✅ Task Overview: Cloudflare Access Configuration for Static Site

🔗 Website URL

- Live Site: https://9af11f77.sagarweb.pages.dev
- Cloudflare Access Gateway (to be configured): https://9af11f77.sagarweb.pages.dev.cloudflareaccess.com/ (this will be active after Access is configured)

🔐 Objective

Configure Cloudflare Access to protect your static site with identity-based authentication and generate a PDF report detailing:

- Access policies
- Security settings
- Integration steps
- Dashboard screenshots (optional)

🛠 Step-by-Step Configuration Guide

1. Add Site to Cloudflare

- Log in to Cloudflare Dashboard
- Add your domain (e.g., https://9af11f77.sagarweb.pages.dev) if not already added
- Verify DNS settings and ensure traffic is proxied through Cloudflare

2. Enable Cloudflare Access

- Go to Zero Trust Dashboard → https://dash.teams.cloudflare.com
- Navigate to Access > Applications
- Click Add Application → Select Self-hosted
- Fill in:
- Application Name: SagarWeb static web
- Subdomain: 9af11f77.sagarweb.pages.dev

- Session Duration: e.g., 24 hours

3. Configure Identity Providers

- Go to Settings > Authentication
- Choose providers like:
- Google Workspace
- GitHub
- One-time PIN
- SAML or OIDC (for enterprise setups)

4. Create Access Policies

- Define rules:
- Allow only specific emails or domains (e.g. @gmail.com)
- Require multi-factor authentication
- Block unknown users
- Example policy:

```
- Action: Allow
  Email Domain: gmail.com
  MFA: Required
```

5. Test Access Flow

- Visit the protected URL
- Confirm login prompt appears
- Validate session behavior and expiration

1. Site Overview

- URL

Cloudflare Website Deployment Security

1. Web Application Firewall (WAF)

- Enable Cloudflare’s WAF to block OWASP Top 10 threats like:
- SQL Injection
- Cross-Site Scripting (XSS)

- Remote File Inclusion

2. Bot Management

- Use Bot Fight Mode or Advanced Bot Management to filter malicious bots and scrapers

3. Rate Limiting

- Set thresholds to prevent brute-force attacks and abuse of login or API endpoints

4. SSL/TLS Configuration

- Use Full (Strict) SSL mode for end-to-end encryption
- Enable Always Use HTTPS and Automatic HTTPS Rewrites

5. Access Control with Cloudflare Zero Trust

- Protect sensitive routes or admin panels using Cloudflare Access
- Enforce identity-based login (Google, GitHub, OTP)
- Apply session duration and MFA policies

6. DNS Security

- Proxy DNS through Cloudflare to hide origin IP
- Enable DNSSEC to prevent spoofing

7. Page Rules & Firewall Rules

- Create rules to:
- Block suspicious countries or IP ranges
- Redirect traffic
- Cache static assets aggressively

8. Security Analytics

- Monitor traffic, threats, and firewall events via Cloudflare dashboard