بنك الجزائر
BANQUE D'ALGERIE

cma

SPÉCIFICATIONS API REST - AUTH SERVICE

Auth Service

pour la Banque d'Algérie

Numéro de version :001

Nom du projet :	Mise à niveau ARTS - RTGS de la Banque d'Algérie
Nom du document :	Spécifications API REST - Auth Service

Table des matières

Historique des versions

Version	Date	Auteur	Commentaires
001	18.06.2025	CMA Small Systems AB	Première version

1 Introduction

Ce document décrit une interface de programmation applicative (API) utilisée pour la gestion des flux de service d'authentification selon le protocole OpenID Connect. L'API REST du service d'authentification est utilisée pour l'échange de messages MX entre les participants et les systèmes CMA (conformément au document de spécification de l'API REST de messagerie).

L'API REST du service d'authentification permet la mise en œuvre des fonctionnalités suivantes :

- Gérer les jetons.
- Demander des certificats.
- Gérer les mots de passe.

2 Flux de mot de passe

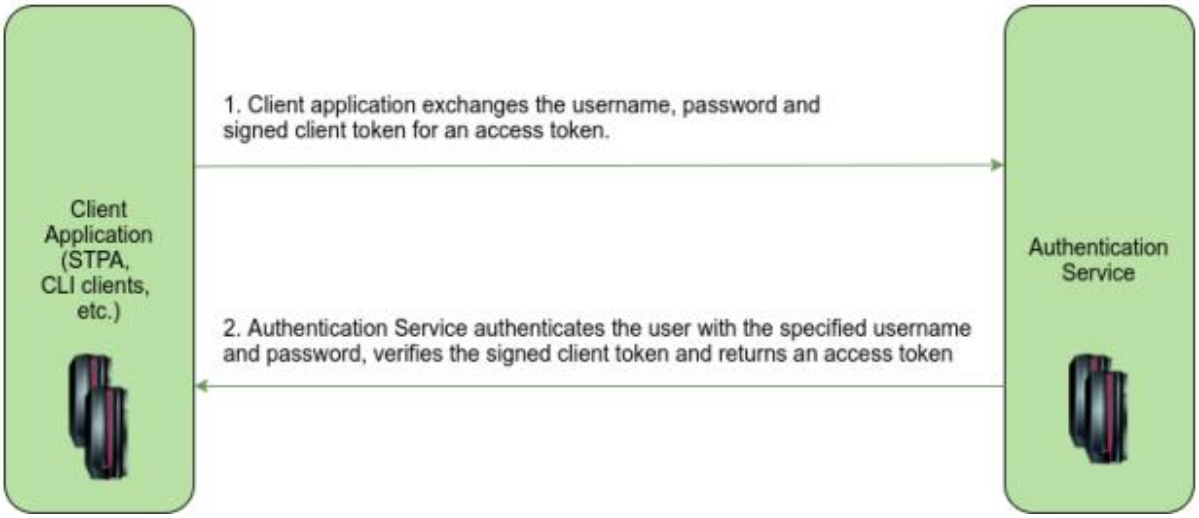

Figure 3.1. Processus de mot de passe avec jeton client

1. L'application cliente génère et signe un jeton client pour l'utilisateur participant, puis l'envoie avec le nom d'utilisateur et le mot de passe dans la requête.

2. Le service d'authentification authentifie l'utilisateur en fonction du nom d'utilisateur et du mot de passe reçus, vérifie la signature du jeton client et renvoie le jeton d'accès en cas de succès.

2.1 Jeton client standard

Le jeton client est utilisé pour les requêtes de jeton d'accès dans le flux de mot de passe. Lorsque les signatures de transport sont activées pour un utilisateur, la signature du jeton est vérifiée.

Le format standard du jeton est lorsque le numéro de série et l'émetteur sont définis dans les revendications asrv_cert_sn et asrv_cert_iss du jeton. Ces revendications sont utilisées pour rechercher le certificat dans les certificats chargés depuis LDAP. Dans certains cas, pour les systèmes sans LDAP, un format de jeton alternatif peut être utilisé, où le certificat lui-même est défini dans le champ x5c de l'en-tête du jeton client. Dans ce cas, le certificat n'est pas recherché, mais la chaîne de certificats est validée. Pour les tests avec les signatures utilisateur désactivées, n'importe quelle clé (par exemple, générée automatiquement) et un faux numéro de série et émetteur peuvent être utilisés pour la signature.

