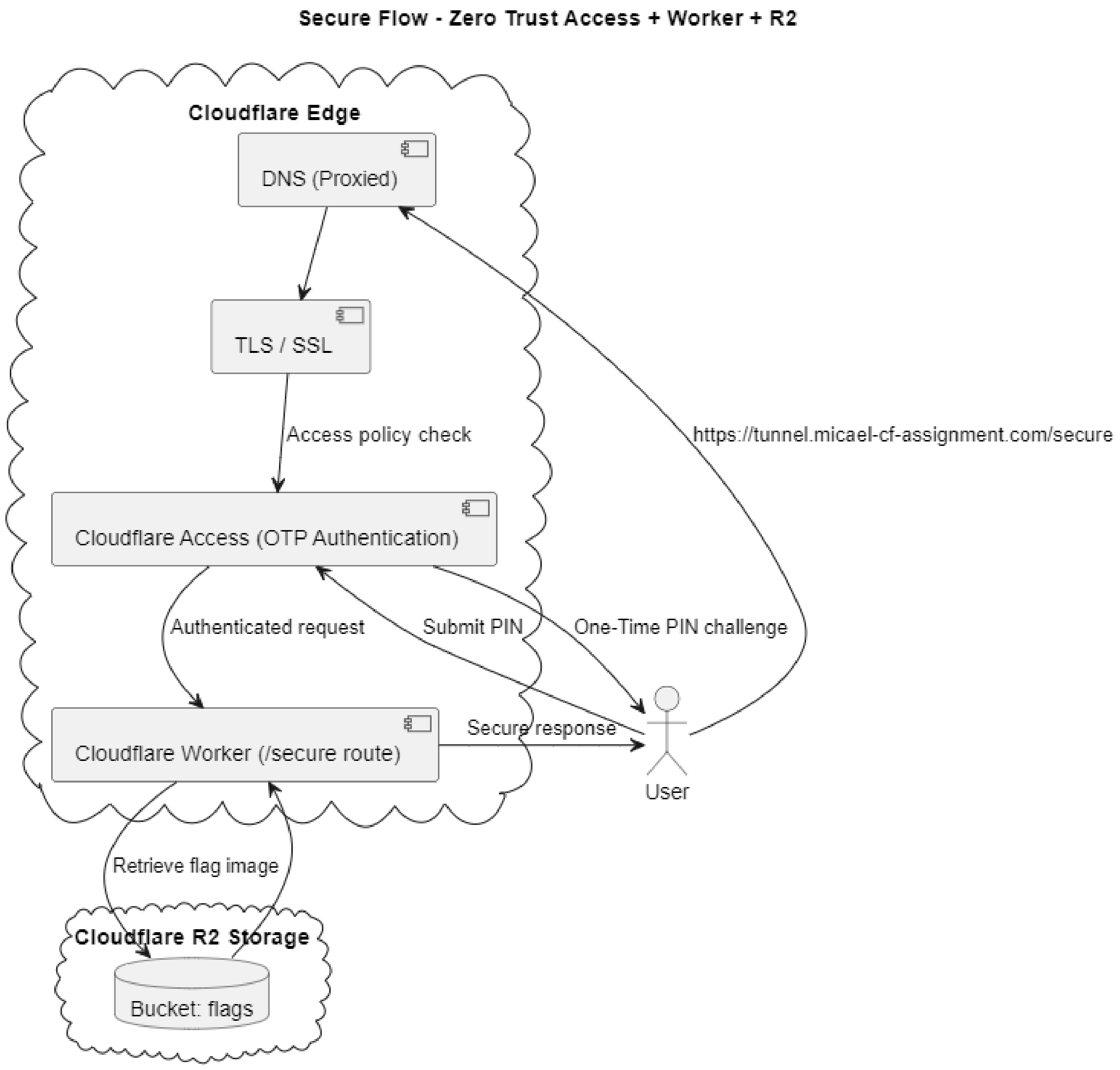


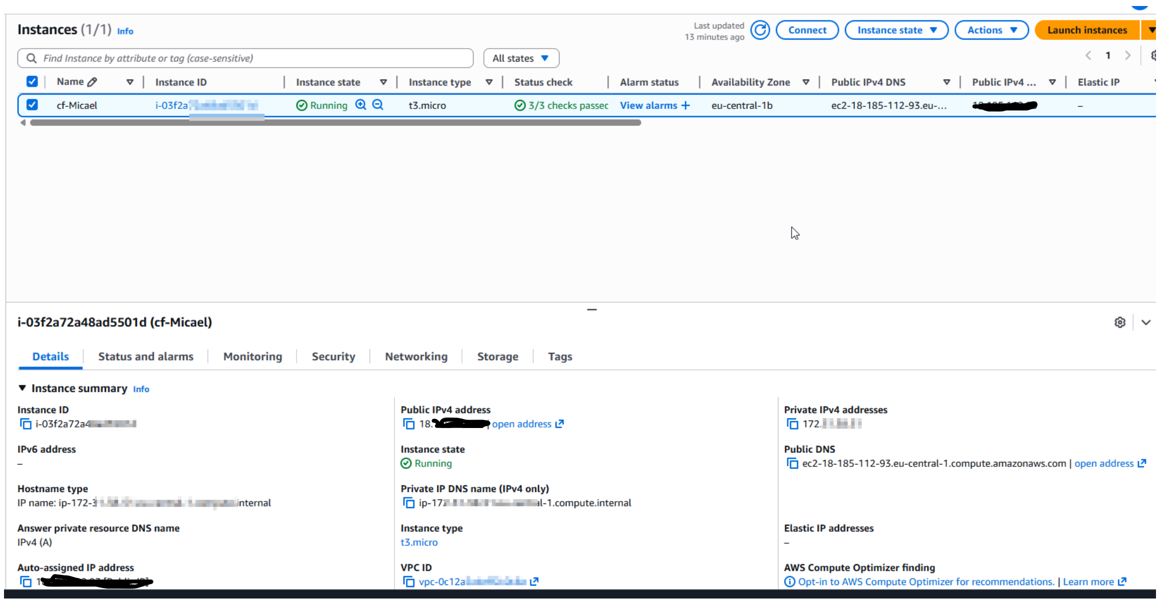
Fig. 3Secure diagram

4. Implementation Steps

Origin Web Server

An origin server was deployed on AWS EC2 using Ubuntu Linux.

A Flask application was created to return all HTTP request headers in the response body. The application was served using Gunicorn as a WSGI server and placed behind an NGINX reverse proxy.


Instances (1/1) Info
Last updated 13 minutes ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
cf-Micael	i-03f2a[redacted]	Running	t3.micro	3/3 checks passec	View alarms +	eu-central-1b	ec2-18-185-112-93.eu-...	[redacted]	–

i-03f2a72a48ad5501d (cf-Micael)

Instance summary Info

Instance ID: i-03f2a72a4[redacted]

Public IPv4 address: 18.[redacted] open address

Private IPv4 addresses: 172.[redacted]

IPv6 address: –

Instance state: Running

Public DNS: ec2-18-185-112-93.eu-central-1.compute.amazonaws.com | open address

Hostname type: IP name: ip-172-3[redacted]internal

Private IP DNS name (IPv4 only): ip-17[redacted]l-1.compute.internal

Answer private resource DNS name: IPv4 (A)

Instance type: t3.micro

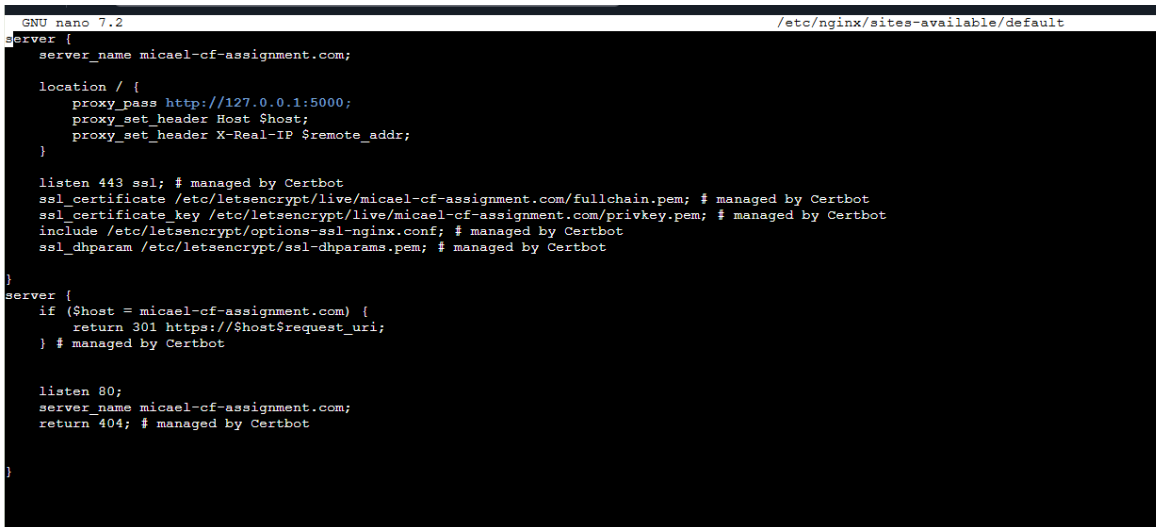
Elastic IP addresses: –

Auto-assigned IP address: 1[redacted]

VPC ID: vpc-0c12a[redacted]

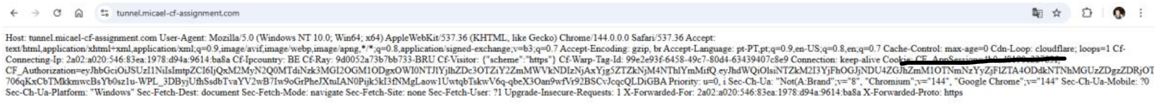
AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Fig. 4EC2 instance running



```
 GNU nano 7.2                                                                    /etc/nginx/sites-available/default
server {
    server_name micael-cf-assignment.com;

    location / {
        proxy_pass http://127.0.0.1:5000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
    }

    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/micael-cf-assignment.com/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/micael-cf-assignment.com/privkey.pem; # managed by Certbot
    include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot

}
server {
    if ($host = micael-cf-assignment.com) {
        return 301 https://$host$request_uri;
    } # managed by Certbot


    listen 80;
    server_name micael-cf-assignment.com;
    return 404; # managed by Certbot


}
```

