



---

## **LoginService-Public\_2\_0 Web Service Reference**

19.07.2017

## Table of contents

<b>Methods</b> .....	<b>3</b>
getAuth .....	4
<b>Complex Types</b> .....	<b>6</b>
<b>types</b> .....	<b>6</b>
Login .....	6
LoginException .....	7
getAuth .....	8
getAuthResponse .....	8
<b>types_authentication</b> .....	<b>9</b>
<b>anonymous</b> .....	<b>9</b>
authentication .....	9
authenticationFault .....	10
<b>Elements</b> .....	<b>11</b>
<b>types</b> .....	<b>11</b>
LoginException .....	11
getAuth .....	11
getAuthResponse .....	12
<b>types_authentication</b> .....	<b>13</b>
authentication .....	13
authenticationFault .....	13

## Methods

Following methods are available for the service **LoginService-Public\_2\_0**

<i>Method name</i>	<i>Description</i>
getAuth	Creates an authentication token for the committed user if user name and password are valid. The authentication token is needed for accessing other DPD Web Services.

## Method: getAuth

### Description

Creates an authentication token for the committed user if user name and password are valid.  
The authentication token is needed for accessing other DPD Web Services.

Sample request:

SOAP:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="http://dpd.com/common/service/types/LoginService/2.0">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:getAuth>
      <delisId><your delisId></delisId>
      <password><your password></password>
      <messageLanguage>de_DE</messageLanguage>
    </ns:getAuth>
  </soapenv:Body>
</soapenv:Envelope>
```

REST:

URL: <server:port><path>/LoginService/V2\_0/getAuth

JSON-Data for POST-Request (for GET-Request data must send with parameter "request" and for JSONP the Parameter "jsonpcallback" can be used, to send a padding function name):

```
{
  "delisId" : "<your delisId>",
  "password": "<your password>",
  "messageLanguage": "en_EN"
}
```

REST-Result, without error:

```
{
  "getAuthResponse": {"return": {
    "delisId": "<your delisId>",
    "customerUid": "<your customerUid>",
    "authToken": "<your authentication token>",
    "depot": "<your customerUid>"
  }},
  "status": {
    "type": "OK",
    "code": "200",
  }
}
```

```
"message": "valid"  
  }  
}
```

REST-Result, if an error occurred:

```
{"status": {  
  "type": "AuthenticationFault",  
  "code": "<Error code, e.g. LOGIN_8>",  
  "message": "<Error message, e.g. The combination of user and password is invalid.>"  
}}
```

Additional REST-State on system error:

Error-Code: 100

Error-Type: SystemFault

Error-Message: <Message, which describe the error>

### Action

<http://dpd.com/common/service/LoginService/2.0/getAuth>

### Input

The input for method getAuth is as follows:

Name	Element
parameters	types:getAuth

### Output

The output for method getAuth is as follows:

Name	Element
parameters	types:getAuthResponse

### Faults

The defined faults for method getAuth are as follows:

Name	Element
AuthenticationFault	types_authentication:authenticationFault

## Complex Types

### Namespace: types

URL: <http://dpd.com/common/service/types/LoginService/2.0>

Namespace: types

### Complex Type: Login

#### Description

Is created when a user logs in and contains its login information.

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
delisId	xsd:string / 1 .. 1 /	The user's DELIS-Id.
customerUid	xsd:string / 1 .. 1 /	The user's customer uid. This is needed for subaccounts, usually this is equal to DELIS-Id
authToken	xsd:string / 1 .. 1 /	The Authtoken, needed for other web service calls.
depot	xsd:string / 1 .. 1 /	The depot, to which the user is assigned.

Namespace: types

### Complex Type: LoginException

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
additionalData	xsd:string / 0 .. 1 /	Additional data for an error.
additionalInfo	xsd:string / 0 .. 1 /	Additional info for an error in user language.
errorClass	xsd:string / 0 .. 1 /	The class which creates the error.
errorCode	xsd:string / 0 .. 1 /	The error code.
fullMessage	xsd:string / 0 .. 1 /	The complete message text for the error with additional info in user language.
language	xsd:string / 0 .. 1 /	Language for messages.
message	xsd:string / 0 .. 1 /	The complete message text for the error in user language.
shortMessage	xsd:string / 0 .. 1 /	The short info text for the error in user language.
systemFullMessage	xsd:string / 0 .. 1 /	The complete message text for the error with additional info in system language.
systemMessage	xsd:string / 0 .. 1 /	The complete message text for the error in system language.
systemShortMessage	xsd:string / 0 .. 1 /	The short info text for the error in system language.

Namespace: types

**Complex Type: getAuth**

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
delisId	xsd:string / 1 .. 1 /	The DELIS-Id of the user.
password	xsd:string / 1 .. 1 /	The password of the user.
messageLanguage	xsd:string / 1 .. 1 / minLength: 5 maxLength: 5	The language (Java format) for messages. "de_DE" for german messages. "en_US" for english messages.

Namespace: types

**Complex Type: getAuthResponse**

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
return	types:Login / 1 .. 1 /	



**Namespace: types\_authentication**

URL: <http://dpd.com/common/service/types/Authentication/2.0>

Namespace: types\_authentication

**Anonymous Complex Type of Element: authentication**

**Description**

The data structure for authentication data.

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
delisId	xsd:string / 1 .. 1 / minLength: 8 maxLength: 10	The delis user id for authentication.
authToken	xsd:string / 1 .. 1 / minLength: 0 maxLength: 64	The token for authentication. Field authToken of Login, as a result of Method "getAuth" of LoginService.
messageLanguage	xsd:string / 1 .. 1 / minLength: 5 maxLength: 5	The language (Java format) for messages. "de_DE" for german messages. "en_US" for english messages.

Namespace: types\_authentication

### Anonymous Complex Type of Element: authenticationFault

#### Description

The data structure for authentication errors.

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
errorCode	xsd:string / 1 .. 1 /	The error code. Valid codes: NoAccessAuthenticationFault=-1 NoRightsAuthenticationFault=-2
errorMessage	xsd:string / 1 .. 1 / minLength: 1 maxLength: 255	The error message text.

## Elements

### Namespace: types

URL: <http://dpd.com/common/service/types/LoginService/2.0>

Namespace: types

### Element: LoginException

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
LoginException	types:LoginException / 1 .. 1 /	Thrown if authentication fails.

Namespace: types

### Element: getAuth

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
getAuth	types:getAuth / 1 .. 1 /	Input parameter of getAuth.

Namespace: types

**Element: getAuthResponse**

<i>Field name</i>	<i>Data type / Occurrence / Range</i>	<i>Description</i>
getAuthResponse	types:getAuthResponse / 1 .. 1 /	Output parameter of getAuth.

**Namespace: types\_authentication**

URL: <http://dpd.com/common/service/types/Authentication/2.0>

Namespace: types\_authentication

**Element: authentication**

Anonymous Complex Type *types\_authentication:authentication*

Namespace: types\_authentication

**Element: authenticationFault**

Anonymous Complex Type *types\_authentication:authenticationFault*