

# IRIS API v2

Τεκμηρίωση, Ιούλιος 2023



Πολεμική Αεροπορία - ΓΕΑ/ΚΜΗ

## Πίνακας Περιεχομένων

1. Εισαγωγή .....	2
2. Γενικά .....	2
3. Authorization .....	2
3.1 Έκδοση authorization token .....	2
4. IRIS API Calls .....	4
4.1 Αποστολή Αλληλογραφίας .....	4
4.1.1 Ανάκτηση ενεργών προφίλ .....	4
4.1.2 Ανάκτηση λίστας φορέων του Συστήματος .....	5
4.1.3 Αποστολή εγγράφου για εισαγωγή στο σύστημα .....	6
4.1.4 Παραδοχές- περιορισμοί σε Αποστολή Αλληλογραφίας .....	8
4.2 Λήψη Αλληλογραφίας .....	8
4.2.1 Ανάκτηση λίστας εγγράφων για παραλαβή .....	9
4.2.2 Ανάκτηση λίστας αρχείων που ανήκουν στο έγγραφο .....	10
4.2.3 Ανάκτηση αρχείου εγγράφου .....	12
4.2.4 Παραδοχές- περιορισμοί στη Λήψη Αλληλογραφίας .....	13
5. Δοκιμές .....	13
6. Ιστορικό Αλλαγών .....	13
6.1 Βελτιώσεις σε σχέση με την έκδοση V.1 .....	14

# 1. Εισαγωγή

Το "ΙΡΙΔΑ" αποτελεί ένα Σύστημα Ηλεκτρονικής Διακίνησης Εγγράφων (ΣΗΔΕ) και μέσω της παρούσας τεκμηρίωσης περιγράφεται ο τρόπος με τον οποίο είναι εφικτή η διασύνδεση του με άλλες εφαρμογές, τόσο για την παραλαβή εισερχόμενης αλληλογραφίας όσο και για την αποστολή της εξερχόμενης. Στο εξής όπου αναφέρεται ο όρος "Σύστημα" εννοείται το ΣΗΔΕ "ΙΡΙΔΑ" και όπου αναφέρεται ο όρος "Εφαρμογή" εννοείται όποια εξωτερική εφαρμογή επιθυμεί να διασυνδεθεί με αυτό.

## 2. Γενικά

Στο παρόν κείμενο γίνεται αναφορά στο σύνολο των λειτουργιών που διατίθεται από το API του συστήματος. Το IRIS API ακολουθεί τη λογική του REST, και για τη χρήση των παρεχόμενων λειτουργιών απαιτείται η αποστολή κατάλληλα διαμορφωμένων HTTP αιτημάτων. Στη συνέχεια του κειμένου, όπου γίνεται αναφορά σε REST resource, για να περιγράψουμε το αίτημα πρόσβασης σε έναν πόρο θα παραλείπεται το σταθερό τμήμα των διευθύνσεων στις οποίες θα αποστέλλονται τα αιτήματα. Για παράδειγμα, το αίτημα λήψης της λίστας των φορέων που εξυπηρετεί το σύστημα είναι:

```
GET /api/v2/external/roots
```

Το οποίο σημαίνει την αποστολή ενός αιτήματος GET στη διεύθυνση [BASE\_URL]/api/v2/external/roots . Στο δοκιμαστικό σύστημα το BASE\_URL είναι το εξής: <https://dev.iridacloud.gov.gr/iris>. Επομένως, η παραπάνω κλήση για τη λήψη των φορέων θα πρέπει να γίνει στο <https://dev.iridacloud.gov.gr/iris/api/v2/external/roots>.

Αντιστοίχως, για το παραγωγικό σύστημα το BASE\_URL είναι το <https://iridacloud.gov.gr/iris>.

## 3. Authorization

Το IRIS API χρησιμοποιεί το πρωτόκολλο [OAuth 2.0](#) για έλεγχο αυθεντικότητας (authentication) και εξουσιοδότηση χρήσης (authorization). Όλες οι κλήσεις προς το API προϋποθέτουν την λήψη του κατάλληλου access token. Για τη λήψη του token πρέπει να υπάρχει ήδη ένας πιστοποιημένος χρήστης του συστήματος, να διαθέτει δηλαδή έγκυρα credentials.

