

Create a Cloudflare, Zero Trust  
Tunnel to your web app in under 5  
minutes.


# Why?

- Control the flow of traffic to your web application
- You want to avoid exposing your ports to the public internet.
- You want to avoid exposing your ip address to the public.
- Authentication
- No more certificate management
- End to End tunnel encryption
- Caching
- Access to other Cloudflare services
- Get to your network from anywhere safely and securely
- Cloudflare Zero Trust is free


# Prerequisites

- Cloudflare Account
- Domain Name (can be purchase via Cloudflare)

# Choose your Cloudflare Zero Trust Team Name

 Cloudflare Zero Trust

Cancel and exit



Choose your team name

Your team name creates a unique domain for your Cloudflare Zero Trust account.  
Don't worry – you can change this later.

conorjwryan-example

.cloudflareaccess.com

Next

# Create Tunnel



← Capetowndev4eva... ▸

Zero Trust overview

Analytics ▾

Gateway ▾

Access ▴

Applications

Access Groups

Service Auth

Tags

**Tunnels**

My Team ▾

Logs ▾

Settings

[← Back to Tunnels](#)

## Create a tunnel

Create a tunnel to connect HTTP web servers, SSH servers, remote desktops, and other protocols safely to Cloudflare.

[Learn more](#)

**Name your tunnel** > Install and run connectors > Route tunnel

### Name your tunnel

Use a descriptive name based on the network you want to connect. We recommend creating only one tunnel for each network.

Tunnel name (Required)

e.g. enterprise-VPC-01

[Back](#)

## Install Tunnel (Docker Example)



← Capetowndev4eva... ▶

 Zero Trust overview

Analytics

Gateway

➡ Access

## Applications

## Access Groups

Service Auth

### Tags

## Tunnels

 My Team

Logs

 Settings


[← Back to Tunnels](#)

## Configure demo

Name your tunnel > **Install and run connectors** > Route tunnel

## Choose your environment

Choose an operating system:

 Windows Debian Red Hat Docker

## Install and run a connector

To connect your tunnel to Cloudflare, copy-paste one of the following commands into a terminal window. Remotely managed tunnels require that you install `cloudflared` 2022.03.04 or later.



**Store your token carefully.** This command includes a sensitive token that allows the connector to run. Anyone with access to this token will be able to run the tunnel.

✕

```
$ docker run cloudflare/cloudflared:latest tunnel --no-autoupdate --token eyJhIjoiaWw3ODBlbmRMTGZyYjA3MzU2YTZlZWZmVjZmZmMTU5MTYtL0J0IiwiaXNjaW9kaXI6ImBhM0M0MmUxLTgzMDctOTItZTFhY2ZhZGMiLCJ1c2Vybm90dXN0eSIsbnVpdjE1bG9kNQAzAwVRZzExUXZvZGVlc0J1NMU5UHQ0REk1YWtpaa1j9
```

[View Frequently Asked Questions](#) 

# Routing and tunnel configuration



← Capetowndev4eva... ▸

Zero Trust overview

Analytics ▾

Gateway ▾

Access ▴

Applications

Access Groups

Service Auth

Tags

Tunnels

My Team ▾

Logs ▾

Settings

## Route Traffic demo

Name your tunnel ▸ Install and run connectors ▸ **Route tunnel**

💡 Route traffic by adding a public hostname or a private network to your tunnel. You can always add more hostnames or networks at a later time.

🔒 Protect your resource by adding an Access policy under [Access > Applications > Self-hosted](#).

Public Hostnames Private Networks

### Add public hostname for demo

Public hostname

Subdomain

demo

Domain (Required)

capetowndev4eva.com ▾

Path

/ (optional) path

Service

Type (Required)

HTTP ▾

URL (Required)

://

web:8000

For example, <https://localhost:8001>

[Additional application settings ▸](#)

# Docker Compose Example

```
(hello-world-001) balazsk@balazs-desktop > ~/code/cloudflare/hello-world-001
version: '3.9'

services:
  web:
    container_name: web
    build: .
    command: python manage.py runserver 0.0.0.0:8000
    ports:
      - 8000:8000
  tunnel:
    container_name: cloudflared
    image: cloudflare/cloudflared:latest
    command: tunnel --no-autoupdate run
    environment:
      - TUNNEL_TOKEN=${TOKEN}
    depends_on:
      - web
networks:
  default:
    name: web
```



Demo