

GABRIELE ROMANATO

Menu

Come implementare CORS da zero in PHP con WordPress

L'implementazione di CORS (Cross-Origin Resource Sharing) in un sito WordPress può essere necessaria quando vuoi permettere ad applicazioni front-end o servizi esterni di accedere a risorse presenti sul tuo sito.

WordPress non gestisce CORS in modo nativo, ma è possibile configurarlo aggiungendo alcune righe di codice PHP nel file `functions.php` del tema attivo oppure tramite un plugin personalizzato.

Cos'è il CORS

CORS è un meccanismo di sicurezza che permette o blocca le richieste HTTP fatte da un dominio diverso rispetto a quello del server che ospita la risorsa. Senza una corretta configurazione, le richieste verranno bloccate dal browser per motivi di sicurezza.

Implementazione base in WordPress

Per permettere le richieste da altri domini, puoi aggiungere i seguenti header HTTP nella risposta. Questo può essere fatto usando gli hook di WordPress come `init` o `rest_api_init`, a seconda delle tue esigenze.

```
add_action('init', function() {
    // Consenti solo da un dominio specifico
    header('Access-Control-Allow-Origin:
https://example.com');
    header('Access-Control-Allow-Methods: GET, POST,
OPTIONS');
    header('Access-Control-Allow-Headers: Content-Type,
Authorization');
});
```

Questa configurazione consente le richieste da `https://example.com`. Per consentire tutte le origini (non consigliato in produzione), puoi usare l'asterisco:

```
header('Access-Control-Allow-Origin: *');
```

Gestire le richieste preflight (OPTIONS)

I browser inviano richieste OPTIONS prima di alcune richieste CORS. È importante rispondere correttamente a queste richieste per evitare errori.

```
add_action('init', function() {
    if ($_SERVER['REQUEST_METHOD'] === 'OPTIONS') {
        header('Access-Control-Allow-Origin:
https://example.com');
        header('Access-Control-Allow-Methods: GET, POST,
OPTIONS');
        header('Access-Control-Allow-Headers: Content-Type,
Authorization');
        header('Access-Control-Max-Age: 86400'); // Cache per
24 ore
        exit;
    }
});
```

Applicare CORS solo alle API REST

Se desideri abilitare CORS solo per le API REST di WordPress, puoi usare l'hook `rest_api_init`:

```
add_action('rest_api_init', function() {
    header('Access-Control-Allow-Origin:
https://example.com');
```

```
header('Access-Control-Allow-Methods: GET, POST,  
OPTIONS');  
header('Access-Control-Allow-Headers: Content-Type,  
Authorization');  
, 15);
```

Consigli di sicurezza

- Evita di usare * su siti in produzione, specialmente con cookie e autenticazione.
- Specifica solo i metodi e gli header necessari.
- Valuta la possibilità di gestire la logica CORS tramite il web server (es. Apache o NGINX) per migliorare le performance.

Conclusione

Implementare CORS in WordPress richiede attenzione, ma con poche righe di codice puoi ottenere il controllo necessario sulle origini che accedono alle tue API o risorse. Ricorda di testare le modifiche in un ambiente di sviluppo prima di applicarle in produzione.