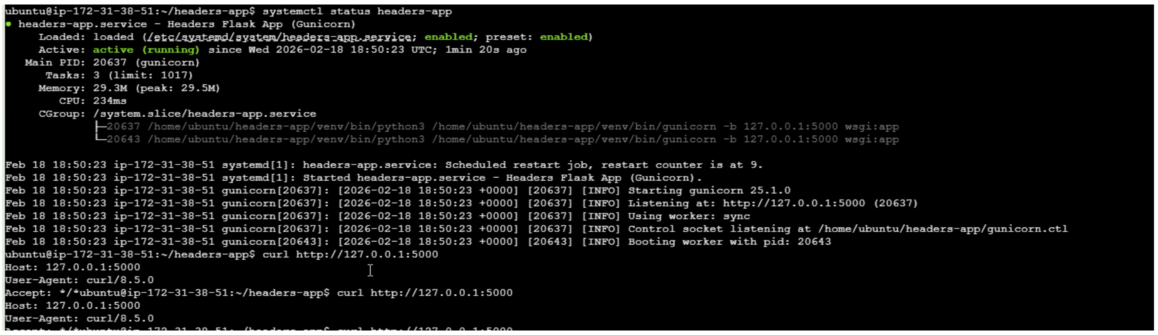
Fig. 5NGINX config file



tunnel.micael-cf-assignment.com

```
Host: tunnel.micael-cf-assignment.com User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7 Accept-Encoding: gzip, br Accept-Language: pt-PT,pt;q=0.9,en-US;q=0.8,en;q=0.7 Cache-Control: max-age=0 Cdn-Loop: cloudflare; loops=1 Cf-
Connecting-Ip: 2a02:a020:546:83ea:1978:d94a:9614:ba8a Cf-Ipcountry: BE Cf-Ray: 9d0052a73b7bb733-BRU Cf-Visitor: {"scheme":"https"} Cf-Warp-Tag-Id: 99e2e93f-6458-49c7-80d4-63439407c8e9 Connection: keep-alive Cook[redacted]
CF_Authorization=eyJhbGciOiJSUzI1NiIsImtpZCI6IjQxM2MyN2Q0MTdiNzk3MGI2OGM1ODgxOWI0NTJlYjlhZDc3OTZiY2ZmMWVkNDIzNjAxYjg5ZTZkNjM4NThlYmMifQ.eyJhdWQiOlsiNTZkM2I3YjFhOGJjNDU4ZGJhZmM1OTNmNzYyZjFlZTA4ODdkNTNhMGUzZDgzZDRjOT
706qKxCbTMkkmwcBsYb0sz1u-WPL_3DByjUfhSsdbTvaYV2wB7Iw9oGrPheJXtuIAN0Pijk5kI3fNMgLaow1UwtqbTskwV6q-qbeX3Oan9wfVr92BSCvJcqcQLDiGBA Priority: u=0, i Sec-Ch-Ua: "Not(A:Brand";v="8", "Chromium";v="144", "Google Chrome";v="144" Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows" Sec-Fetch-Dest: document Sec-Fetch-Mode: navigate Sec-Fetch-Site: none Sec-Fetch-User: ?1 Upgrade-Insecure-Requests: 1 X-Forwarded-For: 2a02:a020:546:83ea:1978:d94a:9614:ba8a X-Forwarded-Proto: https
```

Fig. 6Flask headers output



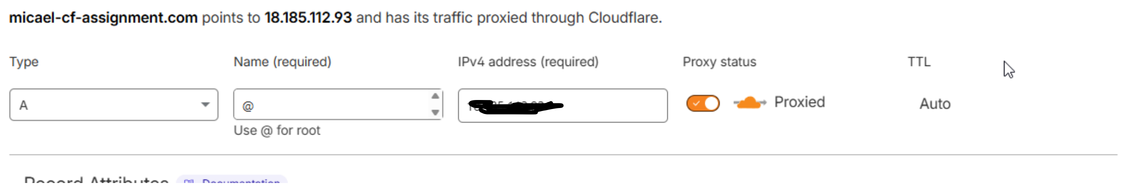
```
ubuntu@ip-172-31-38-51:~/headers-app$ systemctl status headers-app
● headers-app.service - Headers Flask App (Gunicorn)
     Loaded: loaded (/etc/systemd/system/headers-app.service; enabled; preset: enabled)
     Active: active (running) since Wed 2026-02-18 18:50:23 UTC; 1min 20s ago
   Main PID: 20637 (gunicorn)
      Tasks: 3 (limit: 1017)
     Memory: 29.3M (peak: 29.5M)
        CPU: 234ms
     CGroup: /system.slice/headers-app.service
             ├─20637 /home/ubuntu/headers-app/venv/bin/python3 /home/ubuntu/headers-app/venv/bin/gunicorn -b 127.0.0.1:5000 wsgi:app
             └─20643 /home/ubuntu/headers-app/venv/bin/python3 /home/ubuntu/headers-app/venv/bin/gunicorn -b 127.0.0.1:5000 wsgi:app

Feb 18 18:50:23 ip-172-31-38-51 systemd[1]: headers-app.service: Scheduled restart job, restart counter is at 9.
Feb 18 18:50:23 ip-172-31-38-51 systemd[1]: Started headers-app.service - Headers Flask App (Gunicorn).
Feb 18 18:50:23 ip-172-31-38-51 gunicorn[20637]: [2026-02-18 18:50:23 +0000] [20637] [INFO] Starting gunicorn 25.1.0
Feb 18 18:50:23 ip-172-31-38-51 gunicorn[20637]: [2026-02-18 18:50:23 +0000] [20637] [INFO] Listening at: http://127.0.0.1:5000 (20637)
Feb 18 18:50:23 ip-172-31-38-51 gunicorn[20637]: [2026-02-18 18:50:23 +0000] [20637] [INFO] Using worker: sync
Feb 18 18:50:23 ip-172-31-38-51 gunicorn[20637]: [2026-02-18 18:50:23 +0000] [20637] [INFO] Control socket listening at /home/ubuntu/headers-app/gunicorn.ctl
Feb 18 18:50:23 ip-172-31-38-51 gunicorn[20643]: [2026-02-18 18:50:23 +0000] [20643] [INFO] Booting worker with pid: 20643
ubuntu@ip-172-31-38-51:~/headers-app$ curl http://127.0.0.1:5000
Host: 127.0.0.1:5000
User-Agent: curl/8.5.0
Accept: */*ubuntu@ip-172-31-38-51:~/headers-app$ curl http://127.0.0.1:5000
Host: 127.0.0.1:5000
User-Agent: curl/8.5.0
```

Fig. 7systemctl status headers-app

Proxy Through Cloudflare

The domain was onboarded into Cloudflare and DNS records were configured to proxy traffic through Cloudflare's network.

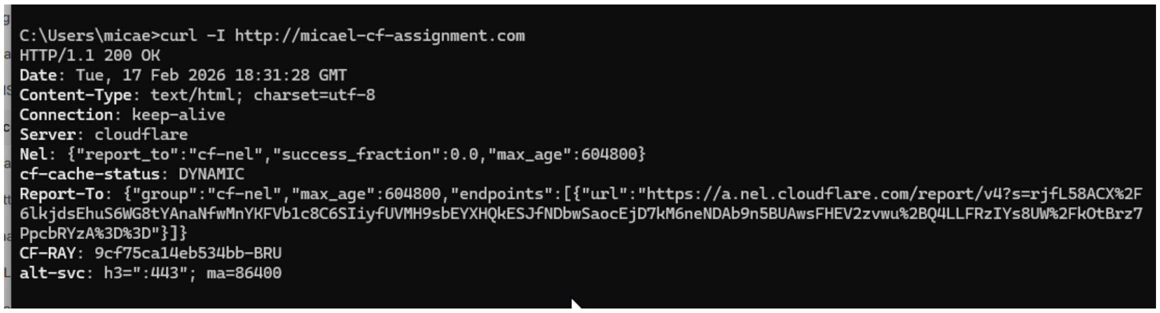
The orange-cloud proxy mode was enabled to ensure traffic passed through Cloudflare services.



micael-cf-assignment.com points to **18.185.112.93** and has its traffic proxied through Cloudflare.

Type	Name (required)	IPv4 address (required)	Proxy status	TTL
A	@ Use @ for root	[redacted]	Proxied	Auto

Fig. 8Cloudflare DNS records



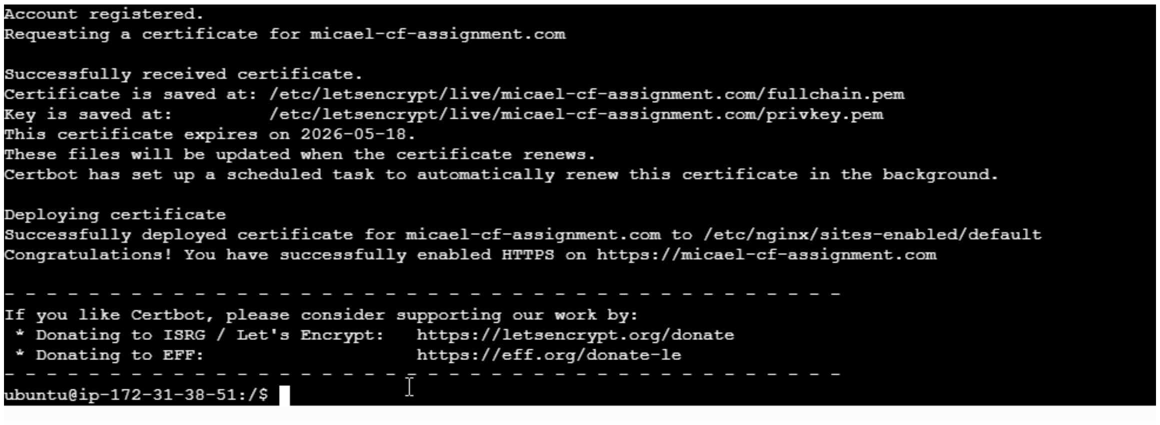
```
C:\Users\micae>curl -I http://micael-cf-assignment.com
HTTP/1.1 200 OK
Date: Tue, 17 Feb 2026 18:31:28 GMT
Content-Type: text/html; charset=utf-8
Connection: keep-alive
Server: cloudflare
Nel: {"report_to":"cf-nel","success_fraction":0.0,"max_age":604800}
cf-cache-status: DYNAMIC
Report-To: {"group":"cf-nel","max_age":604800,"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v4?s=rjfL58ACX%2F
6lkjdsEhuS6WG8tYAnaNfwMnYKFVb1c8C6SIiyfUVMH9sbEYXHQkESJfNDbwSaocEjD7kM6neNDAb9n5BUAwsFHEV2zvwu%2BQ4LLFRzIYs8UW%2FkOtBrz7
PpcbRYzA%3D%3D"}]}
CF-RAY: 9cf75ca14eb534bb-BRU
alt-svc: h3=":443"; ma=86400
```

