**统一身份认证平台**

**集成标准接口规范**

修订记录

| 版本 | 日期 | 作者 | 说明 |
| --- | --- | --- | --- |
| 1.0 | 2023-03-29 |  | 新建 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

[第1章 文档概述 6](#_Toc20819)

[1.1. 范围 6](#_Toc14528)

[1.2. 目的 6](#_Toc22731)

[1.3. 读者 6](#_Toc30501)

[1.4. 定义、术语和缩略语 6](#_Toc23638)

[第2章 应用集成IAM系统准备 7](#_Toc30966)

[2.1. 周边应用准备 7](#_Toc22441)

[2.2. SSO系统准备 7](#_Toc4682)

[第3章 单点认证协议集成 8](#_Toc16620)

[3.1.  OAuth2.0认证集成规范 9](#_Toc12646)

[3.1.1. OAuth2.0认证过程 9](#_Toc25351)

[3.1.2. 接口规范 9](#_Toc7270)

[3.2.  OIDC认证集成规范 15](#_Toc16179)

[3.2.1. OIDC认证过程 15](#_Toc32003)

[3.2.2. 接口规范 16](#_Toc22554)

[3.3. SAML2.0认证集成规范 22](#_Toc13230)

[3.3.1. SAML2.0认证过程 22](#_Toc6936)

[3.3.2. SAML集成规范 24](#_Toc29784)

[3.4. CAS协议认证集成规范 26](#_Toc84)

[3.4.1. CAS认证过程 26](#_Toc666)

[3.4.2. 请求URL实例及参数说明 27](#_Toc201)

# 文档概述

## 范围

本文档包括统一身份认证平台与接入到统一身份认证平台的应用系统在对接过程中所使用的接口与规范的设计。

## 目的

本文档详细定义了统一身份认证平台提供给下游应用系统的被推服务标准接口规范。当下游应用系统在与统一身份认证平台进行数据同步集成时，需要了解该文档的内容，并根据该文档定义的接口规范进行相应接口的开发或调整。

## 读者

本文档读者为本项目所有成员及集成至统一身份认证平台的应用方所有项目成员、项目经理以及统一身份认证平台架构师。

## 定义、术语和缩略语

本文档读者为本项目所有成员及集成至统一身份认证平台的应用方所有项目成员、项目经理以及统一身份认证平台架构师。

| **名词/术语** | **英文全称** | **描述** |
| --- | --- | --- |
| SSO | Single Sign-On | 单点登录，一次登录访问所有应用 |
| IAM | Identity and Access Management | 身份与访问控制 |
| 被推 |  | 统一身份认证平台提供接口给应用系统调用同步数据 |

# 应用集成IAM系统准备

## 周边应用准备

1. 应用系统应先向IAM系统申请注册应用，注册时需要提供应用名称、应用图标、应用访问地址、回调地址（接入Oauth2.0、OIDC协议认证的应用需要提供回调地址，用于接收认证成功后code验证码的地址及认证成功跳转地址）
2. 如果使用SAML2.0协议与IAM系统集成，则需要向IAM系统提供 **SP Metadata文件**

## SSO系统准备

1. 根据应用提供的数据及同步范围开通应用的身份数据同步功能
2. 应用如果使用Oauth2.0、OIDC协议与IAM系统集成单点认证，需要反馈单点认证凭证参数

|  |  |
| --- | --- |
| **属性名** | **说明** |
| client\_id | 1001 应用注册ID |
| client\_secret | 1001 应用注册密码 |



1. 应用如果使用SAML2.0协议与IAM系统集成单点认证，需要反馈IDP Metadata文件给第三方应用系统



# 单点认证协议集成

IAM系统支持Oauth2.0、SAML2.0、OIDC、CAS四种标准协议的对接支持，以下是四种协议集成规范

##  OAuth2.0认证集成规范

### OAuth2.0认证过程



1. 用户访问应用系统地址浏览器重定向请求到SSO的认证地址，并且提供相应的参.
2. OAuth完成用户认证，并为app提供token code。
3. 浏览器跳转到app并提供code参数。
4. 在获取code后，应用通过oauth2.0/accessToken用code换取访问token。
5. 用户在获取访问token后，通过调用接口获取认证用户信息。
6. 应用获取用户信息，认证完成，用户正常访问app。

