sogei

SSO:
SINGLE SIGN-ON TRA EQ ED EQS

Compilato:

Rivisto:

Approvato:

Data: gg mese anno

Versione:

Classificazione: [ripetere la classificazione di riservatezza impostata]

Distribuito a:

INDICE

1. INTRODUZIONE

In questo documento si prende in esame una possibile soluzione per la realizzazione del Single Sign-On tra il portale istituzionale di ADER e le applicazioni.

1.1 PREMESSA

Il presente documento contiene gli elementi fondamentali per comprendere il problema e per realizzarlo con la soluzione proposta. Tra gli argomenti trattati ci sono una descrizione delle asserzioni SAML, una introduzione sulla tecnologia della autenticazione esterna prevista dal TAM di IBM, una analisi dell'architettura proposta ed una descrizione dei passi necessari per realizzarla. Il documento prende in esame anche l'applicazione che realizza il riconoscimento, la validazione e l'estrazione delle credenziali dal token proveniente dalle richieste che giungono dal portale istituzionale.

1.2 STRUTTURA DEL DOCUMENTO

La struttura del documento ha quindi la seguente struttura:
Capitolo 2 – Specifiche e requisiti per il Single Sign-On: in questo capitolo sono descritti i requisiti per l'integrazione delle applicazioni all'interno del portale istituzionale di ADER in modo che l'utente collegato alla intranet non debba effettuare una nuova login per accedere ai servizi.
Capitolo 3 – Architettura della soluzione: in questo capitolo viene esaminata la soluzione che fa uso del concetto di EAI di WebSEAL di IBM.
Capitolo 4 – Interfacce presentate all'utente: sono riportate le interfacce presentate all'utente quando si presentano le situazioni di errore previste oppure quando deve scegliere una delle username associate al suo codice fiscale..
Capitolo 5 – Azioni da svolgere sul WebSEAL: in questo capitolo sono descritti i passi necessari affinché il modulo di autenticazione esterno sia correttamente configurato. L'istanza di WebSEAL da utilizzare deve essere una nuova istanza mentre le componenti di Policy Server, Authorization Server ed il registro degli utenti Ldap devono essere quelli già presenti nell'infrastruttura di EqS.
Capitolo 6 – Deploy delle applicazioni: in questo capitolo si prende in esame l'installazione e la configurazione delle applicazione EAI e dell'applicazione che consente il collaudo dell'applicazione EAI.
Capitolo 7 – Dry Test e Stress Test: In questo capitolo si prendono in esame tutti i passi necessari per eseguire un pre-collaudo di verifica della correttezza sul funzionamento delle componenti installate e la predisposizione per eseguire gli stress test.

1.3 DOCUMENTI DI RIFERIMENTO

Nel documento si fa riferimento a concetti che possono essere approfonditi nei seguenti riferimenti:

DR1 OASIS Security Services (SAML) TC | OASIS
http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security

DR2 Using Tivoli Access Manager for eBusiness WebSEAL without a user registry
http://www.ibm.com/developerworks/tivoli/library/t-tamwebsealur/

DR3 An Introduction to the Java EE 5 Platform
http://java.sun.com/developer/technicalArticles/J2EE/intro_ee5/

DR4 WebSEAL Administration Guide
http://publib.boulder.ibm.com

DR5 OpenSAML
https://wiki.shibboleth.net/confluence/display/OpenSAML/OSTwoUserManual

1.4 DEFINIZIONI, ACRONIMI, ABBREVIAZIONI

Sono di seguito riportate le informazioni su tutti i termini, gli acronimi, le abbreviazioni e le definizioni necessarie alla comprensione di questo documento.

Acronimo	Descrizione/Definizione
JEE	Java Enterprise Edition
SAML	Security Assertion Markup Language
TAM	Tivoli Access Manager

2. SPECIFICHE E REQUISITI PER IL SINGLE SIGN-ON

In questo capitolo sono descritti i requisiti per l'integrazione delle applicazioni all'interno del portale istituzionale di ADER in modo tale che l'utente collegato alla intranet non debba effettuare una nuova login per accedere ai servizi.

2.1 MODALITÀ DI COMUNICAZIONE DELLE CREDENZIALI

Si propone una soluzione di Single Sign-On che consenta agli utenti di ADER autenticati sul dominio Microsoft interno, e di conseguenza sul portale interno ADER, di essere riconosciuti al momento dell'accesso ad applicazioni web esterne senza la necessità di reinserire le proprie credenziali.
Tale soluzione non contempla invece l'autenticazione in SSO tra l'applicazione esterna ed il portale interno ADER, garantendo in tal modo la centralizzazione delle politiche di sicurezza nel dominio ADER.
Per ottenere il Single Sign-On si realizza uno scenario di Identity Federation in cui Sogei ha il ruolo di Identity Provider (IdP) ed il fornitore dell'applicazione ha il ruolo di Service Provider (SP).
Sogei come IdP produce un'asserzione SAML, la inserisce in una SAML Authentication Response, firma la SAML Response, applica una codifica Base64 e invia il messaggio codificato all'applicazione chiamata in modalità HTTP POST.
E' utilizzata la versione 1.1 dello standard SAML.
L'accesso dell'utente nello scenario di federazione proposto avviene nei seguenti passi:

- dal portale ADER l'utente richiede l'accesso all'applicazione web esterna
- il portale invia in modalità HTTP POST, secondo quanto stabilito dallo standard SAML 1.1, una SAML Authentication Response, firmata e codificata in Base64, alla URL, che deve essere predisposta nell'applicazione esterna per accettare asserzioni SAML (es. www.sirfinequi.it/SAMLconsumer). Tale operazione può essere resa trasparente all'utente se la configurazione del browser prevede l'abilitazione dell'esecuzione di codice Javascript;
- l'applicazione esterna verifica la firma della SAML Response ed estrae dall'asserzione l'identificativo dell'utente; in particolare si fa presente che è utilizzato come identificativo il codice fiscale; se l'identificativo utilizzato dall'applicazione esterna fosse diverso, l'applicazione esterna dovrà eseguire un mapping fra tale identificativo ed il codice fiscale;
- l'applicazione esterna, se la verifica dell'asserzione ha avuto esito positivo, consente l'accesso dell'utente all'utente senza chiedere una nuova autenticazione

Il certificato con cui sarà firmata ogni SAML Response sarà comunicato da Sogei in modalità sicura da concordarsi.

