BankID Relying Party Guidelines

Version: 3.5

2020-10-26

1 Introduction

This document contains guidelines for Relying Parties (RP, Förlitande Part in Swedish) when using BankID in their own services.

Please check *https://www.bankid.com/utvecklare/rp-info* and verify that you have the latest version of this document.

1.1 Versions

Version	Date	Change
1.x		Historical versions
< 2.14		Historical versions. Please contact teknikinfo@bankid.com if you need this information.
2.14	2017-02-07	A new url must be used to access the service (appapi2.bankid.com). A new issuer certificate must be used as trusted issuer to be able to access appapi2.bankid.com appapi2.bankid.com accepts only TLS1.1 and TLS1.2. Fingerprint supported for Android. Fingerprint supported for signing by using requirementAlternatives. Editorial and minor improvements.
2.15	2017-03-21	The parameter autostarttoken for iOS was wrong.
2.16	2017-11-14	An upcoming change of IP addresses to access the service added.
3.0	2018-02-16	Version 5 of the service introduced. • JSON replaces XML/SOAP • A cancel method introduced Recommendation to start the iOS app using universal links. New IP addresses for the test environment. Description of how breaking changes are managed in the API. Editorial and minor improvements.
3.1	2018-06-13	New IP addresses for the production environment. A unique recommended user message for the alreadyInProgress error added. Changed the recommended user messages, "Login" replaced with "identification". Editorial and minor improvements.
3.2	2018-09-03	Support for QR codes. A clarification related to the default behaviour for the autoStartTokenRequired requirement.
3.2.1	2018-09-04	Minor editorial.
3.2.2	2019-04-04	Clarification about Apple's review process and minor editorial changes.
3.2.3	2020-05-07	Launch Android: preferred url, do not set package.
3.4	2020-06-08	Support for animated QR Removed historical paragraphs related to change of root CA. Editorial
3.5	2020-10-26	Possible to apply simple formatting of the text presented with the user by using parameter userVisibleDataFormat in method sign.

1.2 Terms and Definitions

Term	Description
BankID Security Application BankID app	The client software that needs to be installed in the end user's mobile device or personal computer (PC). The same term is used for PCs and mobile platforms. BankID app is the short form used in this document. In Swedish the client software installed on PCs is called "BankID säkerhetsprogram", "BankID-programmet" or "BankID-appen". In Swedish the client software installed on mobile platforms is called "BankID säkerhetsapp" or "BankID-appen"
RP	Relying Party that uses the BankID web service to provide authentication and signing functionality to the end user.

1.3 How it Works

To be able to use BankID's identification and signature features users must install the BankID app in a mobile device or PC. They also need to order a BankID from their bank. An RP uses the BankID identification or signature services via a web service API described in this document. The web service API can only be accessed by a RP that has a valid SSL client certificate. The RP certificate is obtained from the bank that the RP has purchased the BankID service from.

If the BankID app is installed on the same device as the RP service executes in, the BankID app can be launched automatically by the RP service. In this case, the users do not need to enter their personal number in the RP service. If, on the other hand, the RP service is used in a web browser on a PC and the users want to use a Mobile BankID the users will have to manually launch the BankID app on their mobile device. In this case, the users need to provide their personal number in the RP service.

1.4 Client Platforms

BankID is available for Windows, macOS, Android and iOS platforms. Smartcards are supported for Windows and macOS only. Detailed information on platform support can be found at https://support.bankid.com.

2 Use Cases

There are a number of use cases that can be implemented using the BankID solution. In this document, we describe the most common use cases to keep it simple and to give the reader a basic understanding of the solution.

- If the BankID app is installed on another device than the user uses to access the service, and the RP supports QR code, the users must manually start their BankID app and scan the QR code. In this case, the users do not need to provide their personal number.
- If the BankID app is installed on another device, and the RP does not support QR code, the users must provide their personal number and manually start the BankID app.
- If the BankID app is installed on the same device the user uses to access the service, the RP should help the user to start the BankID app automatically. In this case, the users do not need to provide their personal number.
- If the BankID app is installed on the same device, but the BankID app cannot be automatically started, the user must provide their personal number and manually start the BankID app on the same device. RP:s should consider this use case as a fallback in case the automatic start fails.

To make the user experience consistent the RP should use the recommended messages and error messages in *Recommended User Messages.*

The possibilities to restrict the type of BankID that can be used and how to define other requirements are described in *Requirement*.

2.1 Basic Use Cases

The following basic use cases exist:

A. The user accesses the service using a browser on a personal computer. Users should be asked if they want to login or sign using "BankID on this computer" or "Mobile BankID". Message RFA19 should be used.
    a. Users that select to use BankID on this computer does not need to enter their personal number and the RP must start the BankID app on the computer. See chapter *Launching*.
    b. Users that select Mobile BankID, and the RP does not support QR code, must enter their personal number start the BankID app manually on their mobile device.
    c. Users that select Mobile BankID, and the RP supports QR code, must start their BankID app manually on their mobile device and scan the QR code.

B. The user accesses the service using a browser on a mobile device. Users should be asked if they want to login or sign using "Mobile BankID on this device" or "Mobile BankID on another device". Message RFA20 should be used.
    a. Users that select to use this device do not need to enter their personal number and the RP must start the BankID app on the mobile device. See chapter *Launching*.
    b. Users that select to use another device, and the RP does not support QR code, must enter their personal number and start the BankID app manually on the other device.

   c. Users that select to use another device, and the RP supports QR code, must start their BankID app manually on the other device and scan the QR code.

C. The user accesses the service using a native app on a mobile device. In this case, the user most likely wants to use a BankID on the same device. The RP may however provide possibilities to use another device in this case as well.
   a. The users do not need to enter their personal number and the RP app launches BankID App programmatically (see *Launching the BankID App from Native App on Mobile Device*).
   b. Users that select to use another device, and the RP does not support QR code, must enter their personal number and start the BankID app manually on the other device.
   c. Users that select to use another device, and the RP supports QR code, must start their BankID app manually on the other device and scan the QR code.

In some cases, it may be impossible to start the BankID app automatically. The reason could be browsers blocking it or that the RP app does not have the capabilities to launch external URL:s. In this case, the users can always start the BankID app manually. In this case, the users need to enter their personal number.

2.2 Flow of Events

1. Users that select "another device" are asked to enter their personal number, if it's not already saved or known by the RP. As an alternative to entering personal number, the RP may support QR codes that the user scans.
2. The RP uses the auth or sign method of the web service API to initiate the order. The web service returns an `autoStartToken` and an `orderRef`. If the user selected "another device", RP should set condition `certificatePolicies` to "1.2.752.78.1.5" to restrict the order to Mobile BankID only.
3. If the user selected "same device" the RP tries to start the BankID app. The `autoStartToken` must be used in the start command if the personal number is not provided in the web service call, see *Launching*. Once the BankID app has finished execution, focus will be returned to the browser/app.
4. If the user selected "another device", the RP informs the user to start the BankID app manually.
5. If the RP supports QR code, the RP creates a QR code, which the user scans.
6. The RP service displays a progress indicator.
7. The auth or sign order is displayed in the BankID app. The RP name (as stated in the RP certificate) is displayed. The user enters personal security code or cancels the order.
8. The RP periodically uses the collect method of the web service API, until a final response is received and continuously updates the message displayed to the user. See *Recommended User Messages*.
9. RP removes the progress indicator.

2.3 Exceptions

1. The web service call in step 2 fails. The use case is cancelled and the RP shall instruct the user according to *Recommended User Messages*. The RP must not try to start the BankID app.
2. The collect call in step 8 fails. The use case is cancelled and RP shall instruct the user according to *Recommended User Messages*.
3. The automatic start in step 3 fails due to different reasons:
   - The user has not installed the BankID app
   - Erroneous start command
   - User did not allow the browser to launch the URL

   The web browser will inform the user that the URL cannot be opened. `hintCode` "startFailed" will be returned to the RP as response to the collect call in step 8 if the automatic start of the BankID app has not been completed within a certain time limit (30 seconds). The RP shall instruct the user according to *Recommended User Messages*.
4. The automatic start in step 3 is successful but the user has no BankID of correct type. The BankID app will display an error message. `hintCode` "started" will be returned to the RP as response to the collect call in step 8. RP shall instruct the user according to *Recommended User Messages*.
5. In step 4, the user fails to start the BankID app manually or no BankID of correct type exists in the started client. Different hint codes will be delivered to RP as response to the collect call in step 8. The RP shall instruct the user according to *Recommended User Messages*.
6. In step 8, the user fails to complete the operation within the time limit (3 minutes). `hintCode` "expiredTransaction" is returned from collect.
7. In step 8, the RP decides to cancel the order using the cancel method. The user is informed that the order was cancelled in the BankID app.