### 3.1 Έκδοση authorization token

**Endpoint**

```
POST /api/v2/common/token
```

**Σημαντικό** Το content type πρέπει να είναι application/x-www-form-urlencoded

**i. Parameters**

Δεν γίνεται χρήση Παραμέτρων.

**ii. Request body**

**Τύπος:** multipart/form-data

key	Value	Type	Description
username*	.....	String	To username του χρήστη του συστήματος.
password*	.....	String	To password του χρήστη του συστήματος.

\*Υποχρεωτικό πεδίο.

**iii. Request example**

**POST api/v2/token HTTP/1.1**  
**content-type: application/x-www-form-urlencoded**

username=test&password=test

Παράδειγμα κλήσης με χρήση του cURL:

```
curl -X POST "api/v2/token" -H "accept: text/plain" -H "Authorization: Bearer <token>" -H "Content-Type: multipart/form-data" -F "username=test" -F "password=test"
```

**iv. Server Responses**

Code	Response body	Details
<b>200</b>	<token>	Επιτυχής Κλήση
<b>200</b>	invalid_username_or_password	Λανθασμένα στοιχεία αιτήματος
<b>500</b>	Internal Server Error	Σφάλμα κατά την άντληση των στοιχείων

**v. Notes**

Η χρονική διάρκεια ισχύος του token είναι **οκτώ (8) ώρες**.

## 4. IRIS API Calls

Από τη στιγμή που η εφαρμογή αποκτήσει ένα διακριτικό πρόσβασης (token), τότε μπορεί να πραγματοποιήσει τις παρακάτω κλήσεις, συμπεριλαμβάνοντας το token στο **Authorization: Bearer** HTTP header.

Εάν μια κλήση στο API αποτύχει με ένα σφάλμα που σχετίζεται με το authorization, τότε η λεπτομέρεια του σφάλματος θα συμπεριληφθεί σε ένα WWW-Authenticate όπως το ακόλουθο, σε περίπτωση λήξης του token:

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Bearer error="invalid_token", error_description="The token expired at '04/29/2020 22:14:45'"
```

ή όπως το ακόλουθο, σε περίπτωση εσφαλμένου token:

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Bearer error="invalid_token", error_description="The signature is invalid"
```

### 4.1 Αποστολή Αλληλογραφίας

Όπως αναφέρθηκε και στην εισαγωγή, η εφαρμογή μπορεί να αποστέλλει έγγραφα τα οποία παράγει, προς αποδέκτες - Φορείς του συστήματος. Με τον όρο “Φορέας” εννοείται κάθε ανεξάρτητος Οργανισμός - Παραλήπτης αλληλογραφίας του συστήματος. Για χάρη απλότητας του σχεδιασμού του API, δεν παρέχεται καθολική λίστα των αποδεκτών του συστήματος. Ο τελικός αποδέκτης του εγγράφου θα πρέπει να περιγράφεται σε αυτό και είναι ευθύνη του συστήματος η δρομολόγηση του εγγράφου από τον Φορέα στον τελικά αποδέκτη. Εξ’ ορισμού όλα τα έγγραφα παραλαμβάνονται στον ρόλο της “**Κεντρικής Γραμματείας**” του Φορέα. Έτσι, π.χ αν η εφαρμογή επιθυμεί να στείλει έγγραφο προς τη Διεύθυνση Α’ του ΥΠ.ΕΣ, θα πρέπει να αναγράφεται η Διεύθυνση Α’ στο κείμενο του εγγράφου.

#### 4.1.1 Ανάκτηση ενεργών προφίλ

Όλες οι κλήσεις προς το API απαιτούν (ως header) την ύπαρξη ενός ενεργού προφίλ ("xProfile": "string") . Για τη λήψη του xProfile γίνεται η παρακάτω κλήση από έναν αυθεντικοποιημένο χρήστη της εφαρμογής, ο οποίος δύναται να διαθέτει ένα ή περισσότερα προφίλ.

##### Endpoint

```
GET /api/v2/external/profiles
```

##### i. Parameters

Δεν γίνεται χρήση Παραμέτρων.

##### ii. Request body

Δεν γίνεται χρήση.

