Indice

Elenco delle figure **5**

Elenco delle tabelle **7**

Introduzione **9**

1 Identity e Access Management **12**

1.1 Cenni storici sull'Identity e Access Management 12

1.2 Stato dell'arte 13

1.3 Che cosa si intende per Identity e Access Management 15

1.4 Come gestire l'Identity e Access Management 21

1.5 Tassonomia dell'Identity Management 23

2 Identity e Access Management in Federazione **26**

2.1 Che cosa si intende per federazione 26

2.2 Compiti della federazione 30

2.3 Tassonomia della federazione 31

2.4 Cenni storici sull'Identity e Access Management Federato 32

2.5 Stato dell'arte 33

3 Progetti vari di Identity Management Federato **36**

3.1 Analisi dei progetti 36

3.2 L'Identity Management nelle Università 38

3.2.1 Progetto IDEM 39

3.3 L'identity Management nella pubblica amministrazione 41

3.3.1 Progetto ICAR 43

3.4 L'Identity Management nelle aziende 45

3.5 Analisi critica dei progetti esaminati 47
4 Gestione dell'Identità digitale in federazione attraverso casi d'uso 49
4.1 Definizione dell'identità digitale 49
4.2 Architettura generale 50
4.3 Componenti e ciclo di vita dell'identità 54
4.4 I sistemi di gestione dell'identità 56
5 Realizzazione dell'infrastruttura 59
5.1 Tecnologie utilizzate 59
5.2 Standard sul formato e lo scambio di credenziali 60
5.2.1 SAML 61
5.2.1.1 Asserzioni 62
5.2.1.2 Protocolli 63
5.2.1.3 Binding 64
5.2.1.4 Profili 66
5.2.2 X.509 66
5.3 Single Sign On 68
5.4 Tecnologie applicate all'Identity Management Federato 69
5.4.1 Shibboleth 69
5.4.2 Estensioni dello standard SAML 75
5.4.3 SimpleSAMLphp 76
5.5 Infrastruttura di test 76
6 Problematiche relative all'Identity Management 79
6.1 Problematiche gestionali 79
6.2 Problematiche tecniche 80
7 Risultati ottenuti e sviluppi futuri 82
Conclusioni 86
Bibliografia 88
Glossario 91

Appendice A..... 94

Appendice B..... 125

Elenco delle figure

FIGURA 1: SISTEMA DI IDENTITY MANAGEMENT 16
FIGURA 2: ACCESS MANAGEMENT 17
FIGURA 3: ORGANIGRAMMA 22
FIGURA 4: MODELLO DI IDENTITÀ ISOLATO 24
FIGURA 5: MODELLO CENTRALIZZATO 24
FIGURA 6: IDENTITY E ACCESS MANAGEMENT SENZA FEDERAZIONE 28
FIGURA 7: IDENTITY E ACCESS MANAGEMENT FEDERATO 29
FIGURA 8: ARCHITETTURA GENERALE 50
FIGURA 9: USE CASE 1 51
FIGURA 10: FLUSSO USE CASE 1 51
FIGURA 11: USE CASE 2 52
FIGURA 12: FLUSSO USE CASE 2 53
FIGURA 13: CICLO DELL'IDENTITÀ 54
FIGURA 14: FLUSSO DI VITA DELL'IDENTITÀ SENZA ELIMINAZIONE PRIVILEGI 55
FIGURA 15: FLUSSO DI VITA DELL'IDENTITÀ CON ELIMINAZIONE PRIVILEGI 55
FIGURA 16: FRAMEWORK SAML 61
FIGURA 17: PROFILI, BINDINGS, PROTOCOLLI E ASSERZIONI SAML 62
FIGURA 18: SINTASSI ASSERZIONE SAML 63
FIGURA 19: SAML BINDING HTTP POST 65
FIGURA 20: STRUTTURA DELLA CA 67
FIGURA 21: COMUNICAZIONE FRA IDP, UA E SP 71
FIGURA 22: IDENTITY PROVIDER 72
FIGURA 23: SERVICE PROVIDER 74
FIGURA 24: INFRASTRUTTURA DI TEST 77
FIGURA 25: ALBERO DELLE ENTRY DI LDAP 95
FIGURA 26: COMUNICAZIONE IN SHIBBOLETH TRA IDP E SP 102
FIGURA 27: SERVICE PROVIDER DI SHIBBOLETH 111
FIGURA 28: DIVERSE ORGANIZZAZIONI IN UN'UNICA FEDERAZIONE 115
FIGURA 29: FORM DI AUTENTICAZIONE DI SHIBBOLETH 120
FIGURA 30: LOGIN EFFETTUATO TRAMITE SHIBBOLETH 121
FIGURA 31: METADATI DI JOOMLA 124
FIGURA 32: PROGETTO ICAR 128

Elenco delle tabelle

TABELLA 1: PROGETTI RIGUARDANTI L'IDENTITY E ACCESS MANAGEMENT 14
TABELLA 2: FEDERAZIONI 33
TABELLA 3: GRAFICO FEDERAZIONI IN EUROPA 34
TABELLA 4: RISORSE PER FEDERAZIONE 34
TABELLA 5: RISORSE PER FEDERAZIONE 35
TABELLA 6: PROGETTI DI AIM 37
TABELLA 7: TECNOLOGIE PER L'AIM 38
TABELLA 8: LISTA VENDOR 47

Introduzione

In un qualsiasi contesto organizzativo dove è necessario gestire numerose risorse informatiche, come credenziali di accesso, portali, domini, servizi, utenti, nasce l'esigenza di coordinare in maniera efficace le differenti identità che ne fanno uso. Infatti, in questi sistemi si evidenziano le problematiche relative alla profilazione delle password, alla fruizione dei servizi in ottica inter-organizzativa e alla realizzazione di una infrastruttura che sia in grado sopperire queste difficoltà. Inoltre non esistono meccanismi precisi per identificare il soggetto e il servizio con il quale si sta comunicando: è sempre mancata cioè una infrastruttura per la gestione delle identità, e di conseguenza le imprese, le Pubbliche Amministrazioni e le singole persone hanno nel tempo dovuto sviluppare una serie di soluzioni isolate, parziali e spesso incompatibili per risolvere i loro contingenti problemi di identificazione nelle transazioni. Ecco dunque la necessità di un elemento federativo in grado di coordinare tutte queste identità e con i relativi servizi. Questo vuol dire riconsiderare le tematiche e le alternative non più circoscritte alla singola organizzazione ma al loro insieme. In questo lavoro si effettua una prima analisi su quelle che sono le tecnologie e i principali protocolli utilizzati per poi proporre una propria infrastruttura federata molto flessibile in grado di essere applicata in diversi ambienti.

Gestione degli accessi e delle identità una visione d'insieme

Un sistema di Identity e Access Management pone al centro del dibattito la questione relativa alla gestione delle identità e degli accessi. Studiare le diverse tipologie e applicare le differenti soluzioni significa ottenere dei vantaggi che stanno alla base del successo aziendale e organizzativo. Non a caso questa necessità è particolarmente sentita all'interno delle pubbliche amministrazioni, come gli enti Regionali o delle Università, dove l'efficiente gestione delle identità sta alla base della organizzazione stessa. Poter coordinare in maniera efficiente i diversi ruoli in diversi contesti è quello che il sistema di Identity e Access Management si prefigge di risolvere.

Motivazione per lo studio

In quest'ottica esistono diverse soluzioni che affrontano queste problematiche. Se diamo uno sguardo ai prodotti software ci accorgiamo che sul mercato vi sono una ventina di applicativi che hanno tali caratteristiche. Inoltre se concentriamo la nostra attenzione sulle federazioni ci rendiamo conto che a livello europeo ne esistono una decina. Alcune di queste soluzioni risultano ormai obsolete e non più applicabili alle realtà moderne. Dunque è bene far chiarezza sulle diverse tecnologie utilizzate per poi progettare una propria soluzione.

Obiettivo dello studio

Lo scopo di questa tesi è stato quello inizialmente di studiare e analizzare tutti gli aspetti riguardanti l'Identity e Access Management con riferimenti alle diverse tecnologie adoperate in differenti realtà. Il passo successivo è stato quello di realizzare una infrastruttura con l'obiettivo di renderla il più possibile flessibile ed esportabile in differenti contesti. Inoltre si è scritto un piano di progetto che andrebbe attuato se l'azienda o l'ente regionale in questione intende affrontare una migrazione ad una infrastruttura di Identity e Access Management federata.

Struttura della tesi

Nel primo capitolo sono introdotti i concetti che stanno alla base dell'Identity e Access Management con riferimenti storici e stato dell'arte.

Nel secondo capitolo si riprendono gli argomenti affrontati in precedenza ma sotto un'ottica diversa: quella di federazione, intesa come insieme di organizzazioni.

Nella terza parte vengono analizzati diversi progetti nei quali l'elemento federazione e l'interoperabilità applicativa sono i principali obiettivi.

Nel quarto capitolo si affronta l'identità digitale attraverso un casi d'uso con i quali si analizzano tutte le fasi critiche che un'utenza subisce all'interno di una organizzazione.

Nel quinto capitolo si discutono gli argomenti tecnici indispensabili per la realizzazione di un'infrastruttura di Identity e Access Management federato.

Nel sesto capitolo sono descritte le principali problematiche affrontate relative all'Identity e Access Management.

Infine nel settimo capitolo sono indicati i progetti e sviluppi futuri sempre nell'ambito dell'identità federata.

Capitolo 1

1 Identity e Access Management

Il problema del riconoscimento dell'utente e dell'attribuzione dei relativi permessi è una difficoltà tipica in tutte le situazioni in cui vengono fornite particolari risorse informatiche, come risorse web, accesso a Internet, postazioni computer, posta elettronica, ecc. dove è richiesto una autenticazione e autorizzazione.

Queste problematiche sono gestite da un sistema di Identity e Access Management che si può quindi definire come l'intero processo - applicazione di policy appropriate e impiego di strumenti tecnologici - volto a gestire le informazioni riguardanti le identità degli utenti e controllarne l'accesso alle risorse informatiche.

In questo capitolo sono affrontati i seguenti temi:

- Cenni storici sull'Identity e Access Management;
- Stato dell'arte;
- Che cosa si intende per Identity e Access Management;
- Come gestire l'Identity e Access Management;
- Tassonomia dell'Identity Management.

1.1 Cenni storici sull'Identity e Access Management

Il concetto di Identity e Access Management è costituito da diversi termini ognuno dei quali nel tempo ha assunto specifici significati. Infatti, i primi due, Identity e Access, fanno riferimento a tutte quelle informazioni d'identificazione personale. Questi argomenti furono affrontati per primi dall'OECD[1] Organisation for Economic Co-operation and Devolpment, organizzazione che ha come obiettivo l'integrazione e cooperazione

[1] Sito di riferimento http://www.oecd.org/

economica finanziaria, e dal NIST[2] National Institute of Standards and Technology, un'agenzia di governo americana che si occupa della gestione delle tecnologie, nelle linee guida riguardanti la protezione delle informazioni personali [1].

Il secondo termine, Management, è utilizzato in differenti contesti a seconda dei quali assume ragionevolmente significati diversi. Nel nostro caso fa riferimento alla gestione degli oggetti, attributi ed elementi che costituiscono un'identità. Questi oggetti nello specifico sono definiti OID Object Identifier che hanno l'obiettivo di mappare gli attributi relativi ad una identità.

1.2 Stato dell'arte

Con il Seventh Research Framework Programme FP7[3], un programma istituito dai principali capi di stato europei per il finanziamento di progetti dal 2007 sino al 2013, sono nati diversi piani di lavoro che si riferiscono all'Identity Management. Primo su tutto è il progetto PICOS[4] creato con lo scopo di analizzare e studiare lo stato dell'arte riguardante la privacy, fiducia e gestione delle identità. Un interessante lavoro a riguardo è stato prodotto da Crespo, Mendez e Liesebach dal titolo "Picos – Privacy and Identity Management for Community Services" [2]. Un successivo progetto, sempre finanziato dal FP7 è SWIFT[5] che si focalizza nell'estendere le funzioni di identità alla rete con concetti di privacy, tecnologie e usabilità come chiavi per integrare servizi e infrastrutture di trasporto dei dati. A riguardo sembra molto interessante la pubblicazione dal titolo "Integrating User Identity Management Systems with the Host Identity Protocol" [3]. In seguito, diversi progetti si svilupparono come FIDIS[6], GUIDE[7] e PRIME[8] riguardanti diversi aspetti dell'Identity Management.

[2] Sito di riferimento http://www.nist.gov/index.html
[3] Sito di riferimento http://cordis.europa.eu/fp7/home_en.html
[4] Sito di riferimento https://www.picos-project.eu/
[5] Sito di riferimento http://www.ist-swift.org/
[6] Sito di riferimento http://www.fidis.net/
[7] Sito di riferimento http://istrg.som.surrey.ac.uk/projects/guide/
[8] Sito di riferimento https://www.prime-project.eu/

Nella *Tabella 1* si è effetuato un confronto fra i diversi progetti elencati precedentemente evidenziandone le caratteristiche e i vantaggi / svantaggi di ognuno di essi.

Progetto	Caratteristiche	Vantaggi / Svantaggi
PICOS	Ricerca, sviluppo e progettazione di un sistema di identity management e privacy compliance che supporta le comunicazioni via mobile.	Vantaggi: Numerosa documentazione e paper. Uno dei primi progetti orientato al mobile. Numerosi partner di sviluppo come: Hp, T-mobile, it-objects. Svantaggi: solo ambiente mobile.
SWIFT	Sviluppo soluzioni per la convergenza di identity management tra reti, servizi, applicazioni, operatori, fornitori di servizio.	Vantaggi: sviluppo di nuovi di layer per l'interfacciamento di diverse tecnologie e protocolli. Svantaggi: Documentazione tecnica poco chiara.
FIDIS	Ricerca nei campi delle identità, profilazione, interoperabilità e gestione dei sistemi, forensic, privacy, hight Tech e mobility	Vantaggi: contesto fortemente concentrato a quello europeo. Numerosi paper divisi per aree tematiche. Svantaggi: Nessun accenno di software, piattaforme realizzate o utilizzate.
GUIDE	Ricerca nella creazione di un sistema di identity management in contesto di eGovernament.	Vantaggi: Si analizzano anche aspetti non tecnologici come leggi, questioni socio-politiche e privacy che coinvolgono direttamente il cittadino. Svantaggi: Nessun accenno di software, solamente paper.
PRIME	Sviluppo e ricerca di una soluzione per l'aumento della privacy in un sistema di identity management.	Vantaggi: Raccoglie numerosi tutorial, paper, prototipi. Dispone anche di una ontologia. Svantaggi: Risulta chiuso e poco aggiornato.

Tabella 1: Progetti riguardanti l'identity e access management

Risulta importante sottolineare come a livello accademico le problematiche relative all'Identity e Access Management sono state ampliamente discusse, analizzate, sperimentate anche se purtroppo difficilmente si è avuto un riscontro pratico nella realtà.

1.3 Che cosa si intende per Identity e Access Management

> *"Identity management is involved in the identification, implementation, administration, and termination of identities with access to information systems, buildings, and data within an organization. Identities, for example, are company employees, system users, business partners and technical objects such as network printers and technical users."*[9] [9]

L'Identity Management è pertanto una vasta area tematica legata alla gestione dell'identità digitale degli utenti, in contesto applicativo molto ampio: da quello di e-Business visto in un contesto di azienda, di impresa, di negozio elettronico, fino a quello di e-Governament inteso come pubblica amministrazione.