### 接口规范

#### 集成应用向IAM系统发起请求认证

**接口名称：** OAuth2.0协议请求授权接口

**请求类型：** GET

**请求Url :** <https://cas.dhu.edu.cn/esc-sso/oauth2.0/authorize>

历史对接应用访问的地址已做适配，可以保留下面的地址不改动：

https://cas.dhu.edu.cn/authserver/oauth2.0/authorize

**接口描述:**  该接口用于第三方系统向IAM系统请求授权登录。在用户访问第三方应用系统时，第三方应用未登录状态下，按下方示例方式拼接地址及参数，并通过客户端浏览器重定向至IAM系统发起请求认证授权，授权成功后，IAM系统会重定向至redirect\_uri传递的地址，并携带code

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **说明** | **是否必填** |
| client\_id | 应用注册ID | 必须 |
| response\_type=code | 属性常量 | 必须 |
| redirect\_uri | 应用回调地址，需要进行encodeURI转义传输，必须与注册时提供的一致 | 必须 |
| state | 额外参数，如果传递url或其他特殊字符（如：#）需要转义 | 可选 |

**示例请求：**

https://cas.dhu.edu.cn/esc-sso/oauth2.0/authorize?client\_id=XXXX&response\_type=code&redirect\_uri=http%3A%2F%2FXXX.XXXX.XXXX.XXX%3A8080%2FXXXX%2FXXXX&state=http%3A%2F%[2Fwww.app1.com](http://2Fwww.app1.com)%2Ftodu%2F1w2341123

**请求授权成功后的重定向回调示例：**

<http://XXXX.XXXX.XXXX.XXXX/XXX/XXX?code=123456abcdef>&state=http%3A%2F%[2Fwww.app1.com](http://2fwww.app1.com/)%2Ftodu%2F1w2341123

**重定向回调携带参数**

|  |  |  |
| --- | --- | --- |
| **属性名称** | **说明** | **是否必须** |
| code | 交互token验证码 | 必须 |
| state | 额外参数（如果authorize接口传递了参数，这里就会返回） | 非必须 |

#### code获取access\_token接口

**接口名称：** OAuth2.0协议使用code获取token接口

**请求类型：** POST

**请求Url :**  <https://cas.dhu.edu.cn/esc-sso/oauth2.0/accessToken>

历史对接应用访问的地址已做适配，可以保留下面的地址不改动：

https://cas.dhu.edu.cn/authserver/oauth2.0/accessToken

**接口描述:**  用于第三方系统，在OAuth2.0单点认证流程第一步完成，获取到code后，在第三方服务端使用code获取token

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **是否必填** |
| grant\_type=authorization\_code | 属性常量 | 必须 |
| client\_id | 应用注册ID | 必须 |
| client\_secret | 应用注册密码 | 必须 |
| code | 接收到的code值 | 必须 |
| redirect\_uri | 应用回调地址,需要进行encodeURI转义传输 | 必须 |

**请求示例：**

[https://cas.dhu.edu.cn/esc-sso/oauth2.0/accessToken?grant\_type=authorization\_code&client\_id=1001&client\_secret=1001&code=b87576e1-fd14-47e5-a835-0d601b18387a&redirect\_uri=http%3A%2F%2FXX.XXX.XXX.XXX%3A8080%2FXXX%2FXXXX](http://10.0.10.8/esc-sso/oauth2.0/accessToken?grant_type=authorization_code&oauth_timestamp=1489739169232&client_id=1001&client_secret=1001&code=b87576e1-fd14-47e5-a835-0d601b18387a&redirect_uri=http://XX.XXX.XXX.XXX:8080/XXX/XXXX)

**返回参数列表：**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **类型** |
| access\_token | 返回的token值（AT-13-D61WfBwwEv1ntL0oop1k7uIlbeL2SKgmla1）,用于下一步的获取当前登录账号 | 必须 |
| refresh\_token | 刷新token(在应用配置里，打开【token刷新】的开关，即可返回该参数) | 非必须 |
| token\_type | 固定返回参数，无需理会 | 必须 |
| expires\_in | token有效期，单位：秒 | 必须 |