8. In step 5, the user fails to scan the QR code. `hintCode` "startFailed" will be returned to the RP as response to the collect call in step 8 if the QR code has not been scanned within a certain time limit (30 seconds). The RP shall instruct the user according to *Recommended User Messages*.

3 Launching

3.1 Launching the BankID App From a Browser

The URL syntax is:

```
bankid:///?autostarttoken=[TOKEN]&redirect=[RETURNURL]
```

The URL works on Android and iOS when the built-in web browser is used. The URL works on PCs with all commonly used browsers. Some differences exist on different platforms.

On Android 6 and later and on iOS the preferred URL syntax is:

```
https://app.bankid.com/?autostarttoken=[TOKEN]&redirect=[RETURNURL]
```

Note that on Android the app.bankid.com link works when using the Chrome browser.

Note that the `redirect` parameter must be last in the parameter list. The `autostarttoken` and `rpref` parameters are optional.

Note that the parameter names must be lower case.

Note that if the BankID app is started but no matching web service call to auth or sign has been done, an error message will be displayed in the app.

3.1.1 Behaviour in Different Browsers

3.1.1.1 Internet Explorer

Internet Explorer manipulates the URL in the redirect parameter. In this specification, we state that the RETURNURL must be URL encoded. However, Internet Explorer decodes the content prior passing it to the BankID app. This is why it must be last in the list of parameters. In the same way, Internet Explorer may decode the content of the RETURNURL when the BankID app passes the return URL back to the browser. If the RP includes session information that is affected by URL encoders/decoders, problems may occur. It is recommended to use only URL encoding safe characters in the parameters.

3.1.2 Parameters in the Start URL

Parameter	Description
`autostarttoken`	Optional. Holds the `autoStartToken` that was returned from the web service call. If the user personal number was not included in the web service call the `autostarttoken` must be provided. We strongly recommend to always use the `autostarttoken` when the URL is used to start the client. If it is not included and the user reloads the page or if the page erroneously repeats the start command, the user may get an error claiming that the BankID is missing. The likelihood of this to happen is reduced if `autostarttoken` is used. **Note** that the parameter names must be lower case.

Parameter	Description
Redirect	Required. The BankID app uses the parameter `redirect` to launch the RP web app after completed (including cancelled) auth or sign. The redirect URL must be UTF-8 and URL encoded and should match the web address the user is visiting when RP web app launches the BankID app. It may include parameters to be passed to the browser. For iOS the `redirect` must have a value. For all other platforms it may be empty ("`redirect`="), or set to "null" ("`redirect`=null"). If it is empty or null the BankID app will terminate without launching any URL and the calling application will be in focus. **The general recommendation is to use `redirect`=null when it is possible.** **Note for Windows and macOS** If `redirect` has a value the redirect parameter must be used together with `autostarttoken`. If `autostarttoken` is excluded, the content of `redirect` will be ignored and the behavior will be as if `redirect`=null. **Note for Android** If the user has several browsers installed on an Android device the user is sometimes presented with a question asking what browser to use. BankID recommends that `redirect`=null is used on Android. This ensures the user will return to the browser previously used. **Note for iOS** Redirect=null on iOS results in the RP web or app not being launched after completed auth or sign.
rpref	Optional. Relying Party Reference. Not supported in mobile devices. Any reference the RP wants to use. The value will be included in the resulting signature. A typical use case is to protect a file when it is transported from a client to a server (compute hashsum of the file content in the client, include the hashsum as `rpref`, compare it (server side) with a hashsum of the file content computed in the server). The value must be base64 encoded, URL encoded, and 8 – 255 bytes (after encoding). `rpref` must be used together with `autostarttoken`. If `autostarttoken` is excluded, the content of `rpref` will be ignored.

3.1.2.1 Examples

The RP wants the BankID app to open a browser with the following URL after finishing execution: https://demo.bankid.com/nyademobanken/CavaClientRedirReceiver.aspx?orderRef=bedea56d-7b46-47b1-890b-f787c650bc93&returnUrl=./CavaClientAuth.aspx&Environment=Kundtest. The `autostarttoken` is included. The start URL is:

```
bankid:///?autostarttoken=a4904c4c-3bb4-4e3f-8ac3-0e950e529e5f&
redirect=https%3a%2f%2fdemo.bankid.com%2fnyademobanken%2fCavaClientRedirRecei
ver.aspx%3forderRef%3dbedea56d-7b46-47b1-890b-
f787c650bc93%26returnUrl%3d.%2fCavaClientAuth.aspx%26Environment%3dKundtest
```

3.2 Launching the BankID App from Native App on Mobile Device

3.2.1 Android

```
Intent intent = new Intent();
intent.setAction(Intent.ACTION_VIEW);
intent.setData(Uri.parse("https://app.bankid.com/?autostarttoken=<INSERT
AUTOSTARTTOKEN HERE>&redirect=null ")) ;
intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
startActivity(intent);
```

A valid result is not guaranteed to be returned back from the BankID app to the RP app's Activity. The RP app should rely on the collect call to obtain the result of the auth or sign order. If the BankID app is not present on the device, an `android.content.ActivityNotFoundException` is thrown. RP must inform the user. Message RFA2 should be used.

On Android 5 the URI should use the bankid scheme instead of the https scheme.

3.2.2 iOS

```
let url = URL(string: "https://app.bankid.com/?autostarttoken=<INSERT
AUTOSTARTTOKEN HERE>&redirect=<INSERT YOUR LINK HERE>")
UIApplication.shared.open(url!, options: [.universalLinksOnly:true]) {
(success) in
```

```
    // handle success/failure
}
```

If the BankID app is not present on the device `false` is returned. RP must inform the user. Message RFA2 should be used.

The RP app must register a Universal Link or a custom URL scheme to make it possible for the BankID app to re-launch RP app.

The Apple App Store review process requires login information to a demo account for the app to be approved. This can be a demo account that does not require a BankID to login, or a way to configure the app to use the BankID test environment.

4 QR codes

The typical use case for QR codes is when the user uses "Mobile BankID on another device" (se use cases above), and there is a security concern that the user does not control both devices. When QR codes are used, the user does not need to provide their personal number.

4.1 Static QR

In September 2018, support for Static QR codes was introduced. The RP service generates a QR code based on the `autoStartToken`, presents the QR code to the user and asks the user to scan it using the BankID app. If this is successful, the BankID app will automatically proceed with the identification or signature operation.

```
Note: If personal number is included in the call to the service, RP must
consider setting the requirement tokenStartRequired to true. By this, the
system enforces that no other device than the one started using the QR code
or autostarttoken is used.
```

4.1.1 Details

The static QR code is generated using the following as input:

```
bankid:///?autostarttoken=[TOKEN]
```

- The `redirect` parameter must not be included.
- The `rpref` parameter must not be included.
- The url must not be URL-encoded
- The error correction level can be kept to a minimum. The code is supposed to be read from the screen.
- Colors in the QR code should be kept to a minimum, we recommend to use black.
- Use sufficient margins.
- Avoid to include other information or graphics in the QR code (in example layered logotypes)

If the user fails to scan the QR code within the time limit, `hintCode` "startFailed" will be returned to the RP as response to the collect call. The RP shall instruct the user according to *Recommended User Messages*.

Example

Input bankid:///?autostarttoken=46f6aa68-a520-49d8-9be7-f0726d038c26

Result:

4.2 Animated QR

In June 2020, support for animated QR-codes was introduced.

The RP Service generates a QR code based on qrStartToken and qrStartSecret, presents the QR code to the user and asks the user to scan it using the BankID app. The RP Service regularly update the QR code. If this is successful, the BankID app will automatically proceed with the identification or signature operation.