I principali temi afferenti all'Identity Management sono:

- **Autenticazione**: non solo tramite l'uso di username e password, attraverso l'uso di certificati;
- **Confidenzialità**: capacità del sistema di impedire che una terza parte intercetti e sfrutti dati che si stanno ricevendo o trasmettendo;
- **Autorizzazione:** restringere l'accesso a informazioni private o consentire l'accesso a servizi specifici;
- **Integrità dei dati:** per essere sicuri che nessuno intercetti i dati che si stanno scambiando, può essere importante sapere che nessuno li abbia alterati durante la trasmissione;
- **Prova della fonte:** è possibile compiere specifiche transazioni in cui i dati inviati con la firma digitale sono codificati in modo da dimostrare che i dati sono effettivamente stati inviati;
- **Non ripudio:** fornire la prova incontestabile di un'avvenuta spedizione o di un'avvenuta ricezione di dati in rete**;**
- **User provisioning**: allocazione e revoca di privilegi e profili utente;
- **Formato e interoperabilità**: le identità digitali devono poter comunicare attraverso un unico protocollo di comunicazione;

[9] Sito di riferimento: http://www.secude.com/html/index.php?id=1242&L=9

- **Single Sign On**[10]: sistema specializzato che permette a un utente di autenticarsi una sola volta e accedere a tutte le risorse informatiche alle quali è abilitato;
- **Servizi di directory**: cartelle riservate a diverse tipologie di utenze.

Inoltre, il termine Identity Management viene spesso associato a sistemi informatici che integrano - anche parzialmente - al proprio interno le funzionalità appena elencate.

Nella *figura 1* si esprime il concetto di Identity Management, mediante il quale ad una entità nel momento in cui vengono associati degli attributi si identifica in una identità. Dunque l'identità è relativa all'assegnazione di attributi diversificati al seconda del contesto in cui essa opera.

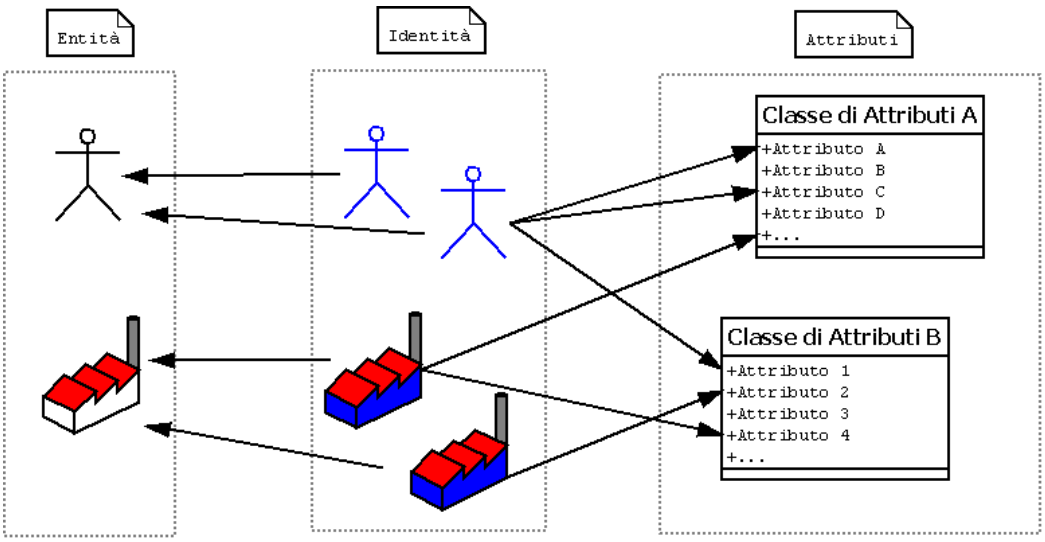

Figura 1: Sistema di Identity Management

L'Access Management si esplicita in un complesso di applicazioni e sistemi che sono utilizzati per gestire l'autenticazione degli utenti, l'accesso - o la restrizione dell'accesso - alle risorse, i profili, le password, e altri attributi che aiutano a definire ruoli e profili degli utenti come visibile nella *figura 2*.

[10] Più semplicemente definito SSO

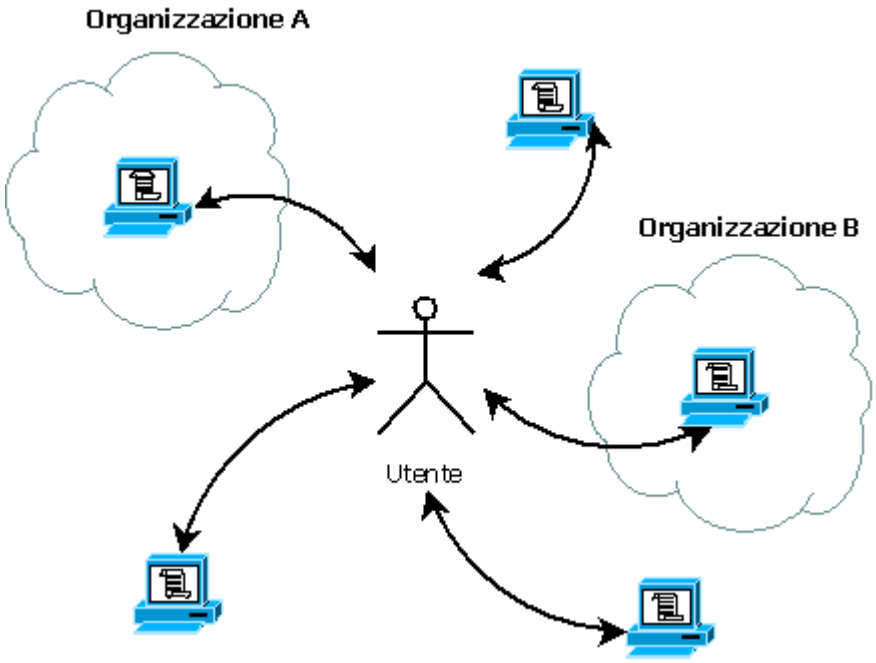

Figura 2: Access Management

Per risolvere questi problemi sia dell'Access sia dell'Identity Management esistono due diverse metodologie:

1. Ogni risorsa gestisce in modo autonomo l'autenticazione e l'autorizzazione ai servizi offerti;
2. La gestione degli accessi è centralizzata.

Nella realtà si possono avere anche casi misti, in cui alcune risorse hanno una gestione degli accessi centralizzata ed esistono risorse che gestiscono gli accessi in proprio; questo caso si può chiaramente ricondurre al primo punto poiché non si ha una vera e propria centralizzazione degli accessi ma sottoinsiemi di risorse che gestiscono autonomamente l'autenticazione e l'autorizzazione.

Il primo metodo, in cui ogni risorsa gestisce in modo autonomo l'autenticazione, comporta evidentemente diversi problemi dal punto di vista del gestore delle risorse:

- le informazioni delle identità sono replicate per ogni risorsa;
- l'aggiunta di una nuova risorsa comporta l'aggiunta di un'intera infrastruttura di accesso;

- la replica delle informazioni su molti sistemi significa dover gestire la sicurezza in molti punti, quindi si ha un minor controllo sulla sicurezza;

Mentre dal punto di vista dell'utente:

- ogni utente si trova ad avere credenziali di accesso diverse secondo la risorsa a cui vuole accedere;
- a causa delle numerose password che si ritrova, è più facile che un utente scelga password troppo semplici o addirittura le annoti su foglietti, il che implica una minor sicurezza per i suoi dati.

Il secondo metodo, in cui la gestione degli accessi è centralizzata, sembra dunque una soluzione a tutti i problemi del primo, infatti l'amministrazione degli utenti viene fatta solo nell'infrastruttura centralizzata, eliminando tutti i problemi del fornitore di servizi. Inoltre poiché gli utenti sono mantenuti in un'unica infrastruttura, si ha una sola chiave di accesso per ogni utente, evitando così la proliferazione delle password. Tuttavia si incontrano problemi non trascurabili, tipici dei sistemi centralizzati: performance e affidabilità. Infatti portando in una sola infrastruttura tutta la gestione degli accesi è indispensabile valutare il carico di richieste a cui sarà sottoposto, quindi dimensionare opportunamente l'intero sistema in modo che riesca a far fronte a tali richieste in tempi sempre accettabili; Inoltre altro aspetto è l'affidabilità, in altre parole fare in modo che l'infrastruttura sia sempre presente anche in caso di guasti accidentali o compromissione del sistema.

Spesso accade che da una situazione dove ogni risorsa gestisce autonomamente le identità e gli accessi si emigri ad un sistema centralizzato per via dei vantaggi che tale sistema porta. Un problema che nasce durante questo passaggio è il consolidamento delle identità digitali. Infatti può capitare che le informazioni sugli utenti arrivino da più fonti ed è fondamentale prevedere una serie di meccanismi che raccolgano tutte le informazioni in un'unica struttura.

Di seguito sono mostrati i passi da compiere per il consolidamento delle identità:

- Analizzare i database esistenti e vedere quali sono autoritativi;

- Decidere quali informazioni prendere, mantenere ed eventualmente aggiungere;
- Consolidare (una persona può essere presente in più fonti) per creare quella che chiamiamo identità digitale;
- Tenere aggiornato automaticamente il database unificato.

I **benefici** ottenuti dal sistema dopo questo processo di consolidamento dell'identità sono diversi:

- Maggiore flessibilità nell'aggiungere una nuova risorsa o modifica di privilegi di accesso a un gruppo di risorse;
- Unico punto per la modifica dei dati;
- Logging consolidato permette di applicare regole di privacy, di conservazione dei dati di auditing, fare dei report, monitorare la sicurezza in modo efficace;
- Eliminazione del problema della disattivazione riguardante la gestione delle identità in sistemi disgiunti.
- Riduzione del numero di credenziali da conoscere da parte dell'utente;
- Organizzazione più efficace poiché può modificare più velocemente i diritti di accesso basandosi sui ruoli;
- Conforme rispetto alla legge sul trattamento dei dati personali[11].

Un altro aspetto fondamentale per l'infrastruttura di rete è il controllo degli accessi. In questo modo è possibile controllare chi può avere accesso alla rete e cosa può fare, quali servizi può effettivamente utilizzare, dopo aver eseguito l'accesso. Tipicamente con il termine AAA, Authentication, Authorization e Accounting, s'intende un framework attraverso il quale si può configurare gli accessi sulla rete.

[11] Decreto legislativo 30 giugno 2003, n°196 Art.1 e Art.3

AAA rappresenta quindi il principale metodo per eseguire il controllo di accesso e fornire i seguenti servizi:

Authentication

Fornisce il servizio d'identificazione degli utenti, attraverso molteplici modalità: login/password, challenge/response, encryption, a seconda del protocollo scelto.

L'autenticazione ci consente di identificare l'utente prima che sia autorizzato all'accesso di rete o ai servizi.

Authorization

Fornisce il servizio per il controllo remoto degli accessi, autorizzazione per utente, autorizzazione per gruppo, autorizzazione per servizio.

Si basa sulla creazione di un insieme di attributi che descrivono le policy associate all'utente. Un database locale o remoto contenente le informazioni per singolo utente servirà per comparare tale set d'informazioni con quelle presenti al suo interno, il risultato sarà ritornato ad AAA che determinerà le azioni che l'utente può o non compiere.

Accounting

Fornisce il servizio d'invio di dati utilizzati per la gestione dei costi, l'auditing, ossia la verifica del sistema di controllo, e il reporting d'identità, tempi di esecuzione, comandi eseguiti, numero di pacchetti e di byte inviati e ricevuti.

In questo modo possiamo tracciare le attività dell'utente, come i servizi utilizzati e le risorse di rete consumate. Questo consente la possibilità di un'analisi a posteriori dei dati collezionati con scopi di auditing e network management.

L'obiettivo finale dichiarato di una qualsiasi soluzione di Identity Access and Management è comunque quello di aumentare la produttività e la facilità d'uso del sistema informatico da parte degli utenti finali,

aumentare il livello generale della sicurezza, diminuendo i costi associati alla gestione degli utenti e delle loro identità, dei loro attributi e delle loro credenziali.

In sintesi, si vuole fornire all'utente uno strumento che supporti il processo che stabilisce chi ha accesso a quali risorse, l'assegnazione delle autorizzazioni, la loro modifica o revoca quando necessario, nonché la gestione del processo stesso e il monitoraggio delle attività, nel rispetto delle politiche di sicurezza.

1.4 Come gestire l'Identity e Access Management

All'interno di un'organizzazione ben strutturata come nelle pubbliche amministrazioni, mettere in luce le problematiche riguardanti l'Identity e Access Management agli organi dirigenziali è un'operazione fondamentale se si vuol iniziare un processo di ottimizzazione delle risorse.

In una qualsiasi azienda pubblica o privata esiste una classificazione dei ruoli ben distinta, tipicamente strutturata come nell'organigramma della *figura 3*. Risulta quindi essenziale la collaborazione e la sinergia tra le diverse figure professionali al fine di attuare un processo di ottimizzazione delle risorse mediante un'architettura di Identity e Access Management che coinvolga l'intera organizzazione.

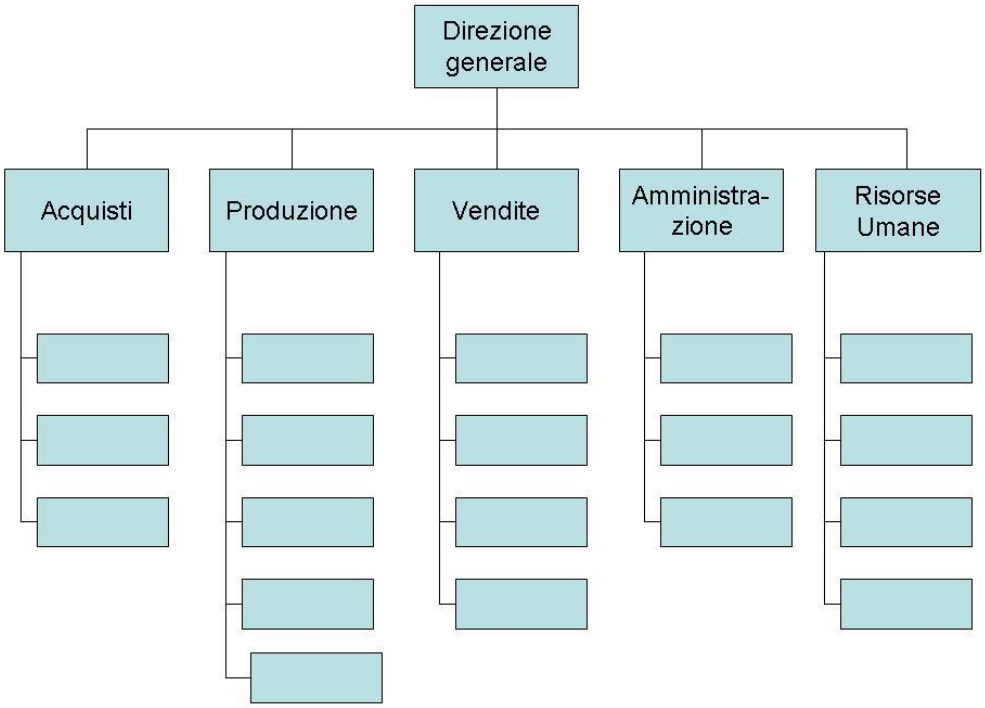

Figura 3: Organigramma

Il primo passo da svolgere è quello di porre la questione all'interno del consiglio di amministrazione, nel caso di aziende medio gradi, o di far presente alle figure dirigenziali, nel caso di piccole realtà, gli aspetti rilevanti che un'infrastruttura ben organizzata può portare. In questa fase è buona norma redigere un piano di progetto in cui si discutano esplicitamente questi punti:

- Le tempistiche con le quali s'intendere procedere. Fissando delle date per riunioni intermedie durante la fase di sviluppo dell'architettura si è in grado di monitorare al meglio i lavori compiuti e focalizzare gli obiettivi;
- Le componenti tecnologiche utilizzate. Ogni volta che s'introducono componenti innovativi è buona norma elencare tutti gli aspetti che ne fanno parte;
- Le figure responsabili. Individuare dei capi progetto sia a livello dirigenziale che tecnico;
- Il periodo di transizione. Identificare un intervallo durante il quale convivono le due infrastrutture: quella precedente e quella appena progettata;

- Il documento di progetto. Una relazione ben dettagliata in cui sono elencate tutte le policy che la nuova infrastruttura deve poter fornire con specifico riferimento agli attributi delle identità, gli use case tipici, i requisiti funzionali e non.

