GOVERNMENT OF ASSAM
INFORMATION TECHNOLOGY
DIRECTORATE OF INFORMATION TECHNOLOGY, ELECTRONICS & COMMUNICATION

# AADHAAR Authentication Services Delivery Platform
# Department Integration Staging API
# Developer Guide

## Version 1.0

**Revision History**

| Date | Version | Description | Author |
|---|---|---|---|
| 14/05/2025 | 1.0 | Initial Version | TA Team |

# TABLE OF CONTENTS


**1. OVERVIEW ..... 4**

PRE-REQUISITES ..... 4
ON BOARDING AND GO-LIVE STEPS ..... 4

**2. SERVICES ..... 5**

AUA SERVICES ..... 5
EKYC SERVICES ..... 5
LANGUAGE CODES ..... 5
STAGING API URLS AND KEYS ..... 5

**3. API DETAILS ..... 6**

*3.1 Demo Graphic Data Basic Authentication ..... 7*

**3.2 GET VAULT DATA OF DEMO BASIC ..... 7**

**3.3 OTP GENERATE ..... 8**

**3.4 AUTHENTICATION OTP ..... 10**

**3.5 GET VAULT DATA ..... 12**

**3.6 UPDATE VAULT DATA ..... 13**

**4. ANNEXURE 1: ENCRYPTION AND DECRYPTION METHODS ..... 15**

**4.1 JAVA ..... 15**

**4.2 .NET ..... 16**

**4.3 PHP ..... 17**

**5. ERROR CODES ..... 18**

ERROR CODES - HANDLED BY DITEC ..... 18
ERROR CODES - HANDLED BY UIDAI ..... 19

**6. ANNEXURE 2: USEFUL LINKS ..... 32**

# 1. Overview

DITEC is the UIDAI registered AUA/KUA entity which allows to provide the AADHAAR based authentication services to the registered Department to perform AADHAAR based authentication services provided by UIDAI. As part of this DITEC has developed AADHAAR based Authentication Service Framework (DASF) which allows the on- boarded Department to perform AADHAAR based authentication as part of the service delivery to the AADHAAR holders.

This document helps the registered Department to understand the DITEC provided Authentication APIs and how they can be integrated with their platforms.

### Pre-Requisites

1. The entities should register with the UIDAI as an Department
2. The Department should follow the UIDAI specified below guidelines as part of availing AADHAAR provided authentication services based on the Department checklist submitted to UIDAI
   a) The Department used applications should follow the UIDAI guidelines
   b) All the Department should strictly follow the data sharing guidelines specified by UIDAI
   c) The Department client application should be Audited by the STQC/CERT-IN agencies
   d) The PID and Request Data encryption can be done at registered biometric devices as per the Registered Device APIs specifications.

Following are the steps followed by Department to integrate and avail the DITEC provided authentication services

### On boarding and Go-Live Steps

1. Collect the Requirements from the Registered Department Entity
2. Finalize the authentication modes, integration and approach
3. Freeze the network connectivity between DITEC Centre and Department location and whitelisting IPs and Ports.
4. Customize the solution
5. Share the Service API URLs and necessary credentials for integration and testing in pre-production environment
6. Application Integration support
7. Conduct end to end test run
8. Rollout and Production migration activities
9. Production IPs, keys integration and configuration
10. Go-Live
11. Run operations

URLs and the access credentials can be provided separately.

# 2. Services

DITEC Authentication Service Framework (DASF) offers two types of authentication services to the registered sub-AUA and KUA agencies to perform the demographic verification and biometric authentication

### AUA Services

Verification Services can be used to verify the Aadhaar holder's demographic / biometric details against UIDAI data at CIDR and provide the Yes/No as response

1. Basic Demographic data - Name Matching
2. Full Demographic data - Demographic data Matching i.e POI (Proof of Identity details) and POA (Proof of Address details) Name, DOB, Gender and Address etc
3. OTP Generation
4. Fingerprint
5. IRIS

### eKYC Services

eKYC Services can used to fetch the electronically available KYC (Know Your Customer) details along with photograph which is available at UIDAI CIDR server. This data can be used as part of Service Delivery to the Aadhaar holders by taking the Aadhaar holder consent. The Aadhaar holders' consent can be given for the following scenarios

1. OTP Validation
2. Fingerprint
3. IRIS

### Language codes

The following **language codes** are used in APIs to identify the client-side programming language or platform from which the request originates. This enables the API to handle requests appropriately based on platform-specific needs such as encryption, formatting, or compatibility.

| Language | Code | Description |
|---|---|---|
| Java | L1 | Used when the API is accessed from a Java-based application |
| .NET | L2 | Used when the API is accessed from a .NET-based application |
| PHP | L3 | Used when the API is accessed from a PHP-based application |

### Staging API URLs and Keys

1. API URL:

   https://stagingaua.assam.gov.in/ekycditecservices/api/v1/{languagecode}{methodname}

2. Authentication URL :

   https://stagingaua.assam.gov.in/ekycditecservices/api/v1/auth/authenticate

3. License key:
   XXXXXXXXXXXXXXX (for this please contact DITEC)

4. Agency Code:
   XXXXXXXXXXXXXXX (for this please contact DITEC)

5. Password:
   XXXXXXXXXXXXXX (for this please contact DITEC)

6. Department PIN:
   XXXX (for this contact DITEC) Example: DITC

# 3. API Details

This document talks about the Integration API Call details which would help the Department to avail the DITEC provided AADHAAR based authentication services

**Authentication:**

**To get JWT token you must authenticate the agency code with password of below process:**

**Authenticate URL :**

https://stagingaua.assam.gov.in/ekycditecservices/api/v1/auth/authenticate

And pass the below json as request body

**Request :**

```
{
    "username": "xxxxxxxxxxx",
    "password": "xxxxx"
}
```

**Reponse:**

```
{
    "token": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
}
```

**Pass this token value as bearer token in authentication header For every API Call is must.**

## 3.1 Demo Graphic Data Basic Authentication:

**Url:** https://stagingaua.assam.gov.in/ekycditecservices/api/v1/{languagecode }/vault/demo-basic

### Encrypt Parameters:

| Encrypt Parameters | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **uid** | String | M | Aadhaar number of your customer | 57***** |
| **name** | string | M | Name of validation of customer | Ven******* |
| **transactionId** | String | M | Must be 16 digits character and starts with four digit subaua pin (DITC) | DITCXXXXXXXXXXXXXXXX |

