



# **PingFederate™**

## **Integration Guide**

**Version 9.1.0**

(October 2024 Prerelease)

**NOTICE TO LICENSEE:**

*This source code and/or documentation ("Licensed Deliverables") are subject to Nok Nok Labs, Inc. intellectual property rights under U.S. and international Copyright laws.*

*These Licensed Deliverables contained herein is PROPRIETARY and CONFIDENTIAL to Nok Nok Labs, Inc. and is being provided under the terms and conditions of a form of Nok Nok Labs, Inc. software license agreement by and between Nok Nok Labs, Inc. and Licensee ("License Agreement") or electronically accepted by Licensee. Notwithstanding any terms or conditions to the contrary in the License Agreement, reproduction or disclosure of the Licensed Deliverables to any third party without the express written consent of Nok Nok Labs, Inc. is prohibited.*

**NOTWITHSTANDING ANY TERMS OR CONDITIONS TO THE CONTRARY IN THE LICENSE AGREEMENT, NOK NOK LABS, INC. MAKES NO REPRESENTATION ABOUT THE SUITABILITY OF THESE LICENSED DELIVERABLES FOR ANY PURPOSE. THEY ARE PROVIDED "AS IS" WITHOUT EXPRESS OR IMPLIED WARRANTY OF ANY KIND. NOK NOK LABS, INC. DISCLAIMS ALL WARRANTIES WITH REGARD TO THESE LICENSED DELIVERABLES, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE. NOTWITHSTANDING ANY TERMS OR CONDITIONS TO THE CONTRARY IN THE LICENSE AGREEMENT, IN NO EVENT SHALL NOK NOK LABS, INC. BE LIABLE FOR ANY SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THESE LICENSED DELIVERABLES.**

*U.S. Government End Users. These Licensed Deliverables are a "commercial item" as that term is defined at 48 C.F.R. 2.101 (OCT 1995), consisting of "commercial computer software" and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (SEPT 1995) and is provided to the U.S. Government only as a commercial end item. Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (JUNE 1995), all U.S. Government End Users acquire the Licensed Deliverables with only those rights set forth herein.*

*Any use of the Licensed Deliverables in individual and commercial software must include, in the user documentation and internal comments to the code, the above Disclaimer and U.S. Government End Users Notice.*

# Table of Contents

- Introduction..... 1
  - Related Documents ..... 1
- Installation..... 2
  - Multi-Tenant Deployments..... 2
- Configuration ..... 2
  - Step 1) Create IdP Adapter Instance ..... 3
  - Step 2) Map Adapter Instance to Grant Contract..... 7
  - Step 3) Set Up a Default Authentication Source ..... 9
  - Step 4) Create Access Token Management Instance ..... 10
  - Step 5) Add Access Token Mapping ..... 12
  - Step 6) Add OpenID Connect Policy ..... 13
  - Step 7) Add OAuth Clients ..... 14
- Sign-in Page..... 15
  - nnlsignin Web Application ..... 15
  - Implement your own PingFederate Sign-in Page ..... 17
- Initial FIDO Registration through OIDC ..... 18
- Appendix A: Sequence Diagrams ..... 19
  - Web App Sign In Integration..... 19
  - Native App Sign In Integration..... 20
- Appendix B: Upgrading ..... 21
  - Upgrading from Previous Version to October 2024 Version..... 21

# Introduction

This document guides you through the installation and configuration of the Nok Nok PingFederate Adapter. Using this Adapter you can implement passwordless authentication on your PingFederate Server, providing a seamless and secure authentication experience for your users.

Your app authenticates a user by sending a JWT to the PingFederate Server. The PingFederate Server then sends the JWT to the Nok Nok PingFederate Adapter for validation. The Server can send this JWT to the Adapter in any one of the following places in its request:

- header
- cookie
- form POST<sup>1</sup>

Upon successful validation of the JWT, the Nok Nok Adapter extracts the sub claim value and stores it in the username attribute of the JWT for the PingFederate Server. See the [Sequence Diagrams](#) for more detail.

The package includes:

- The Nok Nok Adapter and its required JAR libraries
- The Nok Nok Web App SDK and the [nnlsignin web application](#)

## Related Documents

This Guide references several resources that provide additional information about the Nok Nok App SDK and the Nok Nok Authentication Server. You should locate these documents and reference them as you read this Guide.

- **Nok Nok App SDK for JavaScript Developer Guide:** Details how to include strong authentication in a web app using the Nok Nok App SDK.
- **Nok Nok App UI Customization Guide:** Explains how to customize the UI that the Nok Nok App SDK displays. This document contains instructions for editing a cross-platform JSON configuration file to localize and style the UI.
- **Nok Nok S3 Authentication Suite Configuration Guide:** This guide provides step-by-step instructions to perform common post-installation and operational tasks on the Nok Nok Authentication Server. It explains concepts such as RP IDs, facet IDs, session data, and Adaptive Authentication. It is organized around the set up needed for UAF and FIDO2 as well as major features like Adaptive Authentication, FIDO out-of-band (OOB) Authentication and External Authentication.
- **Client API Documentation:** The standard automatically-generated API documentation. Located in the docs/jssdk directory within the Nok Nok JavaScript App SDK deployment package: nn\_web\_app\_sdk\_<version>.<build\_number>.zip.

---

<sup>1</sup> If your end user sign-in page and the Nok Nok Adapter have entirely different domains, then you *must* use form POST for the JWT.

