SINGLE SIGN ON NELLA PIATTAFORMA CAI

SPECIFICHE TECNICHE

30/01/2020

Versione 1.4 - definitivo

SOMMARIO

INTRODUZIONE

Nella piattaforma CAI l'autenticazione è demandata ad un servizio centralizzato dove sono memorizzate le password degli utenti. Inoltre il servizio è predisposto a supportare il Single Sign On: una sessione utente è primariamente tenuta nel servizio stesso e un utente, una volta effettuata log-in per entrare in un'applicazione della piattaforma, potrà accedere ad altre applicazioni della medesima piattaforma senza doversi autenticare di nuovo.

Le applicazioni che a vario titolo devono inserirsi nella piattaforma CAI, pertanto, devono interfacciarsi in modo opportuno con questo servizio centralizzato:

1. Devono implementare un flusso adeguato di colloquio con il servizio di autenticazione (CAS)
2. Qualora le informazioni di accesso restituite dal CAS non siano sufficienti a descrivere l'utente per le esigenze dell'applicazione, una descrizione completa dell'utente si può ottenere attraverso un servizio di autorizzazione.

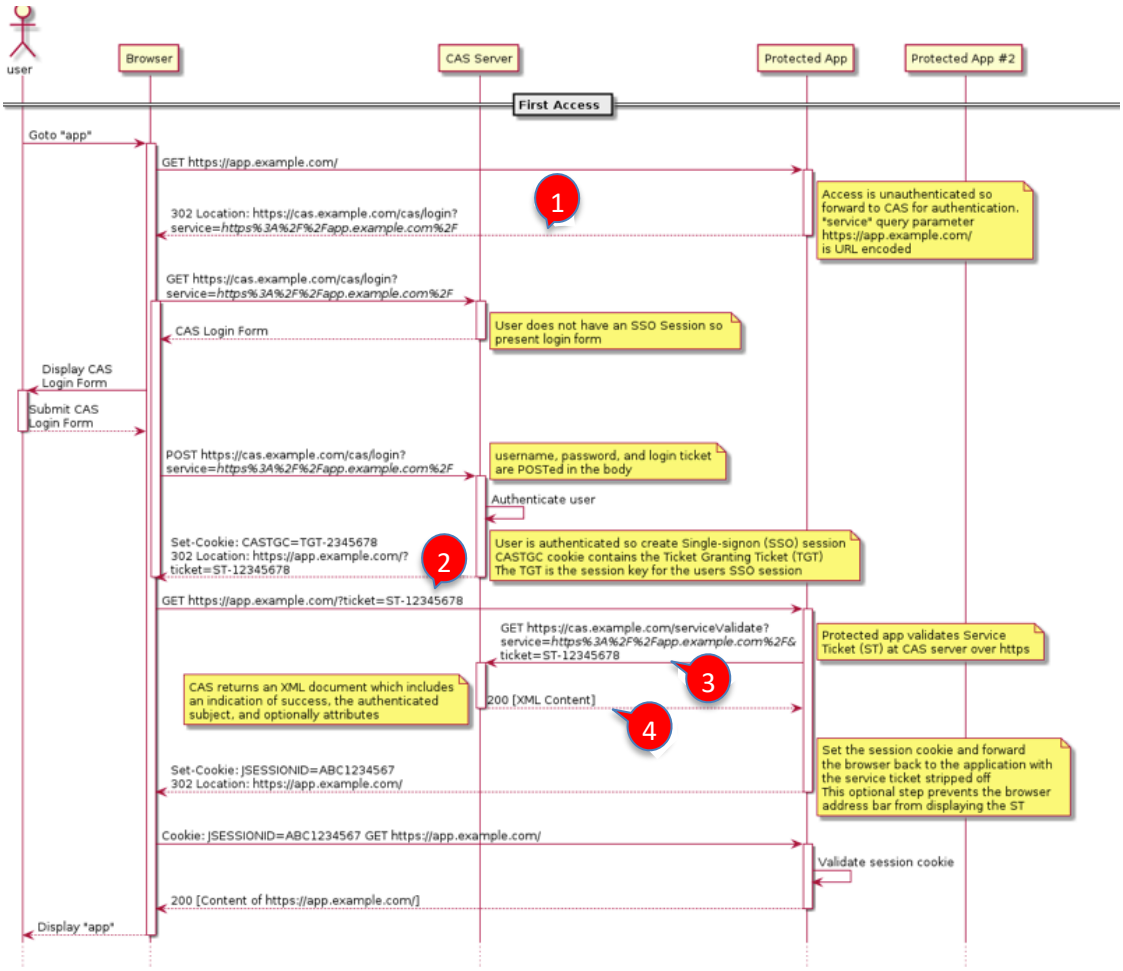
CAS

Il Single Sign On prevede che la sessione utente sia prima di tutto aperta nel servizio di autenticazione centrale (CAS). L'applicazione, verificato che per l'utente non è in piedi una sessione, effettua una redirect al CAS, chiedendo la login. Se la login nel CAS ha successo, il CAS effettua una redirect all'applicazione, passando due informazioni importanti:

1. Un token, che identifica l'utente in tutte le successive chiamate al CAS (per verificare se la sessione utente centralizzata è ancora valida)

2. Un set di dati che descrivono l'utente stesso, quali lo username utilizzato, nome e cognome, ruoli (permissions) e, non ultimo, l'UUID dell'utente, chiave univoca che identifica l'utente stesso all'interno del sistema centrale.

Il flusso in dettaglio può essere visualizzato nella documentazione generale del CAS (https://apereo.github.io/cas/4.2.x/protocol/CAS-Protocol.html) da cui riportiamo la prima immagine per facilità di riferimento.