Solo a titolo di esempio, nel listato sottostante si mostra una SAML Response firmata, prima della codifica in Base64.

```
<?xml version="1.0" encoding="UTF-8"?>
```

SSO:
SINGLE SIGN-ON TRA EQ ED EQS

```
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:1.0:protocol"
        IssueInstant="2011-11-10T07:04:05.976Z"
        MajorVersion="1"
        MinorVersion="1"
        ResponseID="SAML-ac05ed62-3989-4fc6-b3f2-50688e2f8ccf">
        <samlp:Status>
                <samlp:StatusCode Value="samlp:Success"/>
        </samlp:Status>
        <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
                AssertionID="SAML-6658742f-c461-4c88-aa3d-455d6f3916d2"
                IssueInstant="2011-11-10T07:04:05.820Z"
                Issuer="xi50.iamlab.it"
                MajorVersion="1"
                MinorVersion="0">
                <saml:Conditions NotBefore="2011-11-10T07:04:05.804Z"
                             NotOnOrAfter="2011-11-10T07:05:35.804Z"/>
                <saml:AuthenticationStatement
                    AuthenticationInstant="2011-11-10T07:04:05.741Z"
AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:unspecified">
                        <saml:Subject>
                                <saml:NameIdentifier
                      Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
                                GGNFBA99M13H501K</saml:NameIdentifier>
                                   <saml:SubjectConfirmation>
                                           <saml:ConfirmationMethod>
                                     urn:oasis:names:tc:SAML:1.0:cm:bearer
                                   </saml:ConfirmationMethod>
                                   </saml:SubjectConfirmation>
                        </saml:Subject>
                <saml:SubjectLocality IPAddress="111.222.111.222"/>
                </saml:AuthenticationStatement>
        </saml:Assertion>
        <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
                <SignedInfo>
                        <CanonicalizationMethod
                     Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
                        <SignatureMethod
                     Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
                        <Reference URI="#SAML-ac05ed62-3989-4fc6-b3f2-50688e2f8ccf">
                                <Transforms>
                                        <Transform
                  Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
                                        <Transform
                    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
                                </Transforms>
                                <DigestMethod
                               Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>

        <DigestValue>EPvpRSjI8VW5uTJgqF+16Zs/gV4=</DigestValue>
                        </Reference>
                </SignedInfo>
                <SignatureValue>WdXizuHbecfps4.....44MzKw=</SignatureValue>
                <KeyInfo>
                        <X509Data>
                     <X509Certificate>
                       MIIB8DCCAVmgAwIBAgIQ.......XnOT4dcJUTluywS758Kxg=
                     </X509Certificate>
                                <X509IssuerSerial>
                                        <X509IssuerName>
                                  CN=Sogei Certificate Authority, O=Sogei, C=IT
                                </X509IssuerName>
                                        <X509SerialNumber>4</X509SerialNumber>
                                </X509IssuerSerial>
```

```
                        </X509Data>
                    </KeyInfo>
                </Signature>
</samlp:Response>
```

Listato 01 – Esempio di SAML Response

Evidenziato in giallo il Tag XML contenente l'informazione del codice fiscale dell'utente.

2.2 REQUISITO EQS

Ai requisiti del paragrafo precedente occorre aggiungere un requisito posto da EqS su come comunicare il servizio da visualizzare. E' necessario che insieme alla Response SAML ci sia un altro parametro che identifica il servizio a cui l'utente vuole accedere. Il parametro rappresenta un acronimo del servizio che in fase di elaborazione verrà tradotto nell'opportuno indirizzo.

2.3 COSA È SAML 1.1

Security Assertion Markup Language (SAML) è uno standard per lo scambio di dati di autenticazione e autorizzazione (dette asserzioni) tra domini di sicurezza distinti. Tipicamente un identity provider (entità che fornisce informazioni di identità) e un service provider (entità che fornisce servizi). Il formato delle asserzioni SAML è basato su XML chiamate generalmente Token. SAML è mantenuto da OASIS Security Services Technical Committee.
Il problema principale che SAML cerca di risolvere, quindi, è quello del Web Single Sign-On (SSO) tra entità appartenenti a organizzazioni e domini di sicurezza distinti. Di seguito un approfondimento sul tema.