**返回示例：**

{

    "access\_token": "AT-13-D61WfBwwEv1ntL0oop1k7uIlbeL2SKgmla1",//access\_token的值

    "token\_type": "bearer",//固定返回参数

    "expires\_in": 7200,//token有效期，单位：秒

    "refresh\_token": "RT-1-yBEDcdcs6sjxXcTOmAZgyZNryWIzBEWYHfY"//刷新token，根据需要开启，不一定返回

}

#### access\_token获取SSO(二期)当前登录账号接口

**接口名称：** OAuth2.0协议使用token获取登录用户信息接口

**请求类型：** GET

**请求Url :** <https://cas.dhu.edu.cn/esc-sso/oauth2.0/profile>

历史对接应用访问的地址已做适配，可以保留下面的地址不改动：

<https://cas.dhu.edu.cn/authserver/oauthApi/user/profile>

**接口描述:**  用于第三方系统使用token获取当前登录用户信息

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **是否必填** |
| access\_token | 获取到的access\_token的值 | 必须 |

**请求示例：**

 [https://cas.dhu.edu.cn/esc-sso/oauth2.0/profile?access\_token=AT-13-D61WfBwwEv1ntL0oop1k7uIlbeL2SKgmla1](http://10.0.10.8/esc-sso/oauth2.0/profile?access_token=b1d508f3-32e7-4d6d-ac62-87836406704c)

**返回参数列表：**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **类型** |
| id | 用户登录的主账号 | 必须 |
| attributes->email | 邮箱地址 | 非必须 |
| attributes->mobile | 手机号 | 非必须 |
| attributes->username | 姓名 | 非必须 |

**返回数据示例**

{

    "attributes": {

        "email": "800032@paraivew.com",

        "work\_no": "800032",//员工编号

        "mobile": "182XXXXXXXXXX",//手机号

        "id": "1839341111974344339",//用户ID，IAM系统内用户唯一标识编码

        "account\_no": "chenp",//第三方应用系统账号，一般与AD登录账号一致，但可提供兼容，既SSO允许该值与登录电脑、SAD登录的账号不一致

        "user\_name": "陈平",//姓名

        "token\_gtime": 1705484354333,

        "token\_expired": "7200"

    },

    "id": "chenp"//AD登录账号、SSO登录账号

}

#### SSO退出地址

**当系统需要退出时，可调用以下地址：**

对接系统退出自身系统后需要在浏览器地址栏中，重定向到https://cas.dhu.edu.cn/esc-sso/logout?service=http://xxxxx

其中http://xxxxx是应用自身的地址，用户退出后再次登录会跳转到该http://xxxxx地址

##  OIDC认证集成规范

### OIDC认证过程



1. 用户访问应用系统地址浏览器重定向请求到SSO的认证地址，并且提供相应的参.
2. SSO完成用户认证，并为app提供token code。
3. 浏览器跳转到app并提供code参数。
4. 在获取code后，应用通过oidc/accessToken接口用code换取访问access token和id token。
5. 用户在获取access token和id token后，有2种方式获取用户信息，一种是用AccessToken去服务端换取用户信息，一种是直接解析IDToken获取用户信息。应用获取用户信息，认证完成，用户正常访问app。

### 接口规范

#### 集成应用向IAM系统发起请求认证

**接口名称：** OAuth2.0协议请求授权接口

**请求类型：** GET

**请求Url :** https://cas.dhu.edu.cn/esc-sso/oidc/authorize

**接口描述:** 该接口用于第三方系统向IAM系统请求授权登录。在用户访问第三方应用系统时，第三方应用未登录状态下，按下方示例方式拼接地址及参数，并通过客户端浏览器重定向至IAM系统发起请求认证授权，授权成功后，IAM系统会重定向至redirect\_uri传递的地址，并携带code

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **说明** | **是否必填** |
| client\_id | 应用注册ID | 必须 |
| response\_type=code | 属性常量 | 必须 |
| redirect\_uri | 应用回调地址，需要进行encodeURI转义传输，必须与注册时提供的一致 | 必须 |
| oauth\_timestamp | 当前时间格式(1489739502583) | 可选 |