2.1.1 Jeton client standard

2.1.1.1 En-tête du jeton client

```
{
   "typ": "JWT",
   "alg": "RS256"
}
```

2.1.1.2 Corps (payload) du jeton client

```
{
   "iss": "AUTHTESTAXXX",
   "iat": 1634817966,
   "exp": 1634818566,
   "asrv_type": "client",
   "asrv_cert_iss": "cn=AUTHTEST CA,o=CMA,c=SE",
   "asrv_cert_sn": "02 79 6F FB 43 F5 3E B8"
}
```

2.1.1.3 JWT claimes

Claim	Description
typ	Type de média JWS, doit être JWT.
alg	Algorithme cryptographique. Ce doit être RS256
iss	Code utilisateur du client (participant) : AUTHTESTAXXX.
iat	**Issued at** la claim identifie l'heure à laquelle le JWT a été émis (secondes à partir de 1970-01-01T00:00:00Z UTC).
exp	**Exp** (délai d'expiration) la claim identifie le délai d'expiration, après lequel le JWT ne doit pas être accepté pour traitement (secondes à partir du 1970-01-01T00:00:00Z UTC).
asrv_type	Type de jeton. Ce doit être un client.
asrv_cert_iss	Émetteur du certificat de l'utilisateur participant, par exemple : cn=AUTHTEST CA,o=CMA,c=SE.

asrv_cert_sn	Numéro de série du certificat de l'utilisateur participant au format HEX, par exemple : 02 79 6F FB 43 F5 3E B8 or 02796FFB43F53EB8

2.1.2 Jeton client alternatif

3.1.2.1 En-tête du jeton client

```
{
   "alg": "RS256",
   "typ": "JWT",
   "x5c": ["MIIDjjCC..."]
}
```

2.1.2.2 Corps (payload) du jeton client

```
{
   "iss": "AUTHTESTAXXX",
   "iat": 1634817966,
   "exp": 1634818566,
   "asrv_type": "client",
}
```

2.1.2.3 JWT claimes

Claim	Description
typ	Type de média JWS, doit être JWT.
alg	Algorithme cryptographique. Ce doit être RS256
x5c	Le claim x5c (X.509 Certificate Chain) contient le certificat de clé publique X.509, ou la chaîne de certificats, correspondant à la clé utilisée pour signer numériquement le JSON Web Signature (JWS). Ce claim permet de fournir des informations sur la clé de signature directement dans le JWT, facilitant ainsi la validation de la signature.
iss	Code utilisateur du client (participant) : AUTHTESTAXXX

iat	Issued at la claim identifie l'heure à laquelle le JWT a été émis (secondes à partir de 1970-01-01T00:00:00Z UTC).
exp	Exp (délai d'expiration) la claim identifie le délai d'expiration, après lequel le JWT ne doit pas être accepté pour traitement (secondes à partir du 1970-01-01T00:00:00Z UTC).
asrv_type	Type de jeton. Ce doit être un client.

3 API Reference

3.1 Requête de jeton

3.1.1 Requête de jeton pour le flux de code d'autorisation

3.1.1.1 Description de la méthode

Description	Requête pour le jeton. Utilisée pour obtenir des jetons d'accès, d'identité et (en option) de rafraîchissement via le flux
Méthode	POST
Client	Application du participant
Serveur	Serveur d'authentification
URL	https://<auth-server-host:auth-server-port>/token

3.1.1.2 En-têtes de requête

Paramètre	Description
Content-Type	Définissez cet en-tête à application/x-www-form-urlencoded
Accept	Définissez cet en-tête à application/json”
Authorization	Contient le jeton client utilisé pour authentifier un user-agent auprès du serveur
Content-Length	Longueur du corps de la requête, en octets
Host	Adresse et port de l’hôte

3.1.1.3 Variables de la requête

Paramètre	Description
grant_type	Grant type, doit être défini sur authorization_code.
code	Le code d'autorisation.
redirect_uri	L'URI de redirection pour le callback.