To use Animated QR version 5.1 of RP Interface must be used.

```
Note: If personal number is included in the call to the service, RP must
consider setting the requirement tokenStartRequired to true. By this, the
system enforces that no other device than the one started using the QR code
or autostarttoken is used.
```

4.2.1 Details

The QR code is generated by the RP every second using the pattern "bankid.qrStartToken.time.qrAuthCode" as input, where:

- bankid is a fixed prefix
- qrStartToken is from the auth or sign response
- time is the number of seconds since the result from auth or sign was returned

qrAuthCode is computed as $HMAC_{SHA256}$(qrStartSecret, time) where

- time is the number of seconds since the response from auth or sign was returned
- qrStartSecret is from the auth or sign response.

```
Note: The qrStartSecret must not be sent to the client, it is meant to be a
secret shared only between the BankID Service and the RP service.
```

When generating the QR code the following should be considered:

- The error correction level can be kept to a minimum. The code is supposed to be read from the screen.
- Colors in the QR code should be kept to a minimum, we recommend to use black.
- Use sufficient margins.
- Avoid to include other information or graphics in the QR code (in example layered logotypes)

4.2.1.1 Example

In the following example 67df3917-fa0d-44e5-b327-edcc928297f8 is used as qrStartToken and d28db9a7-4cde-429e-a983-359be676944c as qrStartSecret.

Time	QR Data	QR
t=0	bankid.67df3917-fa0d-44e5-b327-edcc928297f8.0.dc69358e712458a66a7525beef148ae8526b1c71610eff2c16cdffb4cdac9bf8	
t=1	bankid.67df3917-fa0d-44e5-b327-edcc928297f8.1.949d559bf23403952a94d103e67743126381eda00f0b3cbddbf7c96b1adcbce2	
t=2	bankid.67df3917-fa0d-44e5-b327-edcc928297f8.2.a9e5ec59cb4eee4ef4117150abc58fad7a85439a6a96ccbecc3668b41795b3f3	

4.2.1.2 Exceptions

Event	Result	Resolution
The QR code is completely irrelevant	The client displays an error message indicating the QR is irrelevant.	The user can only scan relevant QR codes.
The user fails to scan the QR code within the time limit for the login or sign operation	The client displays an error message indicating the operation cannot be completed. The hintCode “startFailed” will be returned to the RP as response to the collect call.	The RP must instruct the user according to *Recommended User Messages*.
The QR code is too old	The client displays an error message indicating the operation cannot be completed. The hintCode “startFailed” will be returned to the RP as response to the collect call.	The RP must instruct the user according to *Recommended User Messages*. The RP must init a new login or sign request. The user must try to login or sign again. If this error occurs frequently, the RP should consider to adjust their implementation of how the QR codes are computed. Especially check the timing.
The QR code is too fresh	The client displays an error message indicating the operation cannot be completed. The hintCode “startFailed” will be returned to the RP as response to the collect call.	The RP must instruct the user according to *Recommended User Messages*. The RP must init a new login or sign request. The user must try to login or sign again. If this error occurs frequently, the RP should consider to adjust their implementation of how the QR codes are computed. Especially check the timing.

4.2.1.3 Sample code

The sample code is included as a detailed description of how to compute the QR:s.

Python

```python
import hashlib
import hmac
import time

qr_start_token = rp_response["qrStartToken"]
# "67df3917-fa0d-44e5-b327-edcc928297f8"

qr_start_secret = rp_response["qrStartSecret"]
# "d28db9a7-4cde-429e-a983-359be676944c"

order_time = time.time()
# (The time in seconds when the response from the BankID service was delivered)

qr_time = str(int(time.time() - order_time))
# ("0" or another string with a higher number depending on order_time and current time)

qr_auth_code = hmac.new(qr_start_secret, qr_time, hashlib.sha256).hexdigest()
# "dc69358e712458a66a7525beef148ae8526b1c71610eff2c16cdffb4cdac9bf8" (qr_time="0")
# "949d559bf23403952a94d103e67743126381eda00f0b3cbddbf7c96b1adcbce2" (qr_time="1")
# "a9e5ec59cb4eee4ef4117150abc58fad7a85439a6a96ccbecc3668b41795b3f3" (qr_time="2")
# (64 chars hex)

qr_data = str.join(".", "bankid", qr_start_token, qr_time, qr_auth_code)
# "bankid.67df3917-fa0d-44e5-b327-edcc928297f8.0.dc69358e712458a66a7525beef148ae8526b1c71610eff2c16cdffb4cdac9bf8" (qr_time="0")
# "bankid.67df3917-fa0d-44e5-b327-edcc928297f8.1.949d559bf23403952a94d103e67743126381eda00f0b3cbddbf7c96b1adcbce2" (qr_time="1")
# "bankid.67df3917-fa0d-44e5-b327-edcc928297f8.2.a9e5ec59cb4eee4ef4117150abc58fad7a85439a6a96ccbecc3668b41795b3f3" (qr_time="2")
```

Java

```java
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.math.BigInteger;
import java.time.temporal.ChronoUnit;

String qrStartToken = rpResponse.getQrStartToken();
// "67df3917-fa0d-44e5-b327-edcc928297f8"

String qrStartSecret = rpResponse.getQrStartSecret();
// "d28db9a7-4cde-429e-a983-359be676944c"

String qrTime = Long.toString(orderTime.until(Instant.now(), ChronoUnit.SECONDS));
// ("0" or another string with a higher number depending on order time and current time)

Mac mac = Mac.getInstance("HmacSHA256");
mac.init(new SecretKeySpec(qrStartSecret.getBytes(StandardCharsets.US_ASCII), "HmacSHA256"));
mac.update(qrTime.getBytes(StandardCharsets.US_ASCII));

String qrAuthCode = String.format("%064x", new BigInteger(1, mac.doFinal()));
// "dc69358e712458a66a7525beef148ae8526b1c71610eff2c16cdffb4cdac9bf8" (qr_time="0")
// "949d559bf23403952a94d103e67743126381eda00f0b3cbddbf7c96b1adcbce2" (qr_time="1")
// "a9e5ec59cb4eee4ef4117150abc58fad7a85439a6a96ccbecc3668b41795b3f3" (qr_time="2")
// (64 chars hex)

String qrData = String.join(".", "bankid", qrStartToken, qrTime, qrAuthCode)
// "bankid.67df3917-fa0d-44e5-b327-edcc928297f8.0.dc69358e712458a66a7525beef148ae8526b1c71610eff2c16cdffb4cdac9bf8" (qr_time="0")
// "bankid.67df3917-fa0d-44e5-b327-edcc928297f8.1.949d559bf23403952a94d103e67743126381eda00f0b3cbddbf7c96b1adcbce2" (qr_time="1")
// "bankid.67df3917-fa0d-44e5-b327-edcc928297f8.2.a9e5ec59cb4eee4ef4117150abc58fad7a85439a6a96ccbecc3668b41795b3f3" (qr_time="2")
```

5 Technical Requirements

Short Name	Requirement
RFT1	When the BankID app is launched with a URL the content of the parameter `redirect` must be UTF-8 and URL encoded.
RFT2	When the BankID app is launched with a URL the URL must not exceed 2000 characters.
RFT3	When the BankID app is launched with a URL the `redirect` URL should use HTTPS.
RFT4	The personal number in the RP web service API must be 12 characters (YYYYMMDDNNNN).
RFT5	When collect returns completed RP shall read and store the values of `signature`, `userInfo` and `ocspResponse`. RP does not need to verify the signature. BankID verifies the signature.

Short Name	Requirement
RFT6	collect should be called every two seconds and must not be called more frequent than once per second.
RFT7	RP should display a progress indicator in its web app when waiting for the complete response from collect.
RFT8	RP must contact BankID's web service API from RP's backend server. RP must NOT contact BankID's web service API from RP's client app.
RFT9	RP should always use the latest version of the web service API, see *Information Regarding the Web Service API*.
RFT10	If the user selects to use Mobile BankID only, the certificatePolicies condition must be set to 1.2.752.78.1.5
RFT11	RP must use the **issuer** of the server cert as trusted root. If the server cert is used as trusted, the RP service will not be able to access the BankID server when the server cert is changed.