Con riferimento all'immagine riportata, nel sistema CAI abbiamo le seguenti specificità:

1. CAS Server
    a. in collaudo: https://prova.cai.it/cai-cas
    b. in produzione: https://accesso.cai.it/cai-cas

   Pertanto, ad esempio in collaudo, la redirect al CAS prenderebbe forma
   `https://prova.cai.it/cai-cas/login?service=https%3A%2F%2Fapp.example.com%2F`

   Nota: in generale, il name server dell'applicazione potrebbe anche essere "localhost" (dove nell'esempio è `app.example.com`), ma il CAS è configurato in modo da non consentire accesso in questa circostanza (pur autenticando con successo l'utente, restituirebbe errore 403, *forbidden*).
2. Restituzione del controllo all'applicazione

Nella figura, una volta che l'utente abbia dato con successo al CAS le proprie credenziali, il CAS restituisce il controllo passando il ticket generato al momento. E' da notare che l'applicazione potrebbe avere una pagina/azione apposita a gestire la restituzione del controllo (in cui si svolgono i passaggi successivi). Nel caso, essa deve essere parte integrante della "registrazione" del servizio nel CAS: essa andrebbe inclusa anche nella prima redirect. Ad esempio, supponendo che l'azione prevista per memorizzare il ticket e tutto quello che consegue si chiami *cas_security_check*, la redirect di cui al punto 1 precedente apparirebbe così:

`https://prova.cai.it/cai-cas/login?service=https%3A%2F%2Fapp.example.com%2Fcas_security_check`

e, di conseguenza, il ritorno dal CAS si vedrebbe avvenire attraverso la redirect

`https://app.example.com/cas_security_check?ticket=ST-12345678`

3. Validazione del ticket

   Si deve usare il ticket ricevuto dal CAS e che si è memorizzato in sessione. Inoltre, se al punto 2 si è indicata una pagina/azione particolare anche qui deve essere riportata

   `https://prova.cai.it/cai-cas/serviceValidate?service=https%3A%2F%2Fapp.example.com%2Fcas_security_check&ticket=ST-12345678`