2.3.1 ASSERZIONI DI AUTENTICAZIONE

I sistemi automatici di gestione delle entità devono avere a disposizione un metodo per creare e distribuire le asserzioni d'identità. Il sistema Kerberos, per esempio, è uno dei sistemi di questo tipo. Lo standard SAML è un linguaggio per esprimere le credenziali. Oltre a rappresentare un metodo per standardizzare tramite una sintassi XML la rappresentazione delle credenziali di sicurezza, SAML definisce anche un protocollo per trasmetterle.
L'uso pratico di questo protocollo è molto semplice. Un client invia una richiesta relativa ad un soggetto ad una autorità SAML, che costituisce delle asserzioni riguardanti l'identità del soggetto nell'ambito di un dato dominio di sicurezza. Il soggetto potrebbe essere, per esempio, una persona identificata dal proprio indirizzo e-mail, dove il dominio di sicurezza sarebbe rappresentato dal

dominio internet a cui appartiene l'indirizzo di posta elettronica. Esiste più di uno schema d'uso di SAML che nel seguito saranno analizzati.
Un'autorità SAML è un servizio on-line in grado di rispondere alle richieste effettuate tramite protocollo SAML. Le risposte SAML sono chiamate asserzioni. Le autorità SAML possono essere di tre tipi: autorità di autenticazione, autorità di attributi e Policy Decision Points (PDP). Ciascuna di queste autorità può restituire tre distinte tipologie di asserzioni, ma quella che nel contesto ADER è coinvolta è quella della autenticazione.
Quando l'autorità di autenticazione SAML riceve una richiesta riguardante le credenziali di un certo soggetto, il risultato viene restituito in una asserzione di autenticazione. Una autorità di autenticazione asserisce che il soggetto S è stato autenticato con il metodo M all'istante di tempo T. Per esempio: "il soggetto Mario Rossi appartenente all'organizzazione ADER.it è stato autenticato utilizzando una password all'istante 2003-05-06T12:20:00-05:00".
Una singola autorità può produrre tutti e tre i tipi di asserzione, oppure solo un sottoinsieme. Le autorità possono essere generatori o consumatori di asserzioni provenienti da altre autorità.
Tutte le asserzioni contengono i seguenti elementi comuni:

- L'ID di chi rilascia e una etichetta temporale;
- L'ID, globale univoco, che lo identifica;
- Il soggetto a cui si riferiscono, incluso un nome, un dominio di sicurezza e, opzionalmente, i dati di autenticazione;
- Opzionalmente, altre informazioni fornite dalla autorità che le ha rilasciate e che vengono chiamate advice (consigli);
- Le condizioni sotto cui le asserzioni sono valide, come la validità temporale (per esempio NotBefore e NotOnOrAfter);
- Le restrizioni di accesso alle asserzioni;
- Le restrizioni di accesso alle risorse eventualmente referenziate dall'asserzione;
- Altre condizioni specifiche dell'applicazione;

Facendo riferimento alla Response SAML riportata nel Listato 01 si ha che il NameIdentifier del Subject è un codice fiscale.
Nella risposta SAML l'autorità di autenticazione asserisce che il soggetto con identificativo GGNFBA99M13H501K è stato autenticato il 10 novembre 2011 alle ore 07:04:05 (AuthenticationInstant) utilizzando un metodo di autenticazione non specificato (AuthenticationMethod), e che questa autenticazione è valida dalle 07:04:05 (NotBefore) alle 07:05:35 (NotOnOrAfter) dello stesso giorno.

2.3.2 UTILIZZO DI SAML

Tipicamente la tecnologia SAML è utilizzata sia per i web browser che per il protocollo SAOP. Per quanto riguarda i web browser e quindi per realizzare il single sign-on tra due siti esistono due modalità: profilo pull e profilo push.
Il profilo pull utilizza frammenti SAML, chiamati token, trasferiti da un sito all'altro tramite stringhe di interrogazione negli URL. Il sito che effettua l'asserzione di identità (cioè il sito sorgente) crea un

link al sito destinazione contenente nell'URL il token; quando un utente clicca l'URL, il sito destinazione riceve il token come parte della richiesta http GET. Il token è una chiave che il sito destinazione può utilizzare per recuperare dal sito sorgente l'effettiva asserzione di identità riguardante l'utente.
Nel profilo push (quello di interesse per questo documento) il sito sorgente crea una form contenente tutti i dati dell'asserzione di identità. Quando l'utente sottomette la form, i dati dell'asserzione di identità vengono inviati al sito destinazione utilizzando una HTTP POST. In questo scenario, il sito sorgente pone una firma digitale sull'asserzione di identità utilizzando il protocollo XML Signature in modo tale che il sito destinazione possa essere sicuro dell'autenticità dei dati di identità ricevuti.

2.3.3 INTEGRITÀ E NON RIPUDIABILITÀ: XML SIGNATURES

Lo standard XML Signatures definisce come utilizzare una sintassi XML per assicurare l'integrità di un documento XML o di una parte di esso. Questo standard non definisce nuovi metodi per le firme, ma specifica invece come una firma digitale possa essere utilizzata all'interno di un documento XML.
Le firme digitali possono essere utilizzate per assicurare l'integrità di un messaggio, dimostrando che il contenuto non è stato modificato come risultato di errori o di manipolazioni maliziose. Le firme digitali impediscono inoltre che chi ha firmato un documento lo possa ripudiare. A proposito di questo si può ricordare che:

- **Integrità**: L'integrità è un requisito fondamentale di una affidabile infrastruttura di gestione delle identità digitali. I sistemi si scambiano credenziali, messaggi, effettuano transazioni e quindi è importante poter dimostrare che i contenuti non siano stati modificati. Si consideri, per esempio, un documento che rappresenta le credenziali di identità di un utente. Per potersi fidare di tali credenziali, è necessario poter verificare che esse siano autentiche e che non siano state modificate.
- **Non ripudiabilità**: La non ripudiabilità è la presentazione di una evidenza incontrovertibile che un messaggio è stato inviato o ricevuto. Se i messaggi o le transazioni fossero contestabili, importanti aspetti legati alle identità digitali verrebbero compromessi e messi in discussione. Queste dispute possono prendere due forme diverse. Si consideri il classico esempio di due corrispondenti, Alice e Giovanni. Nel primo caso Alice nega di averlo ricevuto. Il poter dimostrare che alice ha torto viene chiamata non ripudiabilità del mittente (in genere denominata NRO, non-repudiaton of origin). Nel secondo caso Alice sostiene di aver inviato a Giovanni un messaggio, che invece nega di averlo ricevuto. Il poter dimostrare che Giovanni ha torto viene chiamata non ripudiabilità del destinatario (in genere denominata NRR, non-repudiaton of receipt).