# Installation

Before beginning these steps, make sure that the Nok Nok Server is deployed and operational. Then follow these instructions to install the Nok Nok Adapter on your PingFederate deployment:

1. Copy all JAR libraries from the installation package to the PingFederate server's deploy directory:  
`<pf-install>/pingfederate/server/default/deploy`
2. Replace any older versions of the third-party JARs included in the package to prevent duplication.
3. Prepare a `jwt_config.json` file for the Nok Nok Server tenant. To ensure compatibility with the Nok Nok API Server, make sure the keys and configuration parameters for JWT in the `jwt_config.json` file match those of the API server. If using a symmetric key on the server, you can export the JWT configuration from the API server session plugin. Refer to the *Nok Nok Authentication Suite Configuration Guide*, section *Exporting and Importing Configuration and Objects* for instructions on how to export the API Server configuration.
4. Copy the prepared `jwt_config.json` file to `<pf-install>/pingfederate/server/default/conf/configurations/<TENANT IDENTIFIER>/SessionPlugin/jwt_config.json`, creating the required intermediate directories as needed. See the Nok Nok Adapter's [configuration parameters](#) below.
5. Restart the PingFederate Server. The Nok Nok Adapter appears on the PingFederate Admin Console.
6. Find the `nnlsignin` application inside the Nok Nok Web App SDK package. Deploy the `nnlsignin` web application on a web server. `nnlsignin` can be hosted on an origin that is different from the origin that hosts the Nok Nok Adapter. You will [configure the sign-in page](#) after you configure the Adapter.

## Multi-Tenant Deployments

If your Nok Nok deployment supports multiple tenants, follow these additional installation instructions:

1. Use the PingFederate Admin Console to create and configure a separate realm in your PingFederate server for each Nok Nok Server tenant. The Nok Nok Adapter uses the `TENANT IDENTIFIER` that you configure for a realm in your PingFederate server to find the directory containing the configurations for that realm. See [Step 4. above](#) for details of this directory structure.
2. Also while configuring each realm, make sure that the `SIGN IN ENDPOINT` includes the `tenant_id` parameter. For example, the following value for `SIGN IN ENDPOINT` is needed for the `nnlsignin` app:  
`<SIGN IN ENDPOINT>?tenant_id=<TENANT IDENTIFIER>`
3. Repeat Step 3. and Step 4. in the Installation instructions [above](#) for each tenant.

## Configuration

To configure the Nok Nok Adapter for your PingFederate deployment, follow the instructions below using the PingFederate Admin Console:

## Step 1) Create IdP Adapter Instance

To create an instance of the Nok Nok Adapter, navigate to **Authentication > Integration > IdP Adapters**. Click **Create New Instance**. If the Nok Nok Adapter has been successfully installed, **Nok Nok Universal Adapter** is displayed in the list of available adapters.

The screenshot shows the PingFederate web interface at the URL `local.noknoktest.com:9999/pingfederate/app#/idpAdapterManager`. The navigation menu on the left includes 'Integration', 'IdP Connections', 'IdP Adapters', 'Authentication API Applications', and 'IdP Default URL'. The main content area is titled 'IdP Adapters | Create Adapter Instance' and has tabs for 'Type', 'IdP Adapter', 'Adapter Attributes', 'Adapter Contract Mapping', and 'Summary'. The 'Type' tab is active, showing instructions: 'Enter an Adapter Instance Name and ID, select the Adapter Type, and a parent if applicable. The Adapter Type is limited to the adapters currently installed on your server.' The form fields are 'INSTANCE NAME', 'INSTANCE ID', 'TYPE', and 'PARENT INSTANCE'. The 'TYPE' dropdown menu is open, showing a list of adapters: 'Composite Adapter', 'HTML Form IdP Adapter', 'HTTP Basic IdP Adapter', 'Identifier First Adapter', 'Kerberos Adapter', 'Nok Nok Universal Adapter' (highlighted), 'OpenToken IdP Adapter 2.7.2', 'Passthrough IdP Adapter', 'PingID Adapter 2.12.0', 'PingOne DaVinci IdP Adapter 1.0.1', 'PingOne MFA IdP Adapter 2.0', 'PingOne Risk Management IdP Adapter 1.2', 'PingOne Verify IdP Adapter 1.1', 'Reference ID IdP Adapter 2.0.4', and 'X.509 Certificate IdP Adapter 1.3.1'. The 'Next' button is visible at the bottom right.

Fill in the details and click **Next**:

The screenshot shows the PingFederate Admin Console interface. The left sidebar contains navigation links: Integration, IdP Connections, IdP Adapters (selected), Authentication API Applications, and IdP Default URL. The main content area is titled 'IdP Adapters | Create Adapter Instance'. It features a tabbed interface with 'Type' selected, followed by 'Extended Contract', 'Adapter Attributes', 'Adapter Contract Mapping', and 'Summary'. Below the tabs, a text box instructs the user to 'Enter an Adapter Instance Name and ID, select the Adapter Type, and a parent if applicable. The Adapter Type is limited to the adapters currently installed on your server.' The form contains four input fields: 'INSTANCE NAME' with the value 'nnluniversaladapter', 'INSTANCE ID' with the value 'nnluniversaladapter', 'TYPE' with a dropdown menu showing 'Nok Nok Universal Adapter', and 'PARENT INSTANCE' with a dropdown menu showing 'None'. At the bottom right of the form are 'Cancel' and 'Next' buttons.

In the PingFederate Admin Console, the Nok Nok Adapter provides the following configuration parameters:

- **SIGN IN ENDPOINT** - This is the URL of the sign-in page where the Adapter redirects if the JWT is missing from the request, or if the JWT is present in the request but invalid.
- **TENANT IDENTIFIER** - Specifies the tenant ID used for creating tenant-specific intermediate directories.
- **USE POST METHOD** - Check this box if one or both of the following applies:
  - your sign-in page delivers the JWT to the Adapter as a request parameter in a form POST.
  - your sign-in page and the PingFederate Server have different origins.
- **COOKIE DOMAIN** - If the PingFederate Server and the nnlsignin app have the same parent domain, specify the domain name here. This setting is ignored if "USE POST METHOD" is enabled.
- **ENABLE INLINE REGISTRATION** - When Inline Registration is enabled, end users are prompted to register if they have no credentials already registered. The following are the prerequisites to enabling Inline Registration:
  - The Nok Nok Adapter is used for second-factor authentication. The adapter should be set as part of an adapter chaining flow in the policy to enable multi-step authentication.
  - Your PingFederate policy is configured to permit Inline Registration.

