跨來源資源共用（Cross-Origin Resource Sharing（CORS））是一種使用額外 HTTP 標頭令目前瀏覽網站的使用者代理（en-US)取得存取其他來源（網域）伺服器特定資源權限的機制。當使用者代理請求一個不是目前文件來源，例如來自於不同網域（domain）、通訊協定（protocol）或通訊埠（port）的資源時，會建立一個跨來源 HTTP 請求（cross-origin HTTP request）。

舉個跨來源請求的例子：http://domain-a.com HTML 頁面裡面一個 <img> 標籤的 src 屬性（en-US)載入來自 http://domain-b.com/image.jpg 的圖片。現今網路上許多頁面所載入的資源，如 CSS 樣式表、圖片影像、以及指令碼（script）都來自與所在位置分離的網域，如內容傳遞網路（content delivery networks, CDN）。

基於安全性考量，程式碼所發出的跨來源 HTTP 請求會受到限制。例如，XMLHttpRequest 及 Fetch 都遵守同源政策（same-origin policy）。這代表網路應用程式所使用的 API 除非使用 CORS 標頭，否則只能請求與應用程式相同網域的 HTTP 資源。

跨來源資源共用標準的運作方式是藉由加入新的 HTTP 標頭讓伺服器能夠描述來源資訊以提供予瀏覽器讀取。另外，針對會造成副作用的 HTTP 請求方法（特別是 GET 以外的 HTTP 方法，或搭配某些 MIME types 的 POST 方法），規範要求瀏覽器必須要請求傳送「預檢（preflight）」請求，以 HTTP 的 OPTIONS（en-US）方法之請求從伺服器取得其支援的方法。當伺服器許可後，再傳送 HTTP 請求方法送出實際的請求。伺服器也可以通知客戶端是否要連同安全性資料（包括 Cookies 和 HTTP 認證（Authentication）資料）一併隨請求送出。

簡單請求

部分請求不會觸發 CORS 預檢。這類請求在本文中被稱作「簡單請求（simple requests）」，雖然 Fetch 規範（其定義了 CORS）中並不使用這個述語。一個不觸發 CORS 預檢的請求——所謂的「簡單請求（simple requests）」——其滿足以下所有條件：

- 僅允許下列 HTTP 方法：
    - GET
    - HEAD（en-US）
    - POST

- 除了 user agent 自動設定的標頭（例如 Connection、User-Agent，或是任何請求規範〔Fetch spec〕中定義的「禁止使用的標頭名稱〔forbidden header name〕」中的標頭）之外，僅可手動設定這些於請求規格（Fetch spec）中定義為「CORS 安全列表請求標頭（CORS-safelisted request-header）」的標頭，它們為：
    - Accept
    - Accept-Language（en-US）
    - Content-Language（en-US）
    - Content-Type（但請注意下方的額外要求）
    - Last-Event-ID
    - DPR
    - Save-Data
    - Viewport-Width
    - Width
- 僅允許以下 Content-Type 標頭值：
    - application/x-www-form-urlencoded
    - multipart/form-data
    - text/plain
- 沒有事件監聽器被註冊到任何用來發出請求的 XMLHttpRequestUpload 物件（經由 XMLHttpRequest.upload（en-US）屬性取得）上。
- 請求中沒有 ReadableStream（en-US）物件被用於上傳。