3.1.1.4 Champs de réponse

Chemin	Type	Description
access_token	String	Jeton d’accès émis par le Serveur d'authentification.

token_type	String	Type de jetons émis. Doit être bearer.
expires_in	Number	La durée de vie en secondes du jeton d'accès. Par exemple, la valeur 3600 indique que le jeton d'accès expirera dans une heure à partir du moment où la réponse a été générée.
refresh_token	String	Le jeton de rafraîchissement, qui peut être utilisé pour obtenir de nouveaux jetons d'accès en utilisant la même subvention d'autorisation.
id_token	String	Le jeton d'identité, qui peut être utilisé pour obtenir le code utilisateur et vérifier la signature du jeton.

3.1.1.5 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Basic dGVzdC1jbGllbnQtaWQ6dGVzdC1jbGllbnQtcHdk
Content-Length: 111
Host: auth-service:8000

grant type=authorization code&code=BEaeZ7ZkTHCZ ZAGJrxYfQ&redirect uri=http
%3A%2F%2Flo
calhost%3A8888%2Fcallback
```

3.1.1.6 Réponse HTTP (réussie)

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 2368
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "access_token" : "eyJ4NXQ...",
   "refresh_token" : "eyJ4NXQ...",
   "id_token" : "eyJ4NXQ...",
```

```
   "token_type" : "Bearer",
   "expires in" : 3600
}
```

3.1.2 Requête de jeton pour le flux de mot de passe

3.1.2.1 Description de la méthode

Description	Requête pour obtenir un jeton d'accès via le flux de mot de passe. Utilise des utilitaires et des services qui ne prennent pas en charge le flux de code d'autorisation.
Méthode	POST
Client	Application du participant
Serveur	Serveur d'authentification
URL	https://<auth-server-host:auth-server-port>/token

3.1.2.2 En-têtes de requête

Paramètre	**Description**
Content-Type	Définissez cet en-tête à application/x-www-form-urlencoded
Accept	Définissez cet en-tête à application/json”
Authorization	Contient le jeton client utilisé pour authentifier un user-agent auprès du serveur
Content-Length	Longueur du corps de la requête, en octets
Host	Adresse et port de l’hôte

3.1.2.3 Variables de la requête

Paramètre	Description
grant_type	Réglez ce paramètre sur “password”.
username	Nom d’utilisateur pour l’authentification
password	Participant Mot de passe du Participant

3.1.2.4 Paramètres de réponse

Path	Type	Description
access_token	String	Jeton d’accès émis par le Serveur d'authentification.
token_type	String	Type de jetons émis. Doit être bearer.
expires_in	Number	Durée de vie du jeton d’accès, en secondes.

3.1.2.5 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0e...
Content-Length: 57
Host: auth-service:8000

grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.2.6 Réponse HTTP

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 790
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "access_token" : "eyJ4NXQ...",
   "token_type" : "Bearer",
   "expires_in" : 3600
}
```

3.1.3 Requête de jeton par jeton d'actualisation

3.1.3.1 Description de la méthode

Description	Demande d'obtention d'un jeton d'accès à l'aide d'un jeton d'actualisation
Méthode	POST
Client	Application du participant
Serveur	Serveur d'authentification
URL	https://<auth-server-host:auth-server-port>/token

3.1.3.2 En-têtes de requête

Parameter	Description
Content-Type	Définissez cet en-tête à application/x-www-form-urlencoded

Accept	Définissez cet en-tête à application/json”
Authorization	Contient le jeton client (voir section 3.1 Jeton client) utilisé pour authentifier un user-agent auprès du serveur
Content-Length	Longueur du corps de la requête, en octets
Host	Adresse et port de l’hôte

3.1.3.3 Variables de la requête

Parameter	Description
grant_type	Grant type, doit être défini sur refresh_token.
refresh_token	Le jeton d’actualisation délivré au client.

3.1.3.4 Paramètres de réponse

Path	Type	Description
access_token	String	Jeton d’accès émis par le Serveur d'authentification.
token_type	String	Type de jetons émis. Doit être bearer.
expires_in	Number	Durée de vie du jeton d’accès, en secondes.

3.1.3.5 Requête HTTP

```
POST
/token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Basic dGVzdC1jbGllbnQtaWQ6dGVzdC1jbGllbnQtcHdk
Content-Length: 758
Host: auth-service:8000
grant_type=refresh_token&refresh_token=eyJ4NXQ...
```

3.1.3.6 HTTP response

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 793
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "access_token" : "eyJ4NXQ...",
   "token_type" : "Bearer",
   "expires_in" : 3600
}
```

3.1.4 Dépannage de la requête de jeton

3.1.4.1 En-tête d'autorisation manquant

Si la requête ne comprend pas l'en-tête Authorization, la réponse renverra une erreur 401 avec le message invalid_client.