### iii. Request example

```
GET /api/v2/external/profiles HTTP/1.1
content-type: application/json
Authorization: Bearer <token>
```

Παράδειγμα κλήσης με χρήση του cURL:

```
curl -X GET "api/v2/external/profile" -H "accept: text/plain"-H "Authorization: Bearer <token>
```

### iv. Server Responses

Code	Response body	Details
200	[List of x-profiles]	Επιτυχής Κλήση
401	Unauthorized	invalid_token
500	Internal Server Error	Σφάλμα κατά την άντληση των στοιχείων

### v. Notes

Η επιτυχής κλήση μας επιστρέφει μια λίστα με το σύνολο των ενεργών προφίλ του χρήστη:

```
[
  {
    "positionName": "string",
    "dutyName": "string",
    "xProfile": "string"
  }
]
```

## 4.1.2 Ανάκτηση λίστας φορέων του Συστήματος

Προκειμένου η εφαρμογή να μπορέσει να αποστείλει έγγραφα προς κάποιον φορέα που χρησιμοποιεί το σύστημα, θα πρέπει αρχικά να αποκτήσει την λίστα όλων των φορέων μέσω αυτής της κλήσης προς το API.

### Endpoint

```
GET /api/v2/external/roots
```

#### i. Parameters

Δεν γίνεται χρήση Παραμέτρων.

#### ii. Request body

Δεν γίνεται χρήση.

### iii. Request example

```
GET /api/v2/external/roots HTTP/1.1
content-type: application/json
Authorization: Bearer <token>
```

Παράδειγμα κλήσης με χρήση του cURL:

```
curl -X GET "api/v2/external/roots" -H "accept: text/plain" 'x-profile: profile' -H
"Authorization: Bearer <token>
```

### iv. Server Responses

Code	Response body	Details
200	[List of recipients]	Επιτυχής Κλήση
401	Unauthorized	invalid_token
500	Internal Server Error	Σφάλμα κατά την άντληση των στοιχείων

### v. Notes

Η επιτυχής κλήση μας επιστρέφει μια λίστα που περιέχει το αλφαριθμητικό αναγνωριστικό και το όνομα όλων των φορέων του συστήματος:

```
[
  {
    "Id": "string",
    "Description": "string"
  }
]
```

## 4.1.3 Αποστολή εγγράφου για εισαγωγή στο σύστημα

Εφόσον η εφαρμογή έχει ανακτήσει τη λίστα των φορέων του συστήματος μπορεί να επιλέξει κάποιον ή κάποιους από αυτούς για την αποστολή ενός εγγράφου. Ένα έγγραφο - όπως φαίνεται και παρακάτω στην τεκμηρίωση της κλήσης - μπορεί να περιέχει οσαδήποτε συνοδευτικά αρχεία.

### Endpoint

```
POST /api/v2/external
```

#### i. Parameters

Δεν γίνεται χρήση Παραμέτρων.

## ii. Request body

**Τύπος:** multipart/form-data

key	Type	Description
subject *	string	Το θέμα του εγγράφου.
registrationNumber *	string	Ο αριθμός πρωτοκόλλου του εγγράφου.
sender*	string	Το όνομα του αποστολέα του εγγράφου.
recipients*	array	Οι αποδέκτες του εγγράφου του συστήματος.
files*	array	Τα αρχεία του εγγράφου.

\*Υποχρεωτικό πεδίο.

## iii. Request example

Παράδειγμα κλήσης με χρήση του cURL:

```
curl -X 'POST' '/api/v2/external' -H 'accept: text/plain' -H 'x-profile:profile' -H 'Authorization: Bearer <token>' -H 'Content-Type: multipart/form-data' -F 'Subject=thema' -F 'RegistrationNumber=1234' -F 'Sender=test api' -F 'Recipients=5caaf5f3548e4b3df94d8b4a' -F 'Files=@sample1.pdf;type=application/pdf' -F 'Files=@sample1.pdf;type=application/pdf'
```

Στο συγκεκριμένο παράδειγμα η εφαρμογή έστειλε το έγγραφο με θέμα "test", αποστολέα τον "rolitis" και Αρ.πρωτοκόλλου "123", το οποίο περιείχε δύο συνοδευτικά αρχεία (το "sample1.pdf" και το "sample2.pdf"), προς τους αποδέκτες τους συστήματος, με αναγνωριστικά "5a0000934e1e8b0000859a00" και "6a0000934e1e8b0000859a00".