## IdP Adapters | Create Adapter Instance

Type	IdP Adapter	Extended Contract	Adapter Attributes	Adapter Contract Mapping	Summary
------	-------------	-------------------	--------------------	--------------------------	---------

Complete the configuration necessary to look up user security contexts in your environment. This configuration was designed into the adapter for use at your site.

Set the details necessary for Universal adapter configuration

Field Name	Field Value	Description
SIGN IN ENDPOINT	<input type="text" value="https://ping.noknoktest.com:8443/nnlsignin"/>	Enter sign in page URL
TENANT IDENTIFIER	<input type="text" value="forgerock"/>	Enter tenant identifier
USE POST METHOD	<input checked="" type="checkbox"/>	Check if cross-origin sign in
COOKIE DOMAIN	<input type="text" value="noknoktest.com"/>	Enter cookie domain name
ENABLE INLINE REGISTRATION	<input type="checkbox"/>	Check if inline registration

Navigate to the **Adapter Attributes** tab and select username.

The screenshot shows the PingFederate web interface. The left sidebar has a navigation menu with 'Integration' expanded, showing 'IdP Connections', 'IdP Adapters', 'Authentication API Applications', and 'IdP Default URL'. The main content area is titled 'IdP Adapters | Create Adapter Instance'. It features a tabbed interface with 'Type', 'IdP Adapter', 'Extended Contract', 'Adapter Attributes' (selected), 'Adapter Contract Mapping', and 'Summary'. Below the tabs, there is a text block explaining the purpose of the adapter attributes. A 'UNIQUE USER KEY ATTRIBUTE' section contains a dropdown menu with 'username' selected. Below this is a table with columns 'Attribute', 'Pseudonym', and 'Mask Log Values'. The 'username' attribute is listed with 'Pseudonym' checked and 'Mask Log Values' unchecked. At the bottom, there is a checkbox for 'MASK ALL OGNL-EXPRESSION GENERATED LOG VALUES' and three buttons: 'Cancel', 'Previous', and 'Next'.

Attribute	Pseudonym	Mask Log Values
username	<input checked="" type="checkbox"/>	<input type="checkbox"/>

☐ MASK ALL OGNL-EXPRESSION GENERATED LOG VALUES

[Cancel](#) [Previous](#) [Next](#)

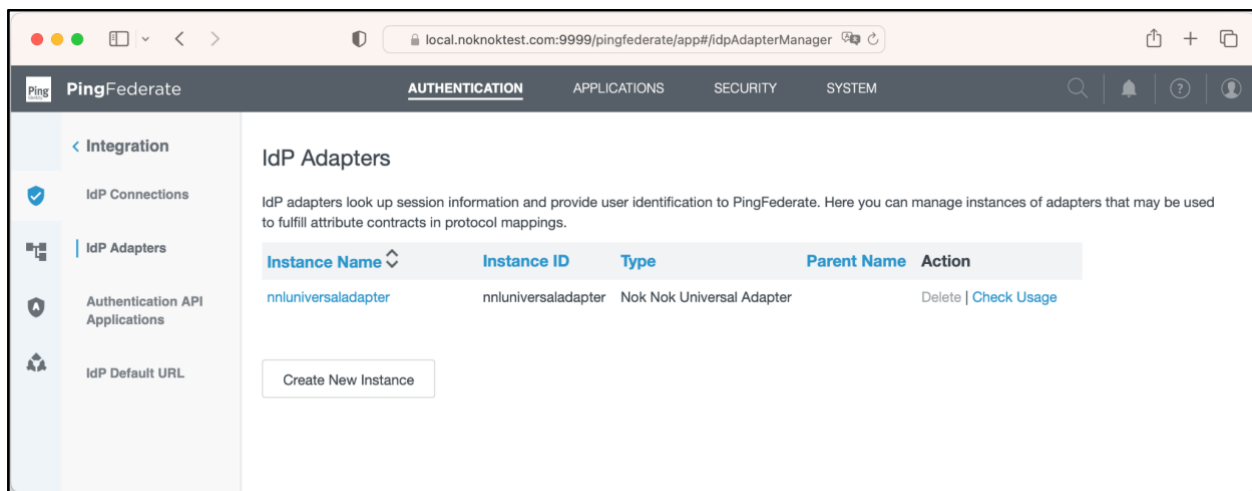


To finish setting up the Nok Nok Adapter, click **Next** to access the summary screen. Take a moment to review the summary and confirm that all information is accurate. After confirming the information, click **Save** to create the adapter instance.