Fig. 9curl showing Cloudflare headers

Full Strict TLS

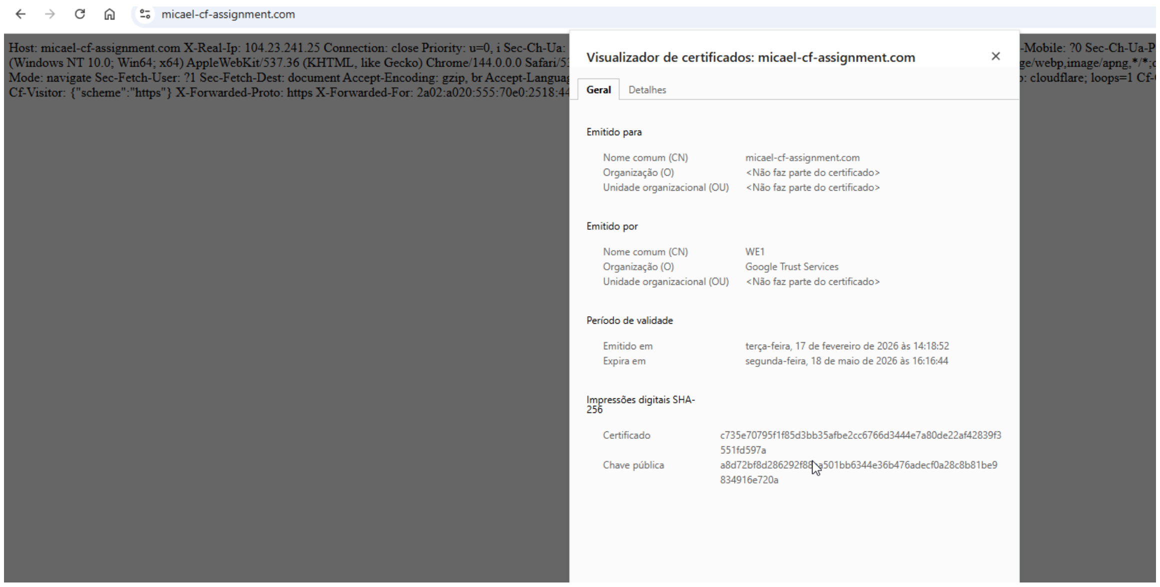
A TLS certificate was generated on the origin server using Let's Encrypt.

Cloudflare SSL mode was configured to Full (Strict), ensuring encrypted communication between Cloudflare and the origin server with certificate validation.



```
Account registered.
Requesting a certificate for micael-cf-assignment.com

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/micael-cf-assignment.com/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/micael-cf-assignment.com/privkey.pem
This certificate expires on 2026-05-18.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for micael-cf-assignment.com to /etc/nginx/sites-enabled/default
Congratulations! You have successfully enabled HTTPS on https://micael-cf-assignment.com

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
If you like Certbot, please consider supporting our work by:
 * Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
 * Donating to EFF:                    https://eff.org/donate-le
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
ubuntu@ip-172-31-38-51:/$
```

Fig. 10App certificate Let's Encrypt



micael-cf-assignment.com

```
Host: micael-cf-assignment.com X-Real-Ip: 104.23.241.25 Connection: close Priority: u=0, i Sec-Ch-Ua:   -Mobile: ?0 Sec-Ch-Ua-P
(Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/5   ge/webp,image/apng,*/*;
Mode: navigate Sec-Fetch-User: ?1 Sec-Fetch-Dest: document Accept-Encoding: gzip, br Accept-Languag   : cloudflare; loops=1 Cf-
Cf-Visitor: {"scheme":"https"} X-Forwarded-Proto: https X-Forwarded-For: 2a02:a020:555:70e0:2518:44
```

Visualizador de certificados: micael-cf-assignment.com

Geral | Detalhes

Emitido para

Nome comum (CN)	micael-cf-assignment.com
Organização (O)	<Não faz parte do certificado>
Unidade organizacional (OU)	<Não faz parte do certificado>

Emitido por

Nome comum (CN)	WE1
Organização (O)	Google Trust Services
Unidade organizacional (OU)	<Não faz parte do certificado>

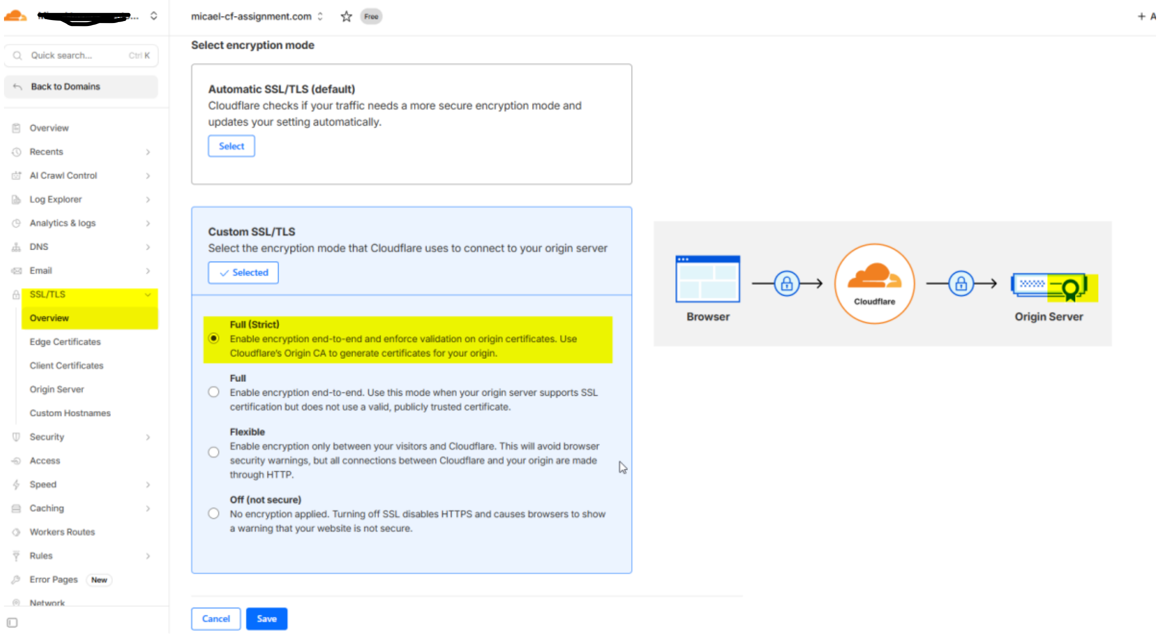
Período de validade

Emitido em	terça-feira, 17 de fevereiro de 2026 às 14:18:52
Expira em	segunda-feira, 18 de maio de 2026 às 16:16:44

Impressões digitais SHA-256

Certificado	c735e70795f1f85d3bb35afbe2cc6766d3444e7a80de22af42839f3551fd597a
Chave pública	a8d72bf8d286292f88[illegible]a501bb6344e36b476adecf0a28c8b81be9834916e720a

Fig. 11Certificate details browser (Cloudflare)


[redacted] micael-cf-assignment.com Free

Select encryption mode

Automatic SSL/TLS (default)
Cloudflare checks if your traffic needs a more secure encryption mode and updates your setting automatically.
Select

Custom SSL/TLS
Select the encryption mode that Cloudflare uses to connect to your origin server
✓ Selected

● **Full (Strict)**
Enable encryption end-to-end and enforce validation on origin certificates. Use Cloudflare's Origin CA to generate certificates for your origin.

○ **Full**
Enable encryption end-to-end. Use this mode when your origin server supports SSL certification but does not use a valid, publicly trusted certificate.

○ **Flexible**
Enable encryption only between your visitors and Cloudflare. This will avoid browser security warnings, but all connections between Cloudflare and your origin are made through HTTP.

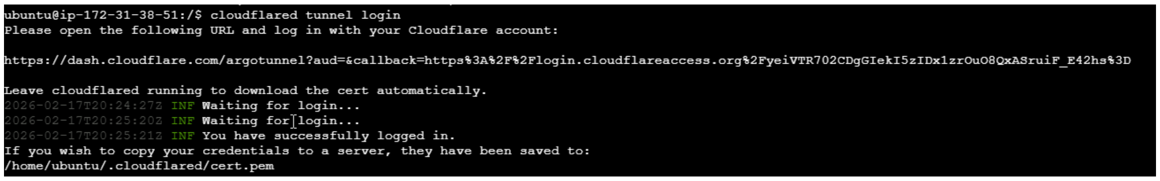
○ **Off (not secure)**
No encryption applied. Turning off SSL disables HTTPS and causes browsers to show a warning that your website is not secure.

Fig. 12Cloudflare SSL settings page

Cloudflare Tunnel

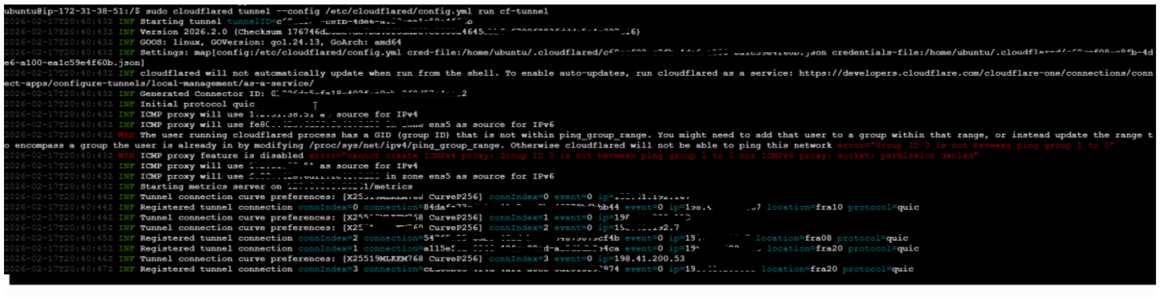
Cloudflare Tunnel was installed and configured on the origin server to securely expose the application without opening inbound firewall ports.