**Options:** M(Mandatory), O (Optional)

### Request Parameters:

| Parameters | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **encData** | String | M | As per Annexure 1 using above encryption perameters pass encryption value here | xxxxxxxxxx |
| **transactionId** | String | M | Give the Transaction id which is given in encrypted data | xxxxxxxxxx |
| **agencyCode** | String | M | Ditec given code | xxxxxxxxxxx |

### Response Parameters:

| Field Name | Type | Description | Response Value |
|---|---|---|---|
| **status** | String | Indicates the status of the transaction 0 means success 1 means failure | "000" |
| **errorCode** | String | Represents an error code | TAE0000 |
| **errorDescription** | String | Describes the nature of the error or the result of the transaction | Success |
| **transactionID** | String | Transaction id | XXXXXXXXX |
| **encResponseData** | String | Encrypted e-KYC data | Encrypt data |

## 3.2 Get Vault Data of Demo basic:

**Url:** https://stagingaua.assam.gov.in/ekycditecservices/api/v1/{ languagecode }/vault/get-demo-basic

| Encrypt Parameters | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **uid** | String | M | Aadhaar number of your customer | 57***** |
| **name** | string | M | Name of validation of customer | Ven******* |
| **transactionId** | String | M | Must be 16 digits character and starts with four digit subaua pin (DITC) | DITCXXXXXXXXXXXXXXXX |

**Options:** M(Mandatory), O (Optional)

### Request Parameters:

| Parameters | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **encData** | String | M | As per Annexure 1 using above encryption perameters pass encryption value here | xxxxxxxxxx |
| **transactionId** | String | M | Give the Transaction id which is given in encrypted data | xxxxxxxxxx |
| **agencyCode** | String | M | Ditec given code | xxxxxxxxxxx |

### Response Parameters:

| Field Name | Type | Description | Response Value |
|---|---|---|---|
| **status** | String | Indicates the status of the transaction 0 means success 1 means failure | "000" |
| **errorCode** | String | Represents an error code | TAE0000 |
| **errorDescription** | String | Describes the nature of the error or the result of the transaction | Success |
| **transactionID** | String | Transaction id | XXXXXXXXX |
| **encResponseData** | String | Encrypted e-KYC data | Encrypt data |

## 3.3 Otp Generate :

Url: https://stagingaua.assam.gov.in/ekycditecservices/api/v1/{languagecode}/otp-generation

## Encrypt Parameters:

| Encrypt Parameters | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **Uid** | String | M | Aadhaar number of your customer | 57***** |
| **Channel** | string | M | "00" – send OTP via both SMS and Email (this is the default)<br>"01" – send OTP via SMS only<br>"02" – send OTP via Email only | 01 |
| **transactionId** | String | M | Must be 16 digits character and starts with four digit subaua pin (DITC) | DITCXXXXXXXXXXXXXX |

**Options:** M(Mandatory), O (Optional)

## Request Parameters:

| Parameters | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **encData** | String | M | As per Annexure 1 using above encryption perameters pass encryption value here | Encryption |
| **transactionId** | String | M | Give the Transaction id which is given in encrypted data | XXXXXXXXXXXX |
| **agencyCode** | String | M | Ditec given code | 12345 |

## Response Parameters:

| Field Name | Type | Description | Response Value |
|---|---|---|---|
| **Status** | string | Indicates the status of the transaction 0 means success 1 means failure | "0", |
| **errorCode** | string | Represents an error code | TAE0000 |
| **errorDescription** | string | Describes the nature of the error or the result of the transaction | Success |
| **transactionId** | string | Transaction id | XXXXXXXXXXXXXXXX |
| **otptrxId** | string | related to an OTP (One-Time Password) | 012363 |
| **Mobile** | string | Represents a mobile number. | 990***** |
| **Email** | string | Represents an email address | xxxxxxx@xxx.com |

## 3.4 Authentication OTP:

URL: https://stagingaua.assam.gov.in/ekycditecservices/api/v1/{languagecode}/authentication-otp

### Encrypt Parameters:

| Parameter Name | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **Uid** | String | M | Aadhaar Number/Virtual ID/UID | XXXXXXX0123 |
| | | | | |
| **dynamicPin** | String | M | Enter the dynamicpin | 012356 |
| | | | | |
| **otpGenTxn** | String | M | Otpgeneration Response "**otptrxId**" | ******** |
| **transactionId** | String | M | Must be 16 digits character and starts with four digit subaua pin (DITC) | DITCXXXXXXXXXXXXXXX |

### Request Parameters:

| Parameter Name | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **encData** | String | M | As per Annexure 1 using above encryption perameters pass encryption value here | Encryption Value |
| **transactionId** | String | M | Give the Transaction id which is given in encrypted data | XXXXXXXXXXXXX |
| | | | | |
| **agencyCode** | String | M | Ditec given code | SUBAUA123 |
| **ekycStatus** | String | M | If eKYC Data as a response set value "y" only authentication set value as "n" | "y" or "n" |
| **citizenConsent** | String | M | This value must in "y" only. | "y" |

### Authentication (ekycstatus ="n" ) Response Parameters:

| Parameter Name | Type | Description | Response Value |
|---|---|---|---|
| **status** | String | Indicates the status of the transaction 0 means success 1 means failure | "0", |
| **errorCode** | String | Represents an error code | TA0001 |
| **errorDescription** | String | Describes the nature of the error or the result of the transaction | Authentication Approved |
| **transactionId** | String | Transaction id | 20151512081025 |
| **authTxnId** | String | Authentication transaction id | Askdf589329893 |

### Authentication (ekycstatus ="Y" ) Response Parameters:

| Field Name | Type | Description | Response Value |
|---|---|---|---|
| **status** | String | Indicates the status of the transaction 0 means success 1 means failure | "0" |
| **errorCode** | String | Represents an error code | TAE0000 |
| **errorDescription** | String | Describes the nature of the error or the result of the transaction | Success |
| **transactionID** | String | Give the Transaction id which is given in encrypted data | XXXXXXXXXXXXXXXX |
| **vaultToken** | String | Vault token | ******* |
| **vaultPassKey** | String | Vault Password Key | ********* |
| **encResponseData** | String | Encrypted e-KYC data | Encrypt data |

That e-KYC sample data json format like below:


```
{
  "uID": "*******",
  "buildingName":
  "*******", "careOf":
  "*******", "district":
  "*******", "dob":
  "*******",
  "gender": "*******",
  "pinCode": "*******",
  "state": "*******",
  "street": "*******",
  "photo": "base64 format of image file",
  "name": "*******",
  "landMark":
  "*******",
  "locality":
  "*******",
  "subDistrict":
  "*******",
  "postOffice":
  "*******",
  "pdf":"*******",
  "village":
  "*******",
  "errorCode": "000",
  "errorMessage":
  "Success",
  "kycTxnId":
  "*******", "tkn":
  "*******",
  "vaultToken":
  "*******",
  " vaultPassKey": "*********"
}
```


You can save vaultToken as UID in your database for further retrieve vault data.

