PACER Authentication
Application Programming Interface (API)
User Guide

May 2025

Version History

Version Number	Date	Update	Page Number
v.1	12/2021	Initial API document	
v.2	04/2025	• Added note for multifactor authentication (MFA) enrollment • Added one-time passcode to authentication service table (for MFA) • Added one-time passcode to authentication examples (for MFA) • Added one-time passcode to Java example (for MFA) • Added one-time passcode generation example (for MFA) • Added one-time passcode to error message example (for MFA)	5 5 6, 7 10 12 14
v.3	5/2025	• Updated the MFA configuration example code and MMA OTP secret key image	13

Contents

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, password, and an optional one-time passcode[1], and returns an authentication token.

> **NOTE:** If you are required or wish to enroll in multifactor authentication (MFA)[2], please see the Appendix section of the PACER User Manual for complete instructions and more detailed information.

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 for searching, 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	qa-pacer.uscourts.gov	pacer.uscourts.gov
Authentication	authenticationurl	qa-login.uscourts.gov	pacer.login.uscourts.gov

Authentication Service

Description	Authenticate using your PACER username and password. Get the authentication token required by all court applications.
Service	/services/cso-auth
Endpoint	To authenticate, use the authenticationurl https://{authenticationurl}/services/cso-auth
Method	POST
Request headers	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. For JSON formatting, use: application/json For XML formatting, use: application/xml
Request body	Use the request body to pass in the PACER username and password for authentication.

[1] This one-time passcode is required if the account is enrolled in multifactor authentication (MFA).

[2] MFA provides an added layer of security to your account by requiring additional verification to log in. Users with filing and all other types of CM/ECF-level access are required to enroll in MFA, while users with PACER-only (case search only) access have the option to enroll.

	An optionally required one-time passcode may be included in the request. An optional client code may 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.[3]
Response	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. 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). Review the error description for any additional information regarding the authentication attempt. 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. 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.[4] 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.[5]

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.

[3] 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.

[4] Please see the PACER Case Locator (PCL) API for specifics on setting header values instead of cookies.

[5] See note 4.

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",
  "otpCode": "youronetimepasscode",
  "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>
       <otpCode>youronetimepasscode</otpCode>
       <redactFlag>1</redactFlag>
</CsoAuth>
```

Response body:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<CsoAuth>
    <nextGenCSO>your128characterauthenticationtokentobeuseduntilexpirationyour128characterauthenticationtokentobeuseduntilexpirationyour128chara</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":"yourpacerusername",
  "password":"yourpacerpassword",
  "otpCode":"youronetimepasscode"
}'
```

Response body:

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

Logout Service

Description	Log out using your PACER authentication token required by all court applications.
Service	/services/cso-logout
Endpoint	To log out, use the authenticationurl https://{authenticationurl}/services/cso-logout
Method	POST
Request headers	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. For JSON formatting, use: application/json For XML formatting, use: application/xml
Request body	Use the request body to pass in the PACER authentication token for logout.
Response	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. Review the error description for any additional information regarding the logout attempt. 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. The **otpCode** in the following example is required if your account is configured to use MFA. The code would be retrieved from your authentication application[6].

```
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\": \"yourpacerusername\","
                   + "\"password\": \"yourpacerpassword\","
                   + "\"otpCode\": \"youronetimepasscode\"}";

   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.

[6] The Administrative Office of the U.S. Courts does not endorse specific authentication apps; however, some options include Authy, DUO Mobile, FreeOTP, Google Authenticator, and Microsoft Authenticator.

```
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);
}
```

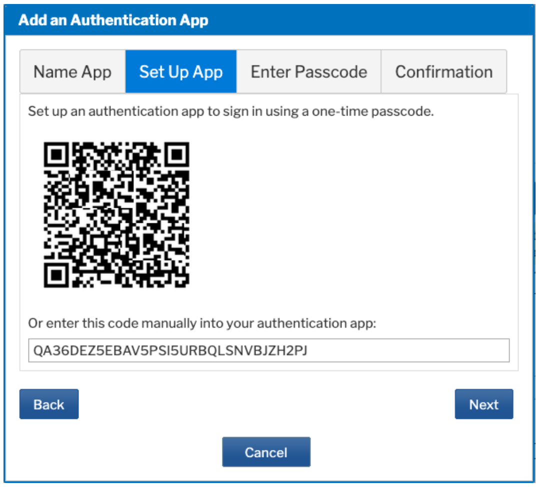
Example: Logging out via the court authentication API using Java

MFA Configuration

MFA is configured via the PACER Manage My Account application, and the one-time passcode (OTP) is retrieved from an authentication application. However, it is also possible to create the OTP programmatically using the secret key and your preferred TOTP library.

```
try {
        // 6-digit TOTP, 30-second time window
        TimeBasedOneTimePasswordGenerator totp =
                        new TimeBasedOneTimePasswordGenerator(Duration.ofSeconds(30));

        // Encoded secret from Manage My Account
        String base32SecretKey = "QA36DEZ5EBAV5PSI5URBQLSNVBJZH2PJ";

        Base32 base32 = Base32.builder().get();

        byte[] decodedKey = base32.decode(base32SecretKey);

        // Convert to SecretKey
        SecretKey secretKey = new SecretKeySpec(decodedKey, totp.getAlgorithm());

        // Generate TOTP using the current time
        Instant now = Instant.now();

        int otp = totp.generateOneTimePassword(secretKey, now);

        // Print the 6-digit code
        System.out.printf("One-Time Password: %06d\n", otp);
}
catch (Exception e) {
        e.printStackTrace();
        System.exit(-1);
}
```

Example: Generate a TOTP using Java



Set up an authentication app to sign in using a one-time passcode.

Or enter this code manually into your authentication app:

QA36DEZ5EBAV5PSI5URBQLSNVBJZH2PJ

OTP secret key[7]

[7] Note that the illustrated QR code and secret code string are not functional and are for exemplar purposes only.

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": "yourpacerusername",
  "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": "your128characterauthenticationtokentobeuseduntilexpirationyour12
8characterauthenticationtokentobeuseduntilexpirationyour128chara",
    "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": "yourpacerusername",
  "password": "yourpacerpassword",
  "clientCode": "yourclientcode"
}
```

Request header:

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

Request body:

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

Response body:

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

```
{
    "nextGenCSO": "your128characterauthenticationtokentobeuseduntilexpirationyour12
8characterauthenticationtokentobeuseduntilexpirationyour128chara",
    "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": "yourpacerusername",
  "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": "yourpacerusername",
  "password": "yourpacerpassword",
  "redactFlag": "1"
}
```

Response body:

The following is an example of a valid user entering an incorrect username, password, or one-time passcode, or not providing a required one-time passcode.

```
{
    "nextGenCSO": "",
    "loginResult": "13",
    "errorDescription": "Invalid username, password, or one-time passcode."
}
```

Error message solution:

The user must provide a proper username and password and/or use their authentication application to retrieve a one-time passcode or provide a valid backup code.