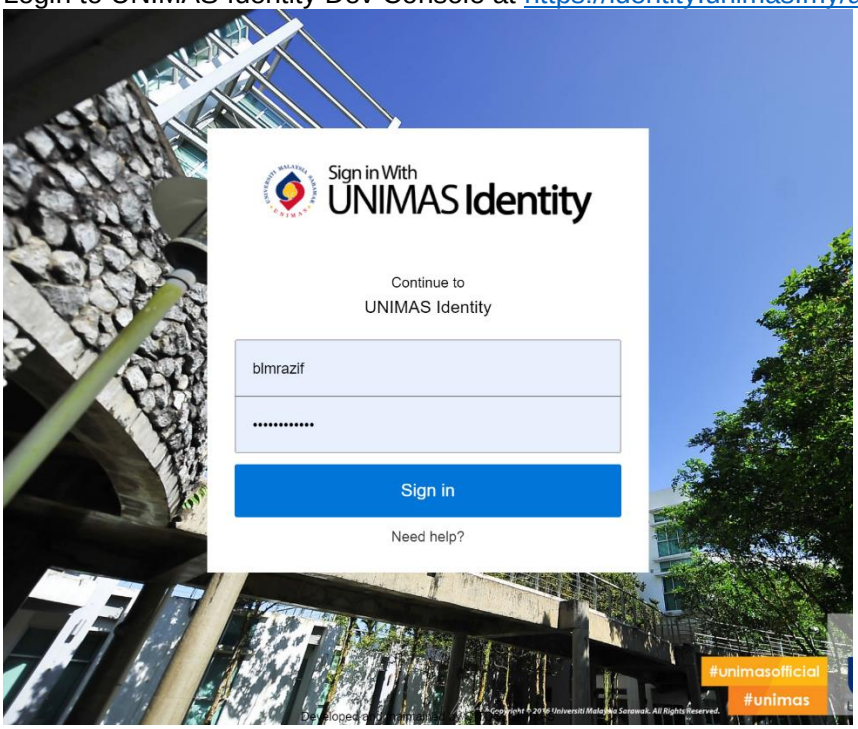
Developer Guides

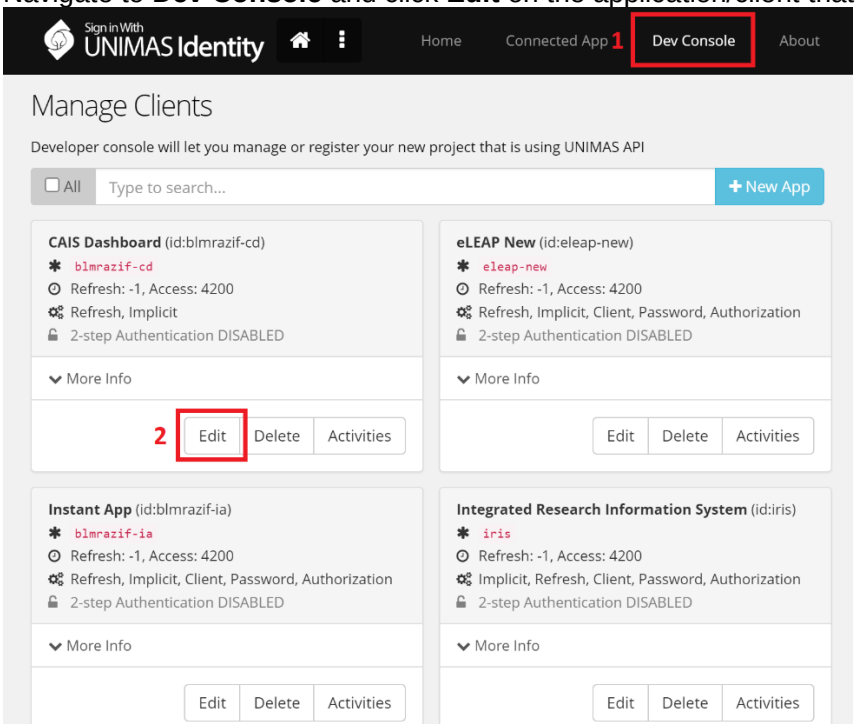
How to enable 2FA in application

UNIMAS Identity 2FA can be enabled on per-application(client) basis and can be limited to only a certain role/authority.