## 3.5 Get Vault Data:

**URL: https://stagingaua.assam.gov.in/ekycditecservices/api/v1/{languagecode}/get-vault-data**

### Encrypt Parameters:

| Parameter Name | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **vaultToken** | String | M | Vault token | ***** |
| **vaultPassKey** | String | M | Vault Password Key | ******** |
| **transactionId** | String | M | Must be 16 digits character and starts with four digit subaua pin (DITC) | DITCXXXXXXXXXXXXXXX |

**Options:** M(Mandatory), O (Optional)

### Request Parameters:

| Parameters | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **encData** | String | M | As per Annexure 1 using above encryption perameters pass encryption value here | Encryption |
| **transactionId** | String | M | Give the Transaction id which is given in encrypted data | 20151512081025 |
| **agencyCode** | String | M | Ditec given code | 12345 |

### Response Parameters:

| Field Name | Type | Description | Response Value |
|---|---|---|---|
| **status** | String | Indicates the status of the transaction 0 means success 1 means failure | "0" |
| **errorCode** | String | Represents an error code | TAE0000 |
| **errorDescription** | String | Describes the nature of the error or the result of the transaction | Success |
| **transactionID** | String | Transaction id | 20151512081025 |
| **encResponseData** | String | Encrypted e-KYC data | Encrypt data |

That e-KYC sample data json format like below:


```
{

  "uID": "********",
  "buildingName":
  "********", "careOf":
  "********", "district":
```

```
"*******", "dob":
"*******",
"gender": "*******",
"pinCode": "*******",
"state": "*******",


"street": "*******",
"photo": "base64 format of image file",
"name": "*******",
"landMark":
"*******",
"locality":
"*******",
"subDistrict":
"*******",
"postOffice":
"*******",
"pdf":"*******",
"village":
"*******",
"errorCode": "000",
"errorMessage":
"Success",
"kycTxnId":
"*******", "tkn":
"*******",
}
```


You can save vault Token as UID in your database for further retrieve vault data.

## 3.6 Update Vault Data:

URL: **https://stagingaua.assam.gov.in/ekycditecservices/api/v1/{languagecode}/update-vault-data**

### Encrypt Parameters:

| Parameter Name | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **Uid** | String | M | Aadhaar Number/Virtual ID/UID | XXXXXXX0123 |
| **dynamicPin** | String | M | Enter the dynamicpin | 012356 |
| **otpGenTxn** | String | M | Otpgeneration Response "**otptrxId**" | ******** |
| **transactionId** | String | M | Must be 16 digits character and starts with four digit subaua pin (DITC) | DITCXXXXXXXXXXXXXXX |

### Request Parameters:

| Parameter Name | Type | Options | Description | Request Value |
|---|---|---|---|---|
| **encData** | String | M | As per Annexure 1 using above encryption | Encryption Value |

| | | | | |
|---|---|---|---|---|
| | | | perameters pass encryption value here | |
| **transactionId** | String | M | Give the Transaction id which is given in encrypted data | 2024121101155 |
| **agencyCode** | String | M | Ditec given code | SUBAUA123 |
| **citizenConsent** | String | M | This value must in "y" only. | "y" |

## Response Parameters:

| Field Name | Type | Description | Response Value |
|---|---|---|---|
| **status** | String | Indicates the status of the transaction 0 means success 1 means failure | "0" |
| **errorCode** | String | Represents an error code | TAE0000 |
| **errorDescription** | String | Describes the nature of the error or the result of the transaction | Success |
| **transactionID** | String | Transaction id | 20151512081025 |
| **vaultToken** | String | Vault token | ******* |
| **vaultPassKey** | String | Updated Vault Password Key | ********* |
| **encResponseData** | String | Updated Encrypted e-KYC data | Encrypt data |

That e-KYC sample data json format like below:


```
{

  "uID": "*******",
  "buildingName":
  "*******", "careOf":
  "*******", "district":
  "*******", "dob":
  "*******",
  "gender":  "*******",
  "pinCode": "*******",
  "state": "*******",
  "street": "*******",
  "photo": "base64 format of image file",
  "name": "*******",
  "landMark":
  "*******",
  "locality":
  "*******",
  "subDistrict":
  "*******",
  "postOffice":
  "*******",
  "pdf":"*******",
  "village":
  "*******",
  "errorCode": "000",
  "errorMessage":
  "Success",
  "kycTxnId":
  "*******", "tkn":
  "*******",


}
```


You can save vaultToken as UID in your database for further retrieve vault data.

# 4. Annexure 1: Encryption And Decryption Methods

## 4.1 Java

The following methods are used for encryption and decryption in the java implementation.

### 1.1 Encryption

```
public static String encBasicValid(String uid, String name, String licenseKey, String agencyCode) {
                String result = "";
                String jsonInString = "{\"uid\":\"" + uid + "\",\"name\":\"" + name + "\"}";
                String plainText = jsonInString;

                logger.info(plainText);

                String password = licenseKey;

                String salt = agencyCode;
                IvParameterSpec ivParameterSpec = AESUtil.generateIv();
                SecretKey key;
                try {

                key = AESUtil.getKeyFromPassword(password, salt);
                result = AESUtil.encryptPasswordBased(plainText, key, ivParameterSpec);
                } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
                }
                return result;
                }
```

### 1.2 Decryption

```
public static String decryptEncString(String encData, String licencekey, String subauaid) {

        String decryptedText = "";
        String password = licencekey;
        String salt = subauaid;

        try {
                SecretKey key = AESUtil.getKeyFromPassword(password, salt);
```

```
                IvParameterSpec ivParameterSpec = AESUtil.generateIv();
                decryptedText = AESUtil.decryptPasswordBased(encData, key, ivParameterSpec);
            } catch (Exception e) {
                e.printStackTrace();
            }
            return decryptedText;
    }
```

## 4.2 .Net

The following methods are used for encryption and decryption in the .net implementation.