L'XML Segnatures è uno standard. Come ogni standard prevede molte opzioni e ciò lo rende apparentemente molto complicato, ma a livello fondamentale è piuttosto semplice. Una firma digitale espressa secondo queste specifiche deve essere contenuta in un elemento `<Segnature/>`, che a sua volta deve consistere delle seguenti tre parti principali:

- L'elemento <SignedInfo/> contiene un riferimento ai dati firmati e ad altre informazioni accessorie quali il metodo utilizzato per la firma, i dati sul digest e ogni altra informazione eventuale applicata;
- L'elemento <SignatureValue/> contiene la firma vera e propria;
- L'elemento <KeyInfo/> contiene i dati relativi alla chiave necessaria per verificare la validità della firma.

Tramite lo standard XML Segnature è anche possibile firmare sezioni di un documento, ed entità diverse sono in grado di firmale la stessa o differenti sezioni di un documento.
Facendo riferimento alla Response SAML del Listato 01 il metodo utilizzato per la firma, il tipo di digest ed i dati del certificato si riferiscono ai formati ed agli algoritmi di crittografia più noti, che vengono semplicemente racchiusi in una struttura XML. Altri elementi richiedono invece alcuni chiarimenti.
Uno di questi è l'elemento <CanonicallizationMethod/>. I documenti XML devono essere resi canonici, visto che esistono vari metodi per rappresentare dei dati utilizzando una sintassi XML. Basti pensare, per esempio, che la sintassi <foo/> è del tutto analoga a <foo></foo>. Ogni documento XML, quindi, deve in qualche modo essere reso in una forma canonica prima che venga firmato, altrimenti la firma non potrà essere verificata. Esistono vari metodi per rendere canonici i documenti XML e l'elemento <SignedInfo/> specifica quale è stato utilizzato.
L'effettiva firma utilizzata per firmare il documento è contenuta nell'elemento <SignatureValue/>. La struttura XML non contiene il contenuto del documento firmato, ma contiene invece, nell'elemento <Reference/>, un riferimento. Nel nostro esempio, l'elemento <Reference/> usa un URI per puntare al documento stesso nel quale è presente se stesso indicando un valore uguale a quello specificato nell'attributo ResponseID del Tag Response. L'elemento <Reference/> specifica anche il tipo algoritmo di digest utilizzato per creare l'hash (<DigestMethod>) ed il valore dell'hash stesso (<DigestValue>).
Per quanto riguarda il **digest** in questo contesto a volte è sufficiente poter determinare se un documento o un messaggio è stato modificato, maliziosamente o per errore, senza dover ricorrere ad algoritmi crittografici dispendiosi dal punto di vista delle risorse di calcolo richieste. In questi casi può essere utilizzata una tecnica matematica chiamata message digest (o anche hash).
Un messaggio digest è una sequenza di lunghezza fissa di bit generata a partire da un messaggio a lunghezza variabile tramite una speciale trasformazione matematica avente le seguenti tre importanti proprietà:

1. *irreversibilità*: sottoponendo il digest ad una trasformazione inversa non è possibile ricercare il messaggio originale. Questa è una ragionevole assunzione per ogni algoritmo che trasforma stringhe di lunghezza variabile in stringhe di lunghezza fissa relativamente corta, visto che la stringa corta non contiene informazioni sufficienti per rappresentare quella più lunga.
2. *non selezionabilità*: trovare un messaggio in grado di generare un dato digest dovrebbe essere matematicamente impossibile.
3. *unicità*: trovare due documenti che producono lo stesso messaggio digest dovrebbe essere matematicamente impossibile.

L'irreversibilità assicura che quando si trasmette un message digest è possibile essere certi che il suo contenuto rimarrà segreto.
La non selezionabilità ed unicità assicurano che non sia possibile sostituire con un messaggio diverso il messaggio con cui è stato creato un digest. Ciò è importante, visto che si sta utilizzando il digest per fornire evidenza dell'integrità di un messaggio. Se si riuscisse a generare un dato digest, si potrebbe infatti sostituire il messaggio originale senza che nessuno se ne accorga.
La tabella che segue enumera alcuni algoritmi per la generazione di digest, la lunghezza in bit dei digest generati e l'autore o detentore dell'algoritmo.

Algoritmo	Dimensione del digest (bit)	Detentore
MD2	128	RSA Data Security, Inc.
MD4	128	RSA Data Security, Inc.
MD5	128	RSA Data Security, Inc.
SHA	160	Governo USA
SHA-1	160	Governo USA

Tabella 1 – Algoritmi per la generazione di digest

Per quanto riguarda la **firma digitale** si utilizza un sistema a chiave pubblica: se si codifica un documento con la chiave privata, chiunque lo può decodificare, sempre che possegga la corrispondente chiave pubblica. A patto che il detentore della chiave provata la abbia tenuta la sicuro, ciò costituisce una forte evidenza sul fatto che questo sia stato effettivamente colui che ha codificato il documento in questione e può quindi servire come firma.
Questo metodo, tuttavia, ha parecchi svantaggi:

- il documento firmato viene reso illeggibile a meno che non venga decodificato con la chiave pubblica. Ciò è troppo scomodo in situazioni in cui si voglia verificare le firme solo occasionalmente;
- la firma ed il documento sono inseparabili. Non c'è modo di inviare la stessa firma insieme ad un altro documento.