Una volta ricevuto la risposta positiva da parte delle figure dirigenziali di solito si costituisce un gruppo di lavoro nel quale si identificano diverse figure come project manager, tecnici, sviluppatori e responsabili del progetto. A questo punto il flusso del lavoro passa nelle mani del capo progetto che ha il compito di portare a termine la missione e di fornire una comunicazioni con le figure dirigenziali. Ovviamente, nel caso in cui l'organizzazione è di piccole dimensioni questi processi non vengono svolti e tutto il carico di responsabilità grava su poche figure. Inoltre in questa fase è buona norma fare una attenta analisi dell'infrastruttura attualmente in uso. Infatti è utile comprendere se realizzarne una ex-nuovo, completamente compatibile con le policy elencate durante il consiglio di amministrazione oppure adattare l'architettura esistente. L'analisi deve valutare se è più oneroso il processo di inserimento del nuovo database per tutte le entità dell'organizzazione oppure modificare quello già esistente.

Realizzando un piano di progetto che tenga conto di tutti questi in cui per ognuno si prevede sempre una fase d'ingegnerizzazione del lavoro prima dell'applicazione pratica si è in grado di compiere la migrazione a un sistema di Identity e Access Management senza troppi problemi.

1.5 Tassonomia dell'Identity Management

A onor di completezza sono specificati di seguito i modelli di Identity Management in base alla concezione di gestione delle relazioni di fiducia tra le parti:

Identity Mamanament isolato

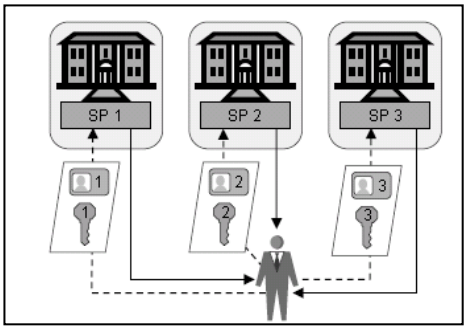

Figura 4: Modello di identità Isolato

Corrisponde al modello di Identity Management attualmente più diffuso. Ogni servizio, come mostrato in *figura 4*, ha un proprio bacino d'utenza indipendente e ad ogni utente viene assegnata una credenziale distinta per ogni servizio a cui fa accesso. Questo approccio semplifica l'Identity Management per i Service Provider, ma presenta seri problemi di usabilità per gli utenti all'aumentare dei servizi utilizzati.

Le relazioni di fiducia coinvolte in questo modello sono piuttosto semplici:

- Il Service Provider garantisce la privacy dell'utente.
- Il Service Provider implementa procedure di registrazione e meccanismi di autenticazione opportuni.
- Il client gestisce responsabilmente le credenziali ricevute dal Service Provider.

Identity Management centralizzato

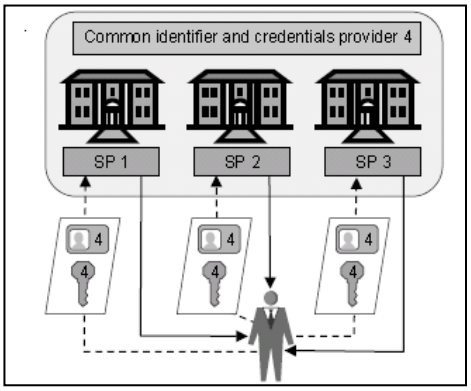

Figura 5: Modello centralizzato

Questo modello di Identity Management, mostrato in *figura 5*, è costituito essenzialmente da un unico Identity Provider che si occupa di identificare gli utenti per conto di una molteplicità di Service Provider. Le informazioni costituenti l'identità digitale di un utente possono anche in questo caso essere distribuite tra i providers, ma l'identificatore ad essa associato è unico e gestito dall'Identity Provider. Come il precedente, anche questo modello permette il Single Sign On, descritto in dettaglio nei successivi capitoli.

Capitolo 2

2 Identity e Access Management in Federazione

In questo capitolo viene descritto l'Identity e Access Management in ottica di federazione. Dunque non si considera più la singola organizzazione ma il loro insieme. Per tanto i problemi, gli aspetti, le caratteristiche viste nel capitolo precedente si ripercorrono nuovamente aggiungendo il componente federativo di unione. Inoltre è proposta un'analisi delle maggiori federazioni indicando i vantaggi che tale approccio comporta.

In questo capitolo sono affrontati i seguenti temi:

- Che cosa si intende per federazione;
- Compiti della federazione;
- Tassonomia della federazione;
- Stati dell'arte.

2.1 Che cosa si intende per federazione

Con il termine **federazione** s'intende un accordo, tra organizzazioni e fornitori di risorse, con il quale i partecipanti decidono di fidarsi reciprocamente, delle informazioni che si scambiano nei processi di autenticazione e autorizzazione, sulla base di regole e linee di condotta stabilite per gestire le relazioni di fiducia.

Spesso si vuole permettere a utenti che appartengono a una certa organizzazione di poter accedere a servizi di altre strutture che fanno parte di una federazione comune. Per fare questo le organizzazioni devono condividere meccanismi per lo scambio d'informazioni sugli utenti e per la gestione degli accessi alle risorse federate.

Con **Identity and Access Management federato** s'intende la gestione delle identità e degli accessi a livello di federazione, quindi non più

limitata a una singola organizzazione ma una gestione che prevede il coinvolgimento di un insieme di organizzazioni. Una infrastruttura di autenticazione e autorizzazione federata è un sistema che permette a un utente, appartenente a una organizzazione che fa parte di una federazione, di potersi autenticare e accedere ai servizi offerti da altre organizzazioni interne alla federazione, utilizzando sempre le stesse credenziali di accesso.

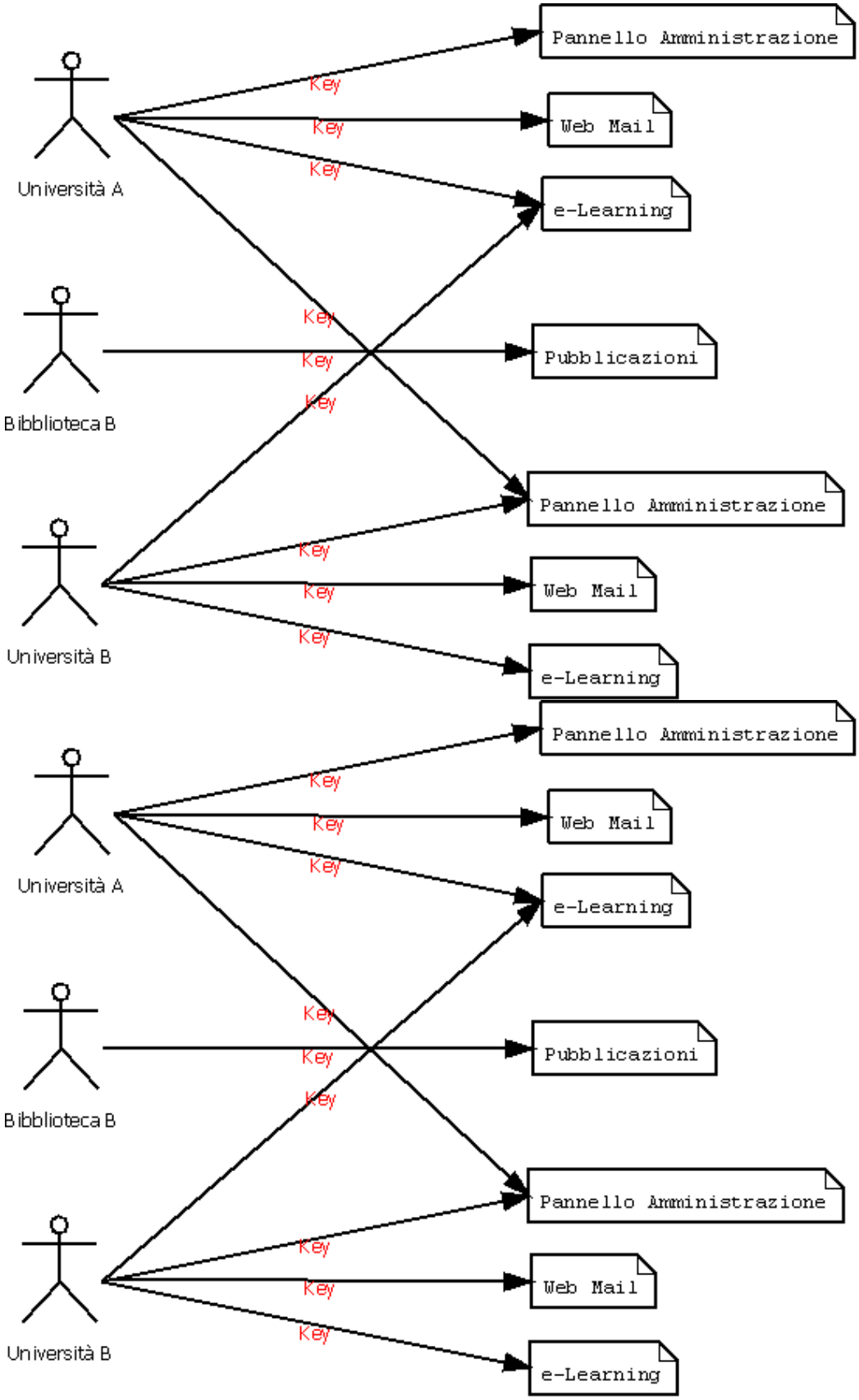

Figura 6: Identity e Access Management senza Federazione

Nella *figura 6* è illustrato un sistema di Identity e Access Management senza federazione: un utente ha una chiave d'accesso differente per ogni servizio che richieda l'autenticazione, anche per utenti esterni. Di conseguenza ogni soggetto è obbligato a gestire oltre che le identità dei propri utenti anche quelli provenienti da altre organizzazioni.

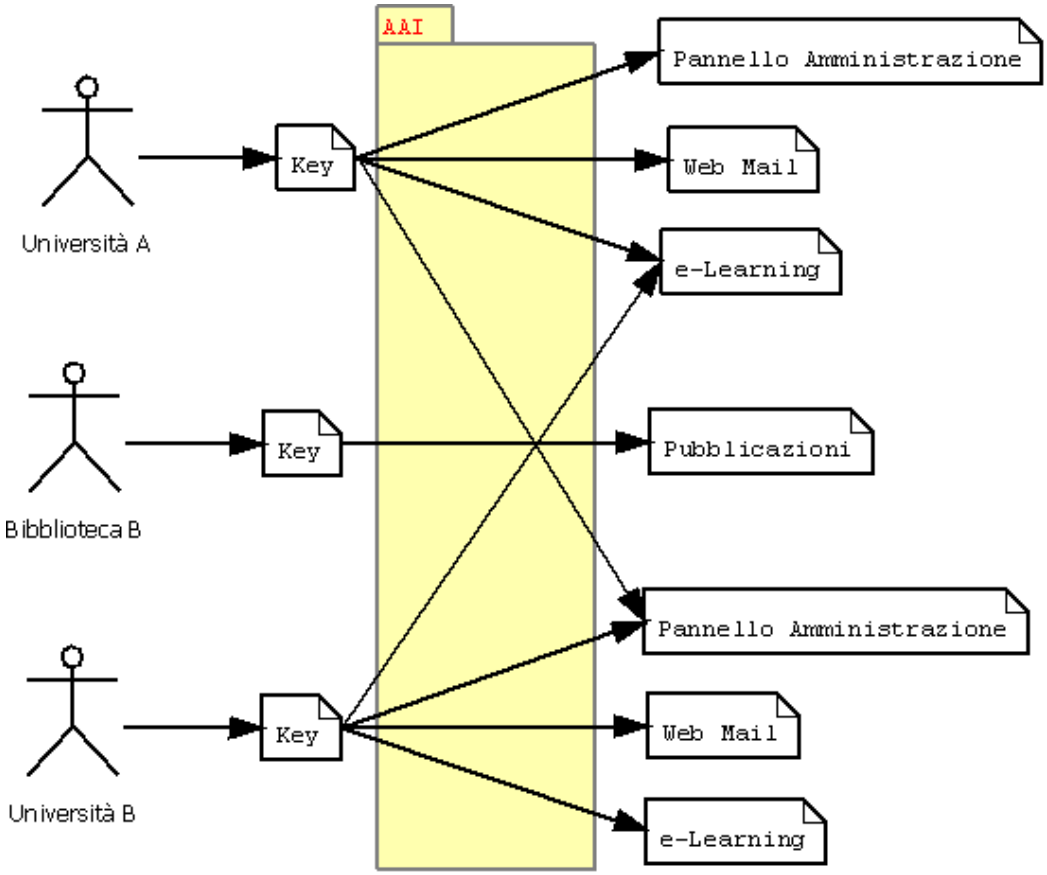

Figura 7: Identity e Access Management Federato

Con un sistema di Identity e Access Management federato, illustrato in *figura 7*, il processo di autenticazione è gestito dalla federazione, in tal modo ogni utente possiede una chiave di accesso unica valida per ogni servizio, e se un utente deve accedere a un servizio di un'organizzazione esterna, il processo di autenticazione preleva le informazioni necessarie dall'organizzazione di appartenenza dell'utente.

Attraverso un sistema di Identity e Access Management federato si ottengono diversi **vantaggi**:

- Possibilità di utilizzare sempre le credenziali della propria organizzazione;
- Maggiore privacy e controllo dei propri dati personali, poiché sono gestiti solamente dall'organizzazione di appartenenza;
- Dati sempre aggiornati;
- Minore carico amministrativo per la gestione delle identità e delle credenziali:
- Più controllo sui sistemi di autenticazione e autorizzazione, quindi maggior sicurezza;
- Scambio d'informazioni e contenuti tra gli utenti in modo più semplice.

Il software più diffuso che permette di implementare un sistema di Identity e Access Management federato a livello web è Shibboleth[12], sviluppato dall'ente internazionale Internet2. Nei successivi capitoli sarà descritto nel maggior dettaglio questo applicativo.

2.2 Compiti della federazione

Il compito principale della federazione è di mantenere sempre aggiornate le risorse disponibili alle diverse utenze che ne fanno parte. Infatti il ruolo che essa costituisce è prettamente a livello organizzativo come coordinatrice tra le organizzazioni. Nello specifico è responsabile di tutta la documentazione che ciascun membro deve avere come:

- la definizione degli attributi;
- il modulo per l'entrata di un nuovo membro;
- il modulo per la cancellazione;
- la documentazione legale in merito al trattamento dei dati personali;
- l'elenco sempre aggiornato dei membri con le rispettive risorse;
- la documentazione tecnica con i requisiti minimi per entrare in federazione.

[12] Sito di riferimento http://shibboleth.internet2.edu/

Al fine di rendere la federazione un organo efficiente è buona norma costituire uno staff sia gestionale sia tecnico che coordini tale sistema.

2.3 Tassonomia della federazione

La federazione dell'identità si può definire come l'insieme di tecnologie, standard e accordi che permettono a un insieme di Service Provider di accettare come validi gli identificatori utente gestiti da un altro insieme di providers, definiti come Identity Provider. Una tale organizzazione di providers, Service e Identity, viene tipicamente definita in una federazione di fiducia. La federazione dell'identità viene realizzata collegando i diversi identificatori utilizzati dai providers della federazione e relativi ad uno stesso utente. Un tale meccanismo implementa intrinsecamente il Single Sign On SSO, ovvero la possibilità per un utente di autenticarsi presso uno qualsiasi dei providers della federazione e, successivamente, di accedere ai servizi di tutti gli altri senza dover rifornire le credenziali di accesso.