### 2.1 Encryption

```
public static string Encrypt(string plainText, string password)
    {
        byte[] salt = Encoding.UTF8.GetBytes("TADITEC@2025"); // Use a fixed or random salt
        byte[] key = GenerateKey(password, salt);
        byte[] iv = GenerateIV();
        using (Aes aesAlg = Aes.Create())
        {
            aesAlg.Key = key;
            aesAlg.IV = iv;
            aesAlg.Mode = CipherMode.CBC;
                        swEncrypt.Write(plainText);
                    }
                    return Convert.ToBase64String(msEncrypt.ToArray());
                }
            }
        }
}
```

### 2.2 Decryption

```
public static string Decrypt(string cipherTextBase64, string password)
    {
        byte[] fullCipher = Convert.FromBase64String(cipherTextBase64);
        byte[] salt = Encoding.UTF8.GetBytes("TADITEC@2025"); // Same fixed salt as used during encryption
        byte[] key = GenerateKey(password, salt);
        byte[] iv = new byte[16];
```

```
        Array.Copy(fullCipher, 0, iv, 0, iv.Length); // Extract IV from the beginning

        byte[] cipherText = new byte[fullCipher.Length - iv.Length];

        Array.Copy(fullCipher, iv.Length, cipherText, 0, cipherText.Length); // Extract the actual cipher

        using (Aes aesAlg = Aes.Create())
        {
            aesAlg.Key = key;
            aesAlg.IV = iv;
            aesAlg.Mode = CipherMode.CBC;
            aesAlg.Padding = PaddingMode.PKCS7;

            using (var decryptor = aesAlg.CreateDecryptor(aesAlg.Key, aesAlg.IV))
            using (var msDecrypt = new MemoryStream(cipherText))
            using (var csDecrypt = new CryptoStream(msDecrypt, decryptor, CryptoStreamMode.Read))
            using (var srDecrypt = new StreamReader(csDecrypt))
            {
                return srDecrypt.ReadToEnd();
            }
        }
    }
```

## 4.3 PHP

The following methods are used for encryption and decryption in the PHP implementation.

### 3.1 Encryption

Use below php syntax for encryption

```
$licenseKey = ‘xxxxxxxxxx’;(given by ditec) $text =’ {"uid": "999999999999","channel":"01" }’ ;

$encrypted = AesCipher::encrypt($licenseKey, $text); Get encrypted data by $encrypted->getData()
```

### 3.2 Decryption

Use below php syntax for decryption

```
$licenseKey = ‘xxxxxxxxxx ’;(given by ditec) $decrypted = AesCipher::decrypt($secretKey, $encryptedtext)

Get decrypted data by $decrypted->getData()
```

# 5. Error Codes

In addition to the above-mentioned Error codes AADHAAR Authentication API provide following error codes as per the UIDAI Specifications.

### Error Codes - handled by DITEC

Following are the Error codes along with the description which are handled by for the DITEC Provided auth Services To update it with the error codes handled at our end

| Code | Description |
|---|---|
| **TAV4000** | Missing or Invalid information |
| **TAV4001** | Citizen Concern Must accepted. |
| **TAIS2000** | An error has occurred, please try again later |
| **TAV2001** | Vault Data Not exist |
| **TAI1500** | No data found |
| **TAE2002** | Resource cannot be updated. |
| **TAI2003** | Content Request found no results |
| **TAI1000** | not valid |
| **TAI1001** | Agency ID not found!! |
| **TAI1002** | Transaction Id already Exist!! |
| **TAV1003** | Service Not available Please contact support |
| **TAV1004** | Your account Deactivate Please contact Admin |
| **TAV1005** | API key Not Found |
| **TAV1006** | These API Not available |
| **TAV001** | Account Inactive |
| **TAV002** | User License Key Inactive |
| **TAV003** | Service Inactive. |
| **TAV004** | Service License Key Expired. |
| **TAV005** | App Settings Error. |
| **TAV006** | Something went wrong. Please try again later. |
| **TAV007** | Invalid Auth Xml. |
| **TAV008** | Input Auth Xml Null. |
| **TAV009** | Prepaid Credits Expired. |
| **TAV010** | Insufficient Balance cannot perform transaction. |
| **TAV011** | Aua License Key Expired. |
| **TAV012** | License key not mapped for your account. |
| **TAV013** | Invalid Service Code. |
| **TAV014** | Auth XSD Validation Failed. |
| **TAV015** | Service Permissions Required |
| **TAV016** | Invalid License Key. |
| **TAV017** | User License Key Blocked. |
| **TAV018** | Service License Key Expired |
| **TAV019** | Account Blocked |
| **TAV020** | Invalid Request |
| **TAV021** | Service License Key Expired. |
| **TAV022** | Invalid Service License Key |
| **TAV023** | Invalid eKyc Xml. |
| **TAV024** | Input eKyc Xml Null. |
| **TAV025** | Insufficient Balance cannot perform transaction. |

| | |
|---|---|
| **TAV026** | KycRes tag missing in UIDAI response. |
| **TAV027** | Kua License Key Expired. |
| **TAV028** | License key not mapped for your account. |
| **TAV029** | eKyc XSD Validation Failed. |
| **TAV030** | Service Permissions Required |
| **TAV031** | Service License Key Expired. |
| **TAV032** | Invalid Service License Key. |
| **TAV033** | Invalid Otp Xml |
| **TAV034** | Input Otp Xml Null. |
| **TAV035** | Insufficient Balance cannot perform transaction. |
| **TAV036** | Service Permissions Required. |
| **TAV037** | Decryption Error. check your api keys |
| **TAV9001** | JWT Token Expire. |