**示例请求：**

https://cas.dhu.edu.cn/esc-sso/oidc/authorize?client\_id=XXXX&response\_type=code&redirect\_uri=http%3A%2F%2FXXX.XXXX.XXXX.XXX%3A8080%2FXXXX%2FXXXX&oauth\_timestamp=1489739502583

**请求授权成功后的重定向回调示例：**

<http://XXXX.XXXX.XXXX.XXXX/XXX/XXX?code=123456abcdef>

**重定向回调携带参数**

|  |  |  |
| --- | --- | --- |
| **属性名称** | **说明** | **是否必须** |
| code | 交互token验证码 | 必须 |

#### code获取access\_token接口

**接口名称：** OIDC协议使用code获取token接口

**请求类型：** POST

**请求Url :** https://cas.dhu.edu.cn/esc-sso/oidc/accessToken

**接口描述:** 用于第三方系统，在OIDC单点认证流程第一步完成，获取到code后，在第三方服务端使用code获取token

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **是否必填** |
| grant\_type=authorization\_code | 属性常量 | 必须 |
| client\_id | 应用注册ID | 必须 |
| client\_secret | 应用注册密码 | 必须 |
| code | 接收到的code值 | 必须 |
| redirect\_uri | 应用回调地址,需要进行encodeURI转义传输 | 必须 |
| oauth\_timestamp | 当前时间格式(1489739502583) | 可选 |

**请求示例：**

[https://cas.dhu.edu.cn/esc-sso/oidc/accessToken?grant\_type=authorization\_code&client\_id=1001&client\_secret=1001&code=b87576e1-fd14-47e5-a835-0d601b18387a&redirect\_uri=http%3A%2F%2FXX.XXX.XXX.XXX%3A8080%2FXXX%2FXXXX](http://10.0.10.8/esc-sso/oauth2.0/accessToken?grant_type=authorization_code&oauth_timestamp=1489739169232&client_id=1001&client_secret=1001&code=b87576e1-fd14-47e5-a835-0d601b18387a&redirect_uri=http://XX.XXX.XXX.XXX:8080/XXX/XXXX)

**返回参数列表：**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **类型** |
| access\_token | 返回的token值（AT-13-D61WfBwwEv1ntL0oop1k7uIlbeL2SKgmla1）,用于下一步的获取当前登录账号 | 必须 |
| refresh\_token | 刷新token(在应用配置里，打开【token刷新】的开关，即可返回该参数) | 非必须 |
| id\_token | 可通过解析id\_token获取当前登录用户信息，但仅有SSO/AD的登录账号，没有其他信息 | 必须 |
| token\_type | 固定返回参数，无需理会 | 必须 |
| expires\_in | token有效期，单位：秒 | 必须 |

**返回示例：**

{

    "access\_token": "AT-16-cE1rCqEx95KcTsOwrGNNQatEiS1cdGNvHge",//access\_token的值

    "token\_type": "bearer",//固定返回参数

    "expires\_in": 7200,//token有效期，单位：秒

    "refresh\_token": "RT-3-FrfmVSknautfgJADIvzr3vWwbx5FUIJqj5M",//刷新token，根据需要开启，不一定返回