4. Un esempio di xml dal CAS del CAI restituito è il seguente:

```
<cas:serviceResponse xmlns:cas='http://www.yale.edu/tp/cas'>
  <cas:authenticationSuccess>
    <cas:user>sara.bollini@gmail.com</cas:user>
    <cas:attributes>
      <cas:attributes>
        <cas:uid>15140</cas:uid>
        <cas:email>sara.bollini@gmail.com</cas:email>
        <cas:roles>ROLE_MESSAGES</cas:roles>
        <cas:lastname>Bollini</cas:lastname>
        <cas:firstname>Sara</cas:firstname>
        <cas:uuid>1c2d9136-95dc-4e4d-a690-29e7d1bc0418</cas:uuid>
        <cas:profile></cas:profile>
        <cas:userUuid>1c2d9136-95dc-4e4d-a690-29e7d1bc0418</cas:userUuid>
      </cas:attributes>
    </cas:attributes>
    <cas:attributes>
      <cas:uid>15140</cas:uid>
      <cas:email>sara.bollini@gmail.com</cas:email>
      <cas:roles>ROLE_MESSAGES</cas:roles>
      <cas:lastname>Bollini</cas:lastname>
      <cas:firstname>Sara</cas:firstname>
      <cas:uuid>1c2d9136-95dc-4e4d-a690-29e7d1bc0418</cas:uuid>
       <cas:profile></cas:profile>
      <cas:userUuid>1c2d9136-95dc-4e4d-a690-29e7d1bc0418</cas:userUuid>
    </cas:attributes>
  </cas:authenticationSuccess>
</cas:serviceResponse>
```

PROFILI UTENTE

L'esempio riportato è per un utente che si sia connesso al sistema utilizzando il profilo principale.

Nel caso in cui l'utente disponesse di profili ulteriori e avesse fatto accesso utilizzando uno di essi, il campo `cas:profile` conterrebbe l'username specifico del profilo selezionato e il campo `cas:uuid` conterrebbe l'uuid del profilo stesso; in tal caso, il campo `cas:userUuid` conterrebbe l'uuid dell'utente (il profilo principale):

```
<cas:profile>sara.bollini@gmail.com#eoop</cas:profile>
<cas:uuid>fce14729-4868-453d-b51e-7a797ce381ed</cas:uuid>
<cas:userUuid>1c2d9136-95dc-4e4d-a690-29e7d1bc0418</cas:userUuid>
```

Nota bene: in tal caso, se si desidera ottenere i dati di dettaglio *dell'utente* (del profilo principale), si utilizzi l'ultimo attributo per ottenerli dal Servizio di Dettaglio Utenti.

SERVIZIO DI DETTAGLIO UTENTI

Se i dati presenti nella risposta del CAS non sono sufficienti per le esigenze dell'applicazione (ad esempio, potrebbe essere necessario sapere di quale Sezione del CAI l'utente sia operatore), è possibile interrogare un servizio (Web Service) preposto a restituire tutti i dettagli di un utente, dato lo UUID che lo identifica.

E' un web service REST con risultato in JSON, la cui chiamata è:

- In collaudo: `https://prova.cai.it/cai-integration-ws/secured/users/<uuid-utente>/full`
- In produzione: `https://services.cai.it/cai-integration-ws/secured/users/<uuid-utente>/full`

Ad esempio, per l'utente di cui sopra, la chiamata sarebbe:

`https://prova.cai.it/cai-integration-ws/secured/users/1c2d9136-95dc-4e4d-a690-29e7d1bc0418/full`