A subdomain tunnel.mydomain.com was configured to route traffic to the local NGINX service.



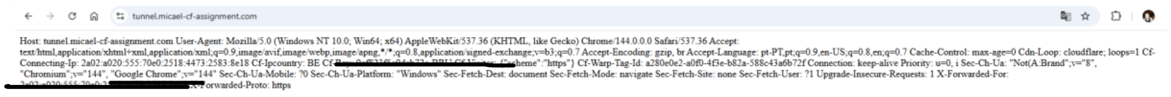
```
ubuntu@ip-172-31-38-51:/$ cloudflared tunnel login
Please open the following URL and log in with your Cloudflare account:

https://dash.cloudflare.com/argotunnel?aud=&callback=https%3A%2F%2Flogin.cloudflareaccess.org%2FyeiVTR702CDgGIekI5zIDx1zrOuO8QxASruiF_E42hs%3D

Leave cloudflared running to download the cert automatically.
2026-02-17T20:24:27Z INF Waiting for login...
2026-02-17T20:25:20Z INF Waiting for login...
2026-02-17T20:25:21Z INF You have successfully logged in.
If you wish to copy your credentials to a server, they have been saved to:
/home/ubuntu/.cloudflared/cert.pem
```

Fig. 13cloudflared running



```
ubuntu@ip-172-31-38-51:/$ sudo cloudflared tunnel --config /etc/cloudflared/config.yml run cf-tunnel
2026-02-17T20:40:43Z INF Starting tunnel tunnelID=c[redacted]-c8fb-4de6-a[redacted]b
2026-02-17T20:40:43Z INF Version 2026.2.0 (Checksum 176746d[redacted]4645[redacted]26)
2026-02-17T20:40:43Z INF GOOS: linux, GOVersion: go1.24.13, GoArch: amd64
2026-02-17T20:40:43Z INF Settings: map[config:/etc/cloudflared/config.yml cred-file:/home/ubuntu/.cloudflared/c[redacted].json credentials-file:/home/ubuntu/.cloudflared/c[redacted]-c8fb-4d
e6-a100-ea1c59e4f60b.json]
2026-02-17T20:40:43Z INF cloudflared will not automatically update when run from the shell. To enable auto-updates, run cloudflared as a service: https://developers.cloudflare.com/cloudflare-one/connections/conn
ect-apps/configure-tunnels/local-management/as-a-service/
2026-02-17T20:40:43Z INF Generated Connector ID: 0[redacted]2
2026-02-17T20:40:43Z INF Initial protocol quic
2026-02-17T20:40:43Z INF ICMP proxy will use 1[redacted] source for IPv4
2026-02-17T20:40:43Z INF ICMP proxy will use fe80[redacted] ens5 as source for IPv6
2026-02-17T20:40:43Z WRN The user running cloudflared process has a GID (group ID) that is not within ping_group_range. You might need to add that user to a group within that range, or instead update the range t
o encompass a group the user is already in by modifying /proc/sys/net/ipv4/ping_group_range. Otherwise cloudflared will not be able to ping this network error="Group ID 0 is not between ping group 1 to 0"
2026-02-17T20:40:43Z WRN ICMP proxy feature is disabled error="cannot create ICMPv4 proxy: Group ID 0 is not between ping group 1 to 0 nor ICMPv6 proxy: socket: permission denied"
2026-02-17T20:40:43Z INF ICMP proxy will use [redacted] as source for IPv4
2026-02-17T20:40:43Z INF ICMP proxy will use [redacted] in zone ens5 as source for IPv6
2026-02-17T20:40:43Z INF Starting metrics server on [redacted]/metrics
2026-02-17T20:40:44Z INF Tunnel connection curve preferences: [X25[redacted] CurveP256] connIndex=0 event=0 ip=[redacted]
2026-02-17T20:40:44Z INF Registered tunnel connection connIndex=0 connection=84da[redacted]bb44 event=0 ip=[redacted] location=fra10 protocol=quic
2026-02-17T20:40:44Z INF Tunnel connection curve preferences: [X25[redacted]68 CurveP256] connIndex=1 event=0 ip=19[redacted]
2026-02-17T20:40:45Z INF Tunnel connection curve preferences: [X25[redacted]68 CurveP256] connIndex=2 event=0 ip=1[redacted]2.7
2026-02-17T20:40:45Z INF Registered tunnel connection connIndex=2 connection=54[redacted]cf4b event=0 ip=19[redacted] location=fra08 protocol=quic
2026-02-17T20:40:45Z INF Registered tunnel connection connIndex=1 connection=a115e[redacted]4ca event=0 ip=19[redacted] location=fra20 protocol=quic
2026-02-17T20:40:46Z INF Tunnel connection curve preferences: [X25519MLKEM768 CurveP256] connIndex=3 event=0 ip=198.41.200.53
2026-02-17T20:40:47Z INF Registered tunnel connection connIndex=3 connection=c[redacted]874 event=0 ip=19[redacted] location=fra20 protocol=quic
```

Fig. 14Tunnel running



tunnel.micael-cf-assignment.com

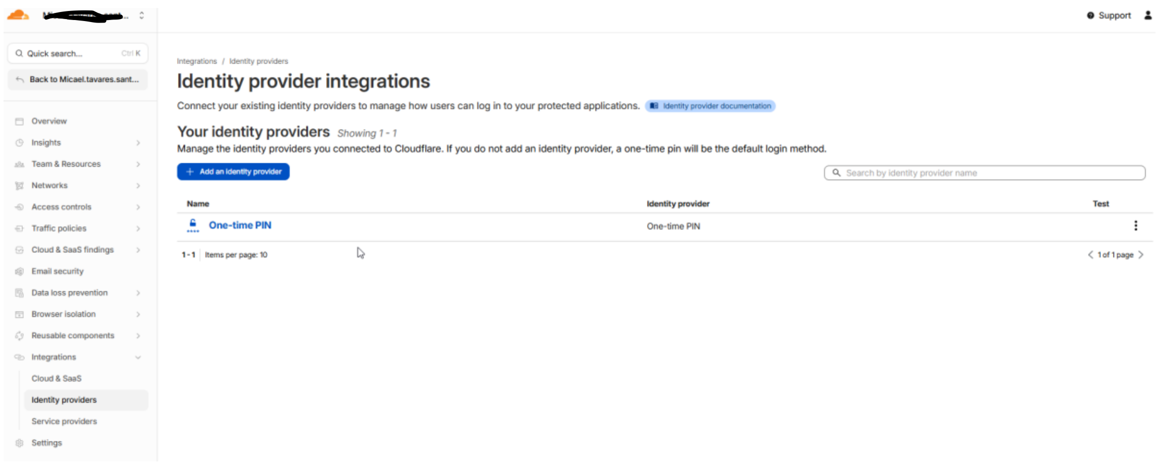
```
Host: tunnel.micael-cf-assignment.com User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7 Accept-Encoding: gzip, br Accept-Language: pt-PT,pt;q=0.9,en-US;q=0.8,en;q=0.7 Cache-Control: max-age=0 Cdn-Loop: cloudflare; loops=1 Cf-
Connecting-Ip: 2a02:a020:555:70e0:2518:4473:2583:8e18 Cf-Ipcountry: BE Cf[redacted]heme":"https"} Cf-Warp-Tag-Id: a280e0e2-a0f0-4f3e-b82a-588c43a6b72f Connection: keep-alive Priority: u=0, i Sec-Ch-Ua: "Not(A:Brand";v="8",
"Chromium";v="144", "Google Chrome";v="144" Sec-Ch-Ua-Mobile: ?0 Sec-Ch-Ua-Platform: "Windows" Sec-Fetch-Dest: document Sec-Fetch-Mode: navigate Sec-Fetch-Site: none Sec-Fetch-User: ?1 Upgrade-Insecure-Requests: 1 X-Forwarded-For:
[redacted]orwarded-Proto: https
```

Fig. 15Browser accessing tunnel URL

Zero Trust IdP

Cloudflare Zero Trust Access was configured using One-Time PIN (OTP) authentication as the Identity Provider.

This allowed secure user authentication without integrating an external identity platform.



Identity provider integrations

Connect your existing identity providers to manage how users can log in to your protected applications.

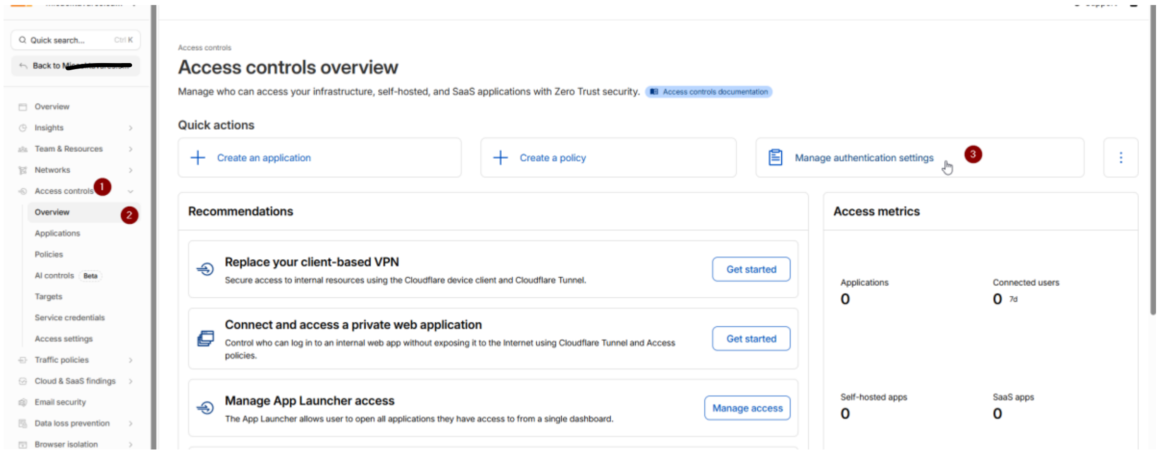
Your identity providers *Showing 1 - 1*

Manage the identity providers you connected to Cloudflare. If you do not add an identity provider, a one-time pin will be the default login method.