Questo approccio di Identity Management introduce nuove classi di fiducia, e in particolare la presenza di accordi di trust tra providers. Le relazioni viste in precedenza valgono anche in questo caso, ma si applicano ai providers che fungono da Identity. I clients e i providers devono inoltre supportare le seguenti relazioni:

All'origine di una richiesta di servizio di un Identity Provider nei confronti di un Service Provider vi deve sempre essere un'esplicita richiesta dell'utente.

La corrispondenza tra gli identificatori collegati in una federazione deve essere corretta, ovvero tutti devono far riferimento a una stessa identità digitale.

La circolarità tra i providers delle informazioni sull'utente deve essere regolata da opportune policy accettate da tutte le parti al momento della federazione dell'identità.

2.4 Cenni storici sull'Identity e Access Management Federato

Nel corso degli anni si sono sviluppati diversi protocolli che hanno portato lo sviluppo di diversi standards utili al fine di poter applicare un Identity e Access Management federato. I primi protocolli a prender vita agli inizi degli anni novanta furono X509, per i certificati digitali, e LDAP Lightweight Directory Access Protocol, per l'accesso alle directory, ampliamente discussi nei capitoli successivi. In quel periodo ci fu una forte domanda da parte dei consumatori riguardo a un protocollo SSO non proprietario. Si svilupparono così due vendor group per soddisfare questa richiesta: la Burton Group[13] e la OASIS[14] responsabile del protocollo SAML. A quel punto anche Microsoft entrò in gioco creando il protocollo Passaport[15] dove ogni utente dispone di una sola credenziale di accesso per tutti i servizi che fanno riferimento a tale sistema. Diverse aziende, tra cui la SUN[16] si ribellano chiedendo un protocollo che consentisse di utilizzare determinati servizi senza dover passare attraverso l'autenticazione creata dagli uomini di Bill Gates: nacque così la Liberty Alliance[17]. Quest'ultima iniziò a creare una serie di protocolli tra cui, insieme alla OASIS, contribuì alla realizzazione di SAML 2.0. Allo stesso tempo, Microsoft insieme ai ricercatori di IBM si unirono per lo sviluppo di specifiche WS-* , come WS-Federation e WS-Trust, nate dalla scia di Passport. In quel periodo furono pubblicate numerose pubblicazioni tra le quali quella di Kim Cameron autore di "The Laws of Identity" [4] dove sono elencate le sette leggi che un'identità per esser tale deve rispettare. Fuori da questo marasma un gruppo di persone tra cui Dick Hardt, Johannes Ernst, Drummond Reede Kaliya Hamlin diedero vita al progetto OpenID con l'obiettivo di realizzare protocolli interoperabili con quelli proprietari. Così furono rivisti i principali prodotti della Liberty Alliance, come SAML 1.0, SAML 1.1 e SAML 2.0 e quelli proprietari WS-Federation, WS-Trust. Attualmente lo standard maggiormente più utilizzato all'interno di un'infrastruttura di Identity e Access Management è senza dubbio SAML 2.0

[13] Sito di riferimento http://www.burtongroup.com/
[14] Sito di riferimento http://www.oasis-open.org/home/index.php
[15] Sito di riferimento www.passport.net/
[16] Sito di riferimento http://it.sun.com/
[17] Sito di riferimento http://www.projectliberty.org/

2.5 Stato dell'arte

Allo stato attuale esistono diverse federazioni nate per rispondere alle più disparate esigenze. La maggior parte di queste federazioni nascono sotto la spinta di un progetto finalizzato alla realizzazione di specifici obiettivi circoscritti alla realtà in cui operano.

Le principali federazioni sono riportate nella *tabella 2* classificandole in base alla nazionalità e il numero dei membri che ne fanno parte.

Nazionalità	Federazioni in produzione	Membri
Italia	Garr IDEM http://progetto-idem.idem.garr.it	10
Svizzera	SWITCH AAI http://www.switch.ch/aai/	38
Regno Unito	The UK Federation http://www.ukfederation.org.uk	87
Finlandia	HAKA http://www.csc.fi/english/institutions/haka	39
Francia	Federation Éducation-Recherche https://federation.renater.fr/	80
Danimarca	WAYF http://www.wayf.dk/wayfweb/forside.html	20
Norvegia	Feide http://www.feide.no/introducing-feide	15
Olanda	SURFfederatie http://www.surfnet.nl/nl/Thema/SURFfederatie	42
Svezia	SWAMI http://www.swami.se/	15

Tabella 2: Federazioni

Nel grafico seguente si nota come le due nazioni Francia e Regno Unito siano le più attive in ambito di organizzazioni correlate alla federazione.

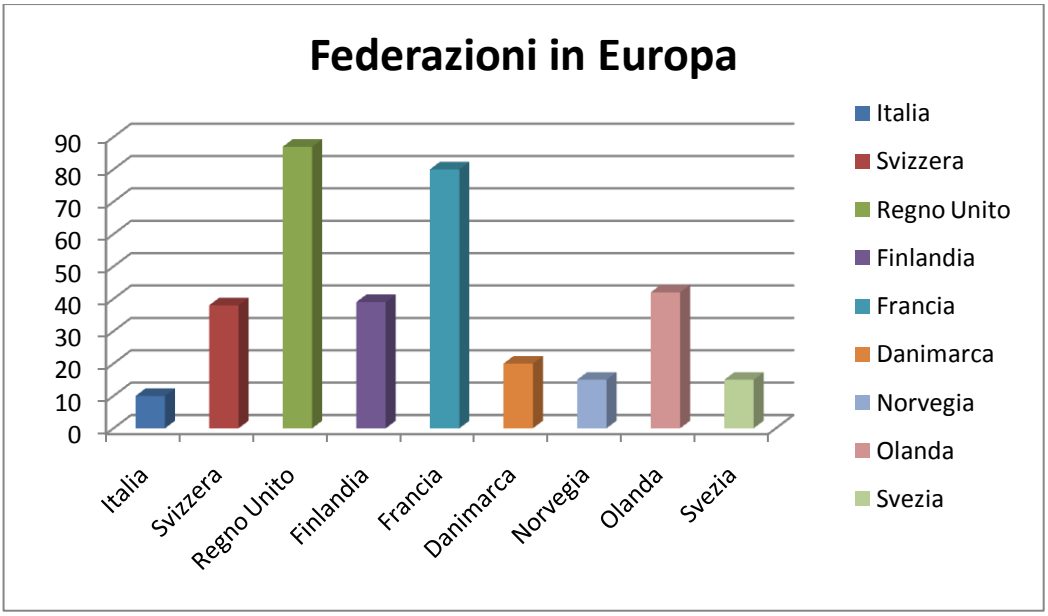

Tabella 3: Grafico Federazioni in Europa

Nella *tabella 4* è invece riportato il numero di risorse che ogni federazione mette a disposizione:

Federazione	Risorse	Link
Garr IDEM	18	https://www.idem.garr.it/index.php/it/servizi/sp
SWITCH	73	http://www.switch.ch/aai/participants/allresources.html
UK Federation	202	http://www.ukfederation.org.uk/content/Documents/AvailableServices
HAKA	34	http://www.csc.fi/english/institutions/haka/Services
Federation Éducation-Recherche	84	https://federation.renater.fr/en/resources
WAYF	41	http://www.wayf.dk/wayfweb/connected_services.html
SURFfederatie	24	http://www.surfnet.nl/nl/Thema/SURFfederatie/diensten/Pages/Default.aspx

Tabella 4: Risorse per federazione

Il numero di risorse rappresentano la scala ideale per classificare la grandezza di ogni federazione. Certamente maggiore è il numero di servizi disponibili, più rilevante è la federazione che ha il compito di coordinare tale struttura. Come era facile intuire dal grafico precedente, le risorse sono maggiori nella federazione britannica.

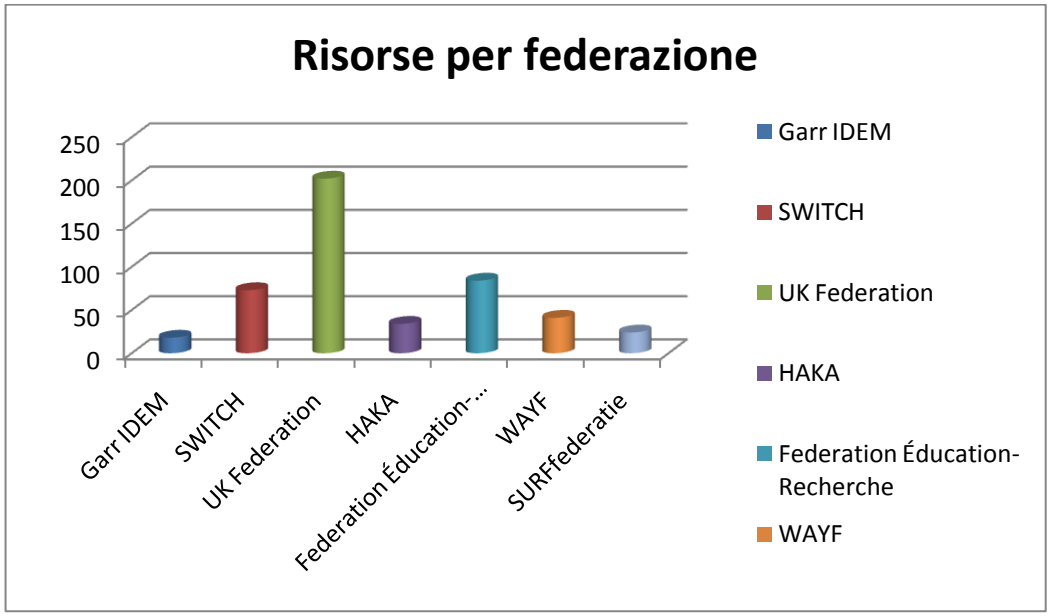

Tabella 5: Risorse per federazione

Tutti questi dati sono stati ricavati dal wiki[18] di TERENA[19], un'associazione che offre una vasta gamma di documentazione e paper utili per i ricercatori al fine di promuovere lo sviluppo delle tecnologie informatiche e di Internet. In particolare è possibile trovare una tabella [20] dove sono elencate tutte le federazioni in questo periodo attive nel mondo.

[18] Sito di riferimento https://refeds.terena.org/index.php/Federations
[19] Sito di riferimento http://www.terena.org/
[20] Sidto di riferimento https://refeds.terena.org/index.php/FederationsTable

Capitolo 3

3 Progetti vari di Identity Management Federato

Gli aspetti ai quali l'Identity e Access Management risponde sono stati ampiamente discussi nei capitoli precedenti. Nel corrente viene presentata un'analisi dei maggiori progetti attualmente vivi su questa tematica. Nello specifico ne sono stati analizzati due che fanno riferimento a due realtà organizzative distinte.

In questo capitolo sono affrontati i seguenti temi:

- Analisi dei progetti;
- L'identity management nelle Università
- L'identity management nella pubblica amministrazione
- L'identity management nelle aziende

3.1 Analisi dei progetti

Allo stato attuale c'è un po' di confusione riguardo alla costituzione di progetti di Identity Management federato. La maggior parte infatti nasce sotto il profilo governativo mediante finanziamenti da parte della Commissione Europea[21]. L'aspetto curioso è che pur affrontando argomenti paradossalmente simili esistono diversi progetti che mirano a coprire specifici dettagli dell'Identity Management federato. Un possibile elenco è mostrato nella *tabella 6*.

[21] Sito di riferimento http://ec.europa.eu/information_society/index_en.htm

Progetto	Continente	Link
OpenID	USA	http://openid.net/
Edugain	Europa	http://www.edugain.org
Stork	Europa	https://www.eid-stork.eu/
PrivacyOS	Europa	https://www.privacyos.eu
PRIME-Life	Europa	http://www.primelife.eu
SWIFT	Europa	http://www.ist-swift.org

Tabella 6: Progetti di AIM

OpenID nasce come fondazione internazionale non-profit di aziende per costituire una relazione di fiducia tra le diverse organizzazioni. OpenID fornisce alla comunità il supporto e l'aiuto al fine di far crescere il progetto. Tra i maggiori vendor spiccano aziende dal calibro di Google, Six Apart, Yahoo, Flickr, Myspace.com, Facebook, Wordpress, VerySign e AOL.

Edugain invece fornisce la tecnologia necessaria per permettere l'interconnessione fra differenti sistemi AAI. Dunque essa gioca un ruolo di confederazione, ovvero federazione di federazioni. Infatti ci sono numerose tecnologie adottate in diverse federazioni: Shibboleth è utilizzato in US, Svizzera, Finlandia, Germania, Gran Bretagna, Ungheria, Italia e Grecia; PAPI è utilizzato in Spagna; A-Select è presente in Olanda; SimpleSAMLphp è adoperato in Danimarca e Norvegia. Scopo della confederazione è la creazione di un ambiente dove differenti utenti provenienti da diverse federazioni possono essere identificati.

Stork è un consorzio che ha come obiettivo quello di stabilire una piattaforma europea di interoperabilità eID che consente ai cittadini di stabilire nuove relazioni elettroniche solamente presentando l'eID nazionale. Successivi dettagli sono presenti all'interno del factsheet [5] disponibile dal sito della Commissione Europea.

PrivacyOS è un progetto nato anch'esso all'interno della Commissione Europea con l'obiettivo di portare aziende, enti pubblici, università europee verso lo sviluppo di una infrastruttura di privacy comune.

PrimeLife deriva dal progetto PRIME[22] e ha come scopo quello di capire come gestire la privacy all'interno delle applicazioni web di internet e come mantenere questo stato duraturo nel tempo.

SWIFT è un'unione europea finanziata dal settimo framework programma FP7[23] della Commissione Europea con il fine di estendere le funzioni dell'identità e delle federazioni alla rete sottolineando gli aspetti dell'usabilità e privacy.

Oltre ai progetti che si riferiscono all'Identity e Access Management federato, è utile fornire un elenco di quelle che sono le tecnologie maggiormente utilizzate. La *tabella 7* mette in luce questo aspetto.

Software	Link
A-Select	http://a-select.surfnet.nl/
Athens	http://www.athensams.net/
SimpleSAMLphp	http://rnd.feide.no/simplesamlphp
LemonLdap	http://lemonldap.objectweb.org/
PAPI	http://papi.rediris.es/
OpenSAML	http://www.opensaml.org/
Shibboleth	http://shibboleth.internet2.edu/
SPOCP	http://www.spocp.org/index.html

Tabella 7: Tecnologie per l'AIM

Il prodotto più conosciuto a livello di Identity Management federato è indubbiamente Shibboleth. Tuttavia come mostrato nella *Tabella 7* esistono diverse soluzioni in grado di fornire il medesimo risultato

3.2 L'Identity Management nelle Università

I grossi centri di ricerca solitamente sono sempre i primi a sperimentare nuove tecnologie al fine di poter risolvere problemi. Anche in questo caso

[22] Sito di riferimento http://www.prime-project.eu/
[23] Sito di riferimento http://cordis.europa.eu/fp7/home_en.html

non poteva mancare un progetto italiano sull'Identity Management federato in campo accademico. Il progetto in questione prende il nome di IDEM che sta per IDEntity Management. Se concentriamo la nostra attenzione sulle diverse università italiane ci rendiamo subito conto che IDEM rappresenta la soluzione principale in ambito di federazione. Le università attualmente presenti sono:

- Politecnico di Milano;
- Università degli Studi di Cagliari;
- Università degli Studi di Ferrara;
- Università degli Studi di Modena e Reggio Emilia;
- Università degli Studi di Padova;
- Università degli Studi di Parma;
- Università degli Studi di Pavia;
- Università degli Studi di Torino;
- Università degli Studi di Urbino Carlo Bo;
- Università degli Studi Roma Tre.

Tra queste università emerge quella di Padova dove è stato integrato esse3, ovvero la piattaforma utilizzata da studenti, segreteria e professori, con lo standard di IDEM.