    "id\_token": "eyJhbGciOiJSUzI1NiIsImtpZCI6ImlhbXNzbyJ9.eyJqdGkiOiIyY2QwODg2MC1lMGMxLTQ4YmUtYWRlZS00MzMzM2Q2NmFiYzEiLCJpc3MiOiJodHRwczovL3Nzby5hcHAuc3dpcmV0ZXN0LmNvbS9lc2Mtc3NvL29pZGMiLCJhdWQiOiI1NzFkOWIxZWFlNDYxNjQxNDZiOCIsImV4cCI6MTcwNjE4MzI3MywiaWF0IjoxNzA2MTc2MDczLCJuYmYiOjE3MDYxNzU3NzMsInN1YiI6InN5c2FkbWluIiwiYXV0aFR5cGUiOiJQV0QiLCJzaWQiOiI5OUU3NjM1QzgyRTlDMjA0NzA2MTA1QjJEODA1NTlCNTE3N0EwOUIxMDA3QkI2NEQxRDE4MDczOENGRTM5MUZGQzEyRkIyMTVENjE4N0RDMDcyM0E5NDBBNUU0RDY1MjgzQjAxQzg1NDI2QkNGRDY4OTQ3N0Y4OUZEOTYxMDFDRTBBMzYxRkJEODEwMENCRjJENzUwQTA5RjYyM0VFQjdEMTZEQTNDNUNFM0IzODM0REU0NUY1N0NEREVDRDJGNkZCNTJDMjhEMTZFMjgwODY3MUJBRjY0QkQ3RkYxNjlCMyIsInN0YXRlIjoiIiwibm9uY2UiOiIiLCJhdF9oYXNoIjoiMnlORTFBZm9sN2RYQWJYSlQ5aUdZZz09IiwicHJlZmVycmVkX3VzZXJuYW1lIjoic3lzYWRtaW4ifQ.Jz6Z66aXOLya4TyyPjBZqM9K6xLhaotDMFYh5pB-Nmy7llf2Gp0DatO2UwsBSCkUwpGys18rz-uV0lgrx-GLLLiMKO1\_Ue-O1zW2SiZYUx271Bf7gFxbTYoyfhtR5QAvBApfnxaut9NSa2VhQcCrAsuhVAg9QCvxTnMY\_YhjVvV\_4A\_pB-yMxePacm34HYW1bvxVO3AamnseJLkRmG1WQJxyB5RekZEFoZyiV7zteyQnuQpuU86IlLKtHVi3oHJkzl57dXOSItMSpyNZKWRIvatSBfsNfURRwCvimjz1pazm4WmNNiwCgVLE-zf-6wTeb4yvHKPMNvFoueOyjorq6g"//id\_token，可根据JWT（JSON Web Token）规则解析，获取当前登录账号

}

#### id\_token 解析获取登录账号

id\_token字段就是上面提到的ID Token, 该ID令牌是一个JSON网络令牌（JWT）包含身份数据。它由应用程序使用，并用于获取用户信息，如用户名，电子邮件等，通常用于UI显示。ID令牌符合IETF RFC 7519标准，包含三部分：标头，正文和签名。 JWT令牌包含Claims，它们是有关实体（通常是用户）的语句（例如名称或电子邮件地址）和其他元数据。 OIDC定义了一组标准的权利要求。这组标准声明包括姓名，电子邮件，性别，出生日期等。但是，如果您要捕获有关用户的信息，并且当前没有最能反映该信息的标准声明，则可以创建自定义声明并将其添加到令牌中。解析后的格式如下所示：



解析后的数据中sub属性的值对应当前登录的账号

#### access\_token获取SSO(二期)当前登录账号接口

**接口名称：** OIDC协议使用token获取登录用户信息接口

**请求类型：** GET

**请求Url :**  https://cas.dhu.edu.cn/esc-sso/oidc/profile

**接口描述:**  用于第三方系统使用token获取当前登录用户信息

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **是否必填** |
| access\_token | 获取到的access\_token的值 | 必须 |

**请求示例：**

 [https://cas.dhu.edu.cn/esc-sso/oidc/profile?access\_token=AT-13-D61WfBwwEv1ntL0oop1k7uIlbeL2SKgmla1](http://10.0.10.8/esc-sso/oauth2.0/profile?access_token=b1d508f3-32e7-4d6d-ac62-87836406704c)

**返回参数列表：**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **类型** |
| id | 用户登录的主账号 | 必须 |
| attributes->account\_no | 从账号、第三方应用系统账号 | 非必须 |
| attributes->email | 邮箱地址 | 非必须 |
| attributes->work\_no | 员工编号 | 非必须 |
| attributes->mobile | 手机号 | 非必须 |
| attributes->id | 用户ID，IAM系统内用户唯一标识编码 | 非必须 |
| attributes->user\_name | 姓名 | 非必须 |
| attributes->token\_gtime | token获取时间 | 非必须 |
| attributes->token\_expired | token有效期 | 非必须 |