## iv. Server Responses

Code	Response body	Details
200	[List of document ids per recipient]	Επιτυχής Κλήση
400	Bad Request	The file is empty
400	Bad Request	The file exceeds <b>32 MB</b>
400	Bad Request	The file type isn't <b>pdf</b> or the file's signature doesn't match the file's extension
400	Bad Request	The request couldn't be processed (Error x)



<b>500</b>	Internal Server Error	Σφάλμα κατά την άντληση των στοιχείων
------------	-----------------------	---------------------------------------

#### v. **Notes**

Η επιτυχής κλήση μας επιστρέφει μια λίστα που περιέχει αντικείμενα της παρακάτω μορφής, όπου (*rootId*) το αλφαριθμητικό αναγνωριστικό του εγγράφου που αποστέλλεται, το όνομα του φορέα (*rootName*) και τον μοναδικό αριθμό πρωτοκόλλου που λαμβάνει σε κάθε φορέα.

```
[
  {
    "rootId": "string",
    "rootName": "string"
    "regNo": "string"
  }
]
```

Επομένως, για το παράδειγμα της παραγράφου 4.2iii αναμένουμε το αποτέλεσμα:

```
[
  {
    "rootId": " 5caaf5f3548e4b3df94d8b4a",
    "rootName": " Προεδρία της Δημοκρατίας",
    "regNo": "4"
  }
]
```

### 4.1.4 Παραδοχές- περιορισμοί σε Αποστολή Αλληλογραφίας

Στην παρούσα έκδοση του IRIS API ισχύουν τα εξής:

- 1) Είναι εφικτή η αποστολή ενός (1) εγγράφου ανά κλήση.
- 2) Λοιπά μεταδεδομένα που ενδεχομένως έχει το έγγραφο (όπως Διαβάθμιση, Τύπος κλπ) δεν μεταφέρονται στην κλήση, αλλά είναι ευθύνη του συστήματος η ενημέρωσή τους στην πλευρά του αποδέκτη.
- 3) Το μέγεθος του κάθε συνοδευτικού αρχείου δεν πρέπει να ξεπερνά τα 32MB.
- 4) Ο τύπος των συνοδευτικών αρχείων πρέπει να είναι pdf ή doc ή docx.

## 4.2 Λήψη Αλληλογραφίας

Η εφαρμογή εκτός από αποστολή αλληλογραφίας προς το σύστημα μπορεί να λάβει αντίστοιχα και εισερχόμενη αλληλογραφία αξιοποιώντας κατάλληλα τις τρεις (3) κλήσεις του API που περιγράφονται στη συνέχεια.

## 4.2.1 Ανάκτηση λίστας εγγράφων για παραλαβή

Η πρώτη κλήση που πρέπει να γίνει, είναι αυτή που επιστρέφει μια λίστα με τα έγγραφα που έχει ο χρήστης της εφαρμογής για παραλαβή.

### Endpoint

```
GET /api/v2/external/inbox/{received}
```

#### i. Parameters

Η παράμετρος *received* (*boolean*) είναι υποχρεωτική και χρησιμοποιείται για να διαφοροποιηθούν τα έγγραφα τα οποία έχουν παραληφθεί (παραλαβή γίνεται με τη χρήση του endpoint `/api/v2/external/document/{id}`) και σε αυτά που δεν έχουν παραληφθεί ακόμα.

#### ii. Request body

Δεν γίνεται χρήση.

#### iii. Request example

```
GET /api/v2/external/inbox/false HTTP/1.1  
content-type: application/json  
Authorization: Bearer <token>
```

Παράδειγμα κλήσης με χρήση του cURL:

```
curl -X GET "api/v2/inbox/false" -H "accept: text/plain" 'x-profile:profile' -H  
"Authorization: Bearer <token>"
```

#### iv. Server Responses

Code	Response body	Details
200	[List of documents]	Επιτυχής Κλήση
400	Bad Request	The request couldn't be processed (Error x)
401	Unauthorized	invalid_token
500	Internal Server Error	Σφάλμα κατά την άντληση των στοιχείων