6 Recommended User Messages

Short Name	Swedish Text	English Text	Event, status, hintCode or errorCode
RFA1	Starta BankID-appen	Start your BankID app.	`status`=pending `hintCode`=outstandingTransaction `hintCode`=noClient
RFA2	Du har inte BankID-appen installerad. Kontakta din internetbank.	The BankID app is not installed. Please contact your internet bank.	The BankID app is not installed in the mobile device.
RFA3	Åtgärden avbruten. Försök igen.	Action cancelled. Please try again.	`errorCode`=cancelled
RFA4	En identifiering eller underskrift för det här personnumret är redan påbörjad. Försök igen.	An identification or signing for this personal number is already started. Please try again.	`errorCode`=alreadyInProgress
RFA5	Internt tekniskt fel. Försök igen.	Internal error. Please try again.	`errorCode`=requestTimeout `errorCode`=maintenance (repeatedly) `errorCode`=internalError
RFA6	Åtgärden avbruten.	Action cancelled.	`status`=failed `hintCode`=userCancel
RFA8	BankID-appen svarar inte. Kontrollera att den är startad och att du har internetanslutning. Om du inte har något giltigt BankID kan du hämta ett hos din Bank. Försök sedan igen.	The BankID app is not responding. Please check that the program is started and that you have internet access. If you don't have a valid BankID you can get one from your bank. Try again.	`status`=failed `hintCode`=expiredTransaction
RFA9	Skriv in din säkerhetskod i BankID-appen och välj Legitimera eller Skriv under.	Enter your security code in the BankID app and select Identify or Sign.	`status`=pending `hintCode`=userSign

Short Name	Swedish Text	English Text	Event, status, hintCode or errorCode
RFA13	Försöker starta BankID-appen.	Trying to start your BankID app.	`status`=pending `hintCode`=outstandingTransaction
RFA14 (A)	Söker efter BankID, det kan ta en liten stund… Om det har gått några sekunder och inget BankID har hittats har du sannolikt inget BankID som går att använda för den aktuella identifieringen/underskriften i den här datorn. Om du har ett BankID-kort, sätt in det i kortläsaren. Om du inte har något BankID kan du hämta ett hos din internetbank. Om du har ett BankID på en annan enhet kan du starta din BankID-app där.	Searching for BankID:s, it may take a little while… If a few seconds have passed and still no BankID has been found, you probably don't have a BankID which can be used for this identification/signing on this computer. If you have a BankID card, please insert it into your card reader. If you don't have a BankID you can order one from your internet bank. If you have a BankID on another device you can start the BankID app on that device.	`status`=pending `hintCode`=started The RP provided the personal number in the web service call (without using `tokenStartRequired`). The user accesses the service using a personal computer.
RFA14 (B)	Söker efter BankID, det kan ta en liten stund… Om det har gått några sekunder och inget BankID har hittats har du sannolikt inget BankID som går att använda för den aktuella identifieringen/underskriften i den här enheten. Om du inte har något BankID kan du hämta ett hos din internetbank. Om du har ett BankID på en annan enhet kan du starta din BankID-app där.	Searching for BankID:s, it may take a little while… If a few seconds have passed and still no BankID has been found, you probably don't have a BankID which can be used for this identification/signing on this device. If you don't have a BankID you can order one from your internet bank. If you have a BankID on another device you can start the BankID app on that device.	`status`=pending `hintCode`=started The RP provided the personal number in the web service call (without using `tokenStartRequired`). The user accesses the service using a mobile device.
RFA15 (A)	Söker efter BankID, det kan ta en liten stund… Om det har gått några sekunder och inget BankID har hittats har du sannolikt inget BankID som går att använda för den aktuella identifieringen/underskriften i den här datorn. Om du har ett BankID-kort, sätt in det i kortläsaren. Om du inte har något BankID kan du hämta ett hos din internetbank.	Searching for BankID:s, it may take a little while… If a few seconds have passed and still no BankID has been found, you probably don't have a BankID which can be used for this identification/signing on this computer. If you have a BankID card, please insert it into your card reader. If you don't have a BankID you can order one from your internet bank.	`status`=pending `hintCode`=started The RP did not provide the personal number in the web service call. The user accesses the service using a personal computer.

Short Name	Swedish Text	English Text	Event, status, hintCode or errorCode
RFA15 (B)	Söker efter BankID, det kan ta en liten stund… Om det har gått några sekunder och inget BankID har hittats har du sannolikt inget BankID som går att använda för den aktuella identifieringen/underskriften i den här enheten. Om du inte har något BankID kan du hämta ett hos din internetbank.	Searching for BankID:s, it may take a little while… If a few seconds have passed and still no BankID has been found, you probably don't have a BankID which can be used for this identification/signing on this device. If you don't have a BankID you can order one from your internet bank	`status`=pending `hintCode`=started The RP did not provide the personal number in the web service call. The user accesses the service using a mobile device.
RFA16	Det BankID du försöker använda är för gammalt eller spärrat. Använd ett annat BankID eller hämta ett nytt hos din internetbank.	The BankID you are trying to use is revoked or too old. Please use another BankID or order a new one from your internet bank.	`status`=failed `hintCode`=certificateErr
RFA17 (A)	BankID-appen verkar inte finnas i din dator eller telefon. Installera den och hämta ett BankID hos din internetbank. Installera appen från din appbutik eller https://install.bankid.com.	The BankID app couldn't be found on your computer or mobile device. Please install it and order a BankID from your internet bank. Install the app from your app store or https://install.bankid.com.	`status`=failed `hintCode`=startFailed RP does not use QR code
RFA17 (B)	Misslyckades att läsa av QR koden. Starta BankID-appen och läs av QR koden. Kontrollera att BankID-appen är uppdaterad. Om du inte har BankID-appen måste du installera den och hämta ett BankID hos din internetbank. Installera appen från din appbutik eller https://install.bankid.com.	Failed to scan the QR code. Start the BankID app and scan the QR code. Check that the BankID app is up to date. If you don't have the BankID app, you need to install it and order a BankID from your internet bank. Install the app from your app store or https://install.bankid.com.	`status`=failed `hintCode`=startFailed RP uses QR code
RFA18	Starta BankID-appen	Start the BankID app	The name of link or button used to start the BankID App
RFA19	Vill du identifiera dig eller skriva under med BankID på den här datorn eller med ett Mobilt BankID?	Would you like to identify yourself or sign with a BankID on this computer or with a Mobile BankID?	The user accesses the service using a browser on a personal computer.
RFA20	Vill du identifiera dig eller skriva under med ett BankID på den här enheten eller med ett BankID på en annan enhet?	Would you like to identify yourself or sign with a BankID on this device or with a BankID on another device?	The user accesses the service using a browser on a mobile device.
RFA21	Identifiering eller underskrift pågår.	Identification or signing in progress.	`status`=pending The `hintCode` is unknown to RP.

Short Name	Swedish Text	English Text	Event, status, hintCode or errorCode
RFA22	Okänt fel. Försök igen.	Unknown error. Please try again.	`status`=failed The `hintCode` is unknown to RP. An error occured. The `errorCode` is unknown to RP.