a cui seguirebbe risposta:


```
{
  "id":15140,
  "username":"sara.bollini@gmail.com",
  "uuid":"1c2d9136-95dc-4e4d-a690-29e7d1bc0418",
  "uuidSocio":"aae5b1ca-d7ab-4ee2-8386-e0b77272ae42",
  "operatore":false,
  "password":null,
  "oldPassword":null,
  "enabled":true,
  "firstName":"Sara",
  "lastName":
  "Bollini",
  "cardCode":"",
  "memberCode":"",
  "phone":null,
  "lastLogin":1508252117831,
  "userAuthorities":
  [
    {
      "id":38,
```

```
      "description":"Utente abilitato alla lettura dei messaggi.",
      "role":"ROLE_MESSAGES",
      "application":"Messaggistica",
      "longDescription":"ROLE_MESSAGES - Utente abilitato alla lettura dei messaggi."
    }
  ],
  "userGroups":[],
  "sectionCode":"9299999",
  "subsectionCode":null,
  "territorialGroupCode":null,
  "regionaleGroupCode":null,
  "operativeEntityCode":null,
  "otcoCode":null,
  "ottoCode":null,
  "centralAdministration":false,
  "aggregatedAuthorities":
  [
    {
      "id":38,
      "description":"Utente abilitato alla lettura dei messaggi.",
      "role":"ROLE_MESSAGES",
      "application":"Messaggistica",
      "longDescription":"ROLE_MESSAGES - Utente abilitato alla lettura dei messaggi."
    }
  ],
  "privacyOk":false,
  "lastEnabled":null,
  "privacyEnabled":false,
  "exposedName":"Sara Bollini",
  "regionalType":false,
  "authorities":
  [
    {
      "authority":"ROLE_MESSAGES"
    }
  ],
  "accountNonLocked":true,
  "credentialsNonExpired":true,
  "accountNonExpired":true,
  "exposedNameWithUsername":"Sara Bollini (sara.bollini@gmail.com)",
  "authoritiesApplicationMap":
  {
    "Messaggistica":[
      {
        "id":38,"description":"Utente abilitato alla lettura dei  messaggi.",
        "role":"ROLE_MESSAGES",
        "application":"Messaggistica",
        "longDescription":"ROLE_MESSAGES - Utente abilitato alla lettura dei messaggi."
      }
    ]
  },
  "sectionType":true,
  "subsectionType":false,
  "territorialType":false,
  "otcoType":false,
  "ottoType":false,
  "operativeEntityType":false,
  "amministrativeEntity":true,
  "technicalEntity":false,
  "profileUuid":"e3c5a9ae-b2a5-481e-849a-9e6ac2a1ddba"
}
```

da cui, ad esempio, si può vedere la sezione di appartenenza (codice 9299999, la “Sezione di Prova”).

Nota: per accedere a questo servizio l’applicazione deve autenticarsi (*basic authentication*) con un utente a cui sia stata data permission *ROLE_USERS_REST_FULL*.

PROFILI UTENTE

Nel caso in cui l’utente disponesse di profili ulteriori, la Sezione di appartenenza, le autorizzazioni concesse, ecc., sono informazioni che possono variare da profilo a profilo. In tali situazioni, accedendo con lo uuid restituito dal CAS, la descrizione dell’utente coerente con il profilo che avesse eventualmente selezionato è costruita mescolando insieme i dati generali dell’utente con i dati specifici del profilo selezionato. Se i dati restituiti sono specifici di un profilo, questo si può evincere dall’attributo **`profileUuid`**, che risulterebbe valorizzato.

Nota: vedi l’ultimo capitolo.

SERVIZIO UTENTI DI ORGANIZZAZIONE

Applicazioni satelliti della piattaforma centrale potrebbero avere la necessità di conoscere gli utenti a cui è stato attribuito un determinato ruolo. Ad esempio, ciò permetterebbe di popolare un menù a tendina da cui selezionare un utente tra quelli di una Sezione (o di un Gruppo Regionale o di una Scuola, ecc.), visualizzando, però, solo quelli abilitati ad una determinata operazione.

E’ pertanto possibile interrogare un servizio (Web Service) preposto a restituire un elenco sintetico (array JSON di record contenenti username, nome – “name” – e cognome – “surname”), dato il codice della Sezione di appartenenza e il ruolo richiesto:

- In collaudo: `https://prova.cai.it/cai-integration-ws/secured/users/list/<codice sezione>/<ruolo>`
- In produzione: `https://services.cai.it/cai-integration-ws/secured/users/list/<codice sezione>/<ruolo>`

Ad esempio, in collaudo, una chiamata possibile sarebbe:

```
https://prova.cai.it/cai-integration-ws/secured/users/list/9220005/ROLE_WEBSITE_SECTION_EDITOR
```

Nota 1: per accedere a questo servizio l’applicazione deve autenticarsi (*basic authentication*) con un utente a cui sia stata data permission *ROLE_USERS_REST_SECTION*.

Nota 2: il servizio restituisce l’elenco degli utenti (array JSON di record, ciascuno con uuid, nome, cognome e username) che abbiano il ruolo indicato in almeno una delle seguenti circostanze:

1. il ruolo vi è stato attribuito direttamente;
2. il ruolo è attribuito ad uno dei gruppi di utenti a cui appartiene l’utente;
3. il ruolo è attribuito ad uno dei gruppi di utenti alla cui appartenenza l’utente è stato delegato.

