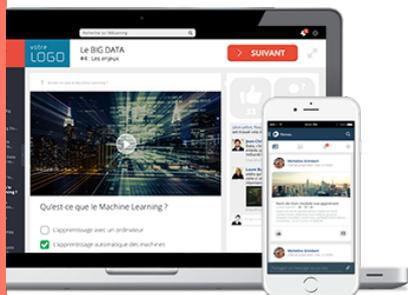


TECHNICAL GUIDE

SSO JWT

At 360Learning, we don't make promises about technical solutions, we make commitments.

This technical guide is part of our Technical Documentation.



360Learning is a Leading European Cloud Solution Provider

"As a leading Cloud Solution Provider, we are strongly committed to providing our clients with high levels of security, SLAs and privacy, both in the contractual engagements we make and the technical infrastructure we build. We comply with French Laws requirements which are the most restrictive in terms of Data, Security & Privacy."

Nicolas Hernandez
CEO, 360Learning

For more information,
please contact us:

product@360learning.com



Dior

HEC
PARIS



SciencesPo
EXECUTIVE EDUCATION

HAVAS
M E D I A

StanleyBlack&Decker

ESSEC
BUSINESS SCHOOL



Capgemini
CONSULTING. TECHNOLOGY. OUTSOURCING

BNP PARIBAS
La banque d'un monde qui change

Table of contents

Introduction	4
Process	4
Standard process	4
SSO Process.....	4
Configuration	4
Setup.....	5
JWT Anatomy.....	5
JWT Resources	6
Route parameter	7
Example	7

Introduction

Single Sign-On (SSO) is an authentication mechanism that allows users to access several applications with only one set of login credentials.

By enabling SSO for your 360Learning application, you become responsible for the authentication of your users: they get authenticated through your own login portal and do not need an additional set of login / password anymore.

Communication between your authentication system and 360Learning can be handled by several technologies including JWT (JSON Web Token).

It is a fairly recent standard but there are a lot of open-source implementations of this specification. You can probably find one that suits your needs!

Process

While using SSO, the process involved is slightly different and requires several browser redirections and message exchanges using JWT.

Standard process

1. An unauthenticated user requests access to your 360Learning space.
2. The user is redirected to the login page of your 360Learning space where he can provide his login and password.
3. 360Learning grants him permission and redirects him to your space.

SSO Process

1. An unauthenticated user requests access to your 360Learning space.
2. 360Learning redirects him to your own login URL.
3. The user gets authenticated using your own authentication process.
4. You create a secured JWT token that contains information about the user.
5. You redirect him to our JWT endpoint with the JWT token
6. 360Learning analyzes the token, grants the user permission and redirects him to your 360Learning space.

Configuration

Setup

To enable SSO for your 360Learning space, contact your designated account manager: he will guide you through the entire integration process.

What you will get from 360Learning:

- Your company ID
- A shared secret that will be used to generate tokens and secure communication between your system and the 360Learning platform.

What you will have to provide:

- The URL of your login portal, for example `https://mycompany.com/sso`. Your portal must comply with the requirements defined in the following sections.
- Test credentials.

JWT Anatomy

Once the user has been authenticated on your side, you must create a token containing the required information about the user.

The token must be appended to the return URL as a query string:
`https://mycompany.360learning.com/?jwt={token}`

The token contains:

- A header:

```
{
  "typ": "JWT",
  "alg": "HS256"
}
```

- HS256 (HMAC SHA 256) is the hashing algorithm that is used to generate the signature.
- This JSON must be base64-encoded.
- A payload containing the following fields:
 - Mandatory
 - An attribute "iat" (issued at).
 - An attribute "jti" (JSON Web Token id).
 - An attribute "email" (the user's email).

- Optional (if the user's account does not exist on your 360Learning space, the following information is used to create the account):
 - An attribute "firstName".
 - An attribute "lastName".
 - An attribute "phone".
 - An attribute "lang". Possible values: en, fr
 - An attribute "role". Possible values: learner, author or admin
 - An attribute "job".
 - An attribute "organization".
 - An attribute "custom".
 - An attribute "keywords". e.g ["label1", "label2", "label3"]
 - This JSON must be base64-encoded.
- A signature: hash of the base64-encoded header and the base64-encoded payload using:
 - The hashing algorithm defined in the header
 - The shared secret provided by 360Learning

```
HMACSHA256(  
  base64UrlEncode(header) + "." +  
  base64UrlEncode(payload),  
  secret)
```

JWT Resources

JWT's full specification is available at <http://tools.ietf.org/html/draft-jones-json-web-token-10> and <https://jwt.io/introduction> provides a more hands on approach to this concept.

As mentioned above, several open-source libraries are available to generate JSON Web Tokens. For example <https://github.com/jwt/ruby-jwt> for Ruby, or <https://github.com/firebase/php-jwt> for PHP or <https://github.com/auth0/node-jsonwebtoken> for Node.js.

Route parameter

While redirecting the user to the 360Learning endpoint, you can also provide a route parameter in the URL, for example the route of a specific group or a specific program. After the authentication, 360Learning will redirect the user to this page.

- Output
`https://mycompany.360learning.com/?company={company_id}&jwt={token}&route={a_specific_route}`
- The user will be redirected to
`https://mycompany.360learning.com/{a_specific_route}`

Example

Node.js sample code, leveraging the node-jsonwebtoken module (tested with version 7.3.0)

```
var router = require("express").Router();
var jwt = require("jsonwebtoken");

router.get("/", function (req, res, next) {

    // Verify in the database that the login / password is correct,
    // or equivalent
    // Get the user data from the database (email, firstName,
    // lastName, phone, role)

    var cid = "4e54273d5d17859d464cb9bc", // Get the 360Learning
    company id from a config file, or equivalent
    key = "oB9eNDdAp2mXpw996gdunxel"; // Get the secret key from a
    config file, or equivalent

    var token = jwt.sign({
        email: "test.user@360learning.com", // Mandatory
        firstName: "Test", // Optional, only used if the account
        does not exists on 360Learning yet
        lastName: "Name", // Idem
        phone: "0123456789", // Idem
        lang: "en", // Idem, possible values: [ en, fr ], default:
        en
        role: "trainee" // Idem, possible values: [ admin, user,
        trainee ], default: trainee
    }, key, { expiresIn: 60 });

    res.redirect("https://app.360learning.com/?company=" + cid +
    "&jwt=" + token + "&route=" +
    encodeURIComponent(req.query.route));

});
```

```
exports.router = router;
```