7 Production Environment

Description	Information
SSL certificate (RP certificate)	Provided by the bank that RP purchases the BankID service from. See section SSL Certificates below.
JSON Web Service URL	https://appapi2.bankid.com/rp/v5.1
Issuer of server certificate	See section SSL Certificates below. The server certificate is issued by the following CA. CN = BankID SSL Root CA v1 OU = Infrastructure CA O = Finansiell ID-Teknik BID AB Certificate: -----BEGIN CERTIFICATE----- MIIFvjCCA6agAwIBAgIITyTh/u1bExowDQYJKoZIhvcNAQENBQAwYjEkMCIGA1UE CgwbRmluYW5zaWVsbCBJRC1UZWtuaWsgQklEIEFCMRowGAYDVQQLDBFJbmZyYXN0 cnVjdHVyZSBDQTEeMBwGA1UEAwwVQmFua0lEIFNTTCBSb290IENBIHYxMB4XDTEx MTIwNzEyMzQwN1oXDTM0MTIzMTEyMzQwN1owYjEkMCIGA1UECgwbRmluYW5zaWVs bCBJRC1UZWtuaWsgQklEIEFCMRowGAYDVQQLDBFJbmZyYXN0cnVjdHVyZSBDQTEe MBwGA1UEAwwVQmFua0lEIFNTTCBSb290IENBIHYxMIICIjANBgkqhkiG9w0BAQEF AAOCAg8AMIICCgKCAgEAwVA4snZiSFI3r64LvYu4mOsI42A9aLKEQGq4IZo257iq vPH82SMvgBJgE52kCx7gQMmZ7iSm39CEA19hlILh8JEJNTyJNxMxVDN6cfJP1jMH JeTES1TmVbWUqGyLpyT8LCJhC9Vq4W3t/O1svGJNOUQIQL4eAHSvWTVoalxzomJh On97ENjXAt4BLb6sHfVBvmB5ReK0UfwpNACFM1RN8btEaDdWC4PfA72yzV3wK/cY 5h2k1RM1s19PjoxnpJqrmn4qZmP4tN/nk2d7c4FErJAP0pnNsll1+JfkdMfiPD35 +qcclpspzP2LpauQVyPbO21Nh+EPtr7+Iic2tkgz0g1kK0IL/foFrJ0Ievyr3Drm 2uRnA0esZ45GOmZhE22mycEX9l7w9jrdsKtqs7N/T46hil4xBiGblXkqKNG6TvAR k6XqOp3RtUvGGaKZnGllsgTvP38/nrSMlszNojrlbDnm16GGoRTQnwr8l+Yvbz/e v/e6wVFDjb52ZB0Z/KTfjXOl5cAJ7OCbODMWf8Na56OTllkrk5NyU/uGzJFUQSvG dLHUipJ/sTZCbqNSZUwboI0oQNO/Ygez2J6zgWXGpDWiN4LGLDmBhB3T8CMQu9J/ BcFvgjnUyhyim35kDpjVPC8nrSir5OkaYgGdYWdDuv1456lFNPNNQcdZdt5fcmMC AwEAAaN4MHYwHQYDVR0OBBYEFPgqsux5RtcrIhAVeuLBSgBuRDFVMA8GA1UdEwEB /wQFMAMBAf8wHwYDVR0jBBgwFoAU+Cqy7HlG1ysiEBV64sFKAG5EMVUwEwYDVR0g BAwwCjAIBgYqhXBOAQQwDgYDVR0PAQH/BAQDAgEGMA0GCSqGSIb3DQEBDQUAA4IC AQAJOjUOS2GJPNrrrqf539aN1/EbUj5ZVRjG4wzVtX5yVqPGcRZjUQlNTcfOpwPo czKBnNX2OMF+Qm94bb+xXc/08AERqJJ3FPKu8oDNeK+Rv1X4nh95J4RHZcvl4AGh ECmGMyhyCea0qZBFBsBqQR7oC9afYOxsSovaPqX31QMLULWUYoBKWWHLVVIoHjAm GtAzMkLwe0/lrVyApr9iyXWhVr+qYGmFGw1+rwmvDmmSLWNWawYgH4NYxTf8z5hB iDOdAgilvyiAF8Yl0kCKUB2fAPhRNYlEcN+UP/KL24h/pB+hZ9mvR0tM6nW3HVZa DrvRz4VihZ8vRi3fYnOAkNE6kZdrrdO7LdBc9yYkfQdTcy0N+Aw7q4TkQ8npomrV mTKaPhtGhA7VICyRNBVcvyoxr+CY7aRQyHn/C7n/jRsQYxs7uc+msq6jRS4HPK8o lnF9usWZX6KY+8mweJiTE4uN4ZUUBUtt8WcXXDiK/bxEG2amjPcZ/b4LXwGCJb+a NWP4+iY6kBKrMANs01pLvtVjUS9RtRrY3cNEOhmKhO0qJSDXhsTcVtpbDr37UTSq QVw83dReiARPwGdURmmkaheH6z4k6qEUSXuFch0w53UAc+1aBXR1bgyFqMdy7Yxi b2AYu7wnrHioDWqP6DTkUSUeMB/zqWPM/qx6QNNOcaOcjA== -----END CERTIFICATE-----
Network information	The BankID app for Android, iOS, macOS and Windows in production connects to the BankID server on the IP address 185.198.4.18 using port 443 and address 185.198.4.19 using port 80. The BankID app for macOS and Windows also connects to 5.150.251.26 using port 80.

8 Test Environment

BankID provides a test environment for an RP to use when developing and testing its service. To be able to use the test environment the RP will need:

1. An SSL certificate (RP certificate) for identification with the BankID web service API.
2. The URL for BankID's web service API.
3. Trust the issuer of the SSL certificate.
4. A test version of the BankID app
5. A BankID for test.

Description	Information
SSL certificate (RP certificate for test)	Available at https://www.bankid.com/bankid-i-dina-tjanster/rp-info. See section SSL Certificates below.
Passphrase for above certificate	qwerty123
JSON Web Service URL	https://appapi2.test.bankid.com/rp/v5.1
Issuer of server certificate	See section SSL Certificates below. CN = Test BankID SSL Root CA v1 Test OU = Infrastructure CA O = Finansiell ID-Teknik BID AB Certificate: -----BEGIN CERTIFICATE----- MIIF0DCCA7igAwIBAgIIhYaxu4khgAwDQYJKoZIhvcNAQENBQAwbDEkMCIGA1UE CgwbRmluYW5zaWVsbCBJRC1UZWtuaWsgQklEIEFCMRowGAYDVQQLDBFJbmZyYXN0 cnVjdHVyZSBDQTEoMCYGA1UEAwwfVGVzdCBCYW5rSUQgU1NMIFJvb3QgQ0EgdjEg VGVzdDAeFw0xNDExMjExMjM5MzFaFw0zNDEyMzExMjM5MzFaMGwxJDAiBgNVBAoM G0ZpbmFuc2llbGwgSUQtVGVrbmlrIEJJRCBBQjEaMBgGA1UECwwRSW5mcmFzdHJ1 Y3R1cmUgQ0ExKDAmBgNVBAMMH1Rlc3QgQmFua0lEIFNTTCBSb290IENBIHYxIFRl c3QwggIiMA0GCSqGSIb3DQEBAQUAA4ICDwAwggIKAoICAQCAKWsJc/kV/0434d+S qn19mIr85RZ/PgRFaUplSrnhuzAmaXihPLCEsd3Mh/YErygcxhQ/MAzi5OZ/anfu WSCwceRlQINtvlRPdMoeZtu29FsntK1Z5r2SYNdFwbRFb8WN9FsU0KvC5zVnuDMg s5dUZwTmdzX5ZdLP7pdgB3zhTnra5ORtkiWiUxJVev9keRgAo00ZHIRJ+xTfiSPd Jc314maigVRQZdGKSyQcQMTWi1YLwd2zwOacNxleYf8xqKgkZsmkrc4Dp2mR5Pkr nnKB6A7sAOSNatua7M86EgcGi9AaEyaRMkYJImbBfzaNlaBPyMSvwmBZzp2xKc9O D3U06ogV6CJjJL7hSuVc5x/2H04d+2I+DKwep6YBoVL9L81gRYRycqg+w+cTZ1TF /s6NC5YRKSeOCrLw3ombhjyyuPl8T/h9cpXt6m3y2xIVLYVzeDhaql3hdi6IpRh6 rwkMhJ/XmOpbDinXb1fWdFOyQwqsXQWOEwKBYIkM6cPnuid7qwaxfP22hDgAolGM LY7TPKUPRwV+a5Y3VPl7h0YSK7lDyckTJdtBqI6d4PWQLnHakUgRQy69nZhGRtUt PMSJ7l4Qtt3B6AwDq+SJTggwtJQHeid0jPki6pouenhPQ6dZT532x16XD+WIcD2f //XzzOueS29KB7lt/wH5K6EuxwIDAQABo3YwdDAdBgNVHQ4EFgQUDY6XJ/FIRFX3 dB4Wep3RVM84RXowDwYDVR0TAQH/BAUwAwEB/zAfBgNVHSMEGDAWgBQNjpcn8UhE Vfd0HhZ6ndFUzzhFejARBgNVHSAECjAIMAYGBCoDBAUwDgYDVR0PAQH/BAQDAgEG MA0GCSqGSIb3DQEBDQUAA4ICAQA5s59/Olio4svHXiKu7sPQRvrf4GfGB7hUjBGk YW2YOHTYnHavSqlBASHc8gGGwuc7v7+H+vmOfSLZfGDqxnBqeJx1H5E0YqEXtNqW G1JusIFa9xWypcONjg9v7IMnxxQzLYws4YwgPychpMzWY6B5hZsjUyKgB+1igxnf uaBueLPw3ZaJhcCL8gz6SdCKmQpX4VaAadS0vdMrBOmd826H+aDGZek1vMjuH11F fJoXY2jyDnlol7Z4BfHc011toWNMxojI7w+U4KKCbSxpWFVYITZ8WlYHcj+b2A1+ dFQZFzQN+Y1Wx3VIUqSks6P7F5aF/l4RBngy08zkP7iLA/C7rm61xWxTmpj3p6SG fUBsrsBvBgfJQHD/Mx8U3iQCa0Vj1XPogE/PXQQq2vyWiAP662hD6og1/om3l1PJ TBUyYXxqJO75ux8IWblUwAjsmTlF/Pcj8QbcMPXLMTgNQAgarV6guchjivYqb6Zr hq+Nh3JrF0HYQuMgExQ6VX8T56saOEtmlp6LSQi4HvKatCNfWUJGoYeT5SrcJ6sn By7XLMhQUCOXcBwKbNvX6aP79VA3yeJHZO7XParX7V9BB+jtf4tz/usmAT/+qXtH CCv9Xf4lv8jgdOnFfXbXuT8I4gz8uq8ElBlpbJntO6p/NY5a08E6C7FWVR+WJ5vZ OP2HsA== -----END CERTIFICATE-----
Test version of BankID app for mobile devices and PCs	See "How to get a test BankID" at https://www.bankid.com/bankid-i-dina-tjanster/rp-info
BankID for test	See "How to get a test BankID" at https://www.bankid.com/bankid-i-dina-tjanster/rp-info
Network information	The BankID app for Android, iOS, macOS and Windows for test connects to the BankID server on the IP address 185.198.6.16 using port 443 and address 185.198.6.14 using port 80. The BankID app for macOS and Windows also connects to 5.150.251.26 using port 80.