Name	Identity provider	Test
One-time PIN	One-time PIN	

1 - 1 Items per page: 10 — 1 of 1 page

Fig. 16IdP OTP



Access controls overview

Manage who can access your infrastructure, self-hosted, and SaaS applications with Zero Trust security.

Quick actions

Recommendations

Replace your client-based VPN
Secure access to internal resources using the Cloudflare device client and Cloudflare Tunnel.

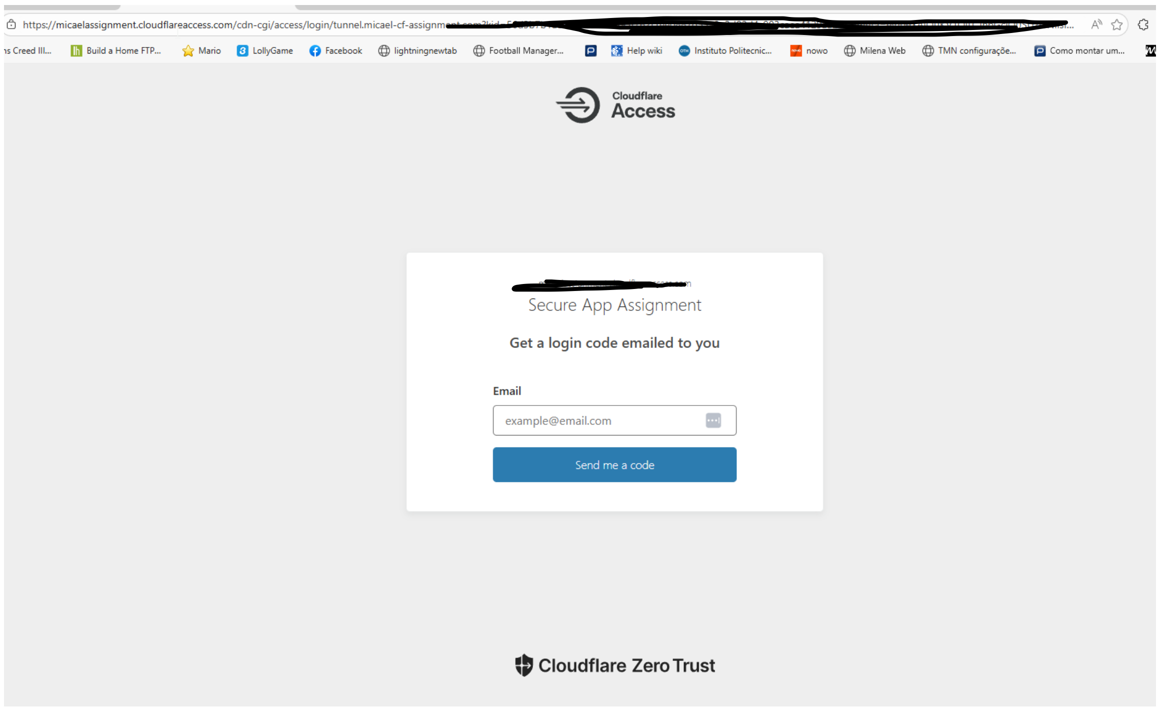
Connect and access a private web application
Control who can log in to an internal web app without exposing it to the Internet using Cloudflare Tunnel and Access policies.

Manage App Launcher access
The App Launcher allows user to open all applications they have access to from a single dashboard.

Access metrics

Applications 0	Connected users 0 7d
Self-hosted apps 0	SaaS apps 0

Fig. 17App config for OTP



https://micaelassignment.cloudflareaccess.com/cdn-cgi/access/login/tunnel.micael-cf-assignm[redacted]

[redacted]

Secure App Assignment

Get a login code emailed to you

Fig. 18Login screen

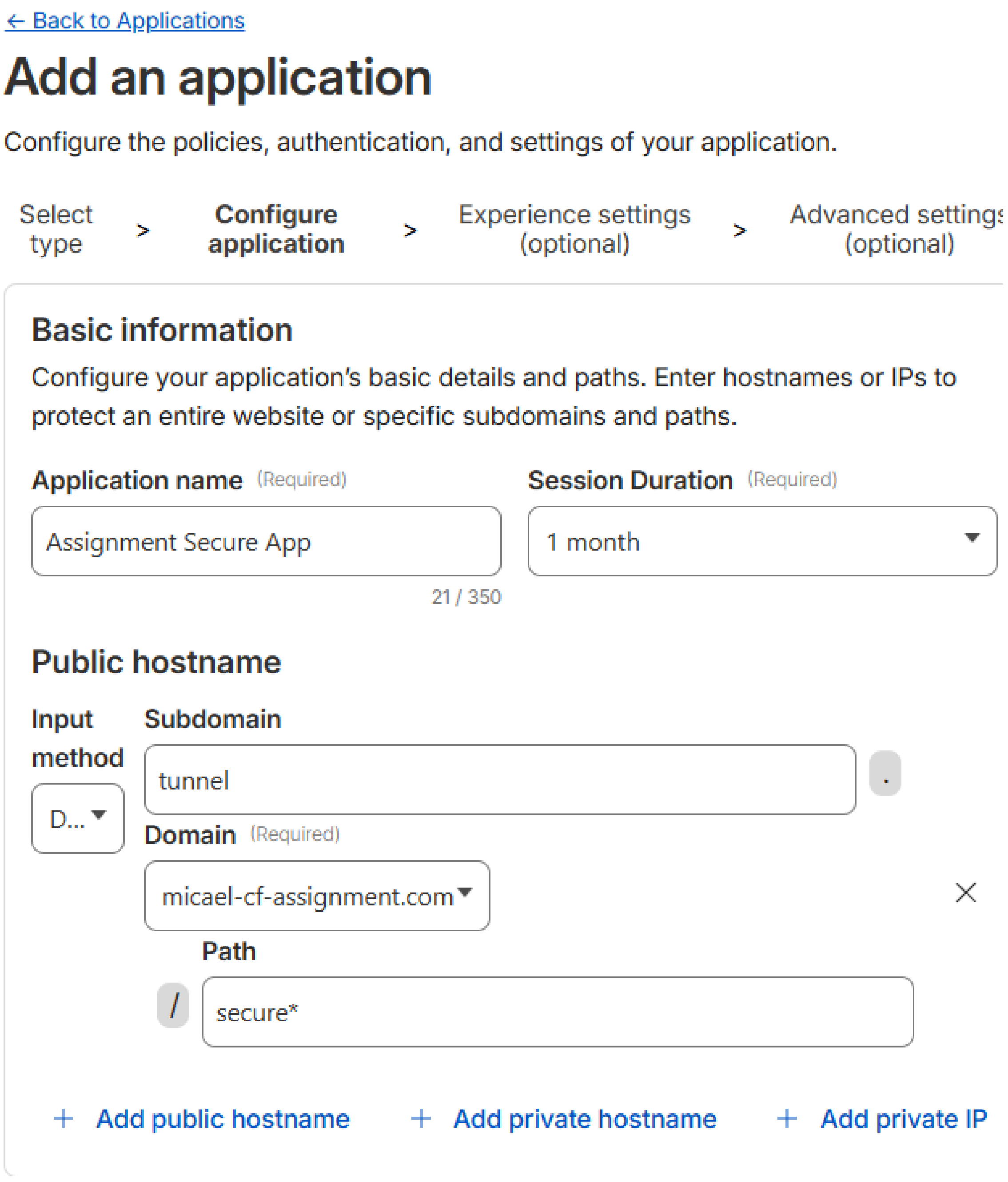
Secure Path Protection

Access policies were configured to protect the /secure path of the tunnel subdomain.

Access was restricted to:

- The project owner email
- Users with @cloudflare.com domain

Direct access to the origin IP was prevented by security group configuration and by routing traffic exclusively through Cloudflare Tunnel.



← Back to Applications

Add an application

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

Select type > **Configure application** > Experience settings (optional) > Advanced settings (optional)

Basic information

Configure your application's basic details and paths. Enter hostnames or IPs to protect an entire website or specific subdomains and paths.

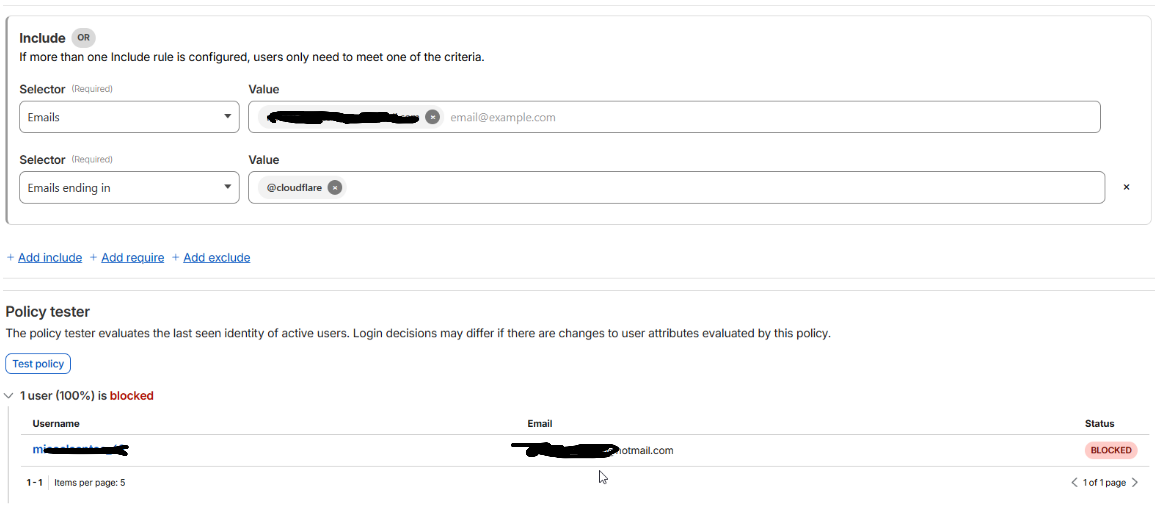
Application name (Required)	Assignment Secure App (21 / 350)
Session Duration (Required)	1 month

Public hostname

Input method	D...
Subdomain	tunnel
Domain (Required)	micael-cf-assignment.com
Path	/ secure*

+ Add public hostname + Add private hostname + Add private IP

Fig. 19Access Policy rule



Include OR

If more than one Include rule is configured, users only need to meet one of the criteria.