IdP Adapters | Create Adapter Instance

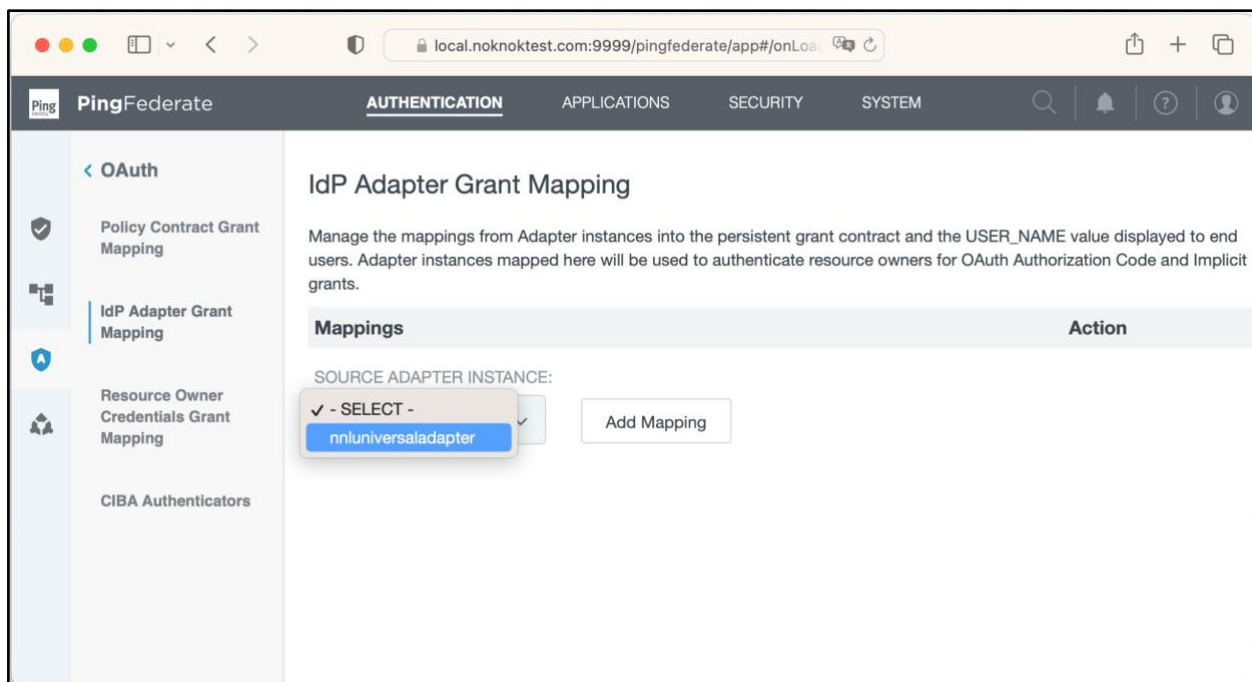
Type	IdP Adapter	Extended Contract	Adapter Attributes	Adapter Contract Mapping	Summary
IdP adapter instance summary information.					
Create Adapter Instance					
Type					
Instance Name	nnluniversaladapter				
Instance ID	nnluniversaladapter				
Type	Nok Nok Universal Adapter				
Class Name	com.noknok.adapter.ping.UniversalAdapter				
Parent Instance Name	None				
IdP Adapter					
Sign in endpoint	https://ping.noknoktest.com:8443/nnlsignin				
Tenant Identifier	forgerock				
Use POST method	true				
Cookie Domain	noknoktest.com				
Enable inline registration	false				
Extended Contract					
Attribute	username				
Adapter Attributes					
Mask all OGNL expression log values	false				
Pseudonym	username				
Unique User Key Attribute	username				
Adapter Contract Mapping					
Attribute Sources & User Lookup					
Data Sources	(None)				
Adapter Contract Fulfillment					
username	username (Adapter)				
Issuance Criteria					
Criterion	(None)				

The Nok Nok Adapter instance is now displayed on the Identity Provider (IdP) Adapters screen.



## Step 2) Map Adapter Instance to Grant Contract

To map the Nok Nok Adapter instance to a persistent grant contract, navigate to **Authentication > OAuth > IdP Adapter Grant Mapping**. From the list, select the Nok Nok Adapter instance and click **Add Mapping**.



Leave the **Attribute Sources** and **User Lookup** blank and click on the **Contract Fulfilment** tab. Add details as below:

The screenshot shows the 'IdP Adapter Grant Mapping' page in the PingFederate console. The 'Contract Fulfilment' tab is selected. The page displays a table for mapping contract items to attribute sources and values.

Contract	Source	Value	Actions
USER_KEY	Adapter	username	None available
USER_NAME	Adapter	username	None available

At the bottom of the page, there are three buttons: 'Cancel', 'Previous', and 'Next'.

No changes are required on the **Issuance Criteria** tab. Click **Next** to see a summary as below. After reviewing the summary, click **Save**.

The screenshot shows the 'IdP Adapter Grant Mapping' page in the PingFederate console, now on the 'Summary' tab. The page displays a summary of the mappings.

**Mapping Summary**

**IdP Adapter Mapping**

**Attribute Sources & User Lookup**

Data Sources	
(None)	

**Contract Fulfillment**

Contract	Value
USER_KEY	username (Adapter)
USER_NAME	username (Adapter)