9 Information Regarding the Web Service API

9.1 SSL Certificates

The RP certificate must be installed/configured in your "key store". It does not need to be verified by your application and the issuer of the RP certificate is not needed. The RP certificate is verified by the BankID server when the channel is established. The BankID server will then present its server certificate to your application. The server certificate needs to be verified by your application. To make that verification possible the issuer of the server certificate needs to be installed/configured in your "trust store". Key stores and trust stores are managed differently depending on your environment and is not explained in this document.

Note that different certificates are used for production and test.

Note that the certificates may need to be converted to a different file format to be accepted by your environment.

Note that your application needs access to your key store and trust store and your application needs to use correct key store and trust store.

Note that line breaks may need to be removed from the issuer of the server certificate pasted from this document.

9.2 Versions

A new version of the web service API will be published on a new URL every time there is a breaking change in the API. RP should always use the latest version of the API. The general rule is that old versions will shut down 2 years after the release of the successor. As new functionality is introduced to the system the behavior of an existing version of the interface may change, e.g. existing faults may also be used in new situations. This document is written for version 5.1 (current version) of the interface.

V	URL	Changes	Release date	End of life
4	https://appapi.bankid.com/rp/v4	Mobile BankID, BankID on file, BankID on Card and Nordea e-leg merged to one solution.	January 2014	March 2019
4	https://appapi2.bankid.com/rp/v4	A new CA issues the server certificate. A new endpoint to access the service. Requires TLS1.1 or TLS1.2	March 2017	February 2020
5	https://appapi2.bankid.com/rp/v5	Http/JSON replaces SOAP/XML cancel introduced.	February 2018	April 2022
5.1	https://appapi2.bankid.com/rp/v5.1	Support for animated QR-codes. New return parameters to support animated QR codes. autoStartTokenRequired deprecated. tokenStartRequired introduced	April 2020	

9.2.1 Breaking Change

The following table describes the general principles for breaking changes. Security reasons may shorten the notice period.

Change	Breaking	Comment
Add optional in-parameter	NO	We may add additional optional in-parameters without prior notice.
Add required in-parameters	YES	We may add additional required in-parameters. This will be done using a new end-point and with a two year notice.
Remove any in-parameter	YES	We may remove support for in-parameters. This will be done using a new endpoint and with a two year notice.
Add return-parameter	NO	We may add additional return-parameters without prior notice. RP must consider this in their implementation. Implementations must **not** discard the complete response if it includes unknown parameters.
Remove any return-parameter	YES	We may remove return-parameters. This will be done using a new endpoint and with a two year notice.

Change	Breaking	Comment
Remove method	YES	We may remove methods. This will be done using a new endpoint and with a two year notice.
Add method	NO	We may add new methods without prior notice.
Change issuer of server certificate	YES	We may change issuer of the server certificate. This will be done using a new endpoint and with a two year notice.
Add new hintCodes	NO	We may add new `hintCode` without prior notice. RP must consider this in their implementation. If RP receives an "unknown" hint code a general message should be presented to the user.
Add new errorCodes	NO	We may add new `errorCode` without prior notice. RP must consider this in their implementation. If RP receives an "unknown" error code a general message should be presented to the user.

9.3 Test Environment

New versions and release candidates are used in the test environment prior to being taken into use in the production environment. Due to this, the content and functionality in the test environment and production environment may temporarily differ.

9.4 HTTP/1.1

The service only supports HTTP/1.1. HTTP/1.0 will not work.

9.5 TLS Versions

appapi2.bankid.com requires TLS1.1 or TLS1.2. We strongly recommend to use TLS1.2.

10 Support

10.1 Developer Support

Please study this guideline carefully before contacting us. Our experience is that all answers are provided in this document. Please also study the FAQ at https://www.bankid.com/bankid-i-dina-tjanster/rp-info. As a last resort, you may contact us using teknikinfo@bankid.com. In non-technical matters, please contact the bank through which you have purchased the BankID service.

10.2 End User Support

Short name	Requirement
RFS1	RP should inform the user what to do in case of lost or forgotten security code (contact the issuer).
RFS2	RP must provide support for its own service.
RFS3	When the user is having problems, the RP should redirect the user to https://test.bankid.com. Users that cannot successfully use their BankID at https://test.bankid.com should be redirected to the issuing bank in case of a BankID related problem and in case of network error to mobile phone carrier or the internet service provider. If the user can successfully identify and sign at https://test.bankid.com, the user should be redirected to the RP user support.

11 Recommended Terminology

Description	Recommended terminology in Swedish	Recommended terminology in English
Mobile BankID	Mobilt BankID	Mobile BankID
BankID Security Application for mobile devices	BankID-appen	The BankID app
BankID Security Application for PCs	BankID-appen or BankID-programmet	The BankID app

Description	Recommended terminology in Swedish	Recommended terminology in English
Security code, password, PIN	Säkerhetskod (för Mobilt BankID) Lösenord (för BankID på fil) PIN (för BankID på kort)	Security code (for Mobile BankID) Password (for BankID on File) PIN (for BankID on Card)
Sign	Skriva under	Sign
Signature	Underskrift	Signature
Identify	Legitimera sig	Identify
Identification/authentication	Legitimering	Identification

12 File Signing

Our recommendation is to use the sign method with the following notes:

1. Present the document to be signed to the user using your own application/website.
2. Compute a message digest of the binary representation of the document.
3. Compile an abstract of the content of the document.
4. Use method sign with `userVisibleData` set to the abstract and `userNonVisibleData` set to the message digest.

The benefits of using this method are that it is available for PCs and mobile devices, that there is no size limitation and that all types of documents can be signed.

13 Verifying Signatures

The signatures (including the certificates) returned from the service are already verified by the service.

Note: The Relying Party does not need to verify the signatures.

It is however possible for the Relying Party to verify them. To do that, the following data is needed:

- The `signature` returned from the service. A specification of the content is delivered to you on request.
- The certificate of the user and intermediate CA:s. These are included in the signature.
- The `ocspResponse` returned from the service.
- The self-signed root certificate. This is delivered to you on request.

The following principle is applicable:

1. Verify the `signature`.
2. Verify the certificates in the certificate chain up to the self-signed root. Note that certificates may have expired at the time of verification if it is later than the time of use.
3. Verify the status of the `ocspResponse` to be ok.
4. Verify the signature of the `ocspResponse`.
5. Verify the certificate of the `ocspResponse` signer and that it is issued by the same CA as the user certificate in question.
6. Verify the nonce included in the `ocspResponse` to be correct by matching it with a hash computed of the `signature`. See *completionData for Completed Orders.*

14 RP Interface Description

In RPv5.1, a JSON based format is used.

- HTTP1.1 is required
- All methods are accessed using HTTP POST to /rp/v5.1/<*method*>.
- HTTP header 'Content-Type' must be set to 'application/json'.
- The parameters including the leading and ending curly bracket is in the body.

14.1 /rp/v5.1/auth and /rp/v5.1/sign

Initiates an authentication or signing order. Use the collect method to query the status of the order. If the request is successful the response includes `orderRef, autoStartToken, qrStartToken and qrStartSecret` (see table below for details).

