



Remote authentication instructions

Last Modified on 12/23/2024 2:42 pm EST

Remote authentication allows you to authenticate readers using your existing systems, usually a 3rd party website or application. This option is best for companies that want to eliminate the need for their readers to manage multiple passwords but do not have the ability to use SAML.

Remote Authentication Overview

Here's how remote authentication works:

- A reader tries to access your knowledge base.
- If they are not already authenticated, the reader is redirected to your specified remote authentication landing page URL.
- Your landing page authenticates the reader in your system.
- After authentication by your system, an API call is made to KnowledgeOwl's remote authentication token request endpoint. If you are utilizing readers and reader groups, that information is passed during this token request.
- Once the login token is received by your system, the reader is redirected to the KnowledgeOwl remote authentication endpoint with this single use token which completes the handshake and the reader gains access.

Prerequisites

Before you enable remote authentication, you'll need three things:

- Appropriate permissions
- The URLs for login/logout of your remote authentication system
- An API key

Appropriate permission

You'll need to be logged in as an author who has one of these two permissions:

- The default Editor role
- A [custom author role](#) with the Update security settings permission

Remote authentication URLs

You'll need to know:

- The login URL for your remote authentication: the URL a reader should visit to enter their credentials
- The logout URL for your remote authentication: the URL a reader should be directed to once they've logged out

API Key

You'll also need an API key with GET permissions. Something like this will work:

API Key Creation ✕

Purpose for this key:

Only alphanumeric characters, spaces, and - _ symbols allowed

Allowed actions for this key:

GET

PUT

POST

DELETE

[Cancel](#)

Refer to [API keys](#) for full steps on creating a new API key.



Use a unique API key

We recommend creating a new API key restricted to GET calls for this type of authentication, and using this key only for this authentication.

Enable remote authentication

Before you can set up the full authentication process, you'll need to enable remote authentication. To do so:

1. Go to **Security and access > Security settings**.
2. In the **Authentication settings** section, under **Content authentication**, select **Remote authentication**.
3. If it won't disrupt current login processes, we also recommend setting the **Unauthenticated access behavior** to **Redirect them to your remote auth login URL**.
4. **Save your changes**.
5. Now go to **Security and access > Single sign-on**.
6. Open the **Remote authentication** tab.

7. Add the URL that readers should be directed to to login to your remote authentication system into the **Remote login URL** field.
8. Add the URL that readers should be directed to once they've logged out of your knowledge base into the **Remote logout URL** field.
9. **Save your changes.**

Create your remote auth token request and script


With remote authentication enabled, configure the calls and script to request access tokens and authenticate your readers.

Step 1: Request token

First, you'll need to hit the KnowledgeOwl API `remotelogin` endpoint to request a token. Use the API key you generated previously to make this call. At minimum, you must pass in the knowledge base's project ID and the reader's username.

Copy your knowledge base ID from **Security and access > Single sign-on > Remote authentication**.

Here are the details on the API endpoint:




Endpoint URL
<https://app.knowledgeowl.com/api/head/remotelogin.json>

Parameters	Example	Required?
project_id	project_id=123a123456789b1234aab1cde	Yes
reader[username]	reader[username]=support@knowledgeowl.com	Yes
reader[ssoid]	reader[ssoid]=12345	No, defaults to username if not specified
reader[groups]	reader[groups]=Internal,Admin	No
reader[first_name]	reader[first_name]=KnowledgeOwl	No
reader[last_name]	reader[last_name]=Support	No
reader[custom1]	reader[custom1]=Red	No
reader[custom2]	reader[custom2]=Orange	No
reader[custom3]	reader[custom3]=Yellow	No
reader[custom4]	reader[custom4]=Green	No

Parameters	Example	Required?
reader[custom5]	reader[custom5]=Blue	No

Step 2: Authenticate

You'll need to use the token generated in Step 1 to authenticate your reader. This endpoint is specific to your knowledge base's URL:



Customize this endpoint URL
[https://support.knowledgeowl.com/help/remote-auth*](https://support.knowledgeowl.com/help/remote-auth)

* Replace "support.knowledgeowl.com" with your KO site URL. For example, if my KO site URL is "myawesomekb.knowledgeowl.com", I'd use `https://myawesomekb.knowledgeowl.com/help/remote-auth`.

If you're unsure of your KO site URL, go to **KB settings > Domain**. Use either your **KnowledgeOwl Sub-domain** or your **Private domain** (if applicable).

Parameters	Example	Required?
n (token received in step 1)	n=1234567890	Yes
r (redirect)	r=/help/contact-us	No, will redirect to the homepage if not specified

Example cURL access token request

Here is an example cURL access token request:

```
//basic authentication without reader information
curl -G -X GET -u {API KEY}:X https://app.knowledgeowl.com/api/head/remotelogin.json -d 'project={Knowledge Base ID}'

//authentication with reader information passed
curl -G -X GET -u {API KEY}:X https://app.knowledgeowl.com/api/head/remotelogin.json -d 'project={Knowledge Base ID}&reader[ssoid]={Unique Reader ID}&reader[username]={Reader Email or Name}'

//example response
{"valid":true,"data":[{"status":"success","token":"12345678"}]}
```

Reader fields can be found in our [API endpoint reference](#).

