Cross-Origin Resource Sharing (CORS)

Last Modified on 09/13/2021 6:33 am EDT

While the SAP Open Connectors UI utilizes browser-based cross-origin resource sharing (CORS) protections, those protections are bypassed if you make calls to any of our APIs which include `/api-v2`. Because the SAP Open Connectors APIs do not offer any inherent CORS protection, users and developers are responsible for the management of any necessary CORS-related protections. As always, we strongly recommend you implement any relevant best practices to ensure security for your account, resources, etc.

Things to Know

Any calls made to the SAP Open Connectors API server (any calls to our server including **/api-v2**, regardless of environment) will **not** return the Access-Control-Allow-* headers associated with the response header, regardless of whether the client sends the header or not.

When the HTTP request provides the Origin header and the origin is safelisted from a CORS perspective by the API, return any Access-Control-* headers with the Origin header's value. This is an instance of same origin policy (SOP); see Additional Information for more.

Resolving CORS Issues

Issues or errors regarding CORS are likely being caused by the connecting application, not SAP Open Connectors. To identify and resolve these errors, check that the application you are attempting to connect with is configured to allow communication outside of its own domain.

Allowed CORS Headers

While the CORS settings block custom headers from being received, custom headers can still be implemented via a backend server or Postman, as those methods do not require CORS validation; in practice, this means that an API defined in a custom connector will work with server-based calls, but may be blocked from a browser. You can change it to a query mapping on the SAP Open Connectors side, but it is necessary as a header; contact Customer Success for additional information.

The following CORS headers are allowed:

- `origin`
- `authorization`
- `accept`
- `content-type`
- `elements-partner-app`
- `elements-user-newpassword`
- `elements-user-password`
- `elements-schedule-request`
- `elements-async-callback-url`
- `elements-session`
- `elements-formula-instance-id`
- `elements-as-team-member`
- `elements-vendor-headers`

- email
- Subsite
- community-import-element
- Elements-Version
- x-amz-server-side-encryption
- x-amz-server-side-encryption-aws-kms-key-id
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-context
- x-amz-server-side-encryption-source-customer-key
- source
- elements-encryption-key
- elements-encryption-algorithm
- elements-element-instance-id

Additional Information

To learn more about implementing best practices for CORS protection or related information, contact Customer Success or see the following documentation:

- W3C: Cross-Origin Resource Sharing Recommendation
- W3C: CORS Specifications and Definitions
- W3C: Same Origin Policy wiki