Example request auth without personal number.

```
POST /rp/v5.1/auth HTTP/1.1
Content-Type: application/json
Host: appapi2.bankid.com
{
  "endUserIp": "194.168.2.25"
}
```

Example request sign with personal number.

```
POST /rp/v5.1/sign HTTP/1.1
Content-Type: application/json
Host: appapi2.bankid.com
{
  "personalNumber":"190000000000",
  "endUserIp": "194.168.2.25",
  "userVisibleData": "IFRoaXMgaXMgYSBzYW1wbGUgdGV4dCB0byBiZSBzaWduZWQ="
}
```

14.1.1 Parameters for auth and sign

Name	Required	Value
`personalNumber`	Optional	The personal number of the user. String. 12 digits. Century must be included. If the personal number is excluded, the client must be started with the `autoStartToken` returned in the response
`endUserIp`	Required	The user IP address as seen by RP. String. IPv4 and IPv6 is allowed. Note the importance of using the correct IP address. It must be the IP address representing the user agent (the end user device) as seen by the RP. If there is a proxy for inbound traffic, special considerations may need to be taken to get the correct address. In some use cases the IP address is not available, for instance for voice based services. In this case, the internal representation of those systems IP address is ok to use.
`requirement`	Optional	Requirements on how the auth or sign order must be performed. See below.

14.1.2 Additional Parameters for sign

Name	Required	Value
`userVisibleData`	Required	The text to be displayed and signed. String. The text can be formatted using CR, LF and CRLF for new lines. The text must be encoded as UTF-8 and then base 64 encoded. 1--40 000 characters after base 64 encoding.
`userNonVisibleData`	Optional	Data not displayed to the user. String. The value must be base 64-encoded. 1-200 000 characters after base 64-encoding.
`userVisibleDataFormat`	Optional	If present, and set to “simpleMarkdownV1”, this parameter indicates that `userVisibleData` holds formatting characters which, if used correctly, will make the text displayed with the user nicer to look at. For further information of formatting options, please study the document **Guidelines for Formatted Text**.

14.1.3 Response from auth and sign

Name	Value
`orderRef`	Used to collect the status of the order. String.

Name	Value
autoStartToken	Used to compile the start url according to chapter 3 in this document. String.
qrStartToken	Used to compute the animated QR code according to section 4.2 in this document. String.
qrStartSecret	Used to compute the animated QR code according to section 4.2 in this document. String.

Example response from auth and sign.

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "orderRef":"131daac9-16c6-4618-beb0-365768f37288",
  "autoStartToken":"7c40b5c9-fa74-49cf-b98c-bfe651f9a7c6",
  "qrStartToken":"67df3917-fa0d-44e5-b327-edcc928297f8",
  "qrStartSecret":"d28db9a7-4cde-429e-a983-359be676944c"
}
```

14.2 /rp/v5.1/collect

Collects the result of a sign or auth order using the `orderRef` as reference. RP should keep on calling collect every two seconds as long as status indicates pending. RP must abort if `status` indicates failed. The user identity is returned when complete.

Example request collect:

```
POST /rp/v5.1/collect HTTP/1.1
Content-Type: application/json
Host: appapi2.bankid.com
{
  "orderRef":"131daac9-16c6-4618-beb0-365768f37288"
}
```

14.2.1 Parameters for collect

Name	Value
orderRef	The `orderRef` returned from auth or sign.

14.2.2 Response from collect

The response will have different content depending on status of the order. The status may be pending, failed or complete.

Name	Value
orderRef	The `orderRef` in question.
status	pending: The order is being processed. `hintCode` describes the status of the order. failed: Something went wrong with the order. `hintCode` describes the error. complete: The order is complete. `completionData` holds user information.
hintCode	Only present for pending and failed orders. See below.
completionData	Only present for complete orders. See below.

14.2.3 hintCode for Pending Orders

The order is pending. RP should use the `hintCode` to provide the user with details and instructions and keep on calling collect until failed or complete.

hintCode	Reason	Action by RP
outstandingTransaction	The order is pending. The client has not yet received the order. The `hintCode` will later change to noClient, started or userSign.	If RP tried to start the client automatically, the RP should inform the user that the app is starting. Message RFA13 should be used. If RP did not try to start the client automatically, the RP should inform the user that she needs to start the app. Message RFA1 should be used.

hintCode	Reason	Action by RP
noClient	The order is pending. The client has not yet received the order.	If RP tried to start the client automatically: This status indicates that the start failed or the users BankID was not available in the started client. RP should inform the user. Message RFA1 should be used. If RP did not try to start the client automatically: This status indicates that the user not yet has started her client. RP should inform the user. Message RFA1 should be used.
started	The order is pending. A client has been started with the `autostarttoken` but a usable ID has not yet been found in the started client. When the client starts there may be a short delay until all ID:s are registered. The user may not have any usable ID:s at all, or has not yet inserted their smart card.	If RP does not require the `autoStartToken` to be used and the user provided her personal number the RP should inform the user of possible solutions. Message RFA14 should be used. If RP require the `autoStartToken` to be used or the user did not provide her personal number the RP should inform the user of possible solutions. Message RFA15 should be used. Note: started is not an error, RP should keep on polling using collect.
userSign	The order is pending. The client has received the order.	The RP should inform the user. Message RFA9 should be used.
We may introduce new hint codes without prior notice. RP must handle unknown hint codes in their implementations.		If an unknown `hintCode` is returned for a pending order, RP should inform the user. Message RFA21 should be used. RP should update their implementation to support the new `hintCode` as soon as possible.

Example response from collect for a pending order:

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "orderRef":"131daac9-16c6-4618-beb0-365768f37288",
  "status":"pending",
  "hintCode":"userSign"
}
```

14.2.4 hintCode for Failed Orders

This is a final state. The order failed. RP should use the `hintCode` to provide the user with details and instructions. The same `orderRef` must not be used for additional collect requests.

hintCode	Reason	Action by RP
expiredTransaction	The order has expired. The BankID security app/program did not start, the user did not finalize the signing or the RP called collect too late.	RP must inform the user. Message RFA8.
certificateErr	This error is returned if: 1) The user has entered wrong security code too many times. The BankID cannot be used. 2) The users BankID is revoked. 3) The users BankID is invalid.	RP must inform the user. Message RFA16.
userCancel	The user decided to cancel the order.	RP must inform the user. Message RFA6.
cancelled	The order was cancelled. The system received a new order for the user.	RP must inform the user. Message RFA3.

hintCode	Reason	Action by RP
startFailed	The user did not provide her ID, or the RP requires `autoStartToken` to be used, but the client did not start within a certain time limit. The reason may be: 1) RP did not use `autoStartToken` when starting BankID security program/app. RP must correct this in their implementation. 2) The client software was not installed or other problem with the user's computer.	The RP must inform the user. Message RFA17.
We may introduce new hint Codes without prior notice. RP must handle unknown hint Codes in their implementations.		If an unknown `hintCode` is returned for a failed order, RP should inform the user. Message RFA22 should be used. RP should update their implementation to support the new `hintCode` as soon as possible.

Example response from collect for a failed request:

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "orderRef":"131daac9-16c6-4618-beb0-365768f37288",
  "status":"failed",
  "hintCode":"userCancel"
}
```

14.2.5 completionData for Completed Orders

This is a final state. The order was successful. The user has provided the security code and completed the order. The `completionData` includes the signature, user information and the OCSP response. RP should control the user information and continue their process. RP should keep the completion data for future references/compliance/audit.

Name	Value
`user`	Information related to the user, holds the following children: • `personalNumber`: The personal number. String. • `name`: The given name and surname of the user. String. • `givenName`: The given name of the user. String. • `surname`: The surname of the user. String.
`device`	Information related to the device, holds the following child: • `ipAddress`: The IP address of the user agent as the BankID server discovers it. String.
`cert`	Information related to the user's certificate (BankID), holds the following children: • `notBefore`: Start of validity of the users BankID. String, Unix ms. • `notAfter`: End of validity of the Users BankID. String, Unix ms. Note: `notBefore` and `notAfter` are the number of milliseconds since the UNIX Epoch, a.k.a. "UNIX time" in milliseconds. It was chosen over ISO8601 for its simplicity and lack of error prone conversions to/from string representations on the server and client side.
`signature`	The signature. The content of the signature is described in BankID Signature Profile specification. String. Base64-encoded. XML signature.
`ocspResponse`	The OCSP response. String. Base64-encoded. The OCSP response is signed by a certificate that has the same issuer as the certificate being verified. The OSCP response has an extension for Nonce. The nonce is calculated as: • SHA-1 hash over the base 64 XML signature encoded as UTF-8. • 12 random bytes is added after the hash • The nonce is 32 bytes (20 + 12)

Example response from collect for a complete order:

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "orderRef":"131daac9-16c6-4618-beb0-365768f37288",
  "status":"complete",
```

```
  "completionData":{
    "user":{
      "personalNumber":"190000000000",
      "name":"Karl Karlsson",
      "givenName":"Karl",
      "surname":"Karlsson"
    },
    "device":{
      "ipAddress":"192.168.0.1"
    },
    "cert":{
      "notBefore":"1502983274000",
      "notAfter":"1563549674000"
    },
    "signature":"<base64-encoded data>",
    "ocspResponse":"<base64-encoded data>"
  }
}
```

14.3 /rp/v5.1/cancel

Cancels an ongoing sign or auth order. This is typically used if the user cancels the order in your service or app.