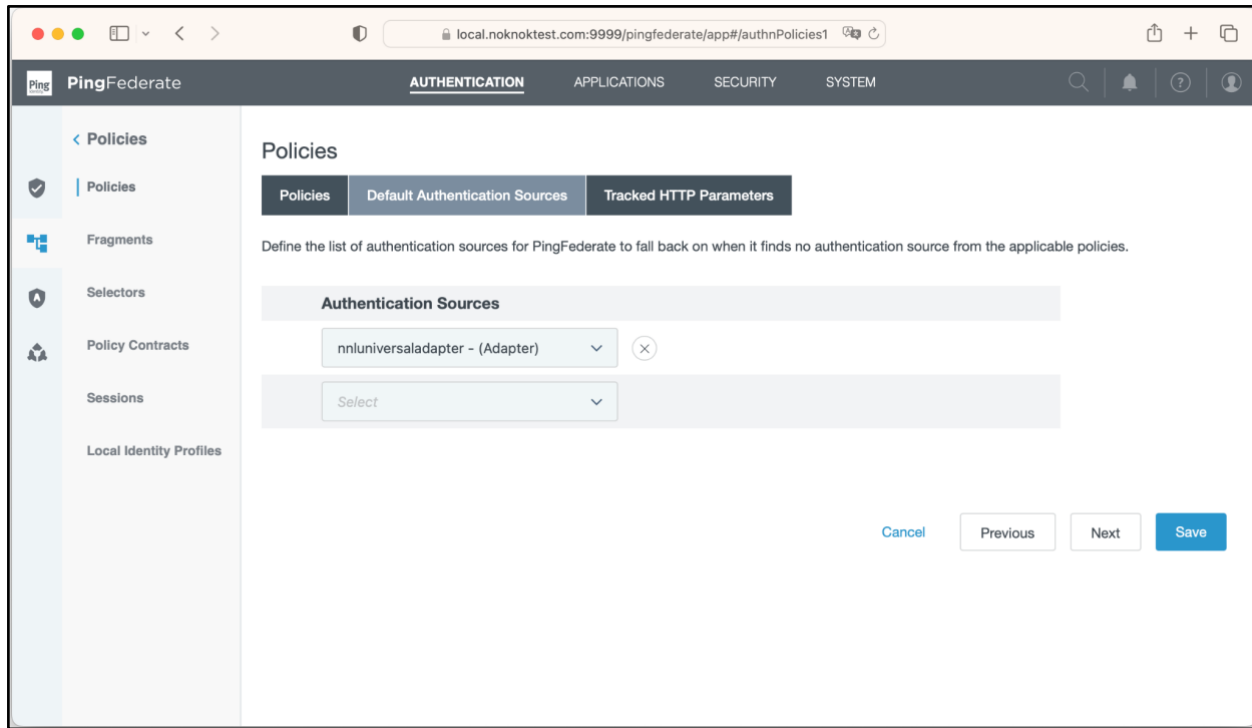
**Issuance Criteria**

Criterion	
(None)	

At the bottom of the page, there are three buttons: 'Cancel', 'Previous', and 'Save'.

### Step 3) Set Up a Default Authentication Source

To add the Nok Nok Adapter as a default authentication source, navigate to **Authentication > Policies > Default Authentication Sources**. Then select the Nok Nok Adapter instance and click **Save**.



## Step 4) Create Access Token Management Instance

To create the access token management instance, navigate to **Applications > OAuth > Access Token Management** and click **Create New Instance**.

The screenshot shows the PingFederate web interface. The top navigation bar includes 'PingFederate', 'AUTHENTICATION', 'APPLICATIONS' (selected), 'SECURITY', and 'SYSTEM'. A left sidebar lists navigation options: '< OAuth', 'Clients', 'Access Token Management' (selected), 'Access Token Mappings', 'OpenID Connect Policy Management', and 'CIBA Request Policies'. The main content area is titled 'Access Token Management | Create Access Token Management Instance'. It features a tabbed interface with 'Type' selected, and sub-tabs for 'Instance Configuration', 'Session Validation', 'Access Token Attribute Contract', and 'Resource URIs'. Below these are 'Access Control' and 'Summary' tabs. A text block instructs the user: 'Enter an Access Token Management Instance Name and ID, select the plugin Access Token Management Type, and a parent if applicable. The types available are limited to the plugins currently installed on your server.' The form contains four fields: 'INSTANCE NAME' with value 'oidcaccessesstokenmanag', 'INSTANCE ID' with value 'oidcaccessesstokenmanag', 'TYPE' with a dropdown menu showing 'Internally Managed Reference Tokens', and 'PARENT INSTANCE' with a dropdown menu showing 'None'. At the bottom right are 'Cancel' and 'Next' buttons.

Type	Instance Configuration	Session Validation	Access Token Attribute Contract	Resource URIs
Access Control	Summary			

Enter an Access Token Management Instance Name and ID, select the plugin Access Token Management Type, and a parent if applicable. The types available are limited to the plugins currently installed on your server.

INSTANCE NAME: oidcaccessesstokenmanag

INSTANCE ID: oidcaccessesstokenmanag

TYPE: Internally Managed Reference Tokens

PARENT INSTANCE: None

Cancel Next

Click **Next** until you reach the **Access Token Attribute Contract** tab. Add the **username** attribute to **Extend the Contract**.

The screenshot shows the PingFederate web interface. The browser address bar indicates the URL is `local.noknokit.com:9999/pingfederate/app#/onAccessToken...`. The interface has a top navigation bar with tabs: AUTHENTICATION, APPLICATIONS, SECURITY, and SYSTEM. The left sidebar shows a navigation menu with options: OAuth, Clients, Access Token Management, Access Token Mappings, OpenID Connect Policy Management, and CIBA Request Policies. The main content area is titled 'Access Token Management | Create Access Token Management Instance'. It features a tabbed interface with tabs: Type, Instance Configuration, Session Validation, Access Token Attribute Contract (selected), and Resource URIs. Below the tabs, there are sub-tabs: Access Control and Summary. A text box explains: 'Provide the names of the attributes that will be carried in (or referenced by) the OAuth access token. For auditing purposes, an attribute may be chosen as the subject.' Below this is a 'Subject Attribute Name' dropdown menu set to 'USER\_KEY'. A table titled 'Extend the Contract' is shown with columns: 'Extend the Contract', 'Multi-Valued' (with a question mark icon), and 'Action'. The table has one row with 'username' in the first column, an unchecked checkbox in the second column, and 'Edit | Delete' in the third column. Below the table is an 'Add' button. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Next'.

Click **Next** until you reach the **Summary** tab. Click **Save**.

## Step 5) Add Access Token Mapping

To add the access token mapping, navigate to **Applications > OAuth > Access Token Mappings**. Select the **Idp Adapter** CONTEXT and ACCESS TOKEN MANAGER, and click **Add Mapping**:

The screenshot shows the PingFederate web interface. The left sidebar is under the 'OAuth' section, with 'Access Token Mappings' selected. The main content area is titled 'Access Token Mappings' and contains a description: 'Manage the attribute mapping(s) used to fulfill the access token attribute contracts. This configuration maps from a persistent grant or other sources into the access token attribute contract. For mappings involving a persistent grant, a default mapping should be configured for each access token manager. The default can be overridden based on the context of the authentication event of the original grant.'