**返回数据示例**

{

    "attributes": {

        "email": "800032@paraivew.com",

        "work\_no": "800032",//员工编号

        "mobile": "182XXXXXXXXXX",//手机号

        "id": "1839341111974344339",//用户ID，IAM系统内用户唯一标识编码

        "account\_no": "chenp",//第三方应用系统账号，一般与AD登录账号一致，但可提供兼容，既SSO允许该值与登录电脑、SAD登录的账号不一致

        "user\_name": "陈平",//姓名

        "token\_gtime": 1705484354333,

        "token\_expired": "7200"

    },

    "id": "chenp"//AD登录账号、SSO登录账号

}

#### SSO退出地址

**当系统需要退出时，可调用以下地址：**

对接系统退出自身系统后需要在浏览器地址栏中，重定向到https://cas.dhu.edu.cn/esc-sso/logout?service=http://xxxxx

其中http://xxxxx是应用自身的地址，用户退出后再次登录会跳转到该http://xxxxx地址

## SAML2.0认证集成规范

### SAML2.0认证过程

下图为SAML2.0认证交互过程图：

配置阶段： 服务方和认证方交换metadata,建立信任



使用阶段： 认证过程



* **流程说明：**
1. 用户试图访问SSO的应用方，如弹性福利系统。
2. 应用方生成一个SAML身份验证请求。SAML请求编码并嵌入到URL。RelayState参数包含编码的合作伙伴应用程序，用户尝试访问的URL也被嵌入在SSO URL。这的RelayState参数，就是要一个不透明的标识符，不作任何修改或检查传回的。
3. 应用方发送重定向到用户的浏览器。重定向URL编码SAML身份验证请求的，应提交到SSO服务。
4. SSO的SAML请求进行解码，并提取两个断言消费服务（ACS）和用户的目标URL（RelayState参数）的URL。
5. SSO的用户进行身份验证。SSO可以通过要求有效的登录凭据，或通过检查有效的会话对用户进行身份验证。
6. SSO生成一个SAML响应，其中包含身份验证的用户的用户名。按照SAML 2.0规范，这种反应是应用方的DSA / RSA密钥数字签名的。SSO SAML响应和RelayState参数进行编码，并将该信息返回到用户的浏览器。
7. 应用方的ACS使用SSO的公钥验证SAML响应。如果成功验证的响应，ACS将用户重定向的目标URL。

用户被重定向的目标URL，并记录在应用方应用程序。

### SAML集成规范

#### SAML相关请求认证接口

##### 处理SP发起的WEB SSO PROFILE POST请求

**接口名称：** 处理SP发起的WEB SSO PROFILE POST请求接口

**请求类型：** POST

**请求Url :** https://cas.dhu.edu.cn/esc-sso/idp/profile/SAML2/POST/SSO

**接口描述:**  用于SP（第三方系统）向SSO发起POST方式认证请求

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **是否必填** |
| SAMLRequest | 由SP发出的AuthnRequest | 必须 |

**返回信息**

重定向跳转**，**跳转到IAM登录页面，如果登录成功返回SAML Response

##### 处理SP发起的WEB SSO PROFILE Redirect请求

**接口名称： 处理SP发起的WEB SSO PROFILE Redirect请求**

**请求类型：** GET

**请求Url :** https://cas.dhu.edu.cn/esc-sso/idp/profile/SAML2/Redirect/SSO

**接口描述:**  用于SP（第三方系统）向SSO发起Redirect方式认证请求

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **是否必填** |
| SAMLRequest | 由SP发出的AuthnRequest | 必须 |

**返回信息**

重定向跳转**，**跳转到IAM登录页面，如果登录成功返回SAML Response

##### IDP发起的单点请求

**接口名称：** 由IAM系统发起请求SAML单点请求

**请求类型：** GET

**请求Url :** https://cas.dhu.edu.cn/esc-sso/idp/profile/SAML2/Unsolicited/SSO

**接口描述:** 如果想直接由IAM系统发起单点认证请求，可以使用该接口

**请求参数列表:**

|  |  |  |
| --- | --- | --- |
| **变量名** | **含义** | **是否必填** |
| providerId | 由SP发出的AuthnRequest | 必须 |