3.1.4.1.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Content-Length: 29
Host: auth-service:8000

grant_type=authorization_code
```

3.1.4.1.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
WWW-Authenticate: Basic realm="auth_service"
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 102
{
   "error" : "invalid_client",
   "error_description" : "Client application cannot be authenticated"
}
```

3.1.4.2 Grant type non pris en charge

Si le paramètre grant_type dans la requête est manquant ou différent de password, refresh_token, ou authorization_code, la réponse renverra une erreur 400 avec le message unsupported_grant_type.

3.1.4.2.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Basic dGVzdC1jbGllbnQtaWQ6dGVzdC1jbGllbnQtcHdk
Content-Length: 29
Host: auth-service:8000

grant_type=client_credentials
```

3.1.4.2.2 Réponse HTTP

```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 90
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "unsupported_grant_type",
   "error_description" : "unsupported grant type"
}
```

3.1.4.3 Changement de mot de passe requis

Si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client et un mot de passe qui doit être changé, la réponse renverra une erreur 420.

3.1.4.3.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 57
Host: auth-service:8000
grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.3.2 Réponse HTTP

```
HTTP/1.1 420 Method Failure
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 98
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_client",
   "error_description" : "User AUTHTESTAXXX must change password"
}
```

3.1.4.4 Certificat non trouvé

Si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client qui est signé avec une clé dont le certificat n'est pas trouvé dans le système, la réponse renverra une erreur 401.

3.1.4.4.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 57
Host: auth-service:8000
grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.4.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 163
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_token",
   "error_description" : "Certificate not found: 0F 73 A6 11 BE 9C 31
19(1113416128033206553) issued by o=CMA,cn=AUTHTEST CA,c=SE"
}
```

3.1.4.5 Certificat expiré

Si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client qui est signé avec une clé dont le certificat est expiré, la réponse renverra une erreur 401.

3.1.4.5.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 57
Host: auth-service:8000

grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.5.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 204
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_token",
   "error_description" : "Certificate is expired:
[cn=AUTHTESTAXXX,o=CMA,c=SE], s/n: [1D DE 55 43 D2 20 D9 41], valid from
[2024-07-25T14:54:17Z] to [2024-07-25T15:49:17Z]"
}
```

3.1.4.6 Certificat révoqué

Si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client qui est signé avec une clé dont le certificat a été révoqué, la réponse renverra une erreur 401.

3.1.4.6.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
OhpgYAH57aybZJCn0p-S2v_A
Content-Length: 57
Host: auth-service:8000

grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.6.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 204
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_token",
   "error_description" : "Certificate is revoked:
[cn=AUTHTESTAXXX,o=CMA,c=SE], s/n: [16 6D 77 3A 7D B0 80 87], valid from
[2024-07-25T15:54:20Z] to [2024-07-25T16:54:20Z]"
}
```

3.1.4.7 Certificat non fiable

Si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client qui est signé avec une clé dont le certificat ne correspond pas à celui spécifié dans les paramètres de la liste de révocation de certificats (CRL), la réponse renverra une erreur 401.

3.1.4.7.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 57
Host: auth-service:8000

grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.7.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 206
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_token",
   "error_description" : "Certificate is untrusted:
[cn=AUTHTESTAXXX,o=CMA,c=SE], s/n: [3B FF BC E1 D2 F9 63 2C], valid from
[2024-07-25T15:54:18Z] to [2024-07-25T16:54:18Z]"
}
```

3.1.4.8 Échec de la validation de la chaîne de certificats

Si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client qui est signé avec une clé dont le certificat ne réussit pas la validation de la chaîne de certificats, la réponse renverra une erreur 401.

3.1.4.8.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 57
Host: auth-service:8000

grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.8.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 221
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_token",
   "error_description" : "Chain validation failed for certificate:
[cn=AUTHTESTAXXX,o=CMA,c=SE], s/n: [60 34 1C 02 0B 1D DC 89], valid from
[2024-07-25T15:54:18Z] to [2024-07-25T16:54:18Z]"
}
```

3.1.4.9 Signature de jeton invalide

Si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client qui est signé avec une clé ne correspondant pas au certificat, la réponse renverra une erreur 401.