Example PHP script to run after authentication

Here is an example PHP script to run after authentication:

```
function remoteAuth()
{
    //shared vars
```

```

/* if a non logged in user attempts to go directly to an article,
our system will redirect them back to your landing page with the
URL variable "r" that equals the original request URI. After
authentication in your system, use this variable to send them
to the originally requested page. */
//grab the value of r
$redirect = $this->_request->getParam('r'); //original request uri
$callurl = "https://app.knowledgeowl.com/api/head/remotelogin.json"; //KnowledgeOwl auth endpoint
$data = array(
    'project' => '{KB ID}', //knowledge base ID
);

//optional data for creating / asserting reader object
$data['reader'] = array(
    'ssoid' => {READER UNIQUE ID} //optional, if not passed in username will be used as ssoid
    'username' => {READER EMAIL OR NAME}, //required
    //'groups' => {KNOWLEDGEOWL GROUP NAMES}, //optional - comma separated list ie - "Group1,Group2"
    //'first_name' => {READER FIRST NAME}, //optional
    //'last_name' => {READER LAST NAME}, //optional
    //'custom1' => {STRING}, //optional
    //'custom2' => {STRING}, //optional
    //'custom3' => {STRING}, //optional
    //'custom4' => {STRING}, //optional
    //'custom5' => {STRING}, //optional
);

//set curl options
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $callurl);
curl_setopt($ch, CURLOPT_USERPWD, '{API Key}':X);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_FOLLOWLOCATION,1);
curl_setopt($ch, CURLOPT_VERBOSE, 0);
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($data));
curl_setopt($ch, CURLOPT_NOPROGRESS, 1);
curl_setopt($ch, CURLOPT_TIMEOUT, 30);

$buffer = curl_exec($ch);

//verify return
if(curl_errno($ch))
{
    //handle curl errors
    print_r("Error: " . curl_error($ch));
}
else
{
    //decode and verify
    $data = json_decode($buffer);
    curl_close($ch);

    //validate response
    if($data->valid && $data->data[0]->status == 'success')
    {
        //grab the token and redirect
        $token = $data->data[0]->token;
    }
}

```



```

Dim resp As AuthTokenResponseObj = JsonConvert.DeserializeObject(Of AuthTokenResponseObj)(strResponse)

For Each atrObj As AuthData In resp.data
    strToken = atrObj.token
Next

myreader.Close()

Dim strURI As String = "http://{yourknowledgeowldomain.com}/help/remote-auth?n=" & strToken
System.Diagnostics.Process.Start(strURI)

End If

End Sub

Public Class AuthTokenResponseObj

    Public Property valid As String
    Public Property data As List(Of AuthData)

End Class

Public Class AuthData

    Public Property status As String
    Public Property token As String

End Class

```



If you have trouble with Newtonsoft json deserializer you can always parse the token out of the response manually i.e.; `strToken = strResponse.Substring(strResponse.IndexOf("token") + 8, 8)`

Javascript example script

```

//only need this if you do not already have jquery loaded
<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
<script>
/**
 * This example assumes that you have already authenticated
 * the end user in your system
 */

/* if a non logged in user attempts to go directly to an article,
our system will redirect them back to your landing page with the
URL variable "r" that equals the original request URI. After
authentication in your system, use this variable to send them
to the originally requested page. */
//grab the value of r
var redirect = "/help"; //default homepage
var pageURL = window.location.search.substring(1);
var urlVariables = pageURL.split('?');

```

```

for (var i = 0; i < urlVariables.length; i++)
{
var parameterName = urlVariables[i].split('=');
if (parameterName[0] == 'r')
{
redirect = parameterName[1];
}
}

//static vars
var token;
var kbEndpoint = "https://{YOUR KNOWLEDGEOWL KB URL}/help/remote-auth"; //where we will be sending them back --
eg https://yoursite.knowledgeowl.com/help/remote-auth
var callURL = "https://app.knowledgeowl.com/api/head/remotelogin.jsonp"; //knowledgeowl auth endpoint
var data = {
_authbykey: '{API KEY}', //KnowledgeOwl API key
project : '{KB ID}' //knowledge base ID
}

//request the token and redirect to desired page
$.ajax({
type: 'GET',
url: callURL,
dataType: 'jsonp',
data: data
}).success(function(data) {
//check for valid response
if(data && data.valid === true) {
//grab the token
token = data.data[0].token;

kbEndpoint += "?n=" + token;
kbEndpoint += "&r=" + redirect;

//optional reader object creation / assertion
kbEndpoint += "&reader[ssoid]=" + {READER UNIQUE ID}; //optional, if no ssoid is passed username will be used
kbEndpoint += "&reader[username]=" + {READER NAME OR EMAIL}; //required
kbEndpoint += "&reader[icon]=" + {READER PICTURE}; //optional
kbEndpoint += "&reader[first_name]=" + {READER FIRST NAME}; //optional
kbEndpoint += "&reader[last_name]=" + {READER LAST NAME}; //optional
kbEndpoint += "&reader[groups]=" + {KNOWLEDGEOWL GROUP NAMES}; //optional, comma seperated list
kbEndpoint += "&reader[custom1]=" + {STRING}; //optional
kbEndpoint += "&reader[custom2]=" + {STRING}; //optional
kbEndpoint += "&reader[custom3]=" + {STRING}; //optional
kbEndpoint += "&reader[custom4]=" + {STRING}; //optional
kbEndpoint += "&reader[custom5]=" + {STRING}; //optional

//redirect user back with the aquired token and initial page request URI
window.location.replace(kbEndpoint);
} else {
//handle empty return error
}
}).error(function(data){
//handle ajax errors
});
</script>

```