Nota 3: la ricerca del ruolo viene fatta nel profilo principale e in tutti gli eventuali secondari; l'esito, in ogni caso, è quello dell'utente (uuid e username sono quelli del profilo principale anche laddove l'appartenenza alla sezione con il ruolo richiesto fosse stata riscontrata in un profilo secondario).

Nota 4: la ricerca avviene analizzando il codice organizzativo passato in input e determinando, in base al suo formato, a cosa esso si riferisca. Ad esempio, un codice di due cifre verrà interpretato come quello che individui un OTCO; uno di quattro cifre individuerà un OTTO; uno di sette che incominci per "95" verrà interpretato per individuare un GR; ecc.

Nota 5: il codice organizzativo viene usato in modo "stretto" per quanto riguarda le gerarchie. Cioè, dato un codice di una struttura radice o intermedia, si considerano utenti che appartengano esattamente a quel livello alla struttura indicata: ad esempio, nel chiedere gli utenti dell'OTCO di codice 05, non verranno considerati tutti quegli utenti (profili) che fossero associati ad un suo OTTO e/o, all'interno di uno di questi, ad un'Unità Operativa specifica. Allo stesso modo, ad esempio, nel chiedere utenti di un GR non verranno considerati gli utenti/profili *di sezione* delle sezioni pur presenti in quel GR.

Nota 6: vedi il capitolo successivo.

PER FARE RIFERIMENTO AI DATI DEGLI UTENTI

Il meccanismo dei profili (autenticazione via CAS e consultazione dei dettagli dell'utente che si è autenticato) partono dal presupposto che le applicazioni si affidino alle informazioni di autorizzazione (ruoli e appartenenza organizzativa) restituiti di volta in volta dal servizio di autorizzazione. Pertanto, ad esempio, si considera che si acquisisca lo **`uuid`** dalla risposta del CAS e con tale **`uuid`** si acceda al servizio delle autorizzazioni: se tale attributo, in realtà si riferisse ad un profilo particolare, il servizio delle autorizzazioni restituirà delle informazioni in ogni caso consistenti – per l'identificazione dell'utente – e coerenti con tale circostanza – la scelta da parte dell'utente di un profilo particolare (una scelta particolare di ruoli e appartenenza organizzativa).

I dati restituiti dai servizi di autorizzazione sono sempre in prima battuta utenti e l'attributo **`uuid`**, così come lo username, il nome e il cognome, sono sempre quelli dell'utente. Dove però l'interrogazione abbia in qualche modo fatto puntare ad un profilo specifico (perché, ad esempio, si sono chiesti utenti di una determinata appartenenza organizzativa; oppure, in ogni caso, quando si utilizza un uuid restituito dal CAS) i dati restituiti dai servizi di autorizzazione contengono in generale *due* uuid:

1. l'**`uuid`** dell'utente e
2. l'attributo **`profileUuid`**, cioè l'**`uuid`** del profilo rispondente al criterio di ricerca (appartenenza organizzativa o uuid restituito dal CAS), attributo sempre presente che, tuttavia, può essere valorizzato a *null*.

Se si devono memorizzare anche solo temporaneamente delle informazioni restituite dai servizi di autorizzazione, si tenga allora conto che, in genere, se non è a *null*, il dato utile è l'attributo **`profileUuid`**.

ESEMPIO

Se si deve presentare in qualche modo l'esito di una ricerca degli utenti dotati di un certo ruolo in una determinata sezione, per ciascun utente restituito si consideri che,

- se il **`profileUuid`** è a *null*, ciò implica che l'utente si connetterà con il ruolo richiesto per la sezione indicata senza aver selezionato un profilo particolare;
- se il **`profileUuid`** non è a *null*, ciò implica che l'utente, per godere del ruolo richiesto nella sezione indicata, si connetterà avendo selezionato quel profilo lì.

Quindi, l'informazione da memorizzare (ad esempio all'interno di un menù a tendina, una *select* html) sarà il **`profileUuid`** e, solo se questo è a *null*, si dovrà utilizzare lo **`uuid`**.