Context	Access Token Manager	Action
CONTEXT: IdP Adapter: nlnuniversaladapter	ACCESS TOKEN MANAGER: oidcaccess token management	<button>Add Mapping</button>

Click **Next** to reach the **Contract Fulfillment** tab. Select Adapter as your **Source**, and select username as your **Value**.

The screenshot shows the 'Access Token Mapping' configuration page with the 'Contract Fulfillment' tab selected. The page has tabs for 'Attribute Sources & User Lookup', 'Contract Fulfillment', 'Issuance Criteria', and 'Summary'. Below the tabs, it says 'Select a Source and Value to map into each item in the Contract list.'

Contract	Source	Value ?	Actions
username	Adapter	username	None available

At the bottom right, there are buttons: [Cancel](#), [Previous](#), and [Next](#).

Click **Next** until you reach the **Summary** tab. Click **Save**.

## Step 6) Add OpenID Connect Policy

To add the OIDC policy, navigate to **Applications > OAuth > OpenID Connect Policy Management** and click **Add Policy**.

The screenshot shows the PingFederate web interface for OpenID Connect Policy Management. The browser address bar shows `local.noknoktest.com:9999/pingfederate/app#/onLoadPolicyManagement`. The interface has a top navigation bar with tabs: AUTHENTICATION, APPLICATIONS (selected), SECURITY, and SYSTEM. A left sidebar contains a menu with: OAuth (selected), Clients, Access Token Management, Access Token Mappings, OpenID Connect Policy Management (highlighted), and CIBA Request Policies. The main content area is titled "OpenID Connect Policy Management | Policy" and has several tabs: Manage Policy (selected), Attribute Contract, Attribute Scopes, Attribute Sources & User Lookup, and Contract Fulfillment. Below these are sub-tabs: Issuance Criteria (selected) and Summary. A text prompt says: "Enter a Policy ID and Name. You may also change general settings for the ID Token." The form fields are: POLICY ID (oidcpolicymanagement), NAME (oidcpolicymanagement), ACCESS TOKEN MANAGER (a dropdown menu is open showing "oidcaccess token management" as the selected option), ID TOKEN LIFETIME (5 minutes), and several checkboxes for token options: INCLUDE SESSION IDENTIFIER IN ID TOKEN, INCLUDE USER INFO IN ID TOKEN, INCLUDE STATE HASH IN ID TOKEN, RETURN ID TOKEN ON REFRESH GRANT, and REISSUE ID TOKEN IN HYBRID FLOW. A "Manage Access Token Managers" button is at the bottom left of the form. "Cancel" and "Next" buttons are at the bottom right.



Click **Next** until you reach the **Contract Fulfillment** tab. Map **sub** claim in **Access Token** to **username** value. Click **Next** until you reach the **Summary** tab. Click **Save**.

The screenshot shows the PingFederate web interface. The left sidebar contains a navigation menu with the following items: OAuth, Clients, Access Token Management, Access Token Mappings, OpenID Connect Policy Management (selected), and CIBA Request Policies. The main content area is titled 'OpenID Connect Policy Management | Policy'. It features a series of tabs: Manage Policy, Attribute Contract, Attribute Scopes, Attribute Sources & User Lookup, and Contract Fulfillment (active). Below these tabs are two sub-tabs: Issuance Criteria and Summary (active). A text instruction reads: 'Fulfill the Attribute Contract with values from the Access Token or from other sources listed.' Below this is a table with the following structure:

Attribute Contract	Source	Value ?	Actions
sub	Access Token	username	None available

At the bottom right of the interface are four buttons: Cancel, Previous, Next, and Save.

## Step 7) Add OAuth Clients

Your Nok Nok Adapter is now ready to use. You can add your OAuth clients by referring to the instructions at [Configuring OAuth Clients - PingFederate](#).

# Sign-in Page

If you are integrating FIDO authentication into a web app, or if you are using a webview to integrate it into your mobile app, then you need to have a sign-in web page for your end users. You can either use the nnlsignin web application included in the Nok Nok Web App SDK package, or you can use your own sign-in page. In either case, the sign-in page and the Nok Nok Adapter can be hosted on the same or on different origins.

The nnlsignin web application included in Nok Nok Web App SDK package implements the required functionality. Test your integration with this nnlsignin web application, even if you intend to implement your own sign-in web page later. This section first describes how to configure the nnlsignin web app, and then tells how to [implement your own PingFederate sign-in page](#).

## nnlsignin Web Application

To use the nnlsignin app as your sign-in page:

1. Extract nnlsignin.war from the Nok Nok Web App SDK package into the webapps directory of your chosen application server.
2. Update the information in nnlsignin/config/config.js:

```
// Configuration object.
var signinConfig = {};

// The version of the Nok Nok JS App SDK the App was built with.
signinConfig.appsdk_version = {{nnl-version}};

// A suffix can be added to this version to point out changes
// to the configuration within the same release.
signinConfig.version = {{nnl-version}};

signinConfig.apiserver = {server-host-where-nnlappsdk-located};

// If you have a single Nok Nok Server tenant, then the tenant_id can be hard
// coded. If your deployment supports multi-tenancy, get this from the
// tenant_id query parameter.
signinConfig.tenant_id = {{tenant-id}};

// Default configuration fields.
signinConfig.nnlappsdk_url = "${apiserver}/nnlappsdk-${appsdk_version}";
signinConfig.storage_endpoint = "${apiserver}/nnlgateway/storage";
signinConfig.reg_endpoint = "${apiserver}/nnlgateway/nnl/${tenant_id}/reg";
signinConfig.auth_endpoint = "${apiserver}/nnlgateway/nnl/${tenant_id}/auth";

// For the OOB method to function, the web_oob_url is mandatory, since code sets
// the QrType to be UNIVERSAL_ANY_RP.
signinConfig.web_oob_url = '${apiserver}/nnlsignin/oobrecv.html';

// If sign-in page and PingFederate Server are hosted on different origins,
// set this to the base URI of the PingFederate Server to enable a form post.
// For example: "/path/to/fedserver" or "${host}/path/to/fedserver".
signinConfig.federation_resume_uri = null;
```

Replace `{{nnl-version}}` with the supplied Web App SDK version number. Replace `{{tenant-id}}` with the supplied tenant name.

