



Il n'y a qu'un seul mode OAUTH qui est implémenté et qui est très proche du *three legs flow* et qui n'est donc pas adapté pour un échange de serveur à serveur.

L'idée principale est que :

l'utilisateur s'authentifie avec le navigateur (et pas en mode BASIC Authentication !) via le serveur OAM. Il faut démarrer le processus par une authentification car l'API suivante ne gère pas la redirection provoquée par la phase d'authentification (sujet à creuser pour voir s'il n'y a pas un contournement... ).

Une fois authentifié, un script javascript peut récupérer un token d'accès en déclenchant l'URL suivante en mode GET:

```
https://<tenant>.documents.us2.oraclecloud.com:443/documents/web?IdcService=GET_OAUTH_TOKEN
```

Le resultat JSON est de la forme:

```
{
  "LocalData":
  {
    "IdcService": "GET_OAUTH_TOKEN",
    "StatusCode": "0",
    "StatusMessage": "You are logged in as 'XXXX57507062C91FXXXXX9015'.",
    "StatusMessageKey": "!csUserLoggedIn,XXXXXX507062C9XXXXXX9015",
    "dUser": "XXXXXXXXXXXXXXXXX57507062CXXXXX4E9015",
    "dUserFullName": "Patrick Monaco",
    "expiration": "604799",
    "idcToken": "111111:XXXXXXXXXXXXXXXXXB5B8634EE2D",
    "localizedForResponse": "1",
```



```
"tokenValue": "zzzzzzzzzzNhdGhldGVydHJp.....  
K8nhD5G8e80TdPxIdogebT  
mIRrLCar8LQkmnAX5PbfnGg"  
},  
"ResultSets":  
}
```

C'est le tokenValue qu'il faut utiliser lors des appels ulterieurs d'apis REST en l'ajoutant systématiquement dans la variable de header:

```
Authorization: Bearer <tokenValue>
```