Selector (Required)	Value
Emails	[redacted] email@example.com
Emails ending in	@cloudflare

+ Add include + Add require + Add exclude

Policy tester

The policy tester evaluates the last seen identity of active users. Login decisions may differ if there are changes to user attributes evaluated by this policy.

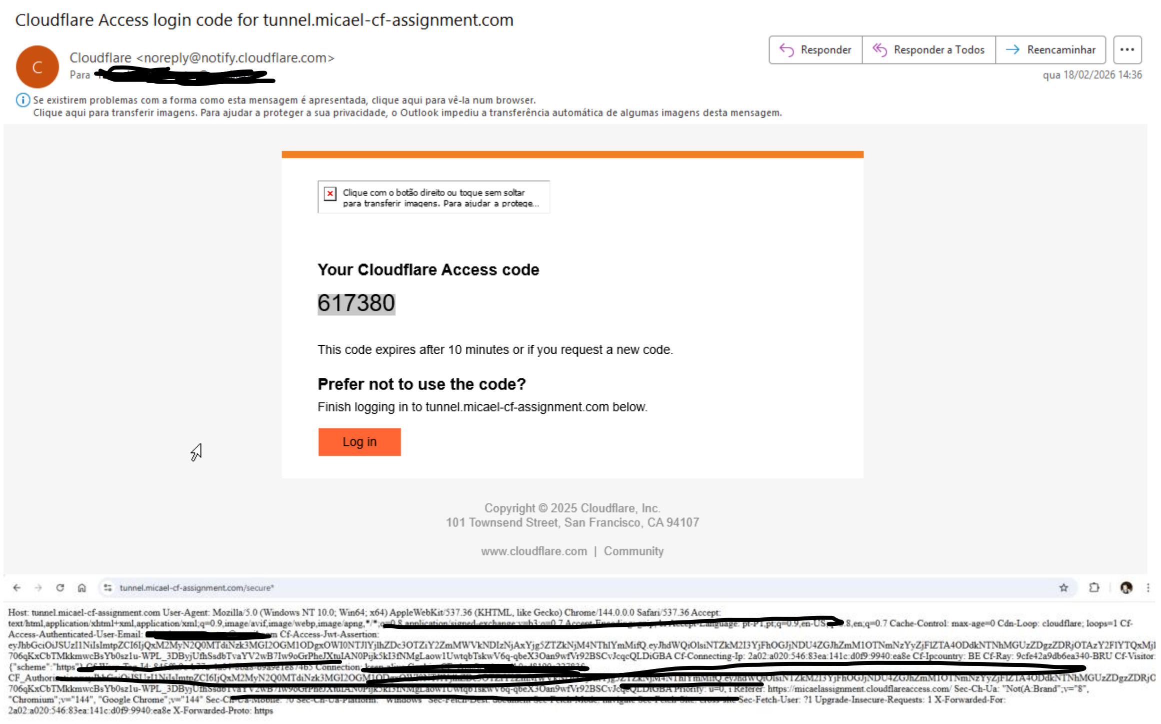
Test policy

1 user (100%) is blocked

Username	Email	Status
[redacted]	[redacted]otmail.com	BLOCKED

1 - 1 Items per page: 5 1 of 1 page

Fig. 20Block test


Cloudflare Access login code for tunnel.micael-cf-assignment.com

Cloudflare <noreply@notify.cloudflare.com>
Para [redacted]

Responder | Responder a Todos | Reencaminhar | ...

qua 18/02/2026 14:36

Se existirem problemas com a forma como esta mensagem é apresentada, clique aqui para vê-la num browser.
Clique aqui para transferir imagens. Para ajudar a proteger a sua privacidade, o Outlook impediu a transferência automática de algumas imagens desta mensagem.

Clique com o botão direito ou toque sem soltar para transferir imagens. Para ajudar a protege...

Your Cloudflare Access code

617380

This code expires after 10 minutes or if you request a new code.

Prefer not to use the code?

Finish logging in to tunnel.micael-cf-assignment.com below.

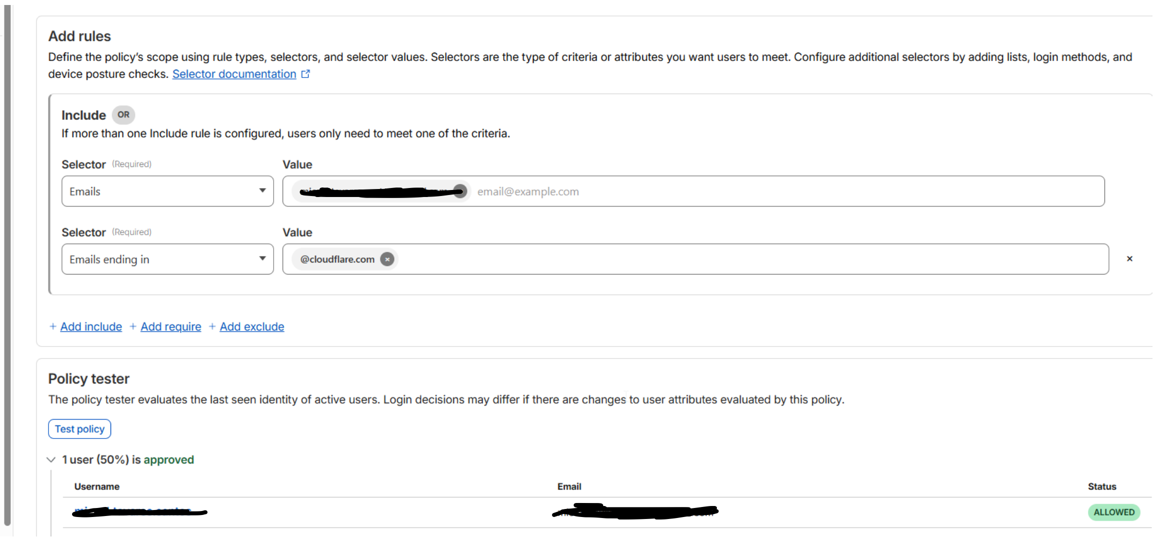
Log in

Copyright © 2025 Cloudflare, Inc.
101 Townsend Street, San Francisco, CA 94107

www.cloudflare.com | Community

tunnel.micael-cf-assignment.com/secure*

```
Host: tunnel.micael-cf-assignment.com User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*; [redacted] .8,en;q=0.7 Cache-Control: max-age=0 Cdn-Loop: cloudflare; loops=1 Cf-
Access-Authenticated-User-Email: [redacted] Cf-Access-Jwt-Assertion:
eyJhbGciOiJSUzI1NiIsImtpZCI6IjQxM2MyN2Q0MTdiNzk3MGI2OGM1ODgxOWI0NTJlYjlhZDc3OTZiY2ZmMWVkNDIzNjAxYjg5ZTZkNjM4NThlYmMifQ.eyJhdWQiOlsiNTZkM2I3YjFhOGJjNDU4ZGJhZmM1OTNmNzYyZjFlZTA4ODdkNTNhMGUzZDgzZDRjOTAzY2FlYTQxMjl
706qKxCbTMkkmwcBsYb0sz1u-WPL_3DByjUfhSsdbTvaYV2wB7Iw9oGrPheJXtuIAN0Pijk5kI3fNMgLaow1UwtqbTskwV6q-qbeX3Oan9wfVr92BSCvJcqcQLDiGBA Cf-Connecting-Ip: 2a02:a020:546:83ea:141c:d0f9:9940:ea8e Cf-Ipcountry: BE Cf-Ray: 9cfe42a9db6ea340-BRU Cf-Visitor:
{"scheme":"https"} [redacted]
CF_Authori[redacted]
706qKxCbTMkkmwcBsYb0sz1u-WPL_3DByjU[redacted] Referer: https://micaelassignment.cloudflareaccess.com/ Sec-Ch-Ua: "Not(A:Brand";v="8",
"Chromium";v="144", "Google Chrome";v="144" Sec-C[redacted] Sec-Fetch-User: ?1 Upgrade-Insecure-Requests: 1 X-Forwarded-For:
2a02:a020:546:83ea:141c:d0f9:9940:ea8e X-Forwarded-Proto: https
```


Add rules

Define the policy's scope using rule types, selectors, and selector values. Selectors are the type of criteria or attributes you want users to meet. Configure additional selectors by adding lists, login methods, and device posture checks. Selector documentation

Include OR

If more than one Include rule is configured, users only need to meet one of the criteria.

Selector (Required)	Value
Emails	[redacted] email@example.com
Emails ending in	@cloudflare.com

+ Add include + Add require + Add exclude

Policy tester

The policy tester evaluates the last seen identity of active users. Login decisions may differ if there are changes to user attributes evaluated by this policy.