**返回信息**

生成SAML Response，重定向跳转**，**跳转至SP（第三方系统）页面。

## CAS协议认证集成规范

###  CAS认证过程

下图为CAS认证交互过程图：



1. 用户访问应用系统地址，应用方浏览器重定向请求到IAM的认证地址，并且提供相应的参数
2. IAM完成用户认证，并为app提供ticket。
3. 浏览器跳转到app并提供ticket参数。
4. 用户在获取访问ticket后，通过调用接口获取认证用户信息。
5. 应用获取用户信息，认证完成，用户正常访问app。

### 请求URL实例及参数说明

#### 对于已对接应用的补充说明：

在2025年5月9日前与统一身份认证平台做过集成对接的应用可保留原来的配置不变，我们已在新版本做过链接适配自动跳转。原链接地址是：

登录地址：https://cas.dhu.edu.cn/authserver/login

认证地址：[https://cas.dhu.edu.cn/authserver/serviceValidate](http://cas.dhu.edu.cn/authserver/serviceValidate)

登出地址：<https://cas.dhu.edu.cn/authserver/logout>

如果是未对接过的应用，请查看下方内容详情。仍有问题可联系021-67792054-25吴老师。

#### 组合认证地址浏览器请求认证及参数,重定向请求认证中心：

URL请求示例:

|  |  |  |
| --- | --- | --- |
| 属性名称 | 说明 | 是否必须 |
| service | 应用注册服务地址 | 必须 |

注：若中存在 & 符号需要进行转义，转义为 %26

回调地址中存在 & 符号之类的特殊字符时，需要使用正则 + 转义 组合配置：

举例：回调地址原本是： <http://localhost:8080/cas/callback?from=para&UserAgentFrom=pc>

1. 应用地址中将 & 符号转义后的地址（%26）：

<http://localhost:8080/cas/callback?from=para%26UserAgentFrom=pc>

1. 回调地址中使用正则匹配（[\S]\*）： [http://localhost:8080/cas/callback?from=para[\S]\*](http://localhost:8080/cas/callback?from=para%5b\\S%5d*)

####  SSO系统通过redirect\_uri浏览器重定向请求反馈ticket给应用

URL示例:

|  |  |  |
| --- | --- | --- |
| 属性名称 | 说明 | 是否必须 |
| ticket | 获取用户信息的凭证 | 必须 |

#### 根据ticket，向认证中心请求获取用户信息（CAS1/CAS2/CAS3）

|  |  |  |
| --- | --- | --- |
| 属性名称 | 说明 | 是否必须 |
| service | 应用注册服务地址 | 必须 |
| ticket | 获取用户信息的凭证 | 必须 |
| format | 返回的格式,默认是xml可选值：json | 非必须 |

##### 3.2.3.1 CAS1（cas1.0版本）

URL示例：

https://cas.dhu.edu.cn/esc-sso/validate?service=http://XXXX.XXXX.XXXX.XXXX/XXX/XXX&ticket=ST-123456abcdef

返回的用户信息：

请求成功：

yes
账号

请求失败：

no

##### 3.2.3.2 CAS2（cas2.0版本）

URL示例:

https://cas.dhu.edu.cn/esc-sso/serviceValidate?service=http://XXXX.XXXX.XXXX.XXXX/XXX/XXX&ticket=ST-123456abcdef&format=xml

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 说明 | 是否必须 |  |
| uid | 用户登录的主账号 | 必须 |  |
| attributes->phone | 用户的手机号 | 非必须 |  |
| attributes->email | 用户的邮箱 | 非必须 |  |
| attributes->usernameattributes->username | 用户的应用账号 | 非必须 |  |
| attributes->usertype | 用户类型编码 | 非必须 | 取值为：jzg,bks,yjs,qtry,hqry,cjxs,lxs |

**返回的用户信息：**

注：解析返回数据得到用户信息,应用系统根据这个字段匹配自身平台的账号数据，让用户通过认证。
Attribute字段里的phone，email，username，usertype字段是可以配置的

