B2C REST APIs

Contents

Introduction

These APIs are provided by the IDP to support the functionalities of the B2C customer portal or the IDP user portal.

B2C API Definition and Use Cases

Get all user devices

This API returns all the user devices associated with the authenticated user. The devices are the ones that are registered using the Device Fingerprint Rule and stored in the SQL database. The endpoints are:

https://<idp>/nidp/risk/rest/session/v1/user/devices
Description: Fetches all user device fingerprints from the SQL DB, **authentication required is user session**.
Method: GET
Sample response: [{"deviceId":"8000","deviceName":"Office Laptop"}]

https://<idp>/nidp/risk/rest/oauth/v1/user/devices
Description: Fetches all user device fingerprints from the SQL DB, **authentication required is OAuth token.**
Method: GET
Sample response: [{"deviceId":"8001","deviceName":"Bob’s iphone"}]

Note: ‘deviceId’ is a unique identifier exposed to the outside world for referring to a device and ‘deviceName’ is the user defined name associated with the device. If no name is specified during registration consent, the ‘deviceName’ contains the user agent string for this device.

Get specific user device

This API returns the queried user device, only if it belongs to the authenticated user. The devices are the ones that are registered using the Device Fingerprint Rule and stored in the SQL databasse. The endpoints are:

https://<idp>/nidp/risk/rest/session/v1/user/devices/{deviceId}

Path Parameter: deviceId – the unique identifier for a device

Description: Fetches the device represented by the deviceId(only if it belongs to the user) from the SQL DB, **authentication required is user session**.

Method: GET

Sample response: [{"deviceId":"8000","deviceName":"Office Laptop"}]

https://<idp>/nidp/risk/rest/oauth/v1/user/devices/{deviceId}

Path Parameter: deviceId – the unique identifier for a device

Description: Fetches the device represented by the deviceId(only if it belongs to the user) from the SQL DB, **authentication required is OAuth token**.

Method: GET

Sample response: [{"deviceId":"8000","deviceName":"Office Laptop"}]

Delete all user devices

This API deletes all the user devices associated with the authenticated/queried user. The devices are the ones that are registered using the Device Fingerprint Rule and stored in the SQL database. The endpoints are:

Invoked by end user:

https://<idp>/nidp/risk/rest/session/v1/user/devices

Path Parameter: deviceId – the unique identifier for a device

Description: Deletes all user device fingerprints(of the authenticated user) from the SQL DB, **authentication required is user session**.

Method: DELETE

Sample success response: {"status":"Delete successful."}

Response when no devices found or an error:: {"status":"Delete failed. Either no records found to delete, or an error occurred."}

https://<idp>/nidp/risk/rest/oauth/v1/user/devices

Path Parameter: deviceId – the unique identifier for a device

Description: Deletes all user device fingerprints(of the authenticated user) from the SQL DB, **authentication required is OAuth token**.

Method: DELETE

Sample response: {"status":"Delete successful."}

Response when no devices found or an error:: {"status":"Delete failed. Either no records found to delete, or an error occurred."}

Invoked by admin:

https://<idp>/nidp/risk/rest/basic/v1/admin/devices?userDN=<URL encoded DN of the user to be deleted>

Querystring parameter: userDN=<URL encoded DN of the user to be deleted>

Description: Deletes fingerprints from the SQL DB, **authentication required is admin credentials.**

Method: DELETE

Sample response: {"status":"Delete successful."}

Response when no devices found or an error:: {"status":"Delete failed. Either no records found to delete, or an error occurred."}

Response when no query parameter found: {"error_message":"Use query parameter userDN, value should be URL encoded DN of the user.} returned with error 400.

