

Network access

You can access the internet from a Snap using the `globalfetch` API.

Steps

1. Request permission to access the internet

Request the [endowment:network-access](#) permission, which exposes the `globalfetch` API to the Snaps execution environment. Add the following to your Snap's manifest file:

```
snap.manifest.json "initialPermissions" :  
{ "endowment:network-access" :  
  {} }
```

2. Use `thefetch`

function

You can now use the `thefetch` function to access the internet. The following example fetches a JSON file from the provided URL.

index.ts async

function

getJson (url :

string)

{ const response =

await

fetch (url) ; return

await response . json () ; } Same-origin policy and CORS fetch requests in a Snap are bound by the browser's [same-origin policy](#) . Since Snap code is executed in an iframe with the `sandbox` property, the browser sends an `Origin` header with the value `null` with outgoing requests. For the Snap to be able to read the response, the server must send an [Access-Control-Allow-Origin](#) CORS header with the value `*` or `null` in the response. Otherwise, you might need [set up a proxy](#) . caution `XMLHttpRequest` isn't available in Snaps, and you should replace it with `fetch` . If your dependencies use `XMLHttpRequest` , you can [patch it away](#) .

Example

See the [@metamask/network-access-example-snap](#) package for a full example of accessing the internet from a Snap. This example exposes a [custom JSON-RPC API](#) for dapps to call the `thefetch` function from a Snap.

[Edit this page](#)