Se invece poniamo lo sguardo in Europa come per i progetti come Uk Federation per l'inghilterra e Switch per la svizzera si nota una situazione speculare all'Italia: unica federazione con lo stesso applicativo, ovvero Shibboleth.

Di fatto nel contesto accademico non esistono differenti soluzioni, ma bensì Shibboleth rappresenta uno standard de facto.

3.2.1 Progetto IDEM

IDEM[24] è il nome del progetto pilota per la costituzione di un sistema di Identity Management federato italiano tra i membri della comunità

[24] Sito di riferimento https://www.idem.garr.it/

GARR[25], gestito dal consorzio stesso. Tra le finalità del progetto, iniziato nel Gennaio 2008, figurano:

- la dimostrazione operativa, non teorica, della fattibilità tecnica e organizzativa e dell'utilità e usabilità della Federazione GARR;
- il rafforzamento dei sistemi di autenticazione e autorizzazione degli enti GARR tramite la condivisione di esperienze, soluzioni adottate e piani di sviluppo, condizione necessaria per aumentare la sicurezza senza penalizzare la flessibilità nella gestione di accesi ai servizi online e lo sviluppo di esperienze nel contesto di autenticazione federata;
- l'aumento della consapevolezza dell'utente nell'uso di servizi on-line;
- stimolare i fornitori di servizi a promuovere criteri di riconoscimento degli utenti sulla base di sistemi di identificazione-utenti logici e non fisici. In tal modo la gestione dei sistemi di accesso ad alcuni servizi (es. ai cataloghi on-line) risulterà più efficiente;
- la promozione del reciproco riconoscimento delle identità nell'ambito della rete di ricerca nel contesto italiano;
- favorire la nascita di servizi a uso della comunità GARR, anche da parte di singoli enti o gruppi di enti di ricerca;
- la predisposizione di accordi e possibili scenari di allargamento dell'iniziativa in contesti nazionali ed europei;
- lo sviluppo di esperienze nel contesto di autenticazione federata;
- il miglioramento in termini di efficienza nella fruizione di servizi in contesti ad elevata o crescente mobilità degli utenti.

E' stato inoltre istituito un comitato tecnico di gestione del progetto comprendente rappresentanti di alcune università ed enti di ricerca aderenti all'iniziativa.

Nell'ambito della Federazione IDEM, GARR agisce da coordinatore, fornisce l'infrastruttura centrale e i servizi e sottoscrive i contratti d'adesione. Inoltre mette a disposizione della Federazione il supporto tecnico per l'implementazione di un Identity Provider e Service Provider,

[25] Sito di riferimento http://www.garr.it/

le realizzazioni di riferimento open source, il servizio WAYF (utilizzato per la localizzare l'IdP di provenienza) la gestione della mappa della federazione definita come i metadati.

Possono aderire alla federazione le organizzazioni che fanno parte della comunità GARR, ovvero università e centri di ricerca e le altre organizzazioni interessate a condividere i propri servizi. Per partecipare alla federazione è necessario registrare almeno un Identity Provider, servizio di gestione e verifica delle identità, o un Service Provider, risorsa accessibile in rete.

I requisiti per aderire alla federazione sono diversi. Per i fornitori di servizi e contenuti:

- adeguamento agli standard tecnici scelti dalla federazione;
- riservatezza verso terzi rispetto alle informazioni utente di cui si entri in possesso tramite i servizi della federazione IDEM;
- realizzazione del servizio in conformità alle specifiche tecniche IDEM GARR.

Per le organizzazioni GARR:

- implementazione di un Identity Provider Shibboleth;
- conservazione dei log per associare un utente ad una sessione di autenticazione;
- disponibilità a fornire informazioni in merito al sistema di accreditamento e gestione degli utenti adottato.

3.3 L'identity Management nella pubblica amministrazione

Il contesto della pubblica amministrazione maggiormente si adatta a quello di federazione dove le diverse regioni per necessità hanno bisogno

di collaborare. Non poteva dunque mancare un progetto simile a quello dell'IDEM ma attivo in ambito diverso. Nasce così ICAR.

Per onor di completezza, prima di scrivere in dettaglio questo progetto, è interessante analizzarne degli altri che stanno prendendo spazio all'interno delle pubbliche amministrazioni. Un esempio è il progetto FedEra, promosso dalla regione Emilia-Romagna e condotto dalla società Lepida Spa per far si che i cittadini e le imprese possano disporre di un'autenticazione federata, tramite la quale accedere ai servizi online forniti da tutti gli enti locali. Se due enti federano i loro sistemi di autenticazione, i cittadini che hanno ricevuto nome utente e password (più in generale le credenziali di autenticazione) per accedere ai servizi on line di uno dei due enti, potranno usarli anche per ottenere i servizi digitali dell'altro; in altri termini il sistema FedERa consentirà agli utenti, con un'unica credenziale di autenticazione, e registrandosi una sola volta, di accedere a tutti i servizi on line messi a disposizione dai vari enti che aderiscono alla federazione. I cittadini possono perciò fare più cose con le stesse credenziali, e gli enti dal canto loro possono fornire servizi senza la necessità di registrare nuovamente gli utenti. Il sistema FedERa è entrato in pre-esercizio ad ottobre 2009: questa fase ha consentito ad alcuni enti, tra cui i Comuni di Ravenna, Castel San Pietro e Carpi, l'Assemblea Legislativa dell'Emilia-Romagna e la Regione Emilia-Romagna (portale geografico e portale imprese) di effettuare prove di integrazione dei propri servizi applicativi. Sempre nel 2009 è stata conclusa con successo una prima sperimentazione con l'Università di Bologna per l'offerta di un servizio rivolto agli studenti. Da gennaio 2010 il sistema FedERa è entrato nella fase di produzione effettiva.

Molto interessante è anche il progetto People[26] che ha l'obiettivo l'integrazione e la centralizzazione di servizi pubblici. Vi sono numerose demo esplicative sviluppate sia in J2EE che in .NET che permettono facilmente di capire cosa è in grado di fare questo framework. J2EE è la piattaforma Java, un linguaggio di programmazione, in grado di realizzare applicazioni web, mentre .NET svolge gli stessi compiti solamente lato Microsoft.

Da sottolineare anche i due progetti istituiti direttamente dal Governo Italiano, ossia la Posta Certificata[27] e ViviFacile[28]. Entrambi hanno l'obiettivo comune di identificare univocamente il cittadino rispetto agli

[26] Sito di riferimento http://www.progettopeople.it/
[27] Sito di riferimento https://www.postacertificata.gov.it/home/index.dot
[28] Sito di riferimento http://www.vivifacile.gov.it

uffici della pubblica amministrazione. La Posta Certificata permette di inviare messaggi di posta elettronica conferendone la piena validità legale nei casi previsti dalla normativa, garantendo data e ora riferiti all'accettazione e alla consegna del messaggio e l'integrità del contenuto trasmesso. ViviFacile si presenta come un portale in cui sono classificati 3 macrogruppi di utenti: cittadino, professionista e azienda. Una volta effettuata la registrazione al macrogruppo di appartenenza si ha accesso ai servizi che la pubblica amministrazione mette a disposizione per quella determinata categoria di utenza. Anche in questo si è cercato di unificare e gestire in maniera efficiente i numerosi servizi.

Infine è interessante citare il progetto della Regione Marche con la CartaRaffaello[29], una smartcard in grado di accedere ai servizi on-line della pubblica amministrazione senza doversi identificare con l´inserimento di username e password in ogni singolo sito, come adesso avviene, ma semplicemente inserendo la card nel lettore e digitando il proprio PIN.

3.3.1 Progetto ICAR

Il progetto di Interoperabilità e Cooperazione Applicativa tra le Regioni dal quale deriva il nome ICAR nasce nel 2004 su iniziativa di 17 regioni e una provincia autonoma con il coordinamento del Comitato Permanente dei responsabili dei Sistemi Informatici delle regioni CPSI, del Centro Interregionale per il Sistema Informatico e il sistema Statistico CISIS.

Lo scopo del progetto è di promuovere l'uso del Sistema Pubblico di Connettività per la cooperazione applicativa interregionale attraverso interventi concreti su due livelli:

- Interventi infrastrutturali (INF). ICAR prevede l'implementazione da parte delle regioni di servizi infrastrutturali per la cooperazione applicativa in sé (INF-1), per la gestione di accordi di servizio (INF-2) e di un sistema federato di autenticazione (INF-3).

[29] Sito di riferimento http://www.cartaraffaello.it/web/

- Casi di studio applicativi (AP). Allo scopo di verificare l'efficacia degli interventi infrastrutturali, il progetto prevede inoltre la stesura di sette casi di studio di specifici domini applicativi relativi alla cooperazione interregionale.

  In particolare:

  - AP-1 "Cooperazioni e Compensazioni Sanitarie Interregionali"
  - AP-2 "Cooperazione tra sistemi di Anagrafe"
  - AP-3 "Area Organizzativa Omogenea"
  - AP-4 "Lavoro e Servizi per l'Impiego"
  - AP-5 "Tassa automobilistica regionale"
  - AP-6 "Osservatorio Interregionale sulla rete distributiva dei carburanti"
  - AP-7 "Sistema Informativo Interregionale di Raccordo con Cinsedo"

L'intervento INF-3 riguarda alcune delle tematiche di questo elaborato e pertanto è descritto in dettaglio.

L'obiettivo di questo intervento è di realizzare un sistema federato di autenticazione ed attribuzione di ruoli che deleghi a componenti distribuiti detti Servizio di Identificazione e di Ruolo di Comunità (SIRC) l'esecuzione di tali operazioni in funzione del dominio di appartenenza dell'utente. In base ad accordi di fiducia tra i domini partecipanti alla federazione, ognuno di essi si impegna di riconoscere come valide le autenticazioni e le qualificazioni effettuate presso il SIRC che serve un altro dominio.

Affinché sia garantita l'interoperabilità tra i vari componenti dell'architettura federata e quindi fortemente distribuita, ICAR individua la necessità di identificare univocamente gli utenti indipendentemente dal particolare meccanismo di autenticazione utilizzato dal dominio di appartenenza. Il progetto prescrive quindi l'utilizzo di un identificatore utente "federato", ovvero valido presso ogni dominio. Lo stesso discorso si applica ai ruoli.

E' previsto che l'integrazione del sistema con i componenti di sicurezza preesistenti nei singoli domini avvenga sviluppando opportuni meccanismi di connessione.

Infine, il progetto ICAR definisce come requisito anche una implementazione di riferimento del servizio, realizzata secondo il paradigma Open Source per garantirne il riuso, la condivisione dell'esperienza e la possibilità di adattarla alle esigenze dei singoli enti.

3.4 L'Identity Management nelle aziende

In ambito aziendale non esistono rilevanti progetti poiché rispetto ad altri contesti come nelle pubbliche amministrazioni e università, non vi è la necessità di condividere risorse con altre realtà. Conseguentemente è più facile trovare aziende con soluzioni specifiche per la propria infrastruttura piuttosto che identità federate.

Tuttavia in ambito vendor esistono diverse aziende che offrono soluzioni nel campo dell'Identity Management. La *tabella 8* riassume una lista delle principali proposte classificandole secondo i seguenti aspetti:

- **User Provisoning** Questi prodotti tipicamente offrono caratteristiche di Workflow, password reset, discover e amministrazione dell'identità;
- **Single Sign On** Tipicamente consente all'utente di autenticarsi e accedere a ulteriori servizi senza chiedere nuovamente le credenziali;
- **Federated Identity Management** Si riferisce all'implementazione dei protocolli SAML, Liberty Alliance e WS-Federation per fornire l'identificazione dell'utente tra diversi domini;
- **Directory** Identifica l'applicativo con il quale sono salvati i dati.

Vendor	User Provisoning	Single Sign On	Federated Identity Management	Directory	Others
Computer Associates	ETrust Identity Manager	- eTrust Access Control - eTrust Site Minder - eTrust Single Sign-On - eTrust TransactionMinder - eTrust Identity and Access Management Toolkit	eTrust® SiteMinder ® Federation Security Services, eTrust TransactionMinder	eTrust Directory	eTrust CA-top Secret Secuity
Cisco		Cisco Clean Acces / Nac appliance			
Citrix		Citrix Password Manager			
HP	Select Identity	Select Access	Select Federation		Select Audit
IBM	IBM Tivoli Identity Manager	- IBM Tivoli Access Manager for e-business - IBM Tivoli Access Manager for operating systems - IBM Tivoli Access Manager for Business Integration	IBM Tivoli Federated Identity Manager	- IBM Tivoly Directory Server - IBM Tivoli Directory Integrator	IBM Tivoli Privacy Manager for e-business, IBM Tivoli Security Compliace Manager, XML Security
Juniper Networks		Unified Access Control			
Microsoft	Microsoft Identity Integration Server 2003 Enterprise Edition	- Microsoft Internet Security and Acceleration Server 2000 Enterprise Edition - Internet Information Services	Active Directory Federation Services	Windows Server 2003 Active Directory	Microsft web Application, Microsoft Clients, Certificate Lifecycle Manager
Novell	Nsure Identity Manager Enhanced Provisionin g Module	- iChain - Novell Nsure SecureLogin	SAML Extension for Novell iChain, Liberty identity provider for Novell eDirectory	eDirectory	
Oracle	Oracle Identity Management	Oracle COREid Access and Identity, Oracle Enterprise SSO	Oracle COREid Federation	Oracle Internet Directory, Oracle Virtual Directory	
SUN Microsystems	Sun Java System Identity Manager	- Sun Java System Access Manager - Solaris - Java System Message Queue Enterprise Edition	Sun Java System Access Manager	- Sun Java System Directory Server Enterprise Edition - Sun Java	Sun Java System Identity Auditor

	System Directory Proxy Server

Tabella 8: Lista vendor

3.5 Analisi critica dei progetti esaminati

Durante lo studio di questi progetti ho riscontrato diversi aspetti che andrebbero trattati in maggior dettaglio.

Se concentriamo l'attenzione sul progetto IDEM possiamo sicuramente afferire che l'organizzazione in generale risulta molto soddisfacente. Infatti vengono proposti eventi a cadenza trimestrale per cercare di acquisire nuovi membri da coinvolgere. La documentazione nel sito è ampia e ben realizzata. Sicuramente è un ottimo punto di partenza per comprendere meglio il software Shibboleth e i documenti utili per costituire una federazione. Molto interessante è il documento DOPAU, documento descrittivo del processo di accreditamento degli utenti, che potrebbe esser preso come linee guida nel caso in cui si volesse istituire un proprio sistema di Identity Management. L'unica considerazione che mi preme sottolineare si riferisce al fatto che le risorse di IDEM sono solamente web. Sarebbe difatti interessante estendere Shibboleth anche a risorse come le cartelle condivise in modo da avere accesso anche attraverso differenti protocolli.

Nel caso del progetto di ICAR possiamo affermare che il progetto di identità digitale federata risulta solamente un task di un lavoro molto più complesso. Conseguentemente la documentazione relativa su alcuni punti risulta un po' vaga e attualmente, all'infuori della Regione Toscana, non è ancora stato implementato questo sistema. Inoltre alcuni componenti, come il local proxy, dal mio punto di vista andrebbe eliminati perché non essenziali per il funzionamento dell'infrastruttura. Sarebbe stato utile anche avere una documentazione in cui fosse stato scritto una configurazione rapida, e poi successivamente un'altra molto più complessa, utile per grandi organizzazioni.

Per tutti gli altri progetti sarebbe auspicabile raggrupparli in un unico grande lavoro al fine di creare una confederazione, ossia una federazione di federazioni. L'auspicio è quello di arrivare ad una identità unica nel web, simile a quella reale come ad esempio accade per il passaporto cartaceo. Un unico documento valido all'interno dei confini europei con il quale è possibile accedere tutti i servizi online e viene ufficialmente riconosciuto, sia che siano semplice applicazioni web sia che siano enti pubblici. In qualche modo è l'obiettivo OpenID esteso anche servizi pubblici con identità federate.