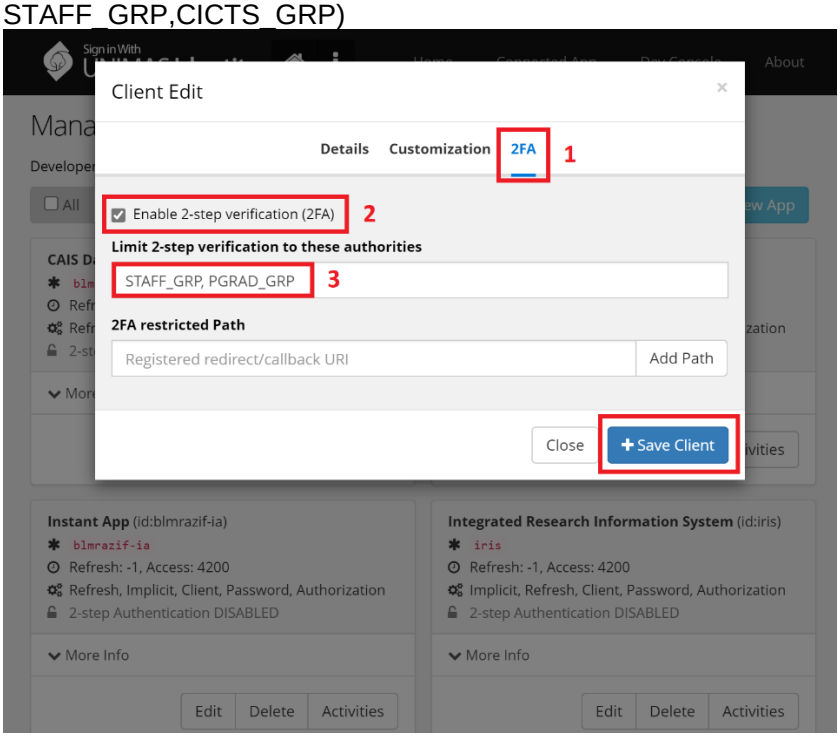
1) Login to UNIMAS Identity Dev Console at https://identity.unimas.my/admin



2) Navigate to **Dev Console** and click **Edit** on the application/client that you want to enable its 2FA



3) Navigate to **2FA** tab and tick **'Enable 2-step verification (2FA)'**. To limit the 2FA only on a certain role, fill in **'Limit 2-step verification..."** field with the role/authority name separated by comma (ie: STAFF_GRP,CICTS_GRP)



4) Click **Save Client** button to finish the process

Make sure to test that the 2FA is working properly as intended in your application.

How to verify OTP via endpoints (for per-transaction 2FA)

We can verify OTP via our OTP's verification endpoints. This would be useful for per-transaction verification. You can prompt the user to key in the OTP and verify the OTP first via *verify-tx endpoint* before performing any secured transaction. The verify-tx endpoint will return a json with property 'valid' = true if the OTP is valid.

POST https://identity.unimas.my/2fa/verify-tx	
Parameters	
otp (parameter)	6-digits OTP number to be verified Example Value | number 123456
Response	

200	OK
	
	Value:
	{
	"valid":boolean,
	"user": "string",
	"timestamp": "string"
	}

401	Unauthorized

Implementation Example:

```
function doTransaction(){
    var otp = window.prompt("Enter OTP:", "");
    if (otp != null && otp != "") {
        $http.post("https://identity.unimas.my/2fa/verify-tx?otp="+otp,{})
        .then(function(res){
            if (res.data.valid){ // check the result
                // perform secured transaction
            }else{
                alert("Invalid OTP");
            }
         });
    }
}
```

The endpoint required secured http request (with access token).

Example response:

```
{
    "valid": true,
    "user": "blmrazif",
    "timestamp": "2020-06-09T05:58:02.314+0000"
}
```

How to get OTP via SMS (for per-transaction 2FA)

Since the per-transaction 2FA requires implementation by the system developer, we also provide endpoints to request OTP via SMS.

GET https://identity.unimas.my/2fa/sms-otp	
Parameters	
<none>	
Response	
200	OK Value: `{ "otpSent": boolean, "success": boolean, "timestamp": "string" }`
401	Unauthorized

The endpoint required secured http request (with access token).

Example response:

```
{
    "otpSent": true,
    "success": true,
    "timestamp": "2020-06-09T05:58:02.314+0000"
}
```

UNIMAS Identity **PassNow** (OTP-*less* verification)

UNIMAS Identity will post the following info:

```
{
    "fid": "string",
    "code": "string",
    "username": "string",
    "clientName": "string",
    "clientId": "string",
    "timestamp": "string"
}
```

Example payload:

```
{
    "fid": "fid_GCXL3XE7OWE3EJ6B6TTDH3OWJAWG2ONX",
    "code": " UR2TU7A ",
    "username": "blmrazif",
    "clientName": "UNIMAS Identity",
    "clientId": "blmrazif-uid",
    "timestamp": "2020-06-10T09:25:01.239+0000"
}
```

To approve, UNIMAS Now may call the following endpoint:

POST https://identity.unimas.my/2fa/verify-push	
Parameters	
fid (parameter)	fid that is provided by UNIMAS Identity Example Value | number faid_GCXL3XE7OWE3EJ6B6TTDH3OWJAWG2ONX
code (parameter)	Code that is provided by UNIMAS Identity Example Value | number UR2TU7A
Response	

200	OK
	
	Value:
	{
	"code": -1536049856,
	"fid": "fid_GCXL3XE7OWE3EJ6B6TTDH3OWJAWG2ONX",
	"timestamp": "2020-06-10T08:29:16.793+0000",
	"success": true
	}

401	Unauthorized

The endpoint required secured http request (with access token).

The way this work is by using 2 different set of code.

One is **fid** which is publicly known id to maintain state and **code** which is only known to UNIMAS Now and Identity's backend.

Security mechanism

Fid will serve as a bridge to route the code back into UNIMAS Identity's otp verifier

Code is a time-based otp which only valid for 30sec

Fid will change everytime otp is prompted or if otp is invalid and reprompted.

Fix redirect in loginSecret page (include params)