### Error Codes - Handled by UIDAI

| Error Code | Description | Provision Required in the Application | Suggested Message to the User | Suggested instructions to the user | Probable Reasons |
|---|---|---|---|---|---|
| 100 | "Pi" (basic) attributes of demographic data did not match | User should be allowed to re- enter his/her personal information attributes like name, lname, gender, dob, dobt, age, phone, email whichever is used for authentication in application | Please re-enter your <name, lname, gender, dob, dobt, age, phone, email>. | Operator should re-enter correct details personal information as per the Aadhaar letter. Ensure correct Aadhaar Information is entered. | One or more personal information attributes not matching. |
| | | | | | |
| 200 | "Pa" (address) attributes of demographic data did not match | User should be allowed to re- enter his/her personal address attribute like co (care of), house, street, lm (land mark), loc (locality), vtc, subdist, dist, state, pc (postal pin code), po (post office) whichever is used for authentication in application | Please re-enter your <co (care of), house, street, lm (land mark), loc (locality), vtc, subdist, dist, state, pc (postal pin code), po (post office)>. | Operator should re-enter correct details personal information as per the Aadhaar letter.<br><br>Ensure correct Aadhaar Information is entered. | One or more personal address attributes not matching. |
| 300 | Biometric data did not match | User should be allowed to give his finger prints "n" number of times. N should be configurable and should be set as per application requirement. (E.g. For Banking Applications it can be set at a maximum of 5 times) | Please give your finger prints again. | Ensure correct Aadhaar number is entered and try authenticating again with another finger; ensure finger is placed correctly; ensure fingers are clean; ensure finger is not very dry; ensure fingerprint scanner is clean.<br><br>After repeated failure, if the resident is genuine, exception handling provision would need to be | Finger print is not given properly, scanner has some dust accumulated, fingers were wet, position of finger not appropriate, scanned finger NFIQ not good |

| | | | | | |
|---|---|---|---|---|---|
| | | | | followed to provide service.<br><br>Please contact UIDAI helpdesk to inform about the issue and to understand the steps for the updation of the biometric information in CIDR. | |
| 310 | Duplicate fingers used | Application should prompt user to try again with distinct fingers. | Please try again with distinct fingers. | Operator should insure that the resident is providing distinct fingers (two different fingers) for "two finger" authentication. | Error occurs when same finger is sent as two or more separate records within same request. For two-finger auth, if resident puts same finger again, then this happens. |
| 311 | Duplicate Irises used | Application should prompt user to try again with distinct irises. | Please try again with distinct irises. | Operator should ensure that the resident is providing distinct irises (two different irises) for authentication. | Error occurs when same iris is sent as two or more separate records within same request. |
| 312 | FMR and FIR cannot be used in same transaction | Application should ensure that authentication request does not mix FMR and FIR in the same transaction e.g. in case of two finger authentication, data for two distinct fingers should either be sent in FMR format or in FIR format. | Technical Exception <No> | Contact technical helpdesk. | Auth packet cannot mix fingerprint "image" records (FIR) and fingerprint "minutiae" records (FMR). AUA app should choose either one or another. FMR is recommended. |
| 313 | Single FIR record contains more than one finger | Application should prompt user to try again by placing single finger. | Please try again by placing Single finger on the authentication device. | Operator should ensure that the resident is providing single finger for authentication. | As per ISO spec, one FIR can contain one or more finger images within itself (like slap, etc). UIDAI currently supports single finger record only. If there is a requirement to send 2 fingers, 2 different biometric records should be sent. |
| 314 | Number of FMR/FIR should not exceed 10 | Application should ensure that one auth request should not contain more than 10 FMR/FIR records. | | | Auth Request has more than 10 finger records |
| 315 | Number of IIR should not exceed 2 | Application should ensure that one auth request should not contain more than 2 IIR records. | | | Auth Request has more than 2 iris records |
| 400 | "OTP" validation failed | Application should have provision for allowing user to provide OTP value again and after some retries (configurable) option to generate OTP again. | Please provide correct OTP value. | If there are repeated failures user is advised to generate new OTP and send the authentication request using the new OTP. | Incorrect OTP value is entered. Input not matching with the value in CIDR. |
| 401 | "Tkn" validation failed | Application should derive the value of tkn (currently only mobile number) from | | | Provided "Tkn" details are not matching with registered values in CIDR. |

| | | | | | |
|---|---|---|---|---|---|
| | | network. This element is meant for self-service transations on mobile (SMS/USSD, etc) where AUA derives the mobile number from the network provider and passes it on as part of API to use it as a factor. | | | |
| 500 | Invalid Skeyencryption | Application should not have hard coded digital certificate information. It should be configurable. | Technical Exception <No><br><br>Note: Application can throw Auth API error code number on screen. So that contact centre or application support helpline can understand the reason. | Contact technical helpdesk. | Use of wrong digital certificate for encryption of AES-256 Key (session key). |
| 501 | Invalid value for "ci" attribute in "Skey" element | Application should not have hard coded "ci" attribute value. It should be configurable. | Technical Exception <> | | Ensure that expiry date of UIDAI certificate used for encryption of Skey is specified as "ci" value. |
| 502 | Invalid Pid Encryption | Application should do extensive testing using UIDAI Test Auth Service to ensure compliance with auth API. | Technical Exception <No> | | Ensure that correct AES encryption has been used.<br><br>Ensure that AES key used for encryption of "Pid" XML was encrypted and specified as value for Skey. |
| 503 | Invalid HMac encryption | Application should do extensive testing using UIDAI Test Auth Service to ensure compliance with auth API. | Technical Exception <No> | | Ensure that correct AES encryption has been used.<br><br>Ensure that AES key used for encryption of "Hmac" was encrypted and specified as value for Skey.<br><br>Ensure that same AES key is used for encryption of Pid and Hmac. |
| 504 | Session key re-initiation required due to expiry or key out of sync | Application should have a provision to send full session key and initiate a new session in case of such failure. | Technical Exception <No> | Please try again. | When Synchronized Session Key scheme is used, this can happen if either session is expired (currently configured to max 4 hrs) or if the key goes out of sync. |
| 505 | Synchronized Skey usage is not allowed | Application should use full skey | Technical Exception <No> | Switch to full skey scheme | This happens when AUA does not have privilage to use SSK scheme |