Capitolo 4

4 Gestione dell'Identità digitale in federazione attraverso casi d'uso

Per affrontare problematiche di gestione dell'identità digitale, è fondamentale capire cosa si intende per **identità digitale**, quali sono le componenti principali di un sistema di gestione e capire le differenze architetturali tra le diverse tipologie di sistemi. Scopo del seguente capitolo sarà di mettere in luce questi aspetti.

In questo capitolo sono affrontati i seguenti temi:

- Definizione dell'identità digitale;
- Architettura generale;
- Componenti e cicli di vita dell'identità;
- Sistemi di gestione dell'identità.

4.1 Definizione dell'identità digitale

Il concetto d'identità digitale deriva dal riconoscimento reciproco tra soggetti digitali, siano essi rappresentazioni di persone, cose o servizi, per personalizzare l'interazione tra essi. Ciò è reso possibile associando ad ogni soggetto digitale, detto principal, un insieme di caratteristiche che lo differenziano dagli altri, dette attributi. L'insieme degli attributi associati a un determinato principal costituisce il profilo utente del soggetto digitale.

Con **gestione dell'identità digitale** o più brevemente gestione dell'identità, si intendono le numerose problematiche che sorgono quando si ha a che fare con l'identità digitale di soggetti fisici. I meccanismi di corrispondenza tra soggetti fisici e principal, la rappresentazione degli attributi dei soggetti, l'integrità e la coerenza dei profili, la gestione dei

privilegi per l'accesso a risorse o servizi sono solo alcuni esempi delle problematiche trattate in questa vasta area tematica.

4.2 Architettura generale

Lo scenario tipico di un'infrastruttura di Identity e Access Management federato è quello mostrato in *figura 8* in cui gli utenti facenti parte di un'organizzazione sono autenticati tramite l'Identity Provider IdP e possono accedere alle risorse localizzate presso i Service Provider SP.

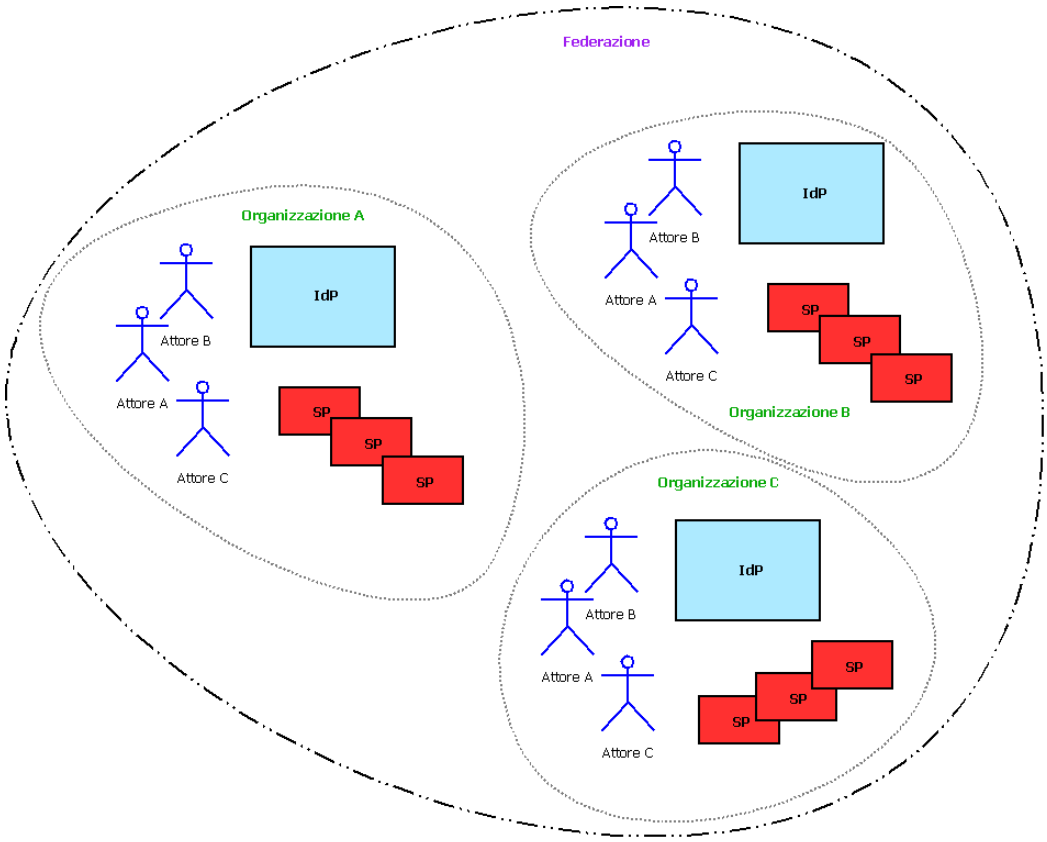

Figura 8: Architettura generale

All'interno di questa architettura base le diverse organizzazioni fanno capo alla federazione, la quale ha il compito di coordinare le differenti risorse a loro interno.

Il caso tipico è quello in cui un utente di una organizzazione vuole accedere a una risorsa della propria struttura proprio come in *figura 9*.

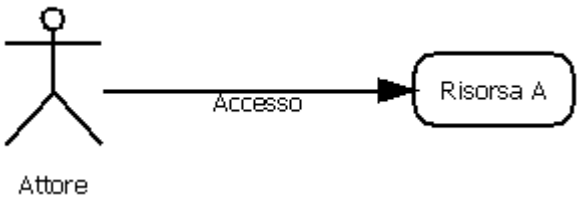

Figura 9: Use case 1

In questo scenario la federazione non risulta essere un elemento significativo, tuttavia il sistema di Identity Access e Management è in grado di fornire il servizio di SSO internamente.

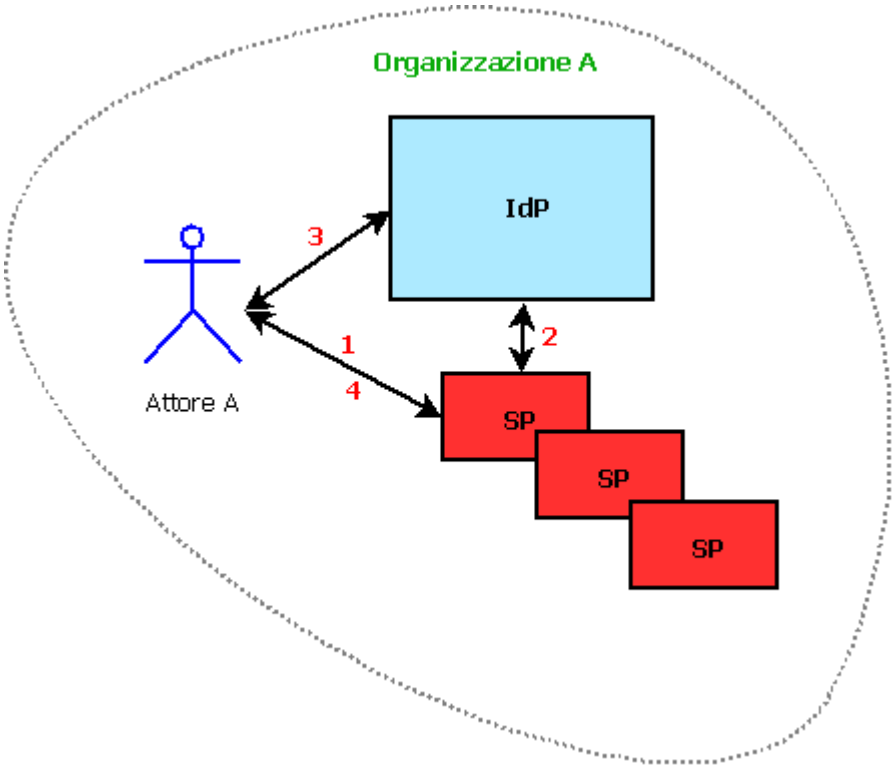

Figura 10: Flusso use case 1

Il flusso dei dati in questo contesto è come quello mostrato in *figura 10* dispiegato nei seguenti punti:

L'utente richiede l'accesso a una risorsa all'interno dello SP;

L'SP contatta l'IdP per ottenere le credenziali di accesso;

L'IdP domanda le credenziali all'utente;

Ad autenticazione avvenuta, l'utente viene indirizzato alla risorsa che aveva precedentemente richiesto.

Un ulteriore caso d'uso è quello in cui l'utente fa accesso ad una risorsa di un'organizzazione diversa, proprio come mostrato nella *figura 11*.

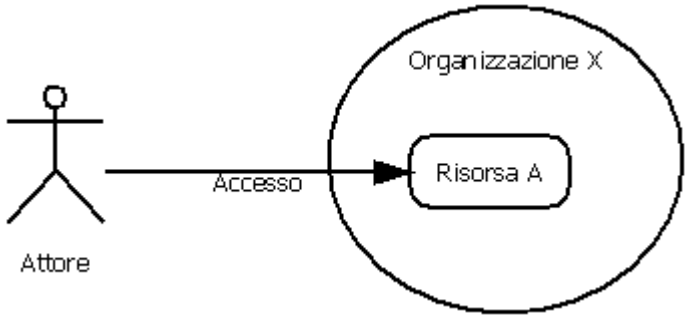

Figura 11: Use case 2

In questo scenario il concetto di federazione risulta essere fondamentale, perché attraverso questo elemento di unione l'utente è in grado di accedere a una risorsa appartenente a una diversa organizzazione.

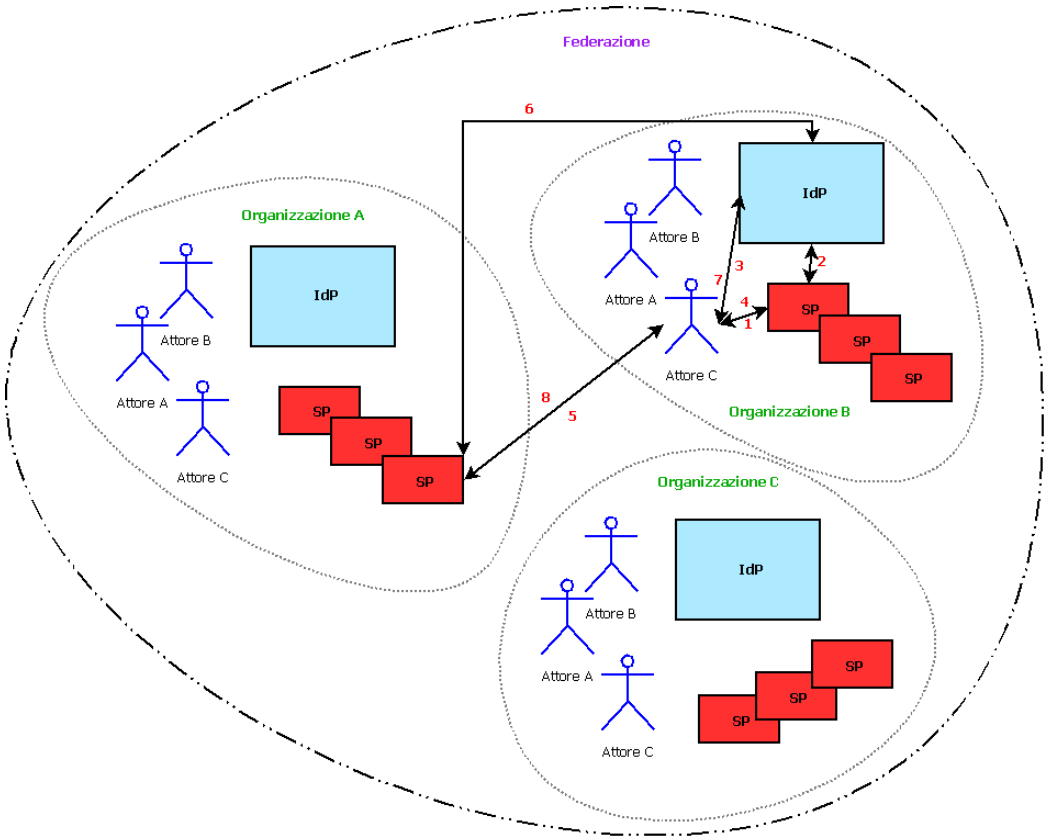

Figura 12: Flusso use case 2

Il flusso dei dati segue un andamento differente rispetto al caso d'uso visto in precedenza. Di seguito in dettaglio i punti mostrati nella *figura 12*:

1. L'utente richiede l'accesso a una risorsa all'interno dello SP;
2. L'SP contatta l'IdP per ottenere le credenziali di accesso;
3. L'IdP domanda le credenziali all'utente;
4. Ad autenticazione avvenuta, l'utente viene indirizzato alla risorsa che aveva precedentemente richiesto;
5. L'utente richiede una risorsa localizzata all'interno di un SP differente dalla propria organizzazione;
6. L'SP contatta l'IdP di appartenenza dell'utente. In questo punto gioca un ruolo fondamentale la federazione, la quale ha il compito specifico di tenere aggiornata la lista degli IdP e SP disponibili all'interno della federazione;
7. L'IdP richiede le credenziali di autenticazione;
8. Una volta autenticato, l'utente è in grado di accedere alla risorsa all'interno di una differente organizzazione.

4.3 Componenti e ciclo di vita dell'identità

La gestione dell'identità digitale è fondamentale quando si tratta di regolamentare e limitare l'accesso a risorse o servizi in base alle caratteristiche del soggetto richiedente.

All'interno di questo tipo di interazione si distinguono tre fasi principali: la fase di autenticazione, durante la quale viene instaurata una corrispondenza tra un soggetto fisico e un principal, e la fase di autorizzazione, durante la quale è deciso se e con che privilegi il principal potrà accedere al servizio o alla risorsa.

Inoltre un sistema di gestione dell'identità solitamente incorpora strumenti per il trasferimento degli attributi utente tra sistemi e per la memorizzazione dello storico degli accessi a risorse o servizi, in modo da poter risalire agevolmente all'identità degli utenti in caso di abusi.

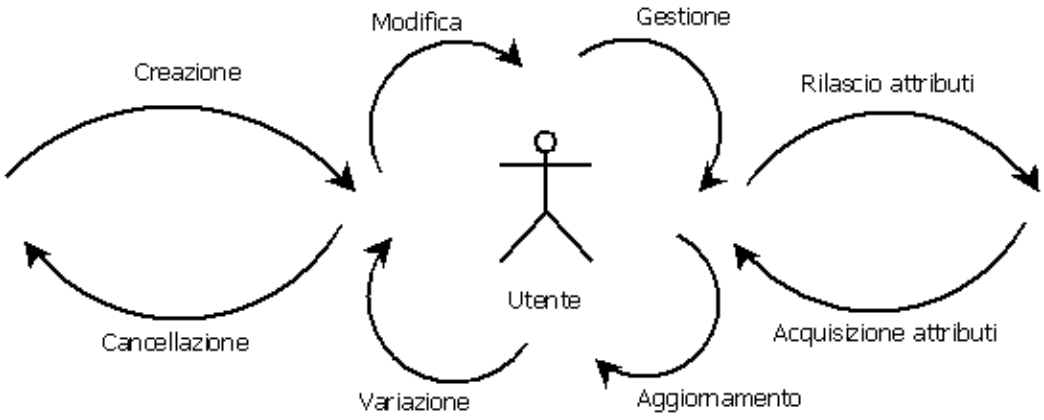

Figura 13: Ciclo dell'identità

Come mostrato in *figura 13* l'identità digitale ha un proprio ciclo di vita il quale prevede la registrazione, l'impiego al servizio, il riutilizzo ma anche l'eliminazione in caso di uscita dell'identità fisica dall'organizzazione.