#### v. Notes

Η επιτυχής κλήση μας επιστρέφει μια λίστα που περιέχει το αλφαριθμητικό αναγνωριστικό του κάθε εγγράφου (*documentId*), τον τίτλο (*title*) που λαμβάνει από το ΣΗΔΕ «Ιριδα», αν έχει αναγνωσθεί το έγγραφο (*received*) και το σύνολο των σχετικών εγγράφων που διαθέτει (*directoryReferences*), όπου κάθε ένα αντικείμενο της λίστας διαθέτει το μοναδικό αναγνωριστικό του σχετικού εγγράφου

(*referenceId*), τον αριθμό πρωτοκόλλου (*referenceRegNo*) που έχει λάβει και τον αποστολέα (*sender*) {που είναι το αρύ που επικοινωνεί με το ΣΗΔΕ «Ίριδα»}

```
[
  {
    "documentId": "string",
    "title": "string",
    "received": "boolean",
    "directoryReferences": [
      {
        "referenceId": "string",
        "referenceRegNo": "string",
        "sender": "string"
      }
    ]
  }
]
```

Παράδειγμα χρήστη εφαρμογής με ένα (1) έγγραφο προς παραλαβή στο inbox του:

```
[
  {
    "documentId": "649bf9e7dc52b809c7188d65",
    "title": "176/Σ.86/28-06-23/ΥΠΟΥΡΓΕΙΟ ΕΣΩΤΕΡΙΚΩΝ",
    "regNo": "176",
    "received": false,
    "directoryReferences": [
      {
        "referenceId": "649bf99bdc52b809c7188cb3",
        "referenceRegNo": "175",
        "sender": "test api"
      }
    ]
  }
]
```

Όπου "649bf9e7dc52b809c7188d65" το μοναδικό αναγνωριστικό του εγγράφου στο Σύστημα, "176/Σ.86/28-06-23/ΥΠΟΥΡΓΕΙΟ ΕΣΩΤΕΡΙΚΩΝ" ο τίτλος του εγγράφου στο Σύστημα, με 176 τον Αρ. Πρωτοκόλλου. Επίσης αυτό το έγγραφο απαντάει στο έγγραφο με μοναδικό αναγνωριστικό "649bf99bdc52b809c7188cb3" που το σύστημα είχε παραλάβει από την Εφαρμογή, το οποίο είχε πρωτοκολληθεί με τον Αρ. 175 και είχε ως αποστολέα τον "test api".

## 4.2.2 Ανάκτηση λίστας αρχείων που ανήκουν στο έγγραφο

Η δεύτερη κλήση, είναι αυτή που επιστρέφει μια λίστα που περιέχει το όνομα και το αλφαριθμητικό αναγνωριστικό των συνοδευτικών με το έγγραφο αρχείων.

### Endpoint

```
GET /api/v2/external/document/{id}
```

#### i. Parameters

**id** Το αλφαριθμητικό αναγνωριστικό του εγγράφου (Υποχρεωτική)

**ii. Request body**

Δεν γίνεται χρήση.

**iii. Request example**

```
GET /api/v2/external/document/0000006f077e044750f00000 HTTP/1.1
content-type: application/json
Authorization: Bearer <token>
```

Παράδειγμα κλήσης με χρήση του cURL:

```
curl -X GET "api/v2/document/external/0000006f077e044750f00000" -H "accept: text/plain" 'x-profile:profile' -H "Authorization: Bearer <token>"
```

**iv. Server Responses**

Code	Response body	Details
200	[List of files]	Επιτυχής Κλήση
400	Bad Request	The request couldn't be processed (Error x)
401	Unauthorized	invalid_token
500	Internal Server Error	Σφάλμα κατά την άντληση των στοιχείων

**v. Notes**

Η επιτυχής κλήση μας επιστρέφει μια λίστα που περιέχει το αλφαριθμητικό αναγνωριστικό του αρχείου και το όνομα όλων των συνοδευτικών αρχείων:

```
[
  {
    "Id": "string",
    "Description": "string"
  }
]
```

Παράδειγμα του εγγράφου με Id: 0000006f077e044750f00000, το οποίο περιέχει δύο (2) συνοδευτικά αρχεία.

```
[
  {
    "id": "0000006f077e044750f11111",
    "description": "test1.pdf"
  },
  {
    "id": "0000006f077e044750f11112",
    "description": "test2.pdf"
  }
]
```

### 4.2.3 Ανάκτηση αρχείου εγγράφου

Η τρίτη και τελευταία κλήση, είναι αυτή που επιστρέφει το συνοδευτικό αρχείο.

#### Endpoint

**GET /api/v2/external/document/{docId}/file/{fileId}**

#### i. Parameters

**docId** Το αλφαριθμητικό αναγνωριστικό του εγγράφου (Υποχρεωτική)

**fileId** Το αλφαριθμητικό αναγνωριστικό του αρχείου (Υποχρεωτική)

#### ii. Request body

Δεν γίνεται χρήση.

#### iii. Request example

```
GET /api/v2/external/document/0000006f077e044750f00000/file/
0000006f077e044750f11111 HTTP/1.1
content-type: application/json
Authorization: Bearer <token>
```

Παράδειγμα κλήσης με χρήση του cURL:

```
curl -X GET
"api/v2/external/document/0000006f077e044750f00000/file/0000006f077e044750f11111
" -H "accept: text/plain" -H 'x-profile:profile' "Authorization: Bearer <token>
```

#### iv. Server Responses

Code	Response body	Details
200	Binary File	Επιτυχής Κλήση
400	Bad Request	The request couldn't be processed (Error x)
401	Unauthorized	invalid_token

Code	Response body	Details
500	Internal Server Error	Σφάλμα κατά την άντληση των στοιχείων

#### v. **Notes**

Η επιτυχής κλήση μας επιστρέφει ένα αντικείμενο τύπου DocumentFileDto με τις απαραίτητες πληροφορίες του συνοδευτικού αρχείου:

```
[
  {
    "fileName": "string",
    "base64ByteArray": "string",
    "mediaType": "string"
  }
]
```

### 4.2.4 Παραδοχές- περιορισμοί στη Λήψη Αλληλογραφίας

Στην παρούσα έκδοση του IRIS API ισχύουν τα εξής:

- 1) Είναι εφικτή η λήψη ενός (1) αρχείου ανά κλήση.
- 2) Λοιπά μεταδεδομένα που έχει το έγγραφο (όπως Διαβάθμιση, Τύπος κλπ) δεν μεταφέρονται στην κλήση, αλλά η αξιοποίηση τους γίνεται με μέριμνα της εφαρμογής, εφόσον αυτό είναι επιθυμητό.
- 3) Μετά την επιτυχή εκτέλεση της κλήσης που περιγράφηκε στην παράγραφο 4.2.1, το inbox του χρήστη της εφαρμογής δεν αδειάζει.
- 4) Μετά την επιτυχή εκτέλεση της κλήσης που περιγράφηκε στην παράγραφο 4.2.2, το έγγραφο αφαιρείται από το inbox του χρήστη, άρα δεν θα εμφανίζεται στα αποτελέσματα της κλήσης της 4.2.2, θα είναι όμως προσβάσιμο μέσω της 4.2.3.
- 5) Δεν υφίσταται περιορισμός στο μέγεθος και στον τύπο των συνοδευτικών αρχείων.

## 5. Δοκιμές

Δοκιμές για την εκτέλεση των κλήσεων μπορούν να γίνουν μέσω του Swagger UI στη διεύθυνση <https://dev.irdacloud.gov.gr/iris/external-docs>

## 6. Ιστορικό Αλλαγών

## 6.1 Βελτιώσεις σε σχέση με την έκδοση V.1

1. Σε κάθε κλήση του API προστέθηκε ως header το x-profile του χρήστη της εφαρμογής.
2. Προστέθηκε η δυνατότητα ανάκτησης μόνο των εγγράφων που δεν έχουν παραληφθεί.
3. Προστέθηκε ο Αρ. Πρωτοκόλλου του Συστήματος μετά την αποστολή εγγράφων από την εφαρμογή.
4. Προστέθηκε η δυνατότητα απάντησης εγγράφου του συστήματος σε περισσότερα του ενός εγγράφων της εφαρμογής.
5. Διορθώθηκε η διαδικασία παραλαβής εγγράφων από την εφαρμογή.