Si può superare questi svantaggi se si combina la crittografia a chiave pubblica con i message digest. Poiché un digest è (almeno entro certi limiti crittografici) unico per ogni dato messaggio, si può creare un digest di un messaggio o documento e quindi firmare il digest invece del messaggio o documento originale. Il messaggio rimane leggibile e firma e messaggio divengono separati e separabili. Questo è ciò che avviene nella response SAML.
Per verificare la firma si può usare la chiave pubblica del mittente per decodificare il digest e quindi applicare lo stesso algoritmo di digest al messaggio firmato. Se i due digest coincidono, allora il messaggio è lo stesso di quello che è stato firmato dal mittente.
Le firme digitali implementate in questo modo forniscono l'evidenza dell'integrità di un documento, visto che se il documento venisse modificato, intenzionalmente o no, il destinatario calcolerebbe un digest diverso da quello originale. La firma digitale fornisce anche un meccanismo di non ripudiabilità, poiché è chiaro che la persona che ha generato il digest originale ha avuto accesso ad una versione identica del documento e, a patto che abbia mantenuto sotto controllo la propria chiave privata, è l'unica persona in grado di produrre la firma.

Nel caso venga usata per scopi di firma, la chiave privata viene a volte chiamata chiave di firma mentre la chiave pubblica chiave di verifica. Tecnicamente, queste chiavi funzionano esattamente allo stesso modo di una normale coppia di chiavi e la terminologia indica semplicemente lo scopo di ognuna delle due chiavi.

2.3.4 DISPOSIZIONE DELLE IDENTITÀ

La tecnologia SAML risolve il problema di come server distinti possono scambiarsi dati di identità, ma non come le identità sono organizzate sui server.
Nel caso specifico del Single Sign-On tra il dominio D1 ed il dominio D2 è necessario che in quest'ultimo l'informazione proveniente dalla Response SAML, cioè il Codice fiscale dell'utente, sia memorizzato come attributo specifico dell'utente nel dominio D2. Questo significa che ogni utente del dominio D2 memorizzato nel repository LDAP deve avere un attributo dal nome **codfiscale** contenente appunto il codice fiscale dell'utente la cui username è valorizzata nell'attributo **uid**.
Per situazioni pregresse può esiste anche la possibilità che più utenze D2 siano associate alla stessa utenza del dominio D1. Questo significa che possono esistere per un Codice Fiscale più di una username D2.

3. ARCHITETTURA DELLA SOLUZIONE

In questo capitolo viene esaminata la soluzione che fa uso del concetto di EAI di WebSEAL di IBM.

3.1 EAI: INTERFACCIA DI AUTENTICAZIONE ESTERNA SU WEBSEAL

Tivoli Access Manager fornisce la modalità denominata "interfaccia di autenticazione esterna" (EAI External Authentication Interface) che consente di estendere il processo di autenticazione per WebSEAL. Tale interfaccia, quindi, permette ad un servizio remoto indipendente di gestire il processo di autenticazione. Le informazioni sull'identità restituite dal servizio sono utilizzate dal WebSEAL per generare le credenziali dell'utente finale.
Questa funzionalità di autenticazione estesa è simile al meccanismo dei moduli di autenticazione personalizzati fornite dalle API C di autenticazione esterna della sicurezza Web del TAM. La differenza sostanziale sta nel fatto che l'interfaccia di autenticazione esterna restituisce le informazioni sull'identità dell'utente in delle header di risposta HTTP anziché tramite le interfaccia del modulo di autenticazione.
Il meccanismo EAI non sostituisce comunque i moduli di autenticazione già incorporati in WebSEAL, ma può offrire una capacità di autenticazione più conveniente e flessibile la dove le esigenze lo richiedano. Occorre notare che a differenza delle API C di autenticazione esterna, l'interfaccia di autenticazione esterna può essere implementata con applicazioni scritte in qualsiasi linguaggio, incluso Java.
In ultima analisi quando si utilizza una EAI l'operazione di autenticazione è eseguita esternamente a WebSEAL che, nel caso trattato in questo documento, da un'applicazione JEE situata su un server WebSphere remoto collegata mediante giunzione.

3.2 CASI D'USO DEL PROCESSO DI AUTENTICAZIONE

Il processo di autenticazione prevede due Use-Case per l'utente come raffigurato dalla figura che segue:

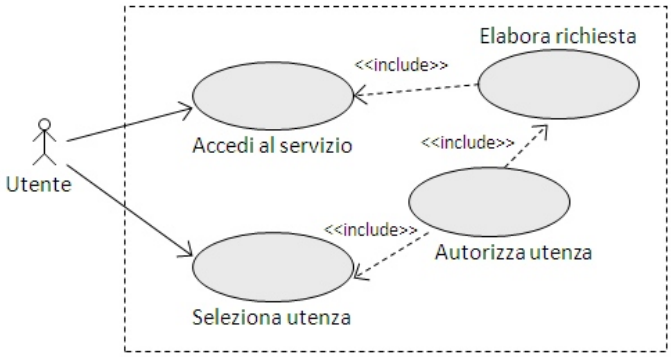

Figura 1 – Casi d'uso

Il primo è utilizzato per accedere al servizio EqS scelto, mentre il secondo è utilizzato quando l'utente deve selezionare l'utenza EqS nel caso il suo identificativo (codice fiscale) corrisponde a più username EqS.
Il caso d'uso di accesso al servizio include quello che ha il compito di elaborare la richiesta proveniente dall'utente (verifica del token SAML e della presenza dell'acronimo del servizio richiesto) che a sua volta include il caso d'uso di autorizzazione la username associata all'utente.

3.3 DIAGRAMMA DELLE ATTIVITÀ DELL'ELABORAZIONE DELLA RICHIESTA

Questo caso d'uso è quello più complesso dell'intero processo e nella figura che segue ne è raffigurato il diagramma delle attività che permette di comprendere quali sono i controlli che implementa.