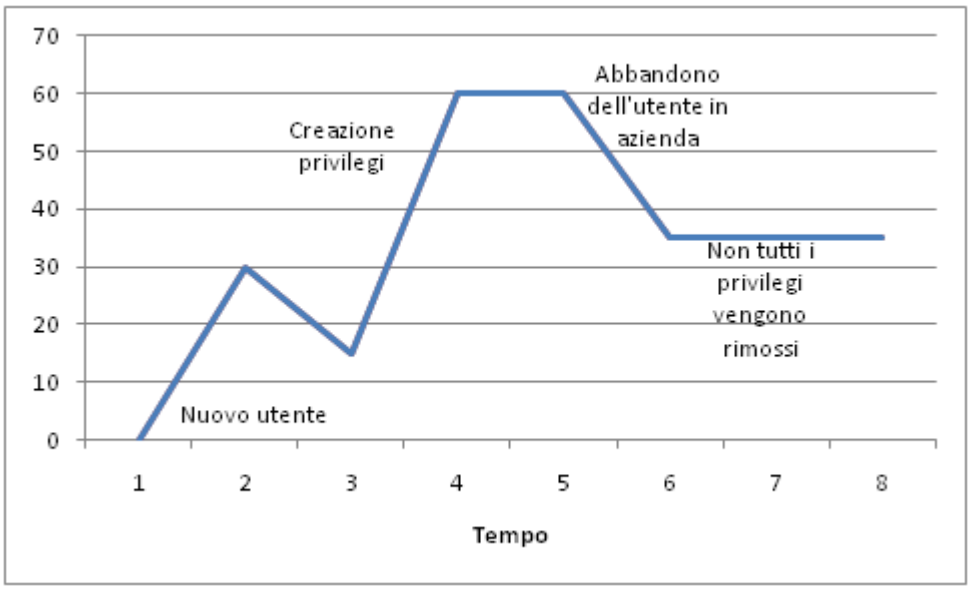

Figura 14: Flusso di vita dell'identità senza eliminazione privilegi

Il concetto di vita dell'identità è un aspetto da non sottovalutare. Infatti nella maggior parte delle realtà si prevede un processo di creazione delle identità ma non di cancellazione, come mostrato in *figura 14*.

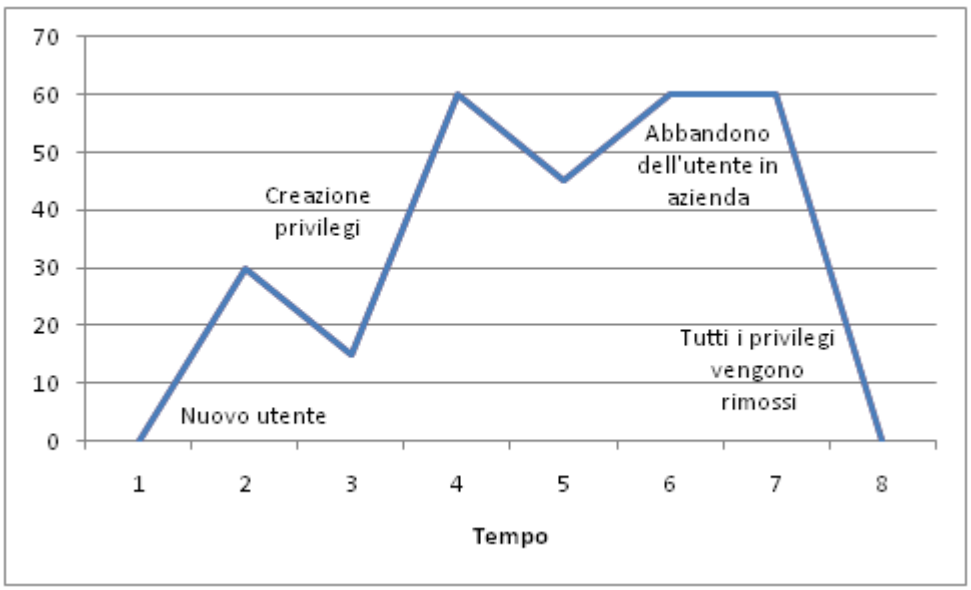

Figura 15: Flusso di vita dell'identità con eliminazione privilegi

All'interno del piano di progetto presentato in consiglio di amministrazione è bene predisporre la procedura per l'eliminazione dell'utente, *figura 15*.

4.4 I sistemi di gestione dell'identità

In questo ambito è possibile effettuare una classificazione dei sistemi di gestione dell'identità, a seconda dell'approccio seguito nella progettazione e nell'implementazione del sistema stesso.

Gestione orientata al servizio

Si ha un approccio orientato al servizio quando il sistema di gestione dell'identità è legato esclusivamente alla risorsa o al servizio locale al sistema stesso. Un esempio è un sito web con un'area pubblicamente visitabile e una riservata agli utenti registrati: sarà quindi implementato un sistema di gestione dell'identità per permettere la registrazione di nuovi utenti, il loro accesso e la personalizzazione dell'area riservata.

Questo approccio, in cui Identity Provider e Service Provider sono accorpati in un unico sistema, è attualmente il più diffuso, grazie anche alla grande disponibilità di sistemi in grado di implementare questa architettura. Alla facilità di configurazione di sistemi di questo tipo si oppone l'aumento del numero di credenziali di cui l'utente deve disporre (una per ogni differente servizio) e l'alta probabilità di incoerenze tra gli attributi dei diversi profili riguardanti l'utente.

Gestione orientata all'istituzione

Si ha un approccio orientato all'istituzione quando lo stesso sistema di gestione dell'identità è utilizzato per accedere a diverse risorse o servizi.

Un esempio è una università che offre a studenti e docenti diversi servizi separati tra loro. Il sistema assocerà ad ogni principal un unico profilo che permetterà al servizio di autorizzare o no l'accesso al sistema di posta elettronica, al pagamento delle tasse universitarie o alla verbalizzazione degli esami.

Questo approccio, che prevede un unico Identity Provider e diversi Service Provider, si sta nel tempo diffondendo tra le grandi istituzioni per via della semplificazione della gestione dei profili e degli attributi.

L'aggiunta di un nuovo servizio interno all'istituzione comporta l'implementazione di un sistema per collegare il servizio al sistema centralizzato.

Gestione federata dell'identità

Si ha un approccio federato alla gestione dell'identità quando istituzioni diverse si accordano per riconoscere reciprocamente i propri principal, in modo che i singoli sistemi di gestione dell'identità interoperino tra loro, mantenendo la gestione dei profili locale alle istituzioni.

Un esempio è una biblioteca universitaria che offre un servizio di consultazione di periodici on-line anche ad altre università: una volta individuata l'istituzione di provenienza del principal, il sistema si metterà in comunicazione con essa per reperirne il profilo.

L'insieme degli Identity Provider, dei Service Provider, degli accordi tra essi e dei meccanismi che permettono loro di interoperare è detta federazione d'identità, o infrastruttura di autenticazione e autorizzazione, detto AAI.

I vantaggi di una gestione federata dell'identità sono i medesimi di una gestione orientata all'istituzione, applicati però in un contesto in cui istituzioni differenti operano tra di loro con uno scopo comune, come l'insegnamento o la ricerca, condividendo risorse e servizi.

Gestione dell'identità orientata all'individuo

Si ha un approccio orientato all'individuo quando il sistema di gestione dell'identità è locale a un dispositivo, hardware o software, in possesso dell'utente.

Un esempio è l'uso di smartcard: il dispositivo di lettura, una volta rilevata la presenza della smartcard, procede all'identificazione del principal e si occupa di comunicarne il profilo, immagazzinato all'interno della carta stessa, ai servizi a cui l'utente vuole accedere.

Questo approccio, che prevede un Identity Provider per ogni utente o punto di accesso e diversi Service Provider, si sta affermando per quanto riguarda servizi quali posta elettronica certificata o internet banking. Il principale vantaggio e la principale differenza architetturale rispetto al modello federato consiste nella decentralizzazione dell'Identity Provider, che evita i costi relativi alla gestione dei profili interna all'istituzione. Tuttavia risulta particolarmente onerosa la fase di creazione dell'infrastruttura, e di distribuzione dei dispositivi agli utenti.

Capitolo 5

5 Realizzazione dell'infrastruttura

In questo capitolo viene presentato un elenco degli standard e delle tecnologie fondamentali per la costituzione di una infrastruttura di Identity e Access Management federato. Queste sono riconducibili a tre fattori principali: lo scambio di credenziali, il protocollo SAML e il Single Sign On con riferimento a Shibboleth.

In questo capitolo sono affrontati i seguenti temi:

- Tecnologie utilizzate
- Standard sul formato e lo scambio di credenziali
- Single Sign On
- Tecnologie applicate all'Identity Management Federato
- Infrastruttura di test

5.1 Tecnologie utilizzate

Al fine di realizzare una infrastruttura di Identity e Access Management federato è necessario introdurre quelle che sono le tecnologie maggiormente utilizzate. In generale è possibile classificare tutta la struttura secondo le seguenti parti:

- **Sistema operativo** dove la scelta può ricadere tra Linux e Windows**;**
- **Server Web** che potrebbe esser Apache o IIS;
- **Web Container** in cui nella maggior parte dei casi si utilizza Tomcat;
- **Protocollo** per lo scambio dei dati dove anche in questo caso si ha una sola scelta SAML;
- **Directory** per il salvataggio che potrebbe essere OpenLDAP, LDAP e ActiveDirectory;

- **Applicativo** con il quale gestire le identità tra cui Shibboleth, PAPI, simpleSAMLphp.

Nei paragrafi successivi saranno discusse in maniera più dettagliata le parti che caratterizzano maggiormente la composizione dell'infrastruttura.

5.2 Standard sul formato e lo scambio di credenziali

Le credenziali, ad eccezione di documento certificato attestante l'identità o altre qualifiche di un soggetto, sono gli elementi informativi centrali nelle problematiche di Identity Management. Qualsiasi soluzione di Identity Management per definizione si trova a dover gestire delle identità digitali, e ognuna di queste ultime non può essere ritenuta tale senza un insieme di credenziali utili per l'autenticazione.

Per tanto è evidente la necessità di una buona standardizzazione e diffusione sul mercato delle tecnologie relative questo aspetto, prima ancora della qualità intrinseca di esse. Si tratta per questo motivo che uno standard, per molti aspetti legacy e limitato come **X.509**, è tuttora il più utilizzato in ambito Internet e non, per l'autenticazione *forte* di identità digitali.

Uno dei limiti maggiori di X.509 è la visione fortemente gerarchica dei soggetti, attivi e passivi, coinvolti nell'autenticazione e delle rispettive relazioni di fiducia, in contrapposizione a modelli decentrati, basati su altri paradigmi, che si vanno affermando soprattutto nelle comunità Internet, ma anche nel mondo istituzionale

Lo standard **SAML** è quindi proposto come compromesso tra la visione completamente *verticale* e quella *orizzontale* del processo di certificazione, che va sotto il nome di approccio federativo. Tale obiettivo viene raggiunto specificando un formato di credenziale, indipendente dal particolare metodo di autenticazione, che permetta ad un'autorità di asserire qualcosa su un soggetto. Il fatto che una seconda autorità riconosca o meno una tale asserzione è in funzione del rapporto di fiducia che ha con la prima.

5.2.1 SAML

Security Assertion Markup Language, SAML[30] è uno standard basato su XML per la creazione e la comunicazione di tokens di sicurezza. Questo protocollo è stato prodotto, a partire dal 2001, dal Security Services Technical Commitee di OASIS. La *figura 16*, tratta dalla presentazione di Maler [8], mostra l'intera architettura dell'intero standard.

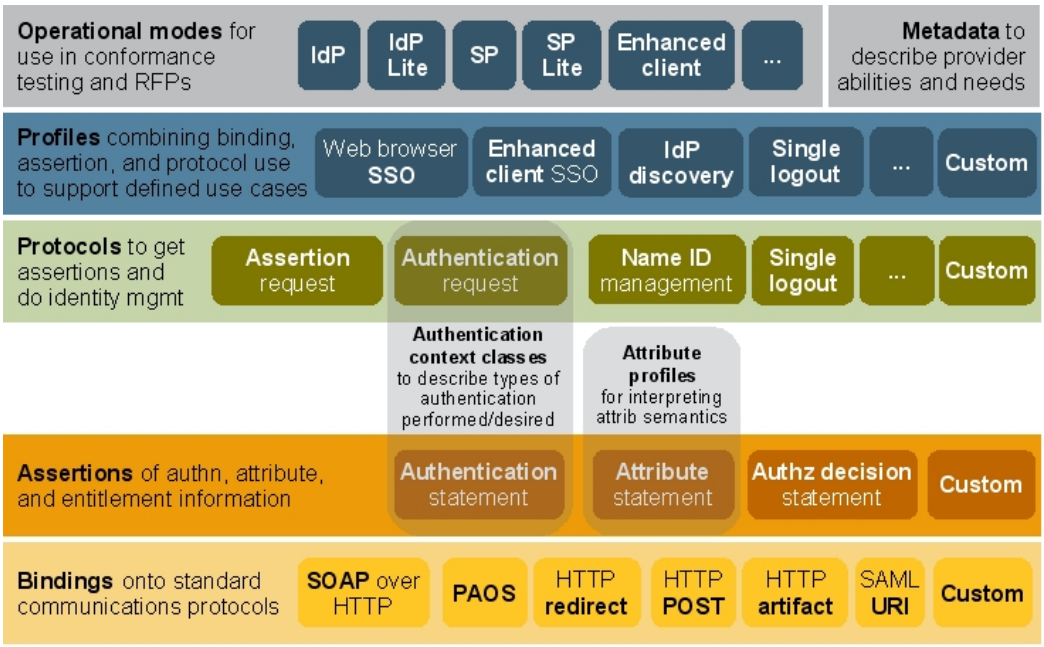

Figura 16: Framework SAML

La più diffusa implementazione Open Source di SAML è OpenSAML[31], Open Source Security Assertion Language implementation.

SAML ha come obiettivo principale la creazione di un framework di supporto per la problematica del SSO e, più in generale, dell'Identity Management. Tuttavia, nel corso degli anni, sono emersi altri utilizzi dello standard come ad esempio la restrizione di deleghe in CAS o la sicurezza di messaggi SOAP.

[30] Sito di riferimento http://saml.xml.org/
[31] Sito di riferimento http://www.opensaml.org/

Lo standard si può suddividere in quattro parti essenziali: la specifica del formato delle credenziali, dette *asserzioni*, i *protocolli* (astratti) per comunicarle, i *bindings* e i *profili*, come si evince in *figura 17*.

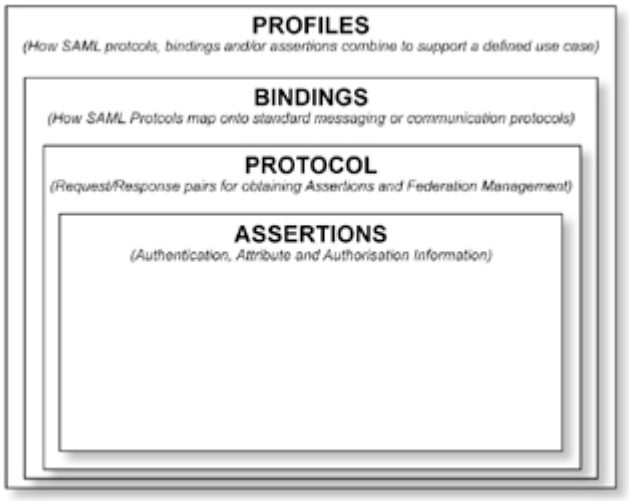

Figura 17: Profili, bindings, protocolli e asserzioni SAML

5.2.1.1 Asserzioni

Le asserzioni sono presenti in tre tipi diversi e sono dichiarazioni di fatti riguardanti l'utente, sia esso una persona fisica o un sistema hardware o software.

Le *asserzioni di autenticazione* sanciscono il fatto che un utente ha provato la propria identità a un opportuno *asserting party***.** Mentre le *asserzioni di attributo* certificano alcuni dati del profilo utente , come ad esempio il ruolo assunto in un'organizzazione.