3.1.4.9.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 57
Host: auth-service:8000

grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.9.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 82
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_token",
   "error_description" : "invalid token signature"
}
```

3.1.4.10 Mauvais numéro de série

Une erreur se produit si vous tentez d'obtenir un jeton d'accès en utilisant un jeton client dans lequel la revendication asrv_cert_sn contient un numéro de série de certificat invalide. Cela se produit généralement lorsque le client, à des fins de test, crée un jeton client signé avec une clé aléatoire et utilise une chaîne factice comme numéro de série, telle que stpa_issuer_name ou tmsx_issuer_name. Si la vérification de signature est désactivée dans l'Autorité de Certification (CA) pour cet utilisateur, un tel jeton fonctionnera, mais si les signatures sont activées, le serveur renverra une erreur avec un code 401.

3.1.4.10.1 Requête HTTP

```
POST /token HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 57
Host: auth-service:8000

grant_type=password&username=AUTHTESTAXXX&password=123456
```

3.1.4.10.2 Réponse HTTP

```
HTTP/1.1 401 Unauthorized
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 76
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "error" : "invalid_token",
   "error_description" : "Bad serial number"
}
```

3.2 Requête UserInfo

3.2.1 Description de la méthode

Description	Requête pour obtenir des informations sur l'utilisateur.
Method	GET
Méthode	Application du participant
Serveur	Serveur d'authentification
URL	https://<auth-server-host:auth-server-port>/userinfo

3.2.2 Paramètres de réponse

Paramètre	Description
sub	Identifiant unique du sujet (utilisateur).

pwd_expires_in	Durée de vie du mot de passe (en secondes).

3.2.3 Exemple de réponse de réussite

3.2.3.1 Requête HTTP

```
GET /userinfo HTTP/1.1
Accept: application/json
Authorization: Bearer eyJ4NXQ...

Host: auth-service:8000
```

3.2.3.2 Réponse HTTP

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 57
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "sub" : "AUTHTESTAXXX",
   "pwd_expires_in" : 864000
}
```

3.2.4 Réponse Non Réussie

3.2.4.1 Mot de passe de l'utilisateur expiré

Lorsque le mot de passe de l'utilisateur est expiré, le champ pwd_expires_in aura une valeur de 0 dans la réponse.

3.2.4.1.1 Requête HTTP

```
GET /userinfo HTTP/1.1
Accept: application/json
Authorization: Bearer eyJ4NXQ..

Host: auth-service:8000
```

3.2.4.1.2 Réponse HTTP

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 52
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "sub" : "AUTHTESTAXXX",
   "pwd_expires_in" : 0
}
```

3.2.4.2 Durée de Vie Illimitée du Mot de Passe

Si la durée de vie du mot de passe de l'utilisateur est illimitée, le champ pwd_expires_in aura une valeur de null dans la réponse.

3.2.4.2.1 Requête HTTP

```
GET /userinfo HTTP/1.1
Accept: application/json
Authorization: Bearer eyJ4NXQ..

Host: auth-service:8000
```

3.2.4.2.2 Réponse HTTP

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
Content-Length: 55
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
{
   "sub" : "AUTHTESTAXXX",
   "pwd_expires_in" : null
}
```

3.3 Requête de Changement de Mot de Passe (avec jeton client)

3.3.1 Description de la Méthode

Description	Requête pour changer le mot de passe de l'utilisateur avec un jeton client.
Méthode	POST
Client	Application du participant
Serveur	Serveur d'authentification
URL	https://<auth-server-host:auth-server-port>/change-password

3.3.2 En-têtes de requête

Parameter	Description
Content-Type	Définissez cet en-tête sur application/x-www-form-urlencoded.
Accept	Définissez cet en-tête sur application/json.
Authorization	Contient le jeton client (voir section 3.1 Jeton client) utilisé pour authentifier un user-agent auprès du serveur
Content-Length	Longueur du corps de la requête, en octets
Host	Adresse et port de l'hôte

3.3.3 Variables de la requête

Paramètre	Description
new_pwd	Nouveau mot de passe que l'utilisateur souhaite définir.
current_pwd	Mot de passe actuel de l'utilisateur.