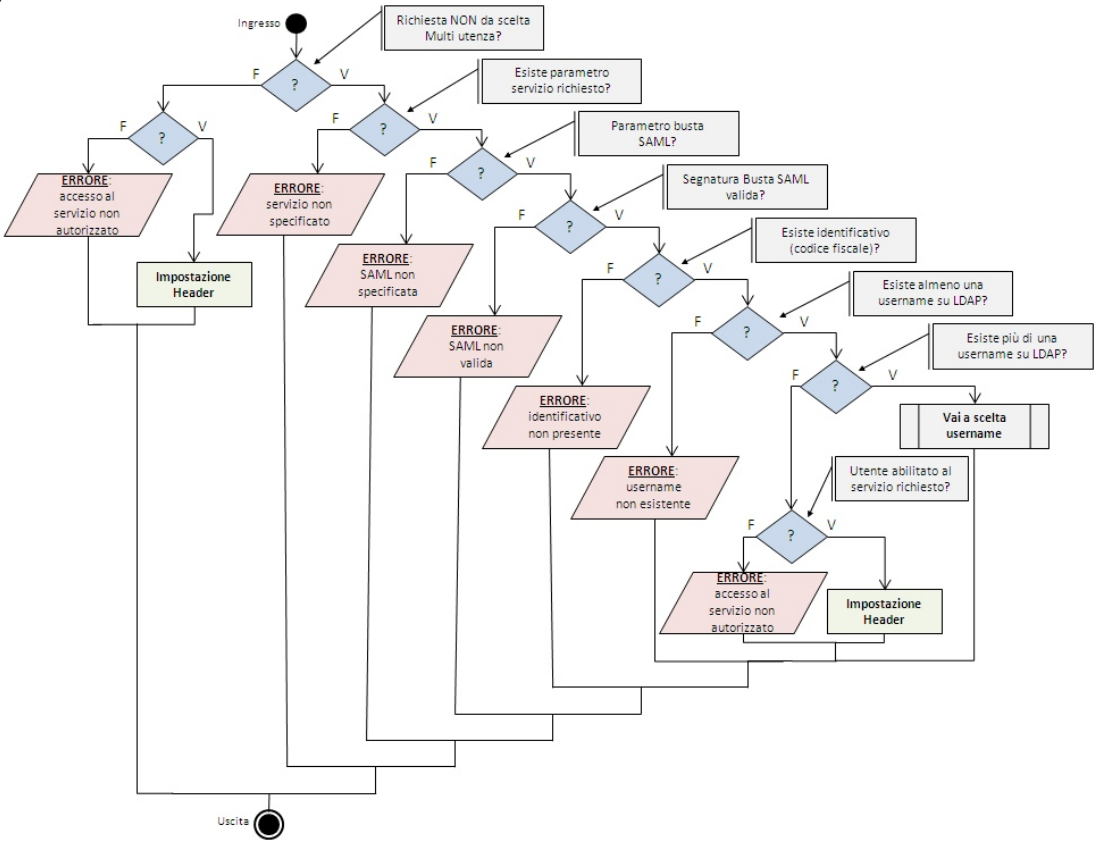

Figura 2 – Diagramma delle attività

In dettaglio la sequenza dei controlli ha il seguente flusso:

- Controllo se la richiesta proviene dal modulo di scelta della username con il quale l'utente vuole accedere al sistema nel caso il suo codice fiscale sia associato a più di una username EqS. Se la richiesta non è di questo tipo si procede con i controlli successivi, altrimenti la username selezionata dall'utente è impostata come header che verrà poi autorizzata dal WebSEAL.
- Controllo di esistenza nella richiesta del parametro contenente l'acronimo del servizio a cui l'utente vuole accedere. In caso di mancanza di tale parametro all'utente viene visualizzata una apposita pagina di errore.
- Controllo si esistenza della parametro contenente la Response SAML di autenticazione codificata in Base64. Se tale parametro non esiste all'utente viene visualizzata una apposita pagina di errore.
- Controllo se la segnatura riportata nella Response SAML è conforme con il certificato pubblico rilasciato. In caso di errore all'utente viene presentata una apposita pagina di errore.
- Controllo dell'esistenza dell'identificativo utente (codice fiscale) all'interno dell'asserzione SAML. Se il Tag contenente l'informazione prevista non esiste viene visualizzata una opportuna pagina di errore.
- Controllo se al codice fiscale dell'utente corrisponde almeno una username del dominio EqS. Se non esiste all'utente viene notificato l'errore con una pagina di che lo informa sul fatto che la sua utenza non esiste nel dominio EqS.
- Controllo se al codice fiscale dell'utente corrisponde una sola utenza. Se si ha una sola utenza si passa al controllo successivo, altrimenti se esistono più username allora questa lista sarà presentata all'utente in modo che possa selezionare quale intende utilizzare per accedere al servizio.
- Controllo se l'utente è abilitato al servizio richiesto. Il controllo consiste nel verificare se la username compare come membro del gruppo LDAP con nome uguale all'acronimo del servizio richiesto. Se il controllo ha esito positivo e quindi l'utente è abilitato al servizio richiesto la username è impostata come header della risposta in modo che WebSEAL possa autorizzare l'accesso, altrimenti l'errore è segnalato all'utente.

3.4 FLUSSO DEL PROCESSO DI AUTENTICAZIONE ESTERNA

Nel diagramma di sequenza che segue viene illustrato il flusso base del processo di autenticazione utilizzando l'interfaccia di autenticazione esterna., cioè quando al codice fiscale corrisponde una sola username. Gli attori di questo scenario di flusso del processo sono:

- Client Web
- WebSEAL.

- Applicazione EAI di autenticazione esterna collegata con una giunzione al WebSEAL.
- Il servizio a cui l'utente vuole accedere.