**Note:** `signinConfig.federation_resume_uri` is a base URI, full or relative, because the `nnlsignin` app concatenates dynamic, session specific parameters with this base URI before submitting the form POST. If your sign-in page is hosted on the same origin as the Nok Nok Adapter, leave this null.

3. Update the following context parameters in the `nnlsignin/WEB-INF/web.xml` file:

Parameter name	Description
<code>quick_mode</code>	Specifies quick authentication mode. Refer to the <code>QuickType</code> enumeration in the JavaScript Client API Docs for a list of possible values. The default value is <code>None</code> .
<code>auth_start_mode</code>	Specifies the mode that the authentication should start in. Refer to the <code>AutoStart</code> enumeration in the JavaScript Client API Docs for a list of possible values. The default value is <code>SIGN_IN</code> .
<code>cookie_domain</code>	Domain to set in the authorization cookie before redirecting to Nok Nok Adapter.

**Note:** When you configure inline registration in the Nok Nok Adapter, set `auth_start_mode` to `AUTO_FIDO` if passkey or security key is used for authentication, or set `auth_start_mode` to `AUTO_ANY` if any authentication method is used.

4. **Optional:** You can choose to use your own strings and images, and modify the style of the user interface for the `nnlsignin` app by editing a JSON UI Configuration file. See the *Nok Nok App UI Customization Guide* for more information.

## Implement your own PingFederate Sign-in Page

Once the Nok Nok Adapter is working with the nnlsignin web application, you can choose to implement your own PingFederate sign-in page. Please refer to the `submitLogin()` function in file `Controller.js` of the nnlsignin app to see how to implement FIDO authentication in your page. The `submitLogin()` function also shows how to do a form POST after successful authentication. That JavaScript code makes use of the Nok Nok Javascript App SDK to initiate FIDO authentication.

The Nok Nok Adapter redirects to your PingFederate sign-in page with the following URL query parameters:

URL Query Parameter	Description
<code>goto &lt;resume_url&gt;</code>	If you are hosting the sign-in page on the same origin as the Nok Nok Adapter, then this parameter is required.
<code>resume_path &lt;relative_url&gt;</code>	If you are hosting the sign-in page on an origin that is different from the Nok Nok Adapter, then this parameter is required.
<code>username</code>	Add this parameter only if the user is pre-authenticated, as is the case with a step-up authentication.
<code>tenant_id</code>	Add this parameter only if your Nok Nok deployment has more than one tenant.

Sample URL query if you are hosting the sign-in page on the same origin:

```
https://<same-origin-host-name>/nnlsignin/?goto=https%3A%2F%2F<same-origin-host-name>%3A9031%2Fas%2FPB2TAsJfGf%2Fresume%2Fas%2Fauthorization.ping&tenant_id=default
```

Sample URL query if you are hosting the sign-in page on a different origin:

```
https://<different-origin-sign-in-page-host-name>/nnlsignin/index.jsp?resume_path=%2Fas%2F1QmfS2twMi%2Fresume%2Fas%2Fauthorization.ping&tenant_id=eval
```

If the authentication is successful, the Nok Nok JavaScript App SDK returns a JWT in `sessionData.sessionKey`.

Your Javascript code sends the JWT to the Nok Nok Adapter in one of the following places:

- Authorization header
- Authorization cookie
- form POST

If you are using the `goto` URL query parameter, send the JWT as the Authorization header or cookie. This header or cookie is sent with the request to the PingFederate resume URL. If you are using the `resume_path` URL query parameter, send the JWT as the token parameter.

For more information on how to add FIDO functionality to your sign-in page, refer to the *Nok Nok App SDK for JavaScript Developer Guide*.

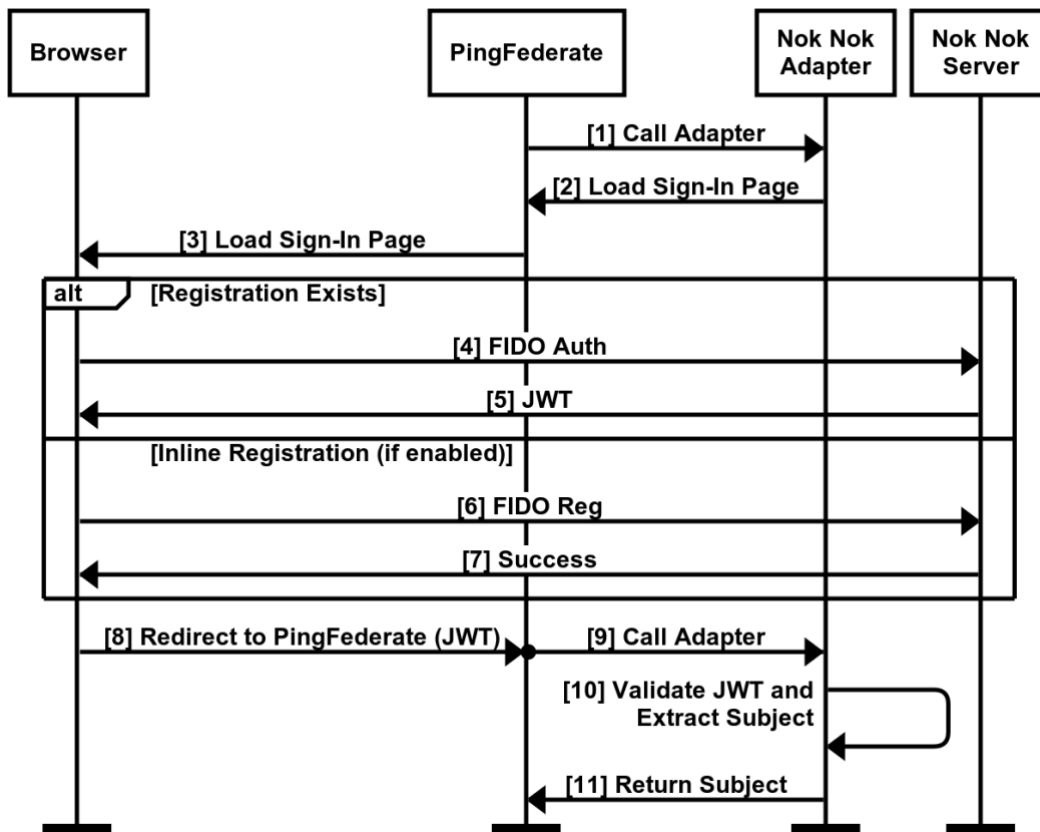
## Initial FIDO Registration through OIDC

