

---

**PACER Authentication  
Application Programming Interface (API)  
User Guide**

---

**December 2021**

## Contents

|                                     |    |
|-------------------------------------|----|
| PACER Authentication API .....      | 3  |
| Authentication Examples .....       | 4  |
| JSON Request.....                   | 5  |
| XML Request .....                   | 5  |
| cURL.....                           | 6  |
| Logout Examples.....                | 7  |
| JSON Request.....                   | 7  |
| XML Request .....                   | 8  |
| cURL.....                           | 8  |
| Java Examples .....                 | 9  |
| Common Error Message Examples ..... | 11 |

## PACER Authentication API

To access court records, the first step is to get an authentication token using your PACER username and password. If you do not have a PACER account, you may register for one at the appropriate account registration URL. The PACER authentication service accepts a valid PACER username and password and returns an authentication token.

The PACER Authentication API provides a way to authenticate automatically and without a user interface. This can help facilitate access for automated systems.

For API testing, a separate PACER QA environment is available. This environment contains test data, and searches are not billable. To access this environment, a QA PACER account is required. This account is separate from any other PACER accounts and can only be used in the QA environment. Register for this type of account at: <https://qa-pacer.uscourts.gov>. For a Production account, register at <https://pacer.uscourts.gov>.

The table below shows the URLs used throughout the document for the QA or Production environments. You should substitute the appropriate URL for the selected environment.

| URL Purpose          | URL Name          | QA URL  | Production URL  |
|----------------------|-------------------|---|---|
| Account registration | registrationurl   | <a href="https://qa-pacer.uscourts.gov">qa-pacer.uscourts.gov</a> | <a href="https://pacer.uscourts.gov">pacer.uscourts.gov</a>             |
| Authentication       | authenticationurl | <a href="https://qa-login.uscourts.gov">qa-login.uscourts.gov</a> | <a href="https://pacer.login.uscourts.gov">pacer.login.uscourts.gov</a> |

### Authentication Service

|                        |   |
|------------------------|---|
| <b>Description</b>     | Authenticate using your PACER username and password. Get the authentication token required by all court applications.   |
| <b>Service</b>         | /services/cso-auth  |
| <b>Endpoint</b>        | To authenticate use the authenticationurl<br><a href="https://{authenticationurl}/services/cso-auth">https://{authenticationurl}/services/cso-auth</a>  |
| <b>Method</b>          | POST  |
| <b>Request headers</b> | Use the request header to specify the format of the request and the response. The Content-type header indicates the format of the request and the Accept header indicates the format of the response.<br>For JSON formatting, use:<br>application/json<br>For XML formatting, use:<br>application/xml |
| <b>Request body</b>    | Use the request body to pass in the PACER username and password for authentication.<br><br>An optional client code can also be included in the request.   |

|                 |   |
|-----------------|---|
|                 | If you are a filer, the request body must also include the redaction flag, with a value of 1. <sup>1</sup>  |
| <b>Response</b> | <p>The authentication API returns a JSON or XML object with three elements: loginResult, nextGenCSO, and errorDescription. If the login is successful, the nextGenCSO will have a 128-byte string of characters.</p> <p>A successful login does not guarantee court search privileges but will allow you to continue to log in and perform other activities (e.g., e-file, request filing privileges).</p> <p>Review the error description for any additional information regarding the authentication attempt.</p> <p>The nextGenCSO token remains valid for an extended period; therefore, it should be used for all subsequent calls while it is valid—until you call the logout service or until you reach the maximum valid account login time. NextGenCSO tokens will be periodically re-issued and should be updated then.</p> <p>If authentication is successful, the nextGenCSO authentication token should be set as a cookie. You should include the nextGenCSO token as a cookie in the header of each request to court systems.<sup>2</sup></p> <p><b>NOTE:</b> If you access legacy (CurrentGen) courts, the nextGenCSO authentication token should also be set as a PacerSession cookie. Once all courts have converted to NextGen, this will be deprecated.</p> <p>If the optional client code information is included, the PacerClientCode cookie should be set and passed in the header of each request to court systems.<sup>3</sup></p> |

**Authentication Examples**

You can retrieve authentication tokens through this service with your PACER username and password. Authentication allows you to supply an optional client code. The client code is only used for court search purposes and is not required for successful authentication. NOTE: If your account requires a client code and you do not enter one, you can successfully log in to e-file and request filing privileges, but you will not be able to search.

Authentication calls require filers to include the redaction flag. This flag indicates the filer complies with filing redaction rules.

---

<sup>1</sup> All filers must redact the following: Social Security or taxpayer identification numbers; dates of birth; names of minor children; financial account numbers; and in criminal cases, home addresses in compliance with Fed. R. App. P. 25(a)(5), Fed. R. Civ. P. 5.2, Fed. R. Crim. P. 49.1, Fed. R. Bankr. P. 9037. This requirement applies to all documents, including attachments.

