CAS Login Ticket

Role

Returns the login ticket (lt) required for a later 3DPassport CAS server authentication.

lt is used as CSRF token to prevent automatic replay of authentication request. It should be passed when authenticating on CAS server.

Request

Method	Path
GET	/login

Query Parameters

Name	Required	Value	Description
action	YES	get_auth_params	

Headers

Name	Required	Value	Description
Accept	YES	application/json	The response MIME format

Response

The successful response (status code 200) is a JSON with these keys:

Key	Description	Value
response	"login"	String
lt	The login ticket	String

Example

The web services has been launched while it is the first connection to the 3DPassport. There is no CASTGC cookie yet, and no 3DPassport session. The response code is 200, and the response body contains the login ticket.

```
##CLIENT REQUEST
##----------------------------------------------------------------
[GET] <3DPASSPORT_URL>/login?action=get_auth_params

##SERVER RESPONSE [200] OK
##--------------------

#RESPONSE HEADER :
#--------------------
HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 06:38:13 GMT
Server: Apache-Coyote/1.1
X-DS-IAM-VERSION: 2
Cache-Control: no-cache
Cache-Control: no-store
Cache-Control: max-age=0
Cache-Control: must-revalidate
Pragma: no-cache
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
Content-Type: text/html;charset=UTF-8
Content-Length: 40
Set-Cookie:
Access-Control-Allow-Credentials: true
Access-Control-Allow-Methods: GET, POST, OPTIONS
Access-Control-Allow-Headers: accept, x-requested-method, origin, x-requested-with, x-request
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive

#RESPONSE BODY
#--------------------
{"response":"login","lt":"LT-1253085-23CvHflVNUgeYe5h4BGyyAs4a7JIgi"}
```

A JSESSIONID is generated, kept by a coocky manager, and the same must be used for the CAS Authentication.

CAS Login Ticket

Role

Returns the login ticket (lt) required for a later 3DPassport CAS server authentication.
lt is used as CSRF token to prevent automatic replay of authentication request. It should be passed when authenticating on CAS server.

Request

Method	Path
GET	/login

Query Parameters

Name	Required	Value	Description
action	YES	get_auth_params	

Headers

Name	Required	Value	Description
Accept	YES	application/json	The response MIME format

Response

The successful response (status code 200) is a JSON with these keys:

Key	Description	Value
response	"login"	String
lt	The login ticket	String

Example

The web services has been launched while it is the first connection to the 3DPassport. There is no CASTGC cookie yet, and no 3DPassport session. The response code is 200, and the response body contains the login ticket.

```
##CLIENT REQUEST
##------------------------------------------------------------
[GET] <3DPASSPORT_URL>/login?action=get_auth_params

##SERVER RESPONSE [200] OK
##--------------------

#RESPONSE HEADER :
#--------------------
HTTP/1.1 200 OK
Date: Fri, 20 Jun 2014 06:38:13 GMT
Server: Apache-Coyote/1.1
X-DS-IAM-VERSION: 2
Cache-Control: no-cache
Cache-Control: no-store
Cache-Control: max-age=0
Cache-Control: must-revalidate
Pragma: no-cache
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
Content-Type: text/html;charset=UTF-8
Content-Length: 40
Set-Cookie:
Access-Control-Allow-Credentials: true
Access-Control-Allow-Methods: GET, POST, OPTIONS
Access-Control-Allow-Headers: accept, x-requested-method, origin, x-requested-with, x-request
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive

#RESPONSE BODY
#--------------------
{"response":"login","lt":"LT-1253085-23CvHflVNUgeYe5h4BGyyAs4a7JIgi"}
```

A JSESSIONID is generated, kept by a coocky manager, and the same must be used for the CAS Authentication.

CAS Authentication

Role

Authenticates a user to the 3D Passport CAS server, and optionally can redirect to a provided service URL too.

The call must be performed with the same JSESSIONID (stored in cookies) as the one generated by a previous login ticket retrieval call.

Request

Method	Path
POST	/login

Query Parameters

Name	Required	Value	Description
service	NO	An URL service. The format is URL-encoded.	The URL of the service to be redirected to after successful authentication. The service URL must use a trusted domain declared in 3DPassport configuration.

Headers

Name	Required	Value	Description
Content-Type	YES	application/x-www-form-urlencoded;charset=UTF-8	The POST body message MIME format

POST Message Body

A string with the following format:

```
lt=<loginticket>&username=<username>&password=<password>
```

Where:

Key	Mandatory	Value	Description
username	YES	string	The username or email (UTF8 en encoded) of the user to authenticate
password	YES	string	The password (UTF8 en encoded) of the user to authenticate
lt	YES	string	The login ticket
rememberMe	NO	string	"no" to keep the CASTGC cookie 2H (OnPremises) or 24H (on cloud) instead of 1 week.