Before a user can sign in with FIDO, the user must first authenticate with an external identity provider and then register a FIDO authenticator. The Nok Nok API Server incorporates built-in OIDC client functionality, facilitating access to the FIDO registration page after a user has successfully authenticated with an external identity provider. For comprehensive guidance on configuring External Identity Providers (IdPs), please refer to the *Nok Nok S3 Authentication Suite Configuration Guide*, section *External Identity Provider Configuration* in Appendix B.

# Appendix A: Sequence Diagrams

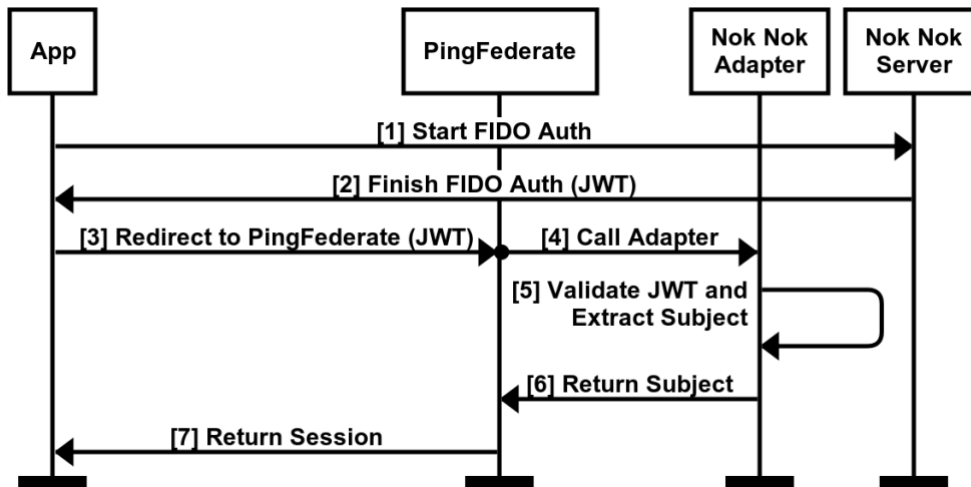
## Web App Sign In Integration

This sequence diagram illustrates the flow for FIDO authentication in a web app.



1. PingFederate calls the Nok Nok Adapter.
2. The Nok Nok Adapter redirects to the sign-in page.
3. PingFederate returns a redirect to the sign-in page
4. If a FIDO registration exists, FIDO Authentication is performed and response sent to Nok Nok Server.
5. The Nok Nok Server returns a JWT.
6. If a FIDO registration does not exist and inline registration is enabled, FIDO registration is performed and the response sent to Nok Nok Server.
7. The Nok Nok Server returns success.
8. Sign in page redirects to PingFederate with the JWT.
9. PingFederate calls the Nok Nok Adapter.
10. The Nok Nok Adapter validates the JWT and extracts the username and any custom attributes from the JWT.
11. This username is returned to the PingFederate.

## Native App Sign In Integration



This sequence diagram illustrates the authentication flow for FIDO authentication in a native app:

1. Using the Nok Nok App SDK, your app sends a "start FIDO authentication" to the Nok Nok Server.
2. The FIDO authentication finishes and the Nok Nok Server returns a JWT.
3. The app redirects to the PingFederate.
4. PingFederate calls the Adapter.
5. The Adapter validates the JWT and extracts the username and any custom attributes from the JWT.
6. This username is returned to PingFederate.
7. PingFederate sends the app the session.

# Appendix B: Upgrading

## Upgrading from Previous Version to October 2024 Version

- In a multi-tenant deployment, the SIGN IN ENDPOINT configuration parameter must now specify the TENANT IDENTIFIER for the target tenant in the Nok Nok Server. In the previous release, the Nok Nok Adapter automatically added the TENANT IDENTIFIER to the SIGN IN ENDPOINT as a URL parameter, but in the current release the customer needs to add the TENANT IDENTIFIER to the SIGN IN ENDPOINT as the URL parameter `tenant_id`. For example, using the PingFederate Admin Console, the customer now configures the SIGN IN ENDPOINT for the `nnlsignin` app as:  
`<SIGN IN ENDPOINT>?tenant_id=<TENANT IDENTIFIER>`
- If your Nok Nok deployment only has one tenant named `default`, you can skip this step because your directory structure has not changed from the previous version. The Nok Nok Adapter now uses the TENANT IDENTIFIER configuration parameter to find the JWT configuration `jwt_config.json` in your directory structure. This enables the Nok Nok Adapter to have different JWT configurations for different tenants. See [Installation](#) section Step 4. for details.