| | | | | | |
|---|---|---|---|---|---|
| 510 | Invalid Auth XML format | Application Authentication request should comply to Authentication API latest version and application should validate its structure before sending it to CIDR for authentication. | Technical Exception <No> | Please ensure that the latest recommended API is used for application development. Refer UIDAI website for the latest version of API.<br><br>If this does not resolve the issue than please contact technical helpdesk. | Non compliance with supported Authentication API version structure in CIDR. |
| 511 | Invalid PID XML format | Application Authentication request should comply to PID XML format defined in Authentication API latest version and structural validation should be done before encryption of PID XML. | Technical Exception <No> | Please ensure that the latest recommended API is used for application development. Refer UIDAI website for the latest version of API.<br><br>If this does not resolve the issue than please contact technical helpdesk. | Non compliance with supported Authentication API version structure in CIDR. |
| 520 | Invalid device | Application should ensure that "tid" attribute in Auth XML has value "public" | Technical Exception <No> | | Using any other value other than "public" (all lower case, no spaces or special char) will result in this error. |
| 521 | Invalid Finger device (fdc in Meta element) | Application should obtain proper code from fingerprint sensor vendor and use it | Technical Exception <No> | | FDC codes are assigned as part of certification and application developer should use proper fdc code given by the fingerprint sensor/extractor vendor |
| 522 | Invalid Iris device (idc in Meta element) | Application should obtain proper code from iris sensor vendor and use it | Technical Exception <No> | | IDC codes are assigned as part of certification and application developer should use proper idc code given by the iris sensor/extractor vendor |
| 530 | Invalid authenticator code | Application should pass valid AUA code in authentication request which is registered with UIDAI. Value of this code should be configurable. | Technical Exception <No> | | AUA code used in Authentication request is invalid<br><br>or<br><br>AUA code used in the Auth URL is not same as the AUA code used in the Auth XML. |
| 540 | Invalid Auth XML version | Application should pass supported valid API version in authentication request. Value of this should be configurable. | Technical Exception <No> | | API version used in Auth XML (Authentication request) is either not supported or invalid. |
| 541 | Invalid PID XML version | Application should pass supported valid API PID XML version in | Technical Exception <No> | | Version of the "Pid" element used |

| | | | | | |
|---|---|---|---|---|---|
| | | authentication request. Value of this should be configurable. | | | In the PID XML (Authentication request) is either not supported or invalid. |
| 542 | AUA not authorized for ASA. | Application should ensure link is in place between AUA-ASA before sending request to CIDR. | | Ensure the authentication request is being sent through the authorized ASA as per the records of UIDAI.<br><br>or<br><br>Please contact UIDAI helpdesk to report the issue and to understand further steps for the updation of ASA-AUA linkage. | This error will be returned if AUA and ASA do not have linking in the portal |
| 543 | Department not associated with "AUA" | Application should ensure Department is added and associated with correct AUA before sending request to CIDR. | | Ensure the authentication request is being sent through the associated AUA as per the records of UIDAI.<br><br>or<br><br>Please contact UIDAI helpdesk to report the issue and to understand further steps for the updation of ASA-AUA linkage. | This error will be returned if Department specified in "sa" attribute is not added as "Sub-AUA" in portal |
| 550 | Invalid "Uses" element attributes | Application should use valid attributes defined in API for <Uses> tag and validation on Auth request should be done before sending request to CIDR. | Technical Exception <No> | | Invalid attributes used in Uses tag.<br><br>This error is typically reported if "bt" attribute has been specified but bio="n" in Uses element. "bt" attribute is required only if bio="y" in Uses element. |
| 561 | Request expired ("Pid->ts" value is older than N hours where N is a configured threshold in authentication server) | AUA application should not store Pid block and in case of application which are using thick client there should be a provision to sync up date with server at start. | 1. In case of Device/Client based Application<br><br>a. Either device date/time is behind current date/time or request is old. Please try again. | Please verify that the device/client date/time is synchronised with Indian Standard Time (IST) and resend the authentication request. | Either Device/Client/Server date/time is behind current one or old stored pid is getting sent. |

| | | | | | |
|---|---|---|---|---|---|
| | | | 2. In case of web based Application<br><br>a. Technical Exception <No> | | |
| 562 | Timestamp value is future time (value specified "Pid->ts" is ahead of authentication server time beyond acceptable threshold) | AUA application should not store Pid block and in case of application which are using thick client there should be a provision to sync up date with server at start. | 1. In case of Device/Client based Application<br><br>a. Either device date/time is ahead current date/time or request is old. Please try again.<br><br>2. In case of web based Application<br><br>a. Technical Exception <No> | Please verify that the device/client date/time is synchronised with Indian Standard Time (IST) and resend the authentication request. | Device/Client/server date/time is ahead than current date/time. |
| 563 | Duplicate request (this error occurs when exactly same authentication request was re-sent by AUA) | Application should ask user to try again. | Please submit your request again. | User is required to send the authentication request once again. | If same "Auth XML" is sent more than once to server, then, 2nd and subsequent requests will fail with this error. |
| 564 | HMAC Validation failed | Application should create HMAC using *SHA-256* | Technical Exception <No> | | HMAC is not calculated using API defined algorithm |
| 565 | License key has expired | Application should have a configurable License key management feature through which one can manage Key without changing application. | Technical Exception <No> | | Current License has expired. |
| 566 | Invalid license key | Application should have a License key management feature through which one can manage Key without changing application. | Technical Exception <No> | | License key used in application is invalid. |
| 567 | Invalid input (this error occurs when some unsupported characters were found in Indian language values, "lname" or "lav") | Application should have client/server level checks to stop users to input unsupported characters. | Technical Exception <No> | | some unsupported characters were found in Indian language values, "lname" or "lav" in Auth request XML |

| | | | | | |
|---|---|---|---|---|---|
| 568 | Unsupported Language | Application should have client/server level checks to restrict users to only select language from API supported local Language. | Technical Exception <No> | | Value of "lang" attribute is not from the list supported by authapi. |
| 569 | Digital signature verification failed (this means that authenticatio n request XML was modified after it was signed) | Application should ensure security of data end to end ie. From client/device to CIDR server by using appropriate communication protocol. | Technical Exception <No> | | Authentication request XML was modified after it was signed. |
| 570 | Invalid key info in digital signature (this means that certificate used for signing the authenticatio n request is not valid ¬it is either expired, or does not belong to the AUA or is not created by a well-known Certification Authority) | Application should have an independent module for signing Auth XML and certificate should be stored and manage outside of the application. | Technical Exception <No> | | Certificate used for signing the authentication request is not valid ¬it is either expired, or does not belong to the AUA or is not created by a well-known Certification Authority |
| 571 | PIN Requires reset (this error will be returned if resident is using the default PIN which needs to be reset before usage) | | Please reset your PIN in UIDAI updation application and use new PIN in this application. | Please change your default PIN through UIDAI updation client and resend your authentication request. | This error will be returned if resident is using the default PIN which needs to be reset before usage. |
| 572 | Invalid biometric position (This error is returned if biometric position value - "pos" attribute in "Bio" element - is not applicable for a given biometric type - "type" | Application should have client level validation to check "type" and corresponding valid "pos" values before creating PID block. | Technical Exception <no> | | This error is returned if biometric position value - "pos" attribute in "Bio" element - is not applicable for a given biometric type - "type" attribute in "Bio" element |