返回的用户信息示例（默认使用xml格式）：

<cas:serviceResponse xmlns:cas='http://www.yale.edu/tp/cas'>
 <cas:authenticationSuccess>
 <cas:user>sysadmin</cas:user>
 <cas:attributes>
 <cas:isFromNewLogin>false</cas:isFromNewLogin>
 <cas:authenticationDate></cas:authenticationDate>
 <cas:phone>18501612606</cas:phone>
 <cas:authenticationMethod>QueryDatabaseAuthenticationHandler</cas:authenticationMethod>
 <cas:successfulAuthenticationHandlers>QueryDatabaseAuthenticationHandler</cas:successfulAuthenticationHandlers>
 <cas:longTermAuthenticationRequestTokenUsed>false</cas:longTermAuthenticationRequestTokenUsed>
 <cas:email>sysadmin@paraview.cn</cas:email>
 <cas:username>sysadmin</cas:username>
 </cas:attributes>
 </cas:authenticationSuccess>
</cas:serviceResponse>

**JSON示例：**

{
 "serviceResponse": {
 "authenticationSuccess": {
 "user": "sysadmin",
 "attributes": {
 "isFromNewLogin": [
 true
 ],
 "authenticationDate": [
 1575645600.289000000
 ],
 "phone": [
 "18501612606"
 ],
 "authenticationMethod": "QueryDatabaseAuthenticationHandler",
 "successfulAuthenticationHandlers": [
 "QueryDatabaseAuthenticationHandler"
 ],
 "longTermAuthenticationRequestTokenUsed": [
 false
 ],
 "email": [
 "sysadmin@paraview.cn"
 ],
 "username": [
 "sysadmin"
 ]
 }
 }
 }
}

##### 3.2.3.3 CAS3（cas3.0版本）

URL示例:

https://cas.dhu.edu.cn/esc-sso/p3/serviceValidate?service=http://XXXX.XXXX.XXXX.XXXX/XXX/XXX&ticket=ST-123456abcdef&format=xml

返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 说明 | 是否必须 |  |
| uid | 用户登录的主账号 | 必须 |  |
| attributes->phone | 用户的手机号 | 非必须 |  |
| attributes->email | 用户的邮箱 | 非必须 |  |
| attributes->username | 用户的应用账号 | 非必须 |  |
| attributes->usertype | 用户类型编码 | 非必须 | 取值为：jzg,bks,yjs,qtry,hqry,cjxs,lxs |

解析返回数据得到用户信息,应用系统根据这个字段匹配自身平台的账号数据，让用户通过认证。
Attribute字段里的phone，email，username，usertype字段是可以配置的

**返回样例**

**XML:**

<cas:serviceResponse
 xmlns:cas='http://www.yale.edu/tp/cas'>
 <cas:authenticationSuccess>
 <cas:user>sysadmin</cas:user>
 <cas:attributes>
 <cas:isFromNewLogin>false</cas:isFromNewLogin>
 <cas:authenticationDate></cas:authenticationDate>
 <cas:phone>18501612606</cas:phone>
 <cas:authenticationMethod>QueryDatabaseAuthenticationHandler</cas:authenticationMethod>
 <cas:successfulAuthenticationHandlers>QueryDatabaseAuthenticationHandler</cas:successfulAuthenticationHandlers>
 <cas:longTermAuthenticationRequestTokenUsed>false</cas:longTermAuthenticationRequestTokenUsed>
 <cas:email>sysadmin@paraview.cn</cas:email>
 <cas:username>sysadmin</cas:username>
 </cas:attributes>
 </cas:authenticationSuccess>
</cas:serviceResponse>

**JSON:**

{
 "serviceResponse": {
 "authenticationSuccess": {
 "user": "sysadmin",
 "attributes": {
 "isFromNewLogin": [
 true
 ],
 "authenticationDate": [
 1575645600.289000000
 ],
 "phone": [
 "18501612606"
 ],
 "authenticationMethod": "QueryDatabaseAuthenticationHandler",
 "successfulAuthenticationHandlers": [
 "QueryDatabaseAuthenticationHandler"
 ],
 "longTermAuthenticationRequestTokenUsed": [
 false
 ],
 "email": [
 "sysadmin@paraview.cn"
 ],
 "username": [
 "sysadmin"
 ]