Response

The response code can be:

- 302, a redirection

  The authentication is successful. A SSO cookie (also named CASTGC) is sent in the response header. The CASTGC is valid for 2 hours (1 week if "Remember me" box is checked).

  About the url of redirection:
  - You have provided an url service as input parameter.

    The url of redirection is the input url service in adjonction with a service ticket. In case of no automatic url redirection you have two mn to realize yourself the redirection.
  - otherwise

    The url of redirection is my-profile page of 3DPassport. See example.
- Error

Example

The web services has been launched, without a service as input, while a login ticket and a 3DPassport session (JSESSIONID) already exist. The call creates a CASTGC kept by a coocky manager.

```
##CLIENT REQUEST
##----------------------------------------------------------------
[POST] <3DPASSPORT_URL>/login
POST Data : lt=LT-15943-Glazcd1ifSa2dasumFj3NeKqLwi5BE&username=uxodtmem&password=0vb3pcds
Content-Type=application/x-www-form-urlencoded;charset=UTF-8

##SERVER RESPONSE [200] OK
##--------------------

Has been redirected. Last Redirect URL : <3DPASSPORT_URL>/my-profile

#RESPONSE HEADER :
#--------------------
HTTP/1.1 200 OK
Date: Thu, 19 Oct 2017 11:11:48 GMT
Server: Microsoft-IIS/7.0
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Headers: x-csrf-token,x-requested-method,x-requested-with,x-request,syc-auth-resourceId,syc-auth-userName,syc-auth-password,cache-control,X-DS-IAM-CSRFTOKEN
Access-Control-Expose-Headers: x-csrf-token,x-requested-method,x-requested-with,x-request,syc-auth-resourceId,syc-auth-userName,syc-auth-password,cache-control,X-DS-IAM-CSRFTOKEN
Access-Control-Allow-Credentials: true
Access-Control-Max-Age: 600
Cache-Control: max-age=0
Cache-Control: no-store
Cache-Control: no-cache
Cache-Control: must-revalidate
X-Content-Type-Options: nosniff
X-DS-IAM-VERSION: 2
X-XSS-Protection: 1; mode=block
Content-Type: text/html;charset=UTF-8
Content-Language: en-US
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Transfer-Encoding: chunked

#RESPONSE BODY
#--------------------

<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />
        <meta name="viewport" content="width=device-width, initial-scale=1" />
        <title>3DPassport - MyProfile</title>
        <link rel="shortcut icon" href="/iam/resources-171006102340/img/3dexperience/favicon.ico" />
        <link rel="apple-touch-icon" href="/iam/resources-171006102340/img/3dexperience/touchicon.png" />
        <link rel="stylesheet" href="/iam/resources-171006102340/css/main-ifwe.css" />
        <script src="https://dsxdev-online.dsone.3ds.com/doconline/English/CAAiamPassport//iam/resources-171006102340/js/libs/modernizr.js"></script>
    </head>
    <body>
        <script id="configData" type="application/json">{"changePasswordUrl":"<3DPASSPORT_URL>/change-my-password","isSocial":false,"browserLocale":"en_US","values":{"zip":"78140","lastName":
        <script src="https://dsxdev-online.dsone.3ds.com/doconline/English/CAAiamPassport//iam/resources-171006102340/js/libs/require.js"></script>
        <script>
            var UWA;
            requirejs.config({
                baseUrl: "/iam/resources-171006102340/js/",
                paths: {
                    qwery: "libs/qwery-v1.0.1-203-g6c2e088",
                    "DS": "libs/AmdLoader.mweb/src/4.requireDs",
                    "css": "libs/AmdLoader.mweb/src/5.requireCss",
                    "text": "libs/AmdLoader.mweb/src/6.requireText",
                    "i18n": "libs/AmdLoader.mweb/src/7.requireI18n",
                    "dsp/config": "empty:",
                    "UWA": "libs/UWA2.mweb/src/js",
                    "DS/UIKIT": "libs/UIKIT.mweb/src",
                    "DS/W3DXComponents": "libs/W3DXComponents.mweb/src",
                    "DS/WebAppsFoundations": "libs/WebAppsFoundations.mweb/src",
                    "DS/VENHammer": "libs/VENHammer.mext/src/1.0.1/hammer",
```

```
                    "DS/CefCommunication" : "libs/CefCommunication.mweb/src"
                }
            });
            define("dsp/config/myProfile", [], function () {
                var configData = JSON.parse(document.getElementById("configData").innerHTML);
                configData.csrfTokenValue = "6YGT-NSA7-9MTB-6HJT-QFBF-EG8Z-32K7-SOG6";
                configData.csrfTokenName = "OWASP_CSRFTOKEN";
                return configData;

            });
            require(["dsp/DSP"], function () {
                require(["myProfile"]);
            });
        </script>
        <script>
            if (typeof widget !== 'undefined') {
                widget.setBody('This is not supposed to happen ! Please check that you are not trying to run an authenticated service as a widget');
            }
        </script>
    </body>
</html>
```