eFax Corporate®

700 S. Flower St, Suite 1500 ▪ Los Angeles CA 90017 ▪ tel: 323.860.9200 ▪ fax: 323.860.9201

Using the eFax® Enterprise API Portal — A Quick Start Guide

VERSION: 2.0

AUTHOR: Consensus Cloud Solutions®, Inc.

DATE: 11/5/2025

Legal Notice

CONTENTS

INTRODUCTION

Welcome to the eFax® Enterprise API!

By now, you should have received a Welcome email with your API details to allow testing via the Enterprise API Developer's Portal or by using Postman. If you did not receive this, please contact your account team Sales Engineer.

This Quick Start Guide provides information needed to start placing API calls and test the overall functionality of the Enterprise API. The Enterprise API Portal is a public facing, comprehensive source of information and will be an essential reference for anyone coding an integration of fax functionality into an application.

On the portal-landing page, there are icons for multiple documents:

- **Getting Started** – a collection of reference documents including error codes, payload limits, supported file types, custom headers, etc.
- **Fax Services API** – an API specification for methods involved in the sending, receiving, listing, and deleting fax images and metadata.
- **Admin API** – an API specification for methods involved in managing users, fax numbers, enabling the API, listing API instances, listing notification URLs, etc. (primarily used by resellers).

IMPORTANT VARIABLES

As a user of the Enterprise API, you should be aware of these important variables:

Variable	Description
AppID	A unique, alphanumeric identifier assigned to each instance of an API on the Enterprise API platform. Customers may have one or several AppIDs (i.e., DEV, QA, PROD), but each AppID, whether on the same customer account or not, is a completely separate environment. It is not possible to regenerate or revise this variable.
API Key	A unique alphanumeric secret used in conjunction with the AppID to generate OAuth2 bearer tokens. Treat the API Key as a sensitive secret; never pass it in the same correspondence as the AppID.
UserID	A unique ID passed as a header in every API call that represents the user/fax number engaged for that call. Only used with the Fax Services API.
Fax Number	A fax (telephone) number for all inbound and outbound traffic via the API. For numbers in North America, the format is 1+areacode+number (13235551212).

GENERATE YOUR BEARER TOKEN

Follow these steps to generate a bearer token using the Stoplight testing tool embedded in the portal:

1. From the [Enterprise API Portal](#) landing page, click **Fax Services API**. The Fax Services API Specification opens.
2. In the left hand menu under **API Integration** heading, click **Generate a New OAuth2 bearer token (POST /tokens)**. The page that opens provides detailed information about using this call.
3. At the bottom of the page, under **Send a Test Request**, use the embedded Stoplight tool to place a call and generate the **bearer token**.
   a. Select the **Settings** tab.
   b. In the **username** field enter your **App ID**; in the **password** field, enter your **API Key**.
   c. Click the **Send** button. The Response tab opens with your **bearer token**.

USING THE PORTAL TO PLACE ADDITIONAL CALLS

In addition to the **Generate a New OAuth2 bearer token**, the API includes many other calls used to send, receive, and manage faxes and metadata. Check the portal for the most current listing of available calls. Commonly used calls include:

- List received faxes based on search criteria (GET /faxes/received)
- List sent faxes based on search criteria (GET /faxes/sent)
- Retrieve a fax's metadata (GET /faxes/{fax_id}/metadata)
- Retrieve a fax's image (GET /faxes/{fax_id}/image)
- Send a fax (POST /faxes)
- Delete a fax (DELETE /faxes/{fax_id}

HEADERS

Headers are an important part of each API call. Each call requires headers for the following:

- **User-id** (as described above).
- **Authorization** – include the token string in this header. Be sure to indicate the token type by including the word **bearer** and a space immediately preceding the token string.
- **Content-type** – calls with a body will need a content-type header that specifies application/json.

Exceptions

- When generating a bearer token a Basic token is used and there are specific requirements for the **content-type** header and **grant_type** in the body of the call. Be sure to review the documentation and examples for this call concerning the construction of a Basic token and the specified **content-type**.
- The call to check operational status of the API requires no headers (GET /health).

Refer to the API Specification for details about other headers contained in the API calls.

QUERY PARAMETERS

For the two List calls (GET /faxes/received and GET /faxes/sent), use the **Query** tab to enter the search parameters.

When using the min and max query parameters, use the ISO 8601 format. If you are testing with the embedded Stoplight tool in the Enterprise API portal, then URL encode these values. Your coded call request to the endpoint also requires URL encoding of date/time values. For example:

- **yyyy-mm-ddThh:mm:ss.fff+|-0000** (basic format)
- **2021-06-09T09:17:32.625-0800** (date and time including milliseconds with -8 hour time zone shift)
- **2021-06-09T09:17:32-0800** (date and time no milliseconds with -8 hour time zone shift)
- **2021-06-09T09%3A17%3A32-0800** (URL encoded)

NOTE: URL encoding is not necessary when testing in Postman.

RETRIEVE AND DELETE CALLS

For the calls to retrieve and delete faxes, it is necessary to enter the **fax_id** in the URL for the call as a path parameter.

In the Stoplight tool on the Enterprise API portal, add the **fax_id** in the **Settings** tab for a test call; in Postman add it directly in the URL.

METHODS FOR RECEIVING FAXES

There are two primary methods for receiving faxes in the eFax® Enterprise API. Users can:

1. Pull faxes using either of the List Received Faxes endpoints in the API.

    The advantage of this approach is that it allows users to take advantage of "out-of-the-box" API functionality to set customized polling intervals (within certain limits) to meet specific business needs.

2. Use the Webhook Notification Function to obtain faxes through an auto-posted URL notification. This method allows eFax Enterprise API Admins to provide a destination URL, of their choosing, where inbound faxes notifications are auto-posted. These notifications include a separate link to download the fax.

    The advantage of this approach is that it is more of a set it and forget it approach. Once setup, inbound faxes flow into your application, as received, to the designated URL. From there the client can program any required backend actions/routing based upon metadata posted to the URL.

    Webhook notifications can be set up to require HMAC authentication.

Refer to **Receiving Faxes** in the **Getting Started** section of the eFax Enterprise API Portal for more information.

POSTMAN

If you are using Postman as your API testing tool, use this link to import a collection that includes our API calls and sample code: https://www.postman.com/collections/268299a822d69a8a05c5.

VIEWING RECEIVED FAXES

For testing purposes, you can send and receive simultaneously on the demo fax number provided to you. Received faxes can be retrieved by API call, can be forwarded to email, and can be viewed via a web portal. If you need delivery via email or access to the web portal to view received faxes, please contact your Sales Engineer.

REVISION HISTORY

Version	Date	Comments
1.0	February 26, 2019	First release version
1.1	March 5, 2019	Incorporated eFax Corporate® logo to cover page.
1.2	March 19, 2019	Documented temporary deactivation of the Generate API Key function.
1.3	May 6, 2019	Added the **Methods for Receiving Faxes** section.
1.4	May 21, 2019	Recent production release of eFax® Enterprise Fax API mandated changes to this guide; new Postman link added.
1.5	May 8, 2020	Updated Welcome message and instructions for generating API key.
1.6	June 16, 2020	Updated Postman collection.
1.7	July 8, 2021	Updated document to reflect changes made to the API in the past year.
2.0	November 6, 2025	Rebranded document with new eFax Corporate logo.