網頁程式設計　HW6-CORS

資工二甲 B0929056 陳冠宇

CORS（Cross-Origin Resource Sharing）稱為跨來源資源共享，是一種用於讓網頁的受限資源能夠被其他域名的頁面存取的一種機制。但某些跨域的請求（例如 Ajax）常常會被 Same-Origin Policy 也就是同源策略所禁止。所謂同源策略中的「同源（Origin)」指的是，瀏覽器中某些行為，必須與來源頁面的「協定（Protocol)」、「網域（Domain)」、「連接埠（Port)」都相同，才能進行。一般而言協定於應用層通常為 Http 或 Https，傳輸層通常為 TCP，網路層通常為 IPv4 或 IPv6。網域通常是由機構名稱加上標準網際網路尾碼所組成。連接埠如果沒有指定的情況下，Http 預設為 80，Https 預設為 443。

同源策略可防止某個網頁上的惡意腳本通過該頁面的文檔對象模型訪問另一網頁上的敏感數據，如果想要在瀏覽器拿到一個網站上的完整內容，在 XHR Level 1 的規範後，我們可以利用 XMLHttpRequest 或是 Fetch API，基本上，同源策略是為了瀏覽器安全性，JSONP（JSON with Padding）則是直接繞過了瀏覽器，除非是為了相容於老舊的瀏覽器，否則不建議使用。然而，CORS 雖是正式規範，瀏覽器也會自動處理細節，但並非保證安全無虞。

當發生 CROS 錯誤時，由於程式碼所發出的跨來源 HTTP 請求會受到限制，這代表網路應用程式所使用的 API 除非有回傳 CROS 的 Header，否則只能請求與應用程式相同來源的 HTTP 資源。在這個情況下，其實請求（Request）已經發出去了，而瀏覽器其實也拿到回應（Response)，但是瀏覽器基於同源政策，因此不把拿到的回應給你的 Javascript 去做進一步的處理。若要開啟跨來源請求，必須在伺服器端做一些設定，像是 Access-Control-Allow-Origin: *（表示允許所有網站發送的請求)，或是如前面所提的 JSONP 是資料格式 JSON 的一種「使用模式」，可以讓網頁從別的網域取得資料，透過 HTML 中<script>標籤的 src 屬性，發送帶有 callback 參數的 GET 請求，服務端將接口返回數據拼湊

到 callback 函數中，返回給瀏覽器並解析執行，從而前端拿到 callback 函數返回的數據。或者是通過 Nginx 配置一個代理服務器域名與 Domain_1 相同但端口不同做跳板機，反向代理訪問 Domain_2 接口。也可以使用 Node.js 中間件（例如 Node + Express + Http-Proxy-Middleware）代理跨域，實現數據的轉發。還有許多種方法，例如 document.domain + iframe 跨域，此方案僅限主域相同，子域不同的跨域應用場景，或是利用 location.hash + iframe 跨域，A 與 B 不同域只能通過 Hash 值單向通訊，B 與 C 也不同域也只能單向通訊，但 C 與 A 同域，所以 C 可通過 parent.parent 訪問 A 頁面所有對象。或是 window.name + iframe 跨域、postMessage 跨域、WebSocket 協議跨域等等…

較為正規化的方式，正如前面所提，使用全球資訊網協會（World Wide Web Consortium）規範的跨來源資源共用規範，透過 Server 在 Http Header 的設定，當伺服器發送 Request 時，如果是屬於簡單跨來源請求（只能是 HTTP、GET、POST、HEAD 方法），則可以直接送出，倘若是屬於預檢請求（Preflight Request）必須經由 Request Header：Access-Control-Request-Method 和 Access-Control-Request-Headers，並進行預檢，確認是否通過伺服器的限制，才會成功發送 Request。

倘若我們無法在伺服器後端加上 Header 的話，則可以透過 Ajax Proxy Server 例如 https://cors-anywhere.herokuapp.com/corsdemo 來通過伺服器端代理實現。