| | | | | | |
|---|---|---|---|---|---|
| | attribute in "Bio" element.) | | | | |
| 573 | Pi usage not allowed as per license | Application should have a configurable business rule which can restrict the usage of Pi attribute based on AUA license authorization. | Technical Exception <No> | | Pi usage not allowed as per license |
| 574 | Pa usage not allowed as per license | Application can have a client level check to restrict/allow entry of "pa" attribute as per license of AUA. | Technical Exception <No> | | Pa usage not allowed as per license |
| 575 | Pfa usage not allowed as per license | Application can have a client level check to restrict/allow entry of "pfa" attribute as per license of AUA. | Technical Exception <No> | | Pfa usage not allowed as per license |
| 576 | FMR usage not allowed as per license | Application can have a client level check to restrict/allow entry of "FMR" attribute as per license of AUA. | Technical Exception <No> | | FMR usage not allowed as per license |
| 577 | FIR usage not allowed as per license | Application can have a client level check to restrict/allow entry of "FIR" attribute as per license of AUA. | Technical Exception <No> | | FIR usage not allowed as per license |
| 578 | IIR usage not allowed as per license | Application can have a client level check to restrict/allow entry of "IIR" attribute as per license of AUA. | Technical Exception <No> | | IIR usage not allowed as per license |
| 579 | OTP usage not allowed as per license | Application can have a client level check to restrict/allow entry of "OTP" attribute as per license of AUA. | Technical Exception <No> | | OTP usage not allowed as per license |
| 580 | PIN usage not allowed as per license | Application can have a client level check to restrict/allow entry of "PIN" attribute as per license of AUA. | Technical Exception <No> | | PIN usage not allowed as per license |
| 581 | Fuzzy matching usage not allowed as per license | Application can have a client level check to restrict/allow entry of "ms" attribute in pi, pa and pfa element as per license of AUA. | Technical Exception <No> | | Fuzzy matching usage not allowed as per license |
| 582 | Local language usage not allowed as per license | Application can have a client level check to restrict/allow entry of local language attribute in pi, pa and pfa element as per license of AUA. | Technical Exception <No> | | Local language usage not allowed as per license |
| 584 | Invalid Pin code in Meta element | Pincode should have a valid value (in lov attribute) | Technical Exception <No> | | If pincode value is not one of the valid values in UIDAI system, this error occurs |

| | | | | | |
|---|---|---|---|---|---|
| 585 | Invalid Geo code in Meta element | Geo code value must be a valid lat.long value in decimal format as per spec (in lov attribute) | Technical Exception <No> | | If geo code does not have proper format as per spec (decimal representation with porecision upto 4 decimal values for lat and long), this error occurs |
| 710 | Missing "Pi" data as specified in "Uses" | Application should validate pid block before encrypting data with API specified PID block structure and "Uses" element attributes values to ensure PID block have all the elements and attributes. Client level validation should also be put to check all mandatory and conditional fields of API XML. | Technical Exception <No> | | Missing "Pi" data as specified in "Uses" |
| 720 | Missing "Pa" data as specified in "Uses" | Same as 710 | Technical Exception <No> | | Missing "Pa" data as specified in "Uses" |
| 721 | Missing "Pfa" data as specified in "Uses" | Same as 710 | Technical Exception <No> | | Missing "Pfa" data as specified in "Uses" |
| 730 | Missing PIN data as specified in "Uses" | Same as 710 | Technical Exception <No> | | Missing PIN data as specified in "Uses" |
| 740 | Missing OTP data as specified in "Uses" | Same as 710 | Technical Exception <No> | | Missing OTP data as specified in "Uses" |
| 800 | Invalid biometric data | AUA to review biometric device being used and whether templates are ISO compliant. | Technical Exception <No> | | FMR value is not ISO compliant –bad header or other issue with templates.<br><br>FIR/IIR value is not compliant, or templates could not be extracted for the given FIR/IIR for matching purposes. |
| 810 | Missing biometric data as specified in "Uses" | Same as 710 | Technical Exception <No> | | Missing biometric data as specified in "Uses" |
| 811 | Missing biometric data in CIDR for the given Aadhaar number | | Your Biometric data is not available in CIDR. | Ensure correct Aadhaar number is entered and try authenticating again.<br><br>After repeated failure, if the resident is genuine, exception handling provision | |

| | | | | | |
|---|---|---|---|---|---|
| | | | | would need to be followed to provide service.<br><br>Please contact UIDAI helpdesk to inform about the issue and to understand the steps for the updation of biometric information in CIDR. | |
| 812 | Resident has not done "Best Finger Detection". Application should initiate BFD application to help resident identify their best fingers. See Aadhaar Best Finger Detection API specification. | Application should make provison to initiate BFD application based on the error code to help resident identify their best fingers. | You have not done best finger detection so kindly proceed with the BFD process for successful authentication. | Refer Aadhaar Best Detection API specifications for details on the BFD process. | Resident has not done "Best Finger Detection". |
| 820 | Missing or empty value for "bt" attribute in "Uses" element | Same as 710 | Technical Exception <No> | | Missing or empty value for "bt" attribute in "Uses" element |
| 821 | Invalid value in the "bt" attribute of "Uses" element | Same as 710 | Technical Exception <No> | | Invalid value in the "bt" attribute of"Uses" element |
| 901 | No authentication data found in the request (this corresponds to a scenario wherein none of the auth data -Demo, Pv, or Bios – is present) | Application should validate that User giveatleast one auth factor before encryption of PID block. | Technical Exception <No> | | All factors of Auth are optional. Hence, it is possible to attempt an auth without specify any values for any of the factors –Pi, Pa, Pfa, Bio or Pv. If none of these elements have any value that can be used for authentication purposes, then, this error will be reported. |
| 902 | Invalid "dob" value in the "Pi" element (this corresponds to a scenarios wherein | Application should have a client level check to check dob date format and age business rules specified (Current Rule is that age should not be less than 0 and greater than 150 years) | Please enter dob in specified date format or enter age in specified range. | Re-enter the date of birth or age and resend a new authentication request. | "dob" attribute is not of the format "YYYY" or "YYYY-MM-DD", or the age of resident is not in valid range. |

