Remote authentication instructions

Last modified on 11/07/2026 12:01 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.

This feature is available on select plans. Remote authentication should be available at the **Pro** plan level. If you're on that plan and having trouble adding remote authentication, please contact us for help.

Remote authentication 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

This feature is available on select plans.

Appropriate permission

You'll need to be logged in as an author who has **Full Admin** rights with one of these two permissions:

- The default Editor role
- A custom author role with the **Update security settings permission**

Remote authentication URLs

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

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 a unique API key used only for this type of authentication. Refer to API keys for full steps on creating a new API key.

Use these settings to create your key:

- **Key purpose:** Add something about this being for remote authentication for the selected knowledge base.
- **Knowledge base access:** Select the knowledge base you're setting up remote authentication for.
- **Resource/action:** Select the following resources and actions:
  - **Remote login:** Read
  - **Remote logout**: Read

Your key setup should look something like this:



Use a unique API key
For security reasons, use this API key only for remote auth reader login/reader logout. Don't combine it with other actions or integrations.

Enable remote authentication

Before you can set up the full authentication process, you'll need to enable remote authentication. This feature is available on select plans. To do so:

1. Go to **Security and access** > **Access & SSO**.
2. In the **Login methods** section, find **Remote authentication** and select **Set up** (or **Manage**, if you've already started). The **Remote authentication** page opens.
3. In step 1, **Connect your app**, add the URL where readers log in to your remote authentication system into the **Remote login URL** field.
4. *Optional:* Add the URL that readers should be directed to once they've logged out of your knowledge base into the **Remote logout URL** field.
5. Select **Save changes**.
6. In step 2, **Turn on remote authentication**, select **Go to who can read**. You're returned to the **Access & SSO** overview.
7. In the **Who can read this knowledge base?** section, select **Your own login system (remote authentication)**.

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

8. Select **Save changes.**

Readers who aren't logged in are now sent to your **Remote login URL.** While remote authentication is required, **Send readers who aren't logged in to** is set to **Remote login URL** for you.

The Remote authentication page, step 1 Connect your app

Only for restricted content
If your knowledge base is public and you only want to send readers to your remote login URL when they open restricted content, leave **Who can read this knowledge base?** set to **Public** and set **Send readers who aren't logged in to** to **Remote login URL** instead.

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 step 1, **Connect your app**, in **Security and access > Access & SSO > 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

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

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



Token limitations
Remote auth tokens are single-use and expire after 60 seconds.

Step 2: Authenticate

Use the token generated in Step 1 to authenticate your reader. This endpoint is specific to your knowledge base's URL. To generate it:

1. Copy the endpoint below.

```
https://{my-kb.knowledgeowl.com}/help/remote-auth
```

2. Replace `{my-kb.knowledgeowl.com}` with the URL of your knowledge base.
3. For example, if my knowledge base URL is `myawesomekb.knowledgeowl.com`, I'd use `https://myawesomekb.knowledgeowl.com/help/remote-auth` as my endpoint.

If you're unsure of your knowledge base 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, but it will redirect to the homepage if not specified

Example cURL access token request

Here is an example cURL access token request:

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

```
//basic authentication without reader information
curl -G -X GET -u {API KEY}:X https://app.knowledgeowl.com/api/head/remotelogin.json -d 'project_id={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_id={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_id' => '{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));
```

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

```
    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;

            $authData = array(
                        'n' => $token,
                        'r' => $redirect
            );

            header("Location: {Your KO Site URL}/help/remote-auth?" . http_build_query($authData));
            exit();
        }
        else
        {
            //handle api errors
        }
    }
}
```

Example vb.net logic to get the token and open the default web browser passing the auth token

This example uses the NewtonSoft package in dot net to deserialize the JSON response. It was installed in Visual Studio using NuGet.

```
Imports System.Net
Imports Newtonsoft.Json

Private Sub GoToHelp()

        Dim request As HttpWebRequest
        Dim response As HttpWebResponse
        Dim strAUTH As String = "123abcd123:X"   ‘ replace 123abcd123 with your API key
        Dim strKBProjectKey As String = "99999"  ‘ replace 99999 with actual value
        Dim strDomainUsername As String = ""

‘Reader Groups to assign (optional)
```

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

```
Reader Groups to assign (optional)
        Dim strGroups As String = "Group 1,Group 2"

        strDomainUsername = Environment.UserDomainName & "\" & Environment.UserName

        If strDomainUsername <> "" Then
            request = DirectCast(HttpWebRequest.Create("https://app.knowledgeowl.com/api/head/remotelogin.json?project_id=" & strKBProjectKey & "&reader[ssoid]=" &
    strDomainUsername & "&reader[username]=" & strDomainUsername & "&reader[groups]=" & strGroups & "&reader[first_name]=" & strDomainUsername), HttpWebRequest)

            request.Method = "GET"
            request.ContentType = "application/json; charset=utf-8"
            request.Headers.Add("Authorization", "Basic " & Convert.ToBase64String(System.Text.Encoding.UTF8.GetBytes(strAUTH)))


            response = DirectCast(request.GetResponse, HttpWebResponse)

            Dim myreader As New IO.StreamReader(response.GetResponseStream)
            Dim strResponse As String = myreader.ReadToEnd
            Dim strToken As String = ""

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

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

    myreader.Close()

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


        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
```



NewtonSoft json deserializer issues

If you have trouble with NewtonSoft json deserializer you can always parse the token out of the

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

response manually for example: `strToken = strResponse.Substring(strResponse.IndexOf("token") + 8, 8)`

Javascript example script

Here is an example Javascript script to run after authentication:

```
//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_id : '{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;
```

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

```
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 acquired token and initial page request URI
window.location.replace(kbEndpoint);
} else {
//handle empty return error
}
}).error(function(data){
//handle ajax errors
});
</script>
```

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.