

eOPF Index Data Feed (IDF) External API — Interface Control Document (ICD)

The provider-facing interface contract for submitting, updating, deleting and retrieving eOPF document index records and their PDF content.

Document version 2.2 — September 2026 • API version v1 • © OPM eOPF, 2026

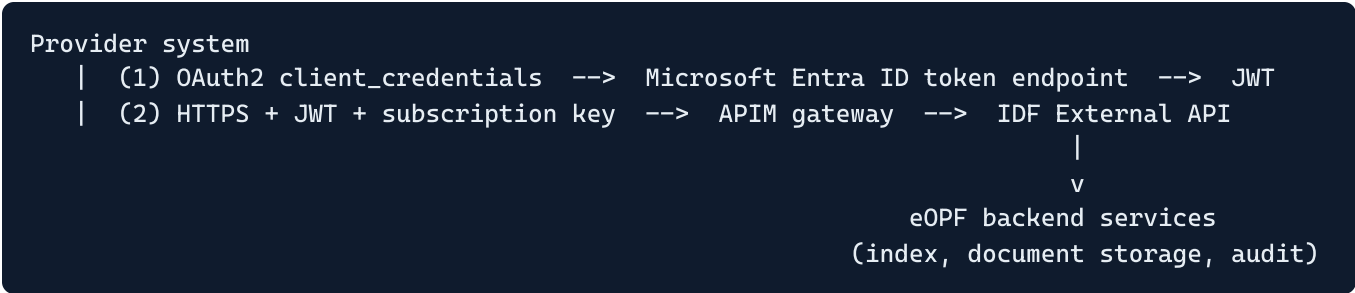
1. Purpose & Scope

The IDF External API is a REST/JSON service through which authorized provider systems submit eOPF document index records — and their associated PDF content — to the U.S. Office of Personnel Management (OPM) electronic Official Personnel Folder (eOPF), and look documents back up for verification. This Interface Control Document describes the external contract: authentication, endpoints, transport and headers, exchanged data, environments, versioning, and error handling.

This document contains no secrets, credentials, connection strings, or internal implementation detail. It is safe to share with integration partners. Client secrets, subscription keys, and provider-specific application IDs are issued privately during onboarding.

2. System Context

A provider system makes outbound HTTPS calls to two endpoints: the Microsoft Entra ID token endpoint (to obtain a JWT) and the IDF API published through Azure API Management (APIM). All eOPF backend systems sit behind the API and are never contacted directly.



3. Interface Inventory

ID	Interface	Direction	Protocol / Format	Purpose
IF-AUTH	Token acquisition	Provider → Microsoft Entra ID	HTTPS, application/x-www-form-urlencoded → JSON	Obtain a JWT bearer token
IF-API	IDF document API	Provider → APIM → IDF API	HTTPS REST, JSON	Add / Update / Delete / Verify / Request documents, identity, discovery

4. IF-AUTH — Authentication Interface

Every request is secured by **two independent layers**; both must be present and valid or the call is rejected at the gateway.

Layer 1 — OAuth 2.0 bearer token (Microsoft Entra ID)

Field	Value	Notes
Flow	<code>client_credentials</code>	Service-to-service; no user interaction
Token endpoint	<code>https://login.microsoftonline.com/3ce07c0b-30c5-4bcd-ad92-238975b649af/oauth2/v2.0/token</code>	OPM tenant (<code>connect.opm.gov</code>)
<code>grant_type</code>	<code>client_credentials</code>	Required
<code>client_id</code>	Your application (client) ID	Required — issued at onboarding
<code>client_secret</code>	Your client secret	Required — store in a secrets manager
<code>scope</code>	Tier-specific (below)	Required

Tier	Valid IDF scope (request exactly this value)
Non-production (TST / UAT)	<code>api://f5eb6fd8-5fbe-4dbf-953e-0145aec1adf3/.default</code>
Production (PRD)	<code>api://2aad6647-c6eb-4195-96b3-c3c66cb7b06d/.default</code>

Scopes are feed-specific. Each eOPF data feed (IDF, EDF, FDF) has its own API audience, and non-production and production differ again. A token requested with another feed's scope, or a non-production token sent to production, is rejected at the gateway with `401`. The scope the IDF gateway expects is also returned by `GET /v1/Auth/config`.

The response JSON contains `access_token`, a JWT typically valid for ~60 minutes. **Cache the token and refresh it shortly before expiry** — do not mint a new token per call. The gateway validates the token's issuer and audience; the API additionally authorizes your token's `appid` and its provisioned subagencies.