Infine i permessi, definiti come *authorization decision*, identificano invece quali operazioni un utente può compiere, come ad esempio l'accesso ad una data pagina di un servizio.

In *figura 18* è schematizzata una asserzione SAML.

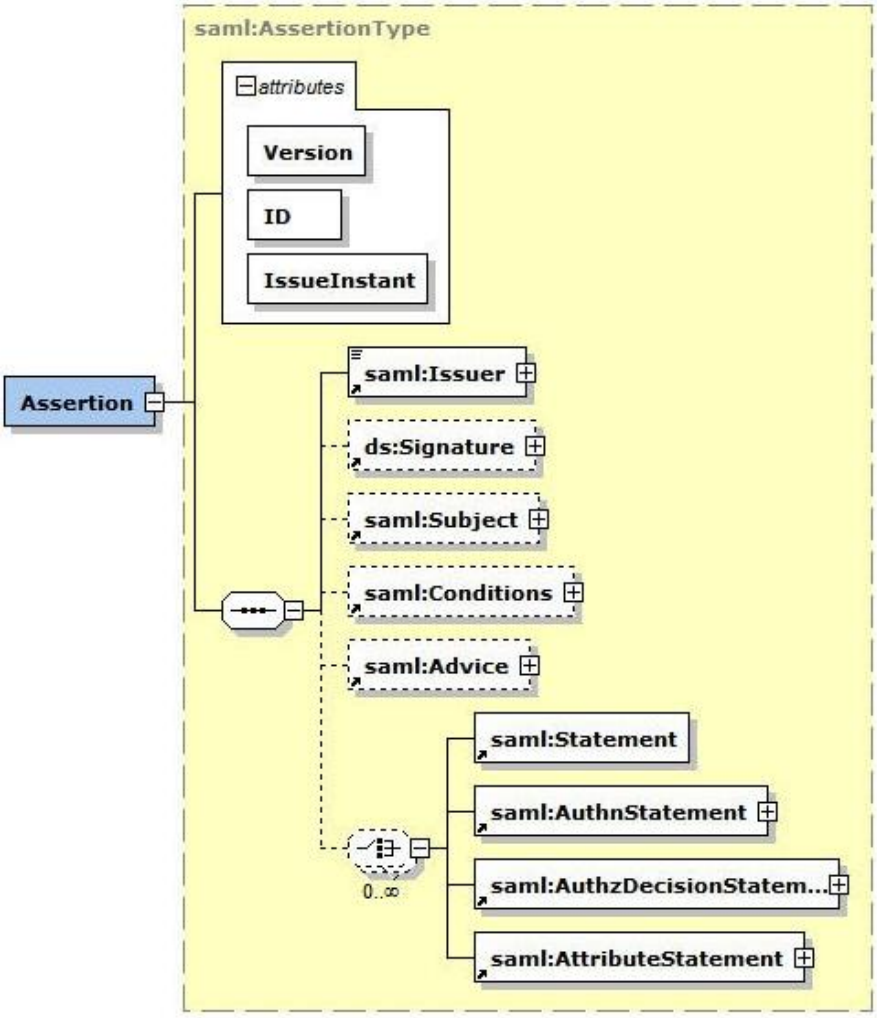

Figura 18: Sintassi asserzione SAML

5.2.1.2 Protocolli

I protocolli definiscono come si richiedono e ricevono le asserzioni contattando un *asserting party*. Sono costituiti da successioni di messaggi chiamati *SAML request* e *SAML response*, codificati in una notazione XML.

I messaggi di richiesta indicano dati sui soggetti relativamente ai quali viene richiesta un'asserzione. Mentre nei messaggi di risposta sono presenti le asserzioni richieste

5.2.1.3 Binding

I protocolli stabiliscono la struttura delle informazioni che possono essere scambiate ma non determinano le specifiche modalità di trasporto.

A tal fine sono stati definiti opportuni *binding* che indicano come realizzare effettivamente in SAML lo scambio di informazioni di sicurezza attraverso alcuni dei principali protocolli di trasporto come per esempio HTTP o SOAP. In pratica stabiliscono come mappare i messaggi di richiesta e risposta SAML sui protocolli standard di comunicazione.

I *binding* di tipo *HTTP POST* e *HTTP Redirect* supportano i casi in cui l'*asserting party* come per esempio un certificatore d'identità e il *relying party* come un fornitore di servizi, devono interagire utilizzando un browser web che agisce da intermediario.

In particolare, l'interazione con browser web nei casi in cui sia necessario inoltrare una richiesta di autenticazione a una *SAML authority* può avvenire rispettivamente mediante una form HTML che contiene una action di input o submit che si traduce in una HTTP POST verso la *SAML authority*

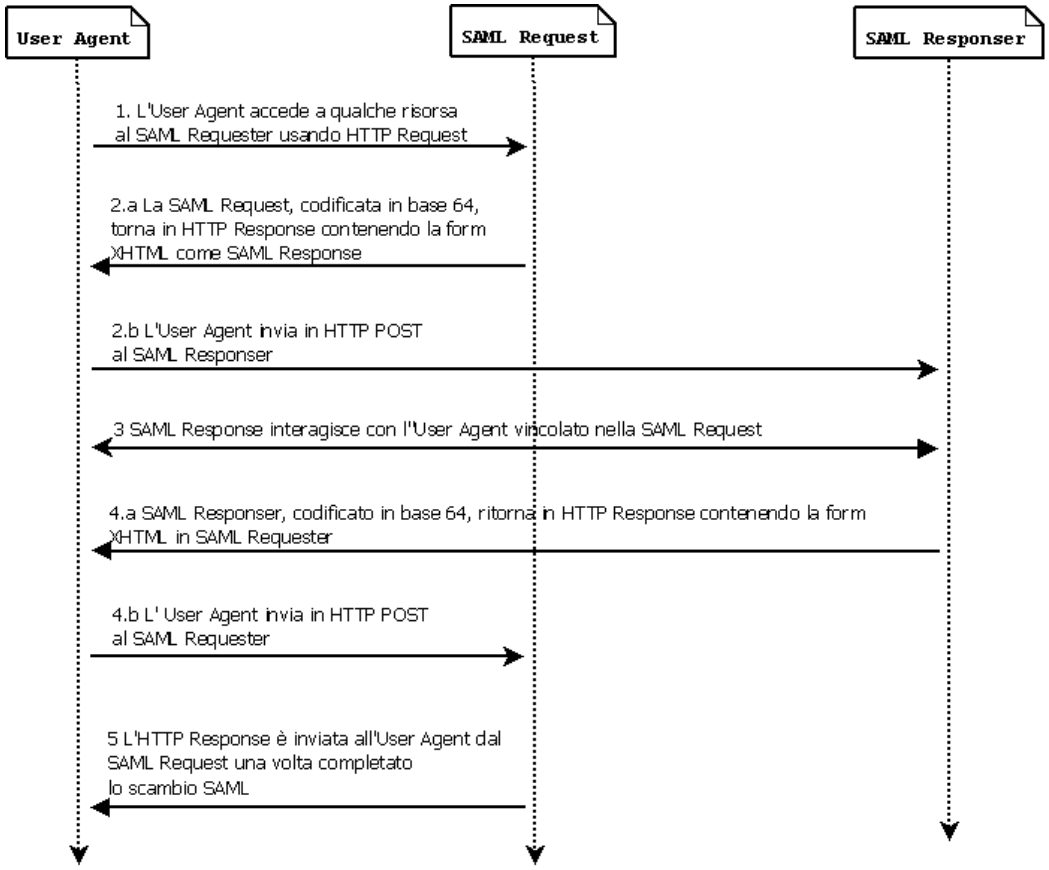

Figura 19: SAML binding HTTP POST

Nel caso di *binding HTTP POST* come in *figura 19*, il messaggio SAML è codificato come un documento XML da inserire nella form HTML restituita all'HTTP client dell'utente, *User Agent*.

Mentre nel *binding HTTP Redirect* la trasmissione del messaggio richiede l'uso di una tecnica di *URL enconding*: l'intero contenuto del messaggio deve essere collocato nell'URL query string associato a un parametro di nome *SAML Request* e quindi trasmesso mediante *HTTP GET*.

5.2.1.4 Profili

I profili corrispondono a un certo numero di scenari d'uso, nei quali è definito come le asserzioni, i protocolli e i binding sono utilizzati in modo combinato per raggiungere gli scopi per i quali tali scenari sono stati pensati

Esempi di profili sono quelli per compiere il Single Sign On tra vari servizi di diverso dominio.

Tra i profili più rilevanti ricordiamo:

- SSO Profiles
- Artifact Resolution Profile
- Attribute Profiles
- Assertion Query/Request Profile
- Name Identifier Mapping Profile

5.2.2 X.509

X.509 è uno standard di ITU-T[32], International Telecommunication Union Telecommunication standard sector, che definisce una PKI. Nel corso degli anni, grazie anche agli adattamenti apportati da IETF[33], Internet Engineering Task Force, è diventato lo standard di fatto per quanto riguarda l'autenticazione *forte* in ambito Internet.

L'elemento informativo di base dello standard è il PKC, ovvero il documento che associa il nome di un soggetto alla sua chiave pubblica. I certificati sono emessi e firmati da una CA su richiesta del soggetto ed hanno una validità temporale fissata.

Una PKI X.509 è tipicamente composta da più CA organizzate in una struttura ad albero, alla radice della quale si trova la root CA. Ciascuna CA è denotata di un certificato emesso e firmato da una CA di livello

[32] Sito di rifermento http://www.itu.int/ITU-T/
[33] Sito di riferimento http://www.ietf.org/dyn/wg/charter/pkix-charter.html

superiore ad eccezione del root che utilizza un certificato auto-firmato. Le foglie dell'albero sono i soggetti per i quali sono stati emessi un certificato, come mostrato in *figura 20*.

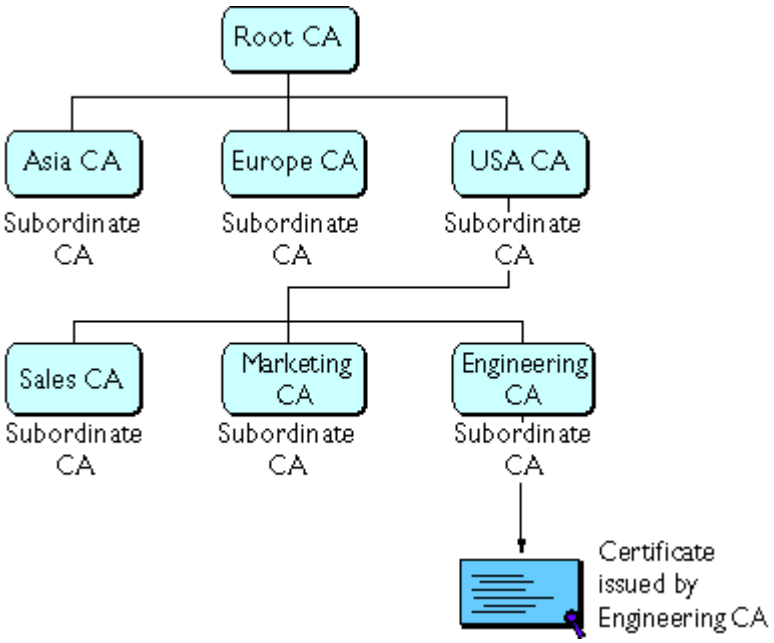

Figura 20: Struttura della CA

Una struttura di questo tipo permette di avere garanzia dell'autenticità di un soggetto fidandosi esclusivamente dell'autenticità della root CA. Chi ha bisogno di questa garanzia può infatti risalire l'albero fino alla radice, verificando che ogni nodo intermedio abbia un certificato firmato dal nodo padre.

In caso di compromissione del soggetto associato ad un certificato, quest'ultimo può essere revocato pubblicandolo in una CRL oppure utilizzando il protocollo OCSP.

Nelle sue ultime evoluzioni, lo standard X.509 specifica un altro tipo di certificato: l'AC. Anziché la chiave pubblica di un soggetto, un AC contiene privilegi o credenziali utili per una autorizzazione per opera di un'opportuna autorità. Gli AC sono emessi e firmati da una Attribute Authority detto AA, e possono essere immagazzinati in un database LDAP.

5.3 Single Sign On

Il Single Sign On è una particolare forma di autenticazione, ed autorizzazione, che permette l'accesso a più risorse protette a partire da un'unica, iniziale, interazione di autenticazione da parte del soggetto fruitore. Esistono varie tipologie di SSO in funzione dei contesti applicativi e dei soggetti coinvolti.

Quella sulla quale si focalizza questa sezione è denominata Single Sign On su web. Il SSO su web si applica esclusivamente all'accesso autenticato a risorse web, statiche o dinamiche, tipicamente effettuato tramite un browser. I soggetti che accedono ad una tale risorsa per la prima volta nell'arco di una stessa sessione, vengono dirottati presso un servizio di autenticazione, più propriamente l'Identity Provider Successivamente, se quest'ultima ha avuto successo, essi vengono rediretti nuovamente presso la risorsa iniziale, nello specifico il Service Provider, a cui verranno comunicati gli aggiornamenti sullo stato di autenticazione di tali soggetti.

L'approccio al SSO, ed all'Identity Management in generale, è di tipo federativo in tutte le soluzioni presentate. Non è imposta dall'alto l'adozione di un particolare Identity Provider, ma vengono invece forniti gli strumenti per stringere legami di fiducia con una federazione di tali provider senza vincoli gerarchici precostituiti.

I **vantaggi del SSO** su web in confronto alle soluzioni tradizionali rispetto ad una credenziale distinta per ogni sito sono evidenti:

- **Usabilità**. Il soggetto fruitore deve effettuare meno interazioni per accedere agli stessi servizi;
- **Sicurezza**. Poiché la fase di autenticazione è un evento che diventa più raro, può essere dedicata maggiore attenzione, eventualmente implementando tecniche *forti* su base crittografica. Per il SSO di tipo federativo questo è vero in quanto gli Identity Provider potrebbero essere gli unici ad avere le informazioni necessarie ad effettuare tale operazione sui propri soggetti;
- **Scalabilità**. Un soggetto che si autentica presso un certo Identity Provider può accedere a risorse protette da un Service Provider che non ha bisogno di tenere internamente informazioni, bensì può

ottenerle dal Service Provider. Tale fenomeno implica una maggiore scalabilità delle soluzioni SSO con il conseguente aumento dei soggetti servibili a parità d'investimento;

- **Incentivi alla federazione**. Un Service Provider è motivato ad aderire ad una federazione dal fatto che può accedere ad un bacino d'utenza più ampio ovvero quello di tutti gli Identity Provider della federazione. Tuttavia un Identity Provider che entri in una federazione garantisce ai propri soggetti una migliore offerta di servizi in termini di qualità, quantità e diversificazione.

5.4 Tecnologie applicate all'Identity Management Federato

Tra le varie tecnologie applicate all'Identity Management federato viste nel *Capitolo 3.1* Shibboleth rappresenta la soluzione open maggiormente diffusa. Non a caso è ampliamente utilizzata nelle più grosse federazioni e nello specifico in quella di IDEM. Nei paragrafi successivi sono affrontate le maggiori tecnologie utilizzate in questo campo.

5.4.1 Shibboleth

Shibboleth, è un progetto inter-universitario del gruppo Middleware Architecture Committee for Education MACE[34], appartenente al consorzio Internet2[35]. Le sue finalità sono la progettazione, la specifica e l'implementazione Open Source di sistemi per la condivisione inter-istituzionale di risorse web soggette a controllo di accesso.

Questo progetto nasce inizialmente per semplificare il problema dell'accesso a contenuti didattici riservati o a pagamento da più campus universitari, ciascuno con una differente infrastruttura di autenticazione.

[34] Sito di riferimento http://middleware.internet2.edu/MACE/
[35] Sito di riferimento http://www.internet