<sup>2</sup> Please see the PACER Case Locator (PCL) API for specifics on setting header values instead of cookies.

<sup>3</sup> See note 2.

## JSON Request

POST: <https://{authenticationurl}/services/cso-auth>

Request header:

```
Content-type: application/json
Accept: application/json
```

Request body:

```
{
  "loginId": "yourpacerusername",
  "password": "yourpacerpassword",
  "clientCode": "testclientcode",
  "redactFlag": "1"
}
```

Response body:

```
{
  "nextGenCSO": "your128characterauthenticationtokentobeuseduntilexpirationyour128characterauthenticationtokentobeuseduntilexpirationyour128chara",
  "loginResult": "0",
  "errorDescription": ""
}
```

## XML Request

POST: <https://{authenticationurl}/services/cso-auth>

Request header:

```
Content-type: application/xml
Accept: application/xml
```

Request body:

```
<?xml version="1.0" encoding="UTF-8"?>
<CsoAuth>
  <loginId>pacerusername</loginId>
  <password>pacerpassword</password>
  <clientCode>pacerclientcode</clientCode>
  <redactFlag>1</redactFlag>
</CsoAuth>
```

### Response body:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CsoAuth>
  <nextGenCSO>your128characterauthenticationtokentobeuseduntilexpirationyour128ch
aracterauthenticationtokentobeuseduntilexpirationyour128chara</nextGenCSO>
  <loginResult>0</loginResult>
  <errorDescription></errorDescription>
</CsoAuth>
```

### cURL

#### Request:

```
curl --location --request POST 'https://{authenticationurl}/services/cso-auth' \
--header 'Content-Type: application/json' \
--data '{
  "loginId": "yourpacername",
  "password": "yourpacerpassword"
}'
```

### Response body:

```
{
  "nextGenCSO": "your128characterauthenticationtokentobeuseduntilexpirationyour128cha
racterauthenticationtokentobeuseduntilexpirationyour128chara",
  "loginResult": "0",
  "errorDescription": ""
}
```

## Logout Service

|                        |  |
|------------------------|--|
| <b>Description</b>     | Log out using your PACER authentication token required by all court applications.  |
| <b>Service</b>         | /services/cso-logout   |
| <b>Endpoint</b>        | To log out, use the authenticationurl<br><a href="https://{authenticationurl}/services/cso-logout">https://{authenticationurl}/services/cso-logout</a>   |
| <b>Method</b>          | POST   |
| <b>Request headers</b> | Use the request header to specify the format of the request and the response. TheContent-Type header indicates the format of the request, and the Accept header indicates the format of the response.<br>For JSON formatting, use:<br>application/json<br>For XML formatting, use:<br>application/xml  |
| <b>Request body</b>    | Use the request body to pass in the PACER authentication token for logout.   |
| <b>Response</b>        | The authentication API returns a JSON or XML object with three elements: loginResult, nextGenCSO, and errorDescription. If the logout is successful, the nextGenCSO has successfully been invalidated.<br><br>Review the error description for any additional information regarding the logout attempt.<br><br>A successful logout means the authentication token is no longer valid for searching or e-filing privileges. |

## Logout Examples

You can invalidate an authentication token through this service. Pass in the nextGenCSO authentication token you want to invalidate. Any subsequent search requests will not recognize this token. The user must authenticate again to get another valid token.

### JSON Request

POST: <https://{authenticationurl}/services/cso-logout>

Request header:

```
Content-type: application/json
Accept: application/json
```

Request body:

```
{
  "nextGenCSO": "yourauthenticationtoken",
}
```

Response body:

```
{
  "loginResult": "0",
  "errorDescription": ""
}
```

## XML Request

POST: <https://{authenticationurl}/services/cso-logout>

Request header:

```
Content-type: application/xml
Accept: application/xml
```

Request body:

```
<?xml version="1.0" encoding="UTF-8"?>
<CsoAuth>
  <nextGenCSO>
    your128characterauthenticationtokentobeuseduntilexpirationyour128characterauthentic
    ationtokentobeuseduntilexpirationyour128chara
  </nextGenCSO>
</CsoAuth>
```

Response body:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CsoAuth>
  <nextGenCSO>your128characterauthenticationtokentobeuseduntilexpirationyour128ch
  aracterauthenticationtokentobeuseduntilexpirationyour128chara</nextGenCSO>
  <loginResult>0</loginResult>
  <errorDescription></errorDescription>
</CsoAuth>
```

## cURL

Request:

```
curl --location --request POST 'https://{authenticationurl}/services/cso-logout' \
--header 'Content-Type: application/json' \
--data '{
  "nextGenCSO": "
  your128characterauthenticationtokentobeuseduntilexpirationyour128characterauthenti
  cationtokentobeuseduntilexpirationyour128chara "
}'
```