| | | | | | |
|---|---|---|---|---|---|
| | "dob" attribute is not of the format "YYYY" or "YYYY-MM-DD", or the age of resident is not in valid range) | | | | |
| 910 | Invalid "mv" value in the "Pi" element | Same as 710 | Technical Exception <No> | | |
| 911 | Invalid "mv" value in the "Pfa" element | Same as 710 | Technical Exception <No> | | |
| 912 | Invalid "ms" value | Same as 710 | Technical Exception <No> | | |
| 913 | Both "Pa" and "Pfa" are present in the authentication request (Pa and Pfa are mutually exclusive) | Same as 710 | | | Attempt to use Pa and Pfa both in the same request can result in this error. |
| 930-939 | Technical error that are internal to authentication server | AUA/ASA should call UIDAI tech support. | Technical Exception <No> | | UIDAI server side issues. UIDAI tech support to review the scenario and take appropriate action. |
| 940 | Unauthorized ASA channel | AUA should consult ASA. | Technical Exception <No> | | |
| 941 | Unspecified ASA channel | AUA should consult ASA. | Technical Exception <No> | | |
| 980 | Unsupported option | AUA to review the auth client to check whether any dev feature is being used in prod | Technical Exception <No> | | Currently this error is not reported. Can be used in future. |
| 997 | Invalid Aadhaar Status | AUA application should have mechanism to handle this scenario as Aadhaar number is valid but its status is not active. | Your Aadhaar number status is not active. Kindly contact UIDAI Helpline. | | Aadhaar number status is either lost, deceased etc. and currently not active. |
| 998 | Invalid Aadhaar Number | AUA application should have a client level validation for Aadhaar number validity ie. should be 12 digits and conform to Verhoeff algorithm. | Ensure you have entered correct Aadhaar number. Please retry with correct Aadhaar number after sometime. | Ensure you have entered correct Aadhaar number. Please retry with correct Aadhaar number after sometime. | If client level validations are done then Aadhaar number does not exist in CIDR. |

| | | | | | |
|---|---|---|---|---|---|
| 999 | Unknown error | | Technical Exception <No> | Please contact authsupport team of UIDAI | |

### 5.1.1 BFD Service Error Codes

| Error Code | Description |
|---|---|
| B-300 | Biometric data did not match |
| B-314 | Number of fingers should not exceed 10 |
| B-500 | Invalid encryption of Skey |
| B-501 | Invalid certificate identifier in "ci" attribute of "Skey" |
| B-502 | Invalid encryption of Rbd |
| B-503 | Invalid encryption of Hmac |
| B-504 | Session key re-initiation required due to expiry or key out of sync |
| B-510 | Invalid "Bfd" XML format |
| B-511 | Invalid "Rbd" XML format |
| B-520 | Invalid device |
| B-530 | Invalid AUA code |
| B-540 | Invalid BFD XML version |
| B-541 | Invalid RBD XML version |
| B-542 | AUA not authorized for ASA. This error will be returned if AUA and ASA do not have linking in the portal |
| B-543 | Sub-AUA not associated with "AUA". This error will be returned if Sub-AUA specified in "sa" attribute is not added as "Sub-AUA" in portal |
| B-561 | Request expired ("Rbd->ts" value is older than N hours where N is a configured threshold in BFD server) |
| B-562 | Timestamp value is future time (value specified "Rbd->ts" is ahead of BFD server time beyond acceptable threshold) |
| B-563 | Duplicate request (this error occurs when exactly same BFD request was re-sent by AUA) |
| B-564 | HMAC Validation failed |
| B-566 | ASA license key has expired or is invalid |
| B-570 | Invalid key info in digital signature (this means that certificate used for signing the BFD request is not valid it is either expired, or does not belong to the AUA or is not created by a CA) |
| B-583 | Best finger detection not allowed as per license |
| B-811 | Missing biometric data in CIDR for the given Aadhaar number |
| B-941 | Unspecified ASA channel |

### 5.1.2 OTP Service Error Codes

| Error Code | Description |
|---|---|
| O-110 | Aadhaar number does not have verified mobile/email |
| O-111 | Aadhaar number does not have verified mobile |
| O-112 | Aadhaar number does not have both email and mobile. |
| O-510 | Invalid "otp" XML format |
| O-520 | Invalid device |
| O-521 | Invalid mobile number |
| O-522 | Invalid "type" attribute |
| O-530 | Invalid AUA code |
| O-540 | Invalid TP XML version |

| | |
|---|---|
| O-542 | AUA not authorized for ASA. This error will be returned if AUA and ASA d not have linking in the portal |
| O-543 | Sub-AUA not associated with "AUA". This error will be returned if Sub-AUA specified in "sa" attribute is not added as "Sub-AUA" in portal |
| O-565 | AUA License key has expired or is invalid |
| O-566 | ASA license key has expired or is invalid |
| O-569 | Digital signature verification failed |
| O-570 | Invalid key info in digital signature (this means that certificate used for signing the TP request is not valid it is either expired, or does not belong to the AUA or is not created by a CA) |
| O-940 | Unauthorized ASA channel |
| O-941 | Unspecified ASA channel |
| O-950 | Could not generate and/or send OTP |
| O-999 | Unknown error |

### 5.1.3 KUA/KSA Service Error Codes

| Error code | Description |
|---|---|
| K-100 | Resident authentication failed |
| K-200 | Resident data currently not available |
| K-540 | Invalid KYC XML |
| K-541 | Invalid KYC API version |
| K-542 | Invalid resident consent ("rc" attribute in "Kyc" element) |
| K-543 | Invalid timestamp ("ts" attribute in "Kyc" element) |
| K-544 | Invalid resident auth type ("ra" attribute in "Kyc" element) |
| K-545 | Resident has opted out of this service |
| K-550 | Invalid "Uses" element attributes – must have either bio or otp enabled for resident authentication |
| K-551 | Invalid "Txn" namespace (should be "UKC") |
| K-552 | Invalid license key |
| K-569 | Digital signature verification failed for KYC XML (means that authentication request XML was modified after it was signed) |
| K-570 | Invalid key info in digital signature for KYC XML (it is either expired, or does not belong to the KUA or is not created by a well known Certification Authority) |
| K-600 | AUA is invalid or not an authorized KUA |
| K-601 | ASA is invalid or not an authorized KSA |
| K-602 | KUA encryption key not available |
| K-603 | KSA encryption key not available |
| K-604 | KSA not allowed to sign |
| K-999 | Unknown error |
| K-955 | Technical error |

# 6. Annexure 2: Useful links

https://uidai.gov.in/images/resource/aadhaar_authentication_api_2_5.pdf

https://uidai.gov.in/images/resource/aadhaar_ekyc_api_2_5.pdf

https://uidai.gov.in/images/resource/aadhaar_otp_request_api_2_5.pdf