Note:

Admin is a user in the config store in this container:

ou=UsersContainer,ou=Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=accessManagerContainer,o=novell

eg:

cn=sspradmin,ou=UsersContainer,ou=Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=accessManagerContainer,o=novell

Delete specific user device

This API deletes the queried user device, only if it belongs to the authenticated user. The devices are the ones that are registered using the Device Fingerprint Rule and stored in the SQL database. The endpoints are:

https://<idp>/nidp/risk/rest/session/v1/user/devices/{deviceId}

Description: Deletes the device represented by the deviceId(only if it belongs to the authenticated user) from the SQL DB, **authentication required is user session.**

Method: DELETE

Sample response: {"status":"Delete successful."}

https://<idp>/nidp/risk/rest/oauth/v1/user/devices/{deviceId}

Description: Deletes the device represented by the deviceId(only if it belongs to the authenticated user) from the SQL DB, **authentication required is OAuth token.**

Method: DELETE

Sample response: {"status":"Delete successful."}

Delete all user history

This API deletes all the user history associated with the authenticated user. The information comprises all the collected data through Risk Based Authentication. The endpoints are:

Invoked by end user:

https://<idp>/nidp/risk/rest/session/v1/user/history

Description: Deletes all user history including fingerprints from the SQL DB, **authentication required is user session.**

Method: DELETE

Sample response: {"status":"Delete successful."}

https://<idp>/nidp/risk/rest/oauth/v1/user/history

Description: Deletes all user history including fingerprints from the SQL DB, **authentication required is OAuth token.**

Method: DELETE

Sample response: {"status":"Delete successful."}

Invoked by admin :

https://<idp>/nidp/risk/rest/basic/v1/admin/history?userDN=<URL encoded DN of the user to be deleted>

Querystring parameter: userDN=<URL encoded DN of the user to be deleted>

Description: Deletes all user history including fingerprints from the SQL DB, **authentication required is admin credentials.**

Method: DELETE

Sample response: {"status":"Delete successful."}

Response when no records found or an error: {"status":"Delete failed. Either no records found to delete, or an error occurred."}

Response when no query parameter found: {"error_message":"Use query parameter userDN, value should be URL encoded DN of the user.} returned with error 400.

Note:

Admin is a user in the config store in this container:

ou=UsersContainer,ou=Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=accessManagerContainer,o=novell

eg:

cn=sspradmin,ou=UsersContainer,ou=Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=accessManagerContainer,o=novell

Get attributes shared with all the SAML 2 providers in an IdP Cluster

https://<idp>/nidp/rest/v1/saml2/sp/attributes

Description: Fetches attributes shared with all the SAML 2 service providers, **authentication required is user session.**

Method: GET

Sample response: [{"displayName":"salesforce","sharedAttributes":["emp_id"]},{"displayName":"office365","sharedAttributes":["email","email_verified"]}], where ‘displayName’ is the name of the provider from the Administration Console and ‘sharedAttributes’ is a list of attribute names shared with each provider.

https://<idp>/nidp/api/saml2/sp

Description: Fetches attributes shared with all the SAML 2 service providers, **authentication required is OAuth. This API will be available in NAM 4.5. For B2C, the nidp_jars in the portal.zip has this functionality implemented.**

Method: GET

Sample response: [{"displayName":"salesforce","sharedAttributes":["emp_id"]},{"displayName":"office365","sharedAttributes":["email","email_verified"]}], where ‘displayName’ is the name of the provider from the Administration Console and ‘sharedAttributes’ is a list of attribute names shared with each provider.

Get consented OAuth clients

https://<idp>/nidp/api/oauth/nam/authzClients

Description: Fetches the list of Oauth clients that the user has given consent to. **Authentication required is OAuth. This API will be available in NAM 4.5. For B2C, the nidp_jars in the portal.zip has this functionality implemented.**

Method: GET

Sample response:

```
{"grants":[
        {"clientId":"bf2fc0b8-526b-4a64-a690-9fcc40752881",
        "clientName":"Digital Car Rental Partner App",
        "scopes":[
                {"name":"profile",
                "desc":"Access your basic profile",
                "claims":["website","birthdate","gender","profile","pre-
        ferred_username","given_name","middle_name","locale","pic-
        ture","zone_info","updated_at","nickname","name","family_name"]},
                {"name":"email",
                "desc":"Access your email address",
                "claims":["email_verified","email"]
                }
        ]
        }
]}
```

Delete content for an OAuth client

https://<idp>/nidp/api/oauth/nam/authzClients/<clientID>

Description: Deletes the consent for the OAuth API. **Authentication required is OAuth. This API will be available in NAM 4.5. For B2C, the nidp_jars in the portal.zip has this functionality implemented.**

Method: DELETE

Sample response:

{"status":"success","msg":"successfully revoked grants to clients"}