Example request cancel:

```
POST /rp/v5.1/cancel HTTP/1.1
Content-Type: application/json
Host: appapi2.bankid.com
{
  "orderRef":"131daac9-16c6-4618-beb0-365768f37288"
}
```

14.3.1 Parameters for cancel

Name	Value
`orderRef`	The `orderRef` from the response from auth or sign. String.

Response from cancel

A successful response contains an empty JSON object.

Example response cancel

```
HTTP/1.1 200 OK
Content-Type: application/json
{}
```

14.4 Errors

The following table describes possible errors, their cause and the action to take by RP.

HTTP	errorCode	Reason	Action by RP
400	alreadyInProgress	An auth or sign request with personal number was sent, but an order for the user is already in progress. The order is aborted. No order is created. Details are found in `details`.	RP must inform the user that an auth or sign order is already in progress for the user. Message RFA4 should be used.
400	invalidParameters	Invalid parameter. Invalid use of method. Using an `orderRef` that previously resulted in completed. The order cannot be collected twice. Using an `orderRef` that previously resulted in failed. The order cannot be collected twice. Using an `orderRef` that is too old. completed orders can only be collected up to 3 minutes and failed orders up to 5 minutes. Details are found in `details`.	RP must not try the same request again. This is an internal error within RP's system and must not be communicated to the user as a BankID error.

HTTP	errorCode	Reason	Action by RP
400	We may introduce new error codes without prior notice. RP must handle unknown error codes in their implementations.		If an unknown `errorCode` is returned, RP should inform the user. Message RFA22 should be used. RP should update their implementation to support the new `errorCode` as soon as possible.
401, 403	unauthorized	RP does not have access to the service.	RP must not try the same request again. This is an internal error within RP's system and must not be communicated to the user as a BankID error.
404	notFound	An erroneously URL path was used.	RP must not try the same request again. This is an internal error within RP's system and must not be communicated to the user as a BankID error.
405	methodNotAllowed, <empty>	Only http method POST is allowed.	RP must not try the same request again. This is an internal error within RP's system and must not be communicated to the user as a BankID error.
408	requestTimeout	It took too long time to transmit the request.	RP must not automatically try again. This error may occur if the processing at RP or the communication is too slow. RP must inform the user. Message RFA5.
415	unsupportedMediaType	Adding a "charset" parameter after 'application/json' is not allowed since the MIME type "application/json" has neither optional nor required parameters. The type is missing or erroneously.	RP must not try the same request again. This is an internal error within RP's system and must not be communicated to the user as a BankID error.
500	internalError	Internal technical error in the BankID system.	RP must not automatically try again. RP must inform the user. Message RFA5.
503	maintenance	The service is temporarily out of service.	RP may try again without informing the user. If this error is returned repeatedly, RP must inform the user. Message RFA5.

Example response from collect with an invalid orderRef:

```
HTTP/1.1 400 Bad Request
Content-Type: application/json
{
  "errorCode":"invalidParameters",
  "details":"No such order"
}
```

14.5 Requirement

RP may use the `requirement` parameter to describe how the signature must be created and verified. A typical use case is to require Mobile BankID or a special card reader. A requirement can be set for both auth and sign orders. The following table describes requirements, their possible values and defaults.

Name	Value	Default
`cardReader`	"class1" - (default). The transaction must be performed using a card reader where the PIN code is entered on the computers keyboard, or a card reader of higher class. "class2" - The transaction must be performed using a card reader where the PIN code is entered on the reader, or a reader of higher class. <no value> - defaults to "class1". This condition should be combined with a `certificatePolicies` for a smart card to avoid undefined behavior.	No special type of card reader required.
`certificatePolicies`	The oid in certificate policies in the user certificate. List of String. One wildcard ”*” is allowed from position 5 and forward ie. 1.2.752.78.* The values for production BankIDs are: "1.2.752.78.1.1" - BankID on file "1.2.752.78.1.2" - BankID on smart card "1.2.752.78.1.5" - Mobile BankID "1.2.752.71.1.3" - Nordea e-id on file and on smart card. The values for test BankIDs are: "1.2.3.4.5" - BankID on file "1.2.3.4.10" - BankID on smart card "1.2.3.4.25" - Mobile BankID "1.2.752.71.1.3" - Nordea e-id on file and on smart card. “1.2.752.60.1.6” - Test BankID for some BankID Banks	If no certificate policies is set the following are default in the production system: 1.2.752.78.1.1, 1.2.752.78.1.2, 1.2.752.78.1.5, 1.2.752.71.1.3 The following are default in the test system: 1.2.3.4.5, 1.2.3.4.10, 1.2.3.4.25, 1.2.752.60.1.6, 1.2.752.71.1.3 If one certificate policy is set all the default policies are dismissed.
`issuerCn`	The cn (common name) of the issuer. List of String. Wildcards are not allowed. Nordea values for production: "Nordea CA for Smartcard users 12" - E-id on smart card issued by Nordea CA. "Nordea CA for Softcert users 13" - E-id on file issued by Nordea CA Example Nordea values for test: "Nordea Test CA for Smartcard users 12" - E-id on smart card issued by Nordea CA. "Nordea Test CA for Softcert users 13" - E-id on file issued by Nordea CA	If issuer is not defined all relevant BankID and Nordea issuers are allowed.
`autoStartTokenRequired`	Deprecated. Will not be possible to use in future versions of the RP API. Use `tokenStartRequired`. If present, and set to true, one of the following methods must be used to start the client: • According to chapter 3 in this document (autoStartToken in an URL). • According to chapter 4.1 in this document (autoStartToken in a static QR) Boolean. To be used if it is important that the BankID App is on the same device as the RP service. If this requirement is omitted or set to false, the client does not need to be started using autoStartToken.	The client does not need to be started using autoStartToken.
`allowFingerprint`	Users of iOS and Android devices may use fingerprint for authentication and signing if the device supports it and the user configured the device to use it. Boolean. No other devices are supported at this point. If set to true, the users are allowed to use fingerprint. If set to false, the users are not allowed to use fingerprint.	true for authentication. false for signing.

Name	Value	Default
`tokenStartRequired`	The tokenStartRequired replaces the autostartTokenRequired. Boolean. If present, and set to true, one of the following methods must be used to start the client: • According to chapter 4.2 in this document (animated QR). • According to chapter 3 in this document (`autoStartToken` in an URL). • According to chapter 4.1in this document (`autoStartToken` in a static QR).	The client does not need to be started using a token.

14.5.1 Example – allowFingerprint for sign

```
POST /rp/v5.1/sign HTTP/1.1
Content-Type: application/json
Host: appapi2.bankid.com
{
  "personalNumber":"190000000000",
  "endUserIp": "192.168.0.1"
  "requirement": {"allowFingerprint":true}
}
```

14.5.2 Example – certificatePolicies for auth with Mobile BankID

```
POST /rp/v5.1/auth HTTP/1.1
Content-Type: application/json
Host: appapi2.bankid.com
{
  "personalNumber":"190000000000",
  "endUserIp": "192.168.0.1"
  "requirement": {"certificatePolicies":["1.2.752.78.1.5"]}
}
```

14.5.3 Example – Combined Requirements

Multiple parameters can be set for a `requirement` (AND). Multiple values can be set for parameter `certificatePolicies` and `issuerCn` (OR).

```
POST /rp/v5.1/auth HTTP/1.1
Content-Type: application/json
Host: appapi2.bankid.com
{
  "personalNumber":"190000000000",
  "endUserIp": "192.168.0.1"
  "requirement": {"certificatePolicies":["1.2.752.78.1.5", "1.2.752.71.1.3",
"1.2.752.78.1.2"], "tokenStartRequired":true}
}
```