Response body:

```
{
  "loginResult": "0",
  "errorDescription": ""
}
```



## Java Examples

The following is an example of calling the PACER authentication service using the Java programming language.

```
try {
    URL url = new URL("https://{authenticationurl}/services/cso-auth");
    HttpURLConnection conn = (HttpURLConnection) url.openConnection();
    conn.setDoOutput(true);
    conn.setRequestMethod("POST");
    conn.setRequestProperty("Content-Type", "application/json");
    conn.setRequestProperty("Accept", "application/xml"); // can also return json

    String authJson = "{\"loginId\": \"yourpacername\", \"
        + \"password\": \"yourpacerpassword\" }";

    OutputStream os = conn.getOutputStream();
    os.write(authJson.getBytes());
    os.flush();

    BufferedReader reader = new BufferedReader(new
        InputStreamReader((conn.getInputStream())));

    // responseBody will contain the response in either JSON or XML format
    String responseLine;
    String responseBody = "";
    while ((responseLine = reader.readLine()) != null) {
        responseBody += responseLine;
    }
}
catch (IOException e) {
    e.printStackTrace();
    System.exit(-1);
}
```

Example: Authenticating a user via the court authentication API using Java

The following is an example of calling the logout service using the Java programming language.

```

try {
    URL url = new URL("https://{authenticationurl}/services/cso-logout");
    HttpURLConnection conn = (HttpURLConnection) url.openConnection();
    conn.setDoOutput(true);
    conn.setRequestMethod("POST");
    conn.setRequestProperty("Content-Type", "application/json");
    conn.setRequestProperty("Accept", "application/xml"); // can also return json

    String authJson = "{\"nextGenCSO\": \""
your128characterauthenticationtokentobeuseduntilexpirationyour128characterauthen
ticationtokentobeuseduntilexpirationyour128chara\" }";

    OutputStream os = conn.getOutputStream();
    os.write(authJson.getBytes());
    os.flush();

    BufferedReader reader = new BufferedReader(new
        InputStreamReader((conn.getInputStream())));

    // responseBody will contain the response in either JSON or XML format
    String responseLine;
    String responseBody = "";
    while ((responseLine = reader.readLine()) != null) {
        responseBody += responseLine;
    }
}
catch (IOException e) {
    e.printStackTrace();
    System.exit(-1);
}

```

Example: Logging out via the court authentication API using Java

## Common Error Message Examples

The following shows sample error messages when calling the authentication API.

Request header:

```
Content-type: application/json
Accept: application/json
```

Request body:

```
{
  "loginId": "yourpacername",
  "password": "yourpacerpassword"
}
```

Response body:

This example shows the error description if your account requires a client code for searching, but one was not provided.

```
{
  "nextGenCSO": "your12characterauthenticationtokentobeuseduntilexpirationyour128characterauthenticationtokentobeuseduntilexpirationyour128chara",
  "loginResult": "0",
  "errorDescription": "A required Client Code was not entered. You may continue to log in and perform other activities (e.g., e-file, request filing privileges), but you will not have PACER search privileges."
}
```

Error message solution:

Add a client code to the request body.

```
{
  "loginId": "yourpacername",
  "password": "yourpacerpassword",
  "clientId": "yourclientId"
}
```

Request header:

```
Content-type: application/json
Accept: application/json
```

Request body:

```
{
  "loginId": "yourpacername",
  "password": "yourpacerpassword"
}
```

Response body:

This example shows the error description you receive when your account is disabled.

```
{
  "nextGenCSO": "your128characterauthenticationtokentobeuseduntilexpirationyour128characterauthenticationtokentobeuseduntilexpirationyour128chara",
  "loginResult": "0",
  "errorDescription": "Although you have a PACER account, your current account has been disabled. If you have any questions or for further details on how to activate your search privileges contact the PACER Service Center at (800) 676-6856 between the hours of 8 AM and 6 PM CT Monday through Friday or by email pacer@psc.uscourts.gov. You may continue to log in and perform other activities (e.g., e-file, request filing privileges), but you will not have PACER search privileges."
}
```

Error message solution:

Contact the PACER Service Center to resolve issues with the court.

Request header:

```
Content-type: application/json
Accept: application/json
```

Request body:

```
{
  "loginId": "yourpacername",
  "password": "yourpacerpassword"
}
```

Response body:

The following is an example of a registered filer not including the redaction flag in the request.

```
{
  "nextGenCSO": "",
  "loginResult": "1",
  "errorDescription": "All filers must redact: Social Security or taxpayer identification numbers; dates of birth; names of minor children; financial account numbers; and in criminal cases, home addresses in compliance with Fed. R. App. P. 25(a)(5), Fed. R. Civ. P. 5.2, Fed. R. Crim. P. 49.1, Fed. R. Bankr. P. 9037. This requirement applies to all documents, including attachments. Please verify that you have read and will comply with the redaction rules."
}
```

Error message solution:

Registered filers must include the redaction flag.

```
{
  "loginId": "yourpacername",
  "password": "yourpacerpassword",
  "redactFlag": "1"
}
```