Test policy

1 user (50%) is approved

Username	Email	Status
[redacted]	[redacted]	ALLOWED

Fig. 21Working Test

Worker + R2

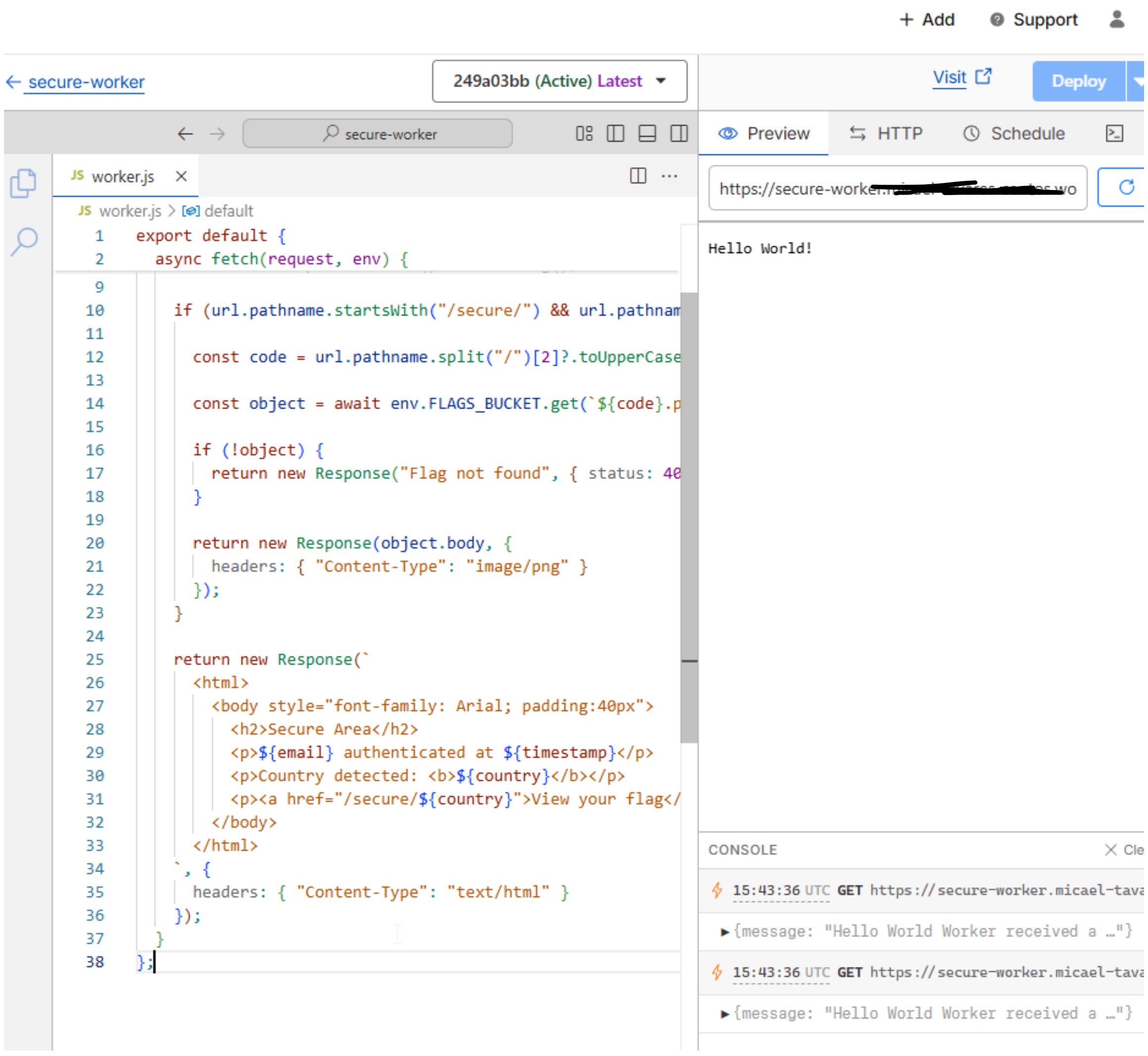
A Cloudflare Worker was created to run on the /secure path.

The Worker:

- Extracts user identity information from Cloudflare Access headers
- Displays authentication metadata

- Retrieves country flag images from a private R2 bucket

The Worker was deployed using the Wrangler CLI and code was uploaded to a public Git repository.



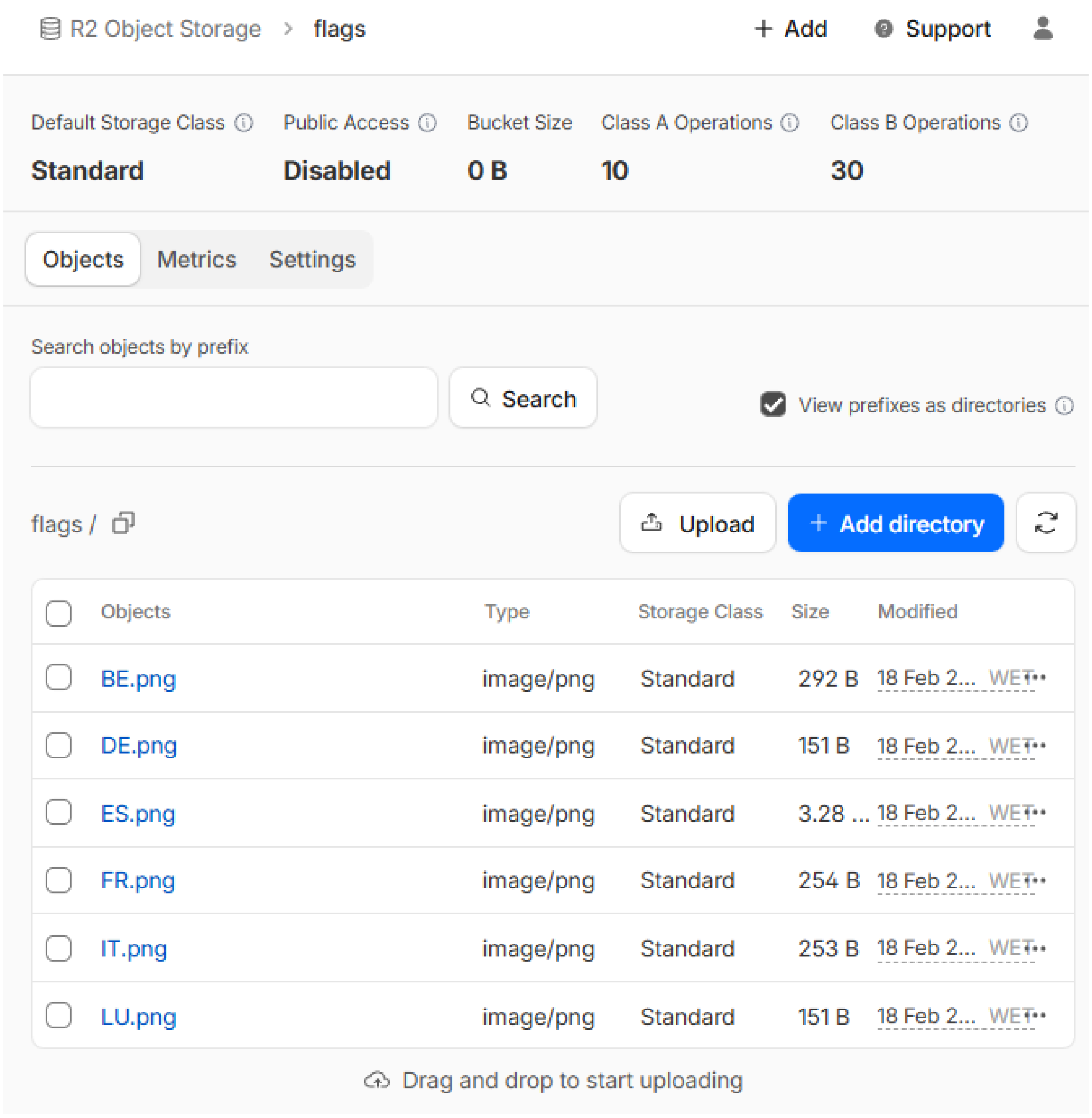
```
export default {
  async fetch(request, env) {

    if (url.pathname.startsWith("/secure/") && url.pathnam

      const code = url.pathname.split("/")[2]?.toUpperCase

      const object = await env.FLAGS_BUCKET.get(`${code}.p

      if (!object) {
        return new Response("Flag not found", { status: 40
      }

      return new Response(object.body, {
        headers: { "Content-Type": "image/png" }
      });
    }

    return new Response(`
      <html>
        <body style="font-family: Arial; padding:40px">
          <h2>Secure Area</h2>
          <p>${email} authenticated at ${timestamp}</p>
          <p>Country detected: <b>${country}</b></p>
          <p><a href="/secure/${country}">View your flag</
        </body>
      </html>
    `, {
      headers: { "Content-Type": "text/html" }
    });
  }
};
```

https://secure-worke[redacted]wo

```
Hello World!
```

```
15:43:36 UTC GET https://secure-worker.micael-tava
▶{message: "Hello World Worker received a …"}
15:43:36 UTC GET https://secure-worker.micael-tava
▶{message: "Hello World Worker received a …"}
```

Fig. 22Worker code



Default Storage Class	Public Access	Bucket Size	Class A Operations	Class B Operations
Standard	Disabled	0 B	10	30