3.3.4 Exemple de réponse de réussite

3.3.4.1 Requête HTTP

```
POST /change-password
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 33
Host: asrv:23432

new_pwd=654321&current_pwd=123456
```

3.3.4.2 Réponse HTTP

```
HTTP/1.1 200 OK
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
```

3.3.5 Réponse Non Réussie

3.3.5.1 Mot de Passe Actuel Invalide (EP174)

3.3.5.1.1 Requête HTTP

```
POST /change-password HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO
Content-Length: 40
Host: asrv:23432

new_pwd=123456&current_pwd=wr0ngPassw0rd
```

3.3.5.1.2 Réponse HTTP

```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
WWW-Authenticate: Bearer realm="auth_service"
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 186
{
   "timestamp" : "2024-07-25T15:54:11.390+00:00",
   "status" : 400,
   "error" : "Bad Request",
   "path" : "/change-password",
   "message" : "Invalid password",
   "errorCode" : "EP174"
}
```

3.3.5.2 Mot de Passe Dupliqué (EP193)

3.3.5.2.1 Requête HTTP

```
POST /change-password HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 33
Host: asrv:23432

new_pwd=123456&current_pwd=123456
```

3.3.5.2.2 Réponse HTTP

```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 192
{
   "timestamp" : "2024-07-25T15:54:11.349+00:00",
   "status" : 400,
   "error" : "Bad Request",
   "path" : "/change-password",
   "errorCode" : "EP193",
   "message" : "Password is duplicated"
}
```

3.3.5.3 Changement de Mot de Passe Non Autorisé (EP211)

3.3.5.3.1 Requête HTTP

```
POST /change-password HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 33
Host: asrv:23432

new_pwd=654321&current_pwd=123456
```

3.3.5.3.2 Réponse HTTP

```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 220
{
   "timestamp" : "2024-07-25T15:54:11.435+00:00",
```

```
   "status" : 400,
   "error" : "Bad Request",
   "path" : "/change-password",
   "errorCode" : "EP211",
   "message" : "Password is not allowed to be changed at this time"
}
```


3.3.5.4 Mot de Passe Trop Court (EP212)

3.3.5.4.1 Requête HTTP

```
POST /change-password HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 32
Host: asrv:23432

new_pwd=short&current_pwd=123456
```

3.3.5.4.2 Réponse HTTP


```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 191
{
   "timestamp" : "2024-07-25T15:54:11.478+00:00",
   "status" : 400,
   "error" : "Bad Request",
   "path" : "/change-password",
   "errorCode" : "EP212",
   "message" : "Password is too short"
}
```

3.3.5.5 Mot de Passe Pas Assez Fort (EP213)

3.3.5.5.1 Requête HTTP

```
POST /change-password HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 31
Host: asrv:23432

new_pwd=weak&current_pwd=123456
```

3.3.5.5.2 Réponse HTTP

```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 199
{
   "timestamp" : "2024-07-25T15:54:11.521+00:00",
   "status" : 400,
   "error" : "Bad Request",
   "path" : "/change-password",
   "errorCode" : "EP213",
   "message" : "Password is not strong enough"
}
```

3.3.5.6 Mot de Passe Pas Assez Fort (EP215)

3.3.5.6.1 Requête HTTP

```
POST /change-password HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 33
Host: asrv:23432

new_pwd=123/45&current_pwd=123456
```

3.3.5.6.2 Réponse HTTP

```
HTTP/1.1 400 Bad Request
Vary: Origin
```

```
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 199
{
   "timestamp" : "2024-07-25T15:54:10.872+00:00",
   "status" : 400,
   "error" : "Bad Request",
   "path" : "/change-password",
   "errorCode" : "EP215",
   "message" : "Password is not strong enough"
}
```

3.3.5.7 Mot de Passe Pas Assez Fort (EP216)

3.3.5.7.1 Requête HTTP

```
POST /change-password HTTP/1.1
Content-Type: application/x-www-form-urlencoded;charset=UTF-8
Accept: application/json
Authorization: Bearer eyJ0eXAiO...
Content-Length: 33
Host: asrv:23432

new_pwd=*12345&current_pwd=123456
```

3.3.5.7.2 Réponse HTTP

```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 199
{
   "timestamp" : "2024-07-25T15:54:11.566+00:00",
   "status" : 400,
   "error" : "Bad Request",
   "path" : "/change-password",
   "errorCode" : "EP216",
```

```
    "message" : "Password is not strong enough"
}
```