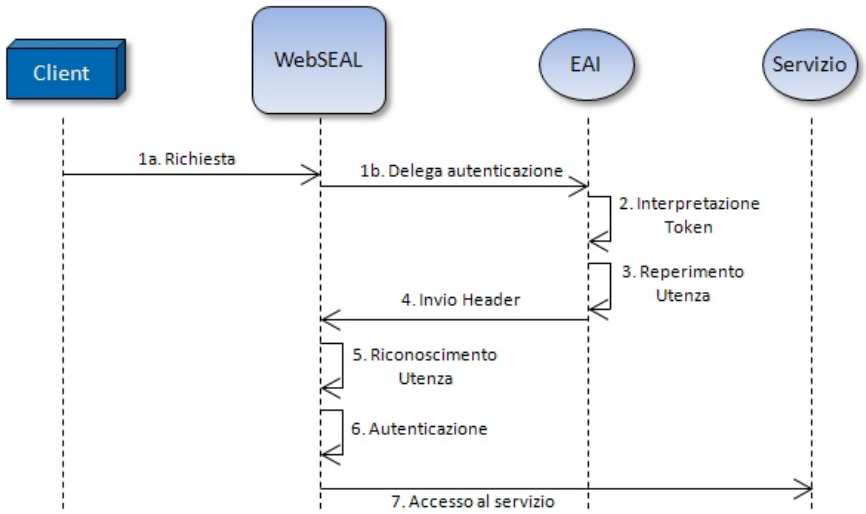

Figura 3 – Sequenza nel caso di singola username

1. In questa fase è avviato il processo di autenticazione.
    a. L'utente con il suo browser richiede di essere attestato nel sistema transitando per la giunzione appositamente predisposta per l'autenticazione esterna.
    b. Il WebSEAL dirotta la chiamata del client verso l'applicazione EAI. Nella chiamata http sono presenti I seguenti dati:
        - Le informazioni sull'identità dell'utente (codice fiscale contenuto nella Response SAML).
        - L'acronimo del servizio che desidera consultare.
2. L'applicazione EAI interpreta le informazioni dell'identità dell'utente verificandone la correttezza secondo i controlli descritti nel paragrafo precedente.
3. L'applicazione EAI effettua un mapping tra l'informazione dell'identità dell'utente pervenuta (codice fiscale) e l'insieme delle username presenti nel registro LDAP del TAM. Il requisito fondamentale è quindi che nel registro LDAP ogni username abbia come attributo secondario una informazione contenente il codice fiscale dell'utente come descritto nel paragrafo "Disposizione delle identità" del capitolo precedente .
4. La username corrispondente al codice fiscale dell'utente è impostata come intestazione nella risposta insieme al riferimento del servizio richiesto dall'utente.
5. WebSEAL recupera dalla header la username inviata dall'applicazione esterna e autentica l'utente.

6. WebSEAL autentica l'utente.
7. Il client è redirezionato automaticamente verso il servizio da lui richiesto.

Il flusso nel caso di più username associate allo stesso codice fiscale è illustrato nella figura che segue:

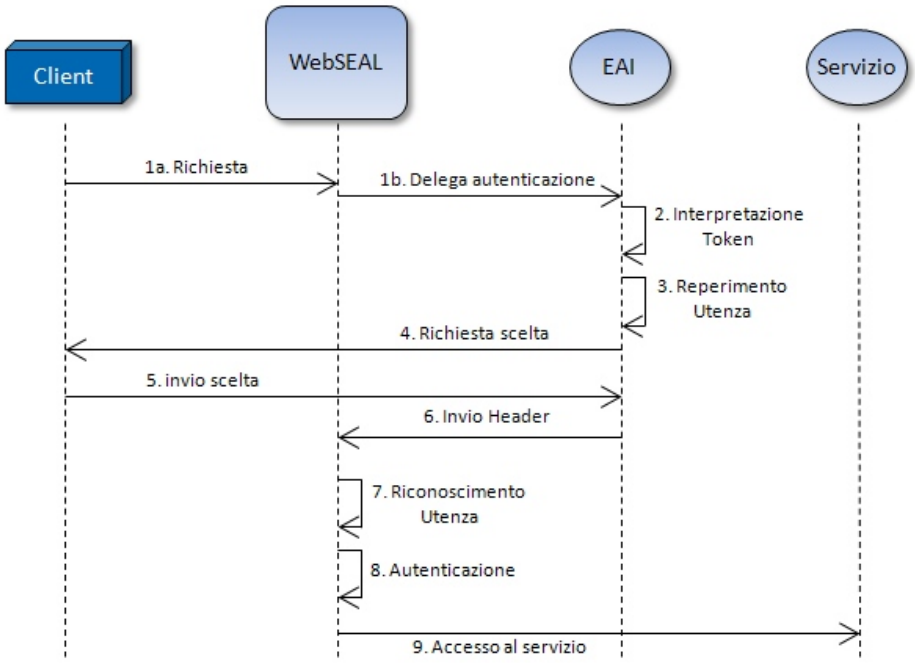

Figura 4 – Sequenza nel caso di username multipla

Il processo è del tutto simile al precedente tranne per il fatto che il passo 3 si ottiene una lista di username composta da più elementi. Per tale ragione è necessario che l'utente selezioni la username con la quale intende accedere al servizio. La scelta da lui effettuata è invita all'applicazione EAI ed i passi che seguono sono del tutto analoghi al caso precedente.
Nella figura che segue è riportato il diagramma di sequenza della navigazione nel caso in cui l'utente debba selezionare una username:

SSO:
SINGLE SIGN-ON TRA EQ ED EQS

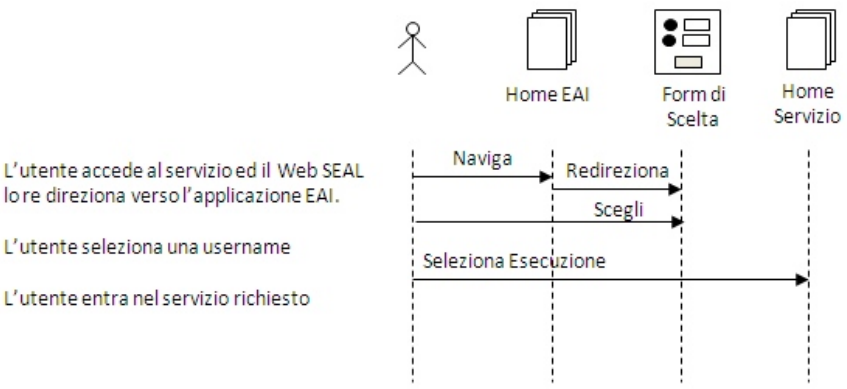

Figura 5 – Sequenza di navigazione nel caso di username multipla

3.5 TOPOLOGIA DEI COMPONENTI

Nella figura che segue viene riportata la topologia dei componenti necessari per la soluzione proposta.

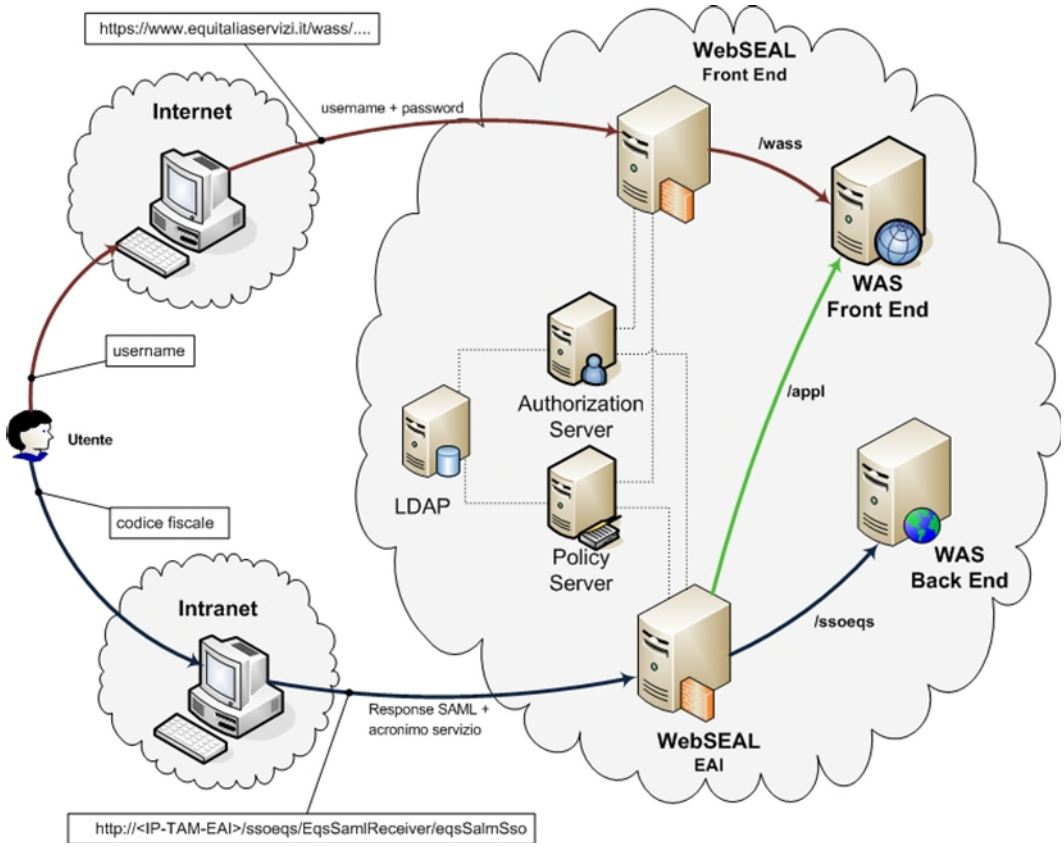

Figura 6 – Topologia dei componenti

La parte superiore della figura riporta la situazione attuale per accedere alle applicazioni installate sui WAS di Front End. Gli utenti accedono da internet mediante connessione https SSL sulla giunzione /wass. Il riconoscimento dell'utente è fatta mediante username e password transitando sul WebSEAL di Front End.
Nella parte bassa della figura è riportata la situazione che riguarda l'accesso delle applicazioni per gli utenti provenienti dalla intranet. L'utente nella prima chiamata ha nella sua richiesta la Response SAML e l'acronimo del servizio richiesto come parametri in POST. Il suo riconoscimento avviene grazie all'applicazione EAI installata sul server WAS di Back End che viene contattata dal WebSEAL EAI sulla giunzione /ssoeqs. Dopo il processo di autenticazione l'utente si collega alle applicazioni installate sul WAS di Front End mediante la giunzione /appl.
Da notare come il WebSEAL di Front End ed il WebSEAL di Back End condividono lo stesso Policy Server e Authorization Server che a loro volta utilizzano un unico LDAP come repository delle utenze di ADER.
In ultima analisi occorre osservare che un utente può accedere ad un determinato servizio sia da internet che dalla intranet. La modalità di autenticazione è naturalmente differente, ma il risultato finale è il medesimo, ovvero l'accesso all'applicazione. Le politiche Eqs sulle password non devono essere modificate: se un utente ha la password scaduta se transita per internet gli sarà chiesto di modificarla come già avviene oggi, mentre se transita dalla intranet non dovrà intraprendere nessuna azione.