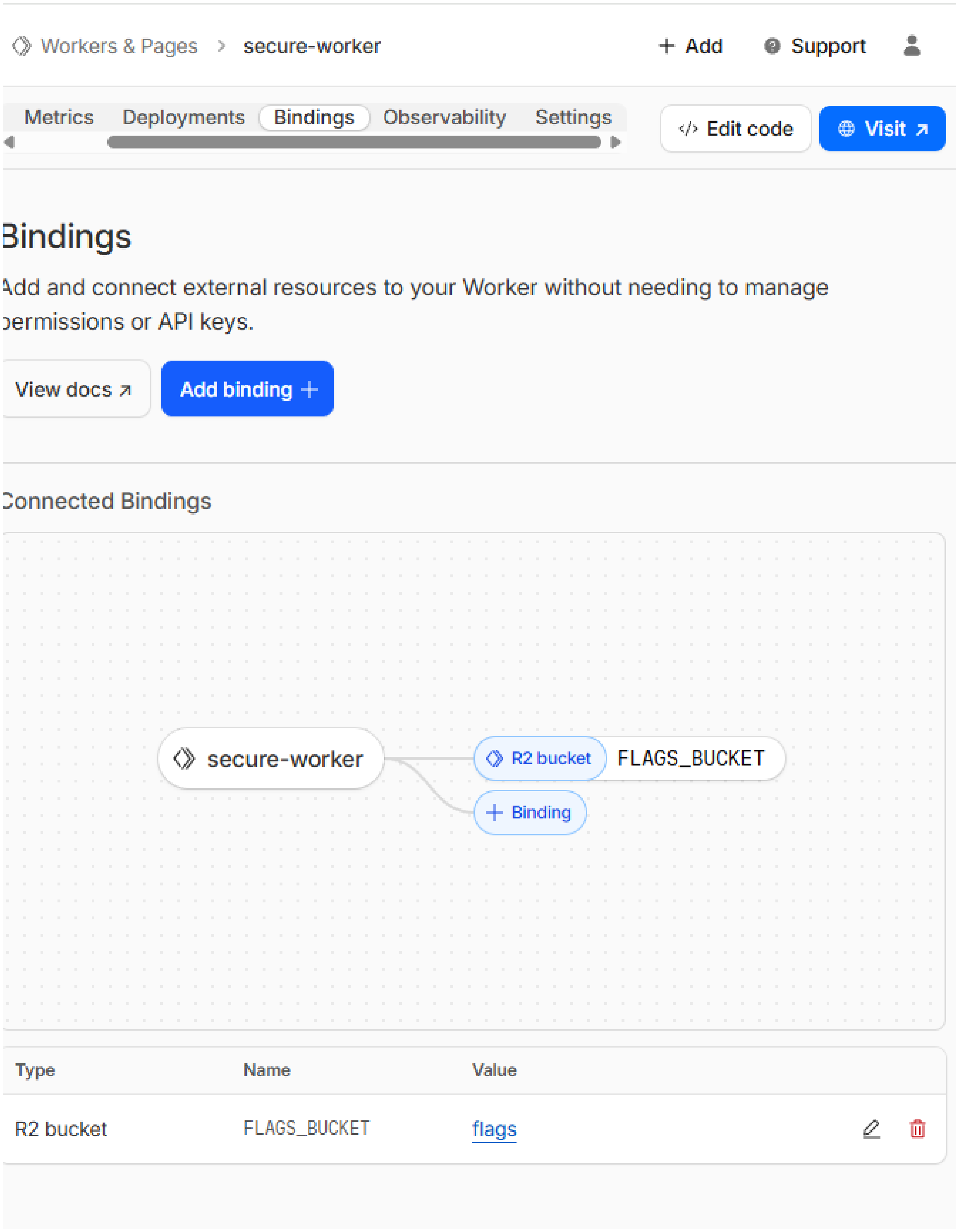
Search objects by prefix

View prefixes as directories ✓

flags /

Objects	Type	Storage Class	Size	Modified
BE.png	image/png	Standard	292 B	18 Feb 2... WET
DE.png	image/png	Standard	151 B	18 Feb 2... WET
ES.png	image/png	Standard	3.28 ...	18 Feb 2... WET
FR.png	image/png	Standard	254 B	18 Feb 2... WET
IT.png	image/png	Standard	253 B	18 Feb 2... WET
LU.png	image/png	Standard	151 B	18 Feb 2... WET

Fig. 23R2 bucket



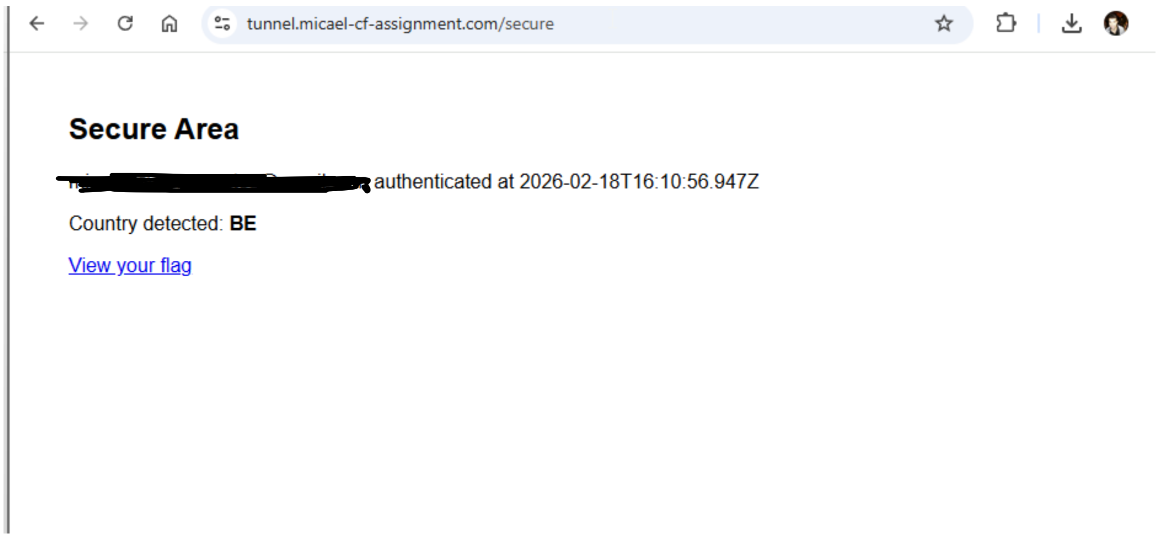
Bindings

Add and connect external resources to your Worker without needing to manage permissions or API keys.

Connected Bindings

Type	Name	Value
R2 bucket	FLAGS_BUCKET	flags

Fig. 24Worker binding


tunnel.micael-cf-assignment.com/secure

Secure Area

[redacted] authenticated at 2026-02-18T16:10:56.947Z

Country detected: **BE**

View your flag

Fig. 25Secure page working

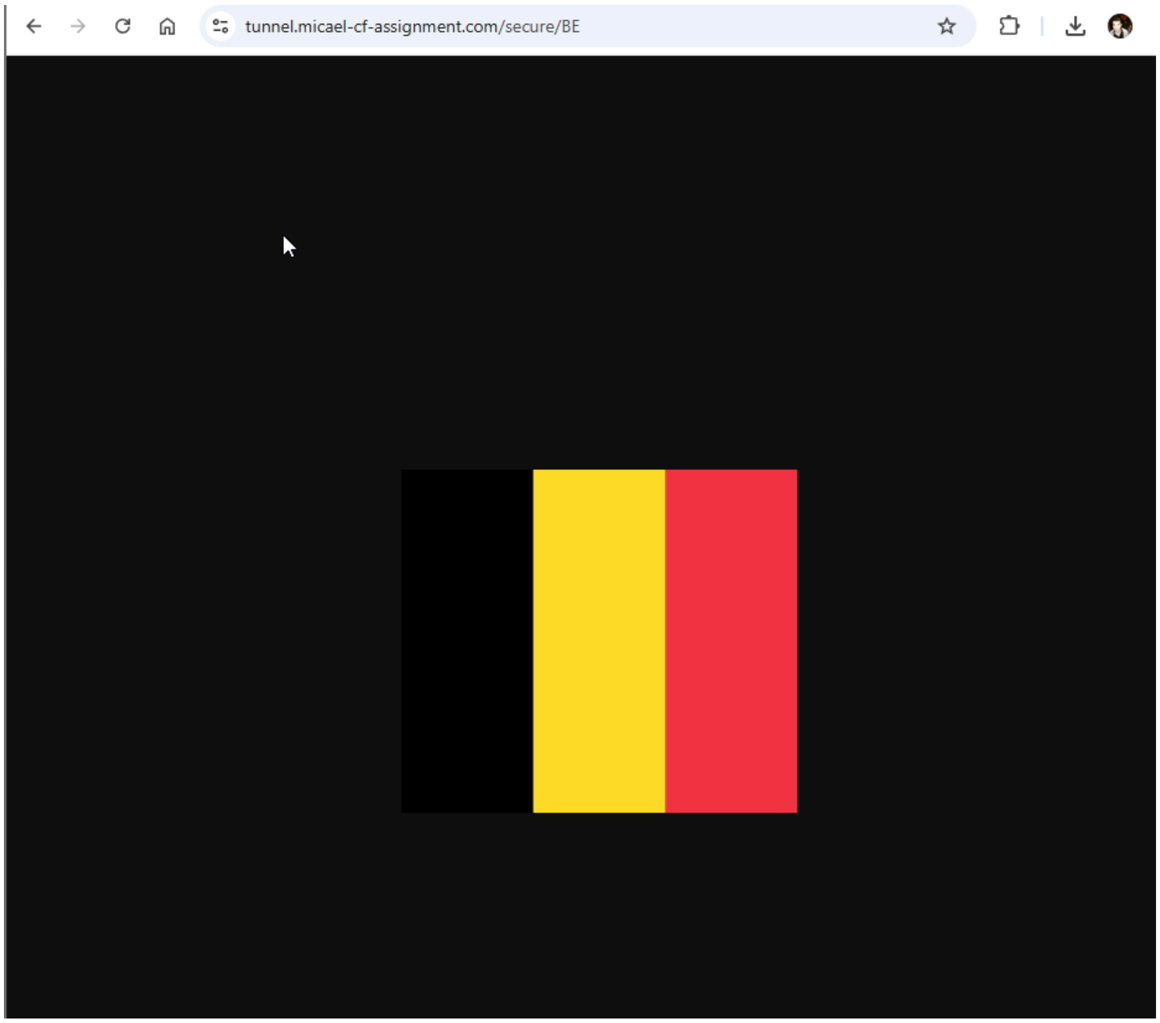

tunnel.micael-cf-assignment.com/secure/BE

Fig. 26Flag displayed

5. Security Controls Implemented

The following security controls were implemented:

- TLS encryption end-to-end
- Full Strict certificate validation
- Cloudflare Tunnel to eliminate exposed ports
- Zero Trust authentication policies
- Identity-based access control
- Private object storage with R2
- Reverse proxy isolation
- Origin firewall restrictions

6. Use Cases

The implemented architecture is relevant for multiple enterprise scenarios:

- Secure remote access to internal applications
- Zero Trust replacement for VPN access
- Protection of legacy applications without modification
- Identity-aware content delivery
- Secure asset delivery using object storage
- Dev/Test environment exposure without public IP risks

7. Knowledge Gaps and Research

During the implementation process, several knowledge gaps were identified and addressed through research and experimentation.

Examples include:

- Cloudflare Tunnel configuration and troubleshooting
- Systemd service configuration for persistent application execution
- Cloudflare Zero Trust Access policy behaviour
- Worker and R2 integration
- SSL Full Strict troubleshooting

Primary research sources included:

- Official Cloudflare documentation
- Community technical resources
- Hands-on testing and validation
- AI-assisted guidance for development efficiency

AI tools were used specifically to accelerate:

- Code generation (Flask and Worker examples)
- Troubleshooting ideas
- Documentation structure guidance

All configurations were validated manually to ensure correctness.

8. Customer Experience Perspective

From a customer perspective, the experience would be highly positive.

Cloudflare provides a unified platform to secure applications without requiring significant infrastructure changes.

The ability to deploy Zero Trust controls, secure tunnels and edge compute capabilities rapidly demonstrates strong value, particularly for organizations seeking to modernize legacy environments or reduce VPN dependency.

The learning curve is moderate, but documentation quality and platform integration significantly reduce operational complexity.

9. Conclusion

This project successfully demonstrates the deployment of a secure application architecture using Cloudflare Application Services.

The implementation shows practical understanding of Cloudflare’s core capabilities including secure connectivity, identity-based access control and edge computing.

The architecture is scalable, secure and aligned with modern Zero Trust principles.

10. Appendix (GitHub + URLs)

Public Application:

https://tunnel.micael-cf-assignment.com

Secure Application:

https://tunnel.micael-cf-assignment.com/secure

GitHub Repository:

https://github.com/MicaelTavaresSantos/cloudflare-assignment-Micael.git