Production uses separate app registrations: production tokens must be requested with the **production** scope using your production `client_id` /secret. A non-production token sent to production is rejected (`401`).

Token request example (non-production)

Replace the placeholders with the client ID and client secret issued to your system at onboarding (Section 10). For production, use your production client ID and secret with the production scope above.

```
$body = @{
    client_id      = '<your-client-id>'
    client_secret  = '<your-client-secret>'      # from your secrets manager; never commit it
    scope          = 'api://f5eb6fd8-5fbe-4dbf-953e-0145aec1adf3/.default'  # IDF non-production
    grant_type     = 'client_credentials'
}
$token = (Invoke-RestMethod -Method Post -Body $body `
    -Uri 'https://login.microsoftonline.com/3ce07c0b-30c5-4bcd-ad92-238975b649af/oauth2/v2.0/token').access_token
```

```
curl -s -X POST 'https://login.microsoftonline.com/3ce07c0b-30c5-4bcd-ad92-238975b649af/oauth2/v2.0/token' \
  -d 'client_id=<your-client-id>' -d 'client_secret=<your-client-secret>' \
  -d 'scope=api://f5eb6fd8-5fbe-4dbf-953e-0145aec1adf3/.default' -d
'grant_type=client_credentials'
```

Layer 2 — APIM subscription key

- Issued per provider under the IDF APIM product; you receive a **primary** and a **secondary** key — send **one** per call.
- Two keys enable zero-downtime rotation: switch to the secondary, rotate the primary, switch back.
- The key is scoped to the gateway/environment, not to your token.

Common request headers

Header	Value	Notes
Authorization	Bearer <jwt>	Required
Ocp-Apim-Subscription-Key	APIM subscription key	Required at the gateway
Content-Type	application/json	On requests with a body
Accept	application/json	Recommended

5. Environments & Base URLs

Env	Base URL (outside OPM)	Status
TST	<code>https://apim-np.opm.gov/eopf-tst/idf-ext/</code>	Available
UAT	<code>https://apim-np.opm.gov/eopf-uat/idf-ext/</code>	Available
PRD	<code>https://apim.opm.gov/eopf/idf-ext/</code>	Live (validated September 2026)

Inside OPM: replace the hostname with `gateway.apim-np.opm.gov` / `gateway.apim.opm.gov` ; paths are identical. Make the base URL configurable per environment — never hardcode one. The interactive API reference and your subscription keys are available on the non-production Developer Portal at `https://developer.apim-np.opm.gov` .

6. IF-API — Operations

Operation	Verb & Route	Request body	Purpose
Add Document	POST <code>/v1/Index/AddDocument</code>	AddDocumentRequest	Index a new document and store its PDF
Update Document	PUT <code>/v1/Index/UpdateDocument</code>	UpdateDocumentRequest	Update an existing document by DocGUID
Delete Document	DELETE <code>/v1/Index/DeleteDocument</code>	DeleteDocumentRequest	Soft-delete a document by DocGUID (requires the <code>IDF.Delete</code> role)
Verify Folder	POST <code>/v1/Index/VerifyFolder</code>	VerifyFolderRequest	Pre-check whether an eOPF folder exists for each { SSN, subagency } pair
Request Document	GET <code>/v1/Index/RequestDocument?docGUID={guid}</code>	—	Look up a single document's index & PDF
Identity (Me)	GET <code>/v1/me</code>	—	Caller identity, roles, authorized subagencies, token state — recommended first call
Documentation	GET <code>/v1/Index/Documentation</code>	—	Endpoint documentation summary; no token required
Auth Config	GET <code>/v1/Auth/config</code>	—	Auth-configuration discovery (token endpoint, expected scope); no token required
Error Catalog	GET <code>/v1/errors</code>	—	Live catalog of HTTP and eOPF-specific error codes; no token required

All routes carry the required `/v1` version segment — there is no unversioned fallback (an unversioned path returns `404`). The OpenAPI/Swagger definition published on the Developer Portal is the authoritative field-level schema.

7. Data Model

AddDocumentRequest / UpdateDocumentRequest

Field	Type	Required	Description
DocGUID	string	Yes	Unique document identifier (GUID)
SSN	string	Yes	Employee SSN the document belongs to (9 digits, no dashes); required on Add and on Update
FormNumber	string	Yes	eOPF form number
Noa1 / Noa2	string	Cond.	Nature-of-action codes; validated against the form and effective date when supplied
EffectiveDate	string	Yes	Document effective date (yyyy-MM-dd)
SubagencyCode	string (4)	Yes	Owning subagency; must be authorized for the caller
AgencyAccountName	string (≤40)	Yes	eOPF agency-account short name the subagency rolls up to (e.g. OM00 → OPM , TR93 → IRS); missing or over-length returns eOPF error 941 . The mapping for your onboarded subagencies is provided by the OPM eOPF integration team.
BatchDesignator	string (≤5)	No	Optional batch designator recorded with the submission
Bytes	byte[] (base64 on the wire)	Cond.	PDF file content — unencrypted PDF only; up to 355 MB on the Add/Update paths

DeleteDocumentRequest

Field	Type	Required	Description
DocGUID	string	Yes	Identifier of the document to delete
SSN	string	Yes	Owning employee SSN (ownership check)

VerifyFolderRequest

Field	Type	Required	Description
Folders	array	Yes	List of { SSN , SubagencyCode } pairs to check

The response returns one result per submitted pair, in order, with `FolderExists` per pair (SSNs are echoed masked). Responses across the API share a common envelope: `Success` , `StatusCode` , `CorrelationId` , and an `ErrorMessages` list of eOPF error codes with descriptions.

8. Versioning

The API uses URL-segment versioning (`/v{version}/...` , currently `/v1/`). Breaking changes ship under a new segment; the previous version keeps its contract. The `api-supported-versions`

response header lists versions currently served; monitor `api-deprecated-versions` and migrate before a sunset.

9. Responses, Errors & Retries

Status	Meaning	Recommended provider action
200 / 2xx	Operation succeeded	—
400	Validation failure (invalid SSN, missing required field, invalid/encrypted PDF, ...) with eOPF error codes in the body	Correct the request; consult <code>GET /v1/errors</code>
401	Missing or invalid token / subscription key	Refresh the token; verify scope/tier and key
403	Token authenticated but not provisioned for the operation or target subagency	Confirm provisioning with the integration team
404	Path not found — usually a missing <code>/v1</code> segment or wrong base URL	Include the version segment; verify the environment base URL
409	Duplicate content (same bytes and identity under a new DocGUID)	Do not resubmit identical documents
413	Payload too large for the path	Respect the 355 MB upload limit; use Add/Update paths for PDF payloads
429	Rate limited	Honor <code>Retry-After</code> ; apply exponential backoff with jitter
5xx	Transient server / gateway error	Retry idempotent calls with backoff; alert if persistent

- Retry only transient failures (429 , 502 – 504 , network interruptions) with exponential backoff plus jitter; honor an explicit `Retry-After` .
- Keep submission concurrency modest and configurable to stay within gateway rate limits.
- Log the `x-Correlation-Id` response header (echoed from the request or generated by the gateway) and include it when reporting issues.

If a call returns HTTP 200 with an HTML body (e.g. a gateway “Invalid URL” page) instead of JSON, the request did not match a published operation — verify the base URL and the `/v1` segment.

10. Data Provider Onboarding for API Access

Access to the IDF External API is requested from, and provisioned by, the OPM eOPF team at eOPFOperations@opm.gov. The steps are:

1. **Complete the Provider Intake Form** and submit it to **eOPFOperations@opm.gov**. The form is a separate document, `IDF-Provider-Intake-Form.pdf`, distributed with this ICD (relative location `../Intake-Form/`). It collects your system details, the environments requested, the subagency codes you will submit for, roles, technical contacts, your secrets-delivery channel, and the public source IP addresses to allow-list. It contains no secrets.
2. **The OPM eOPF team provisions** your Microsoft Entra ID app registration, provider and subagency authorization, and APIM subscription (primary and secondary keys).
3. **The OPM eOPF team delivers your credentials** (client secret and subscription keys) through the secure channel you selected on the form — never by email.
4. **Verify connectivity to the API in TST:**
 - a. acquire a token and call `GET /v1/me` against the TST base URL (Section 5) with the headers in Section 4;
 - b. an HTTP `200` response with your identity JSON confirms both credential layers end to end.

Providers integrate in TST first, promote to UAT for sign-off, and receive production access after UAT sign-off; production is provisioned separately (app registration, gateway allow-list, and APIM subscription) on request through the same mailbox. Send questions about onboarding, credential rotation, or changes to your egress IP addresses to **eOPFOperations@opm.gov**.

This ICD reflects the IDF External API v1 contract. Exact endpoint URLs are environment-specific; subscription keys and credentials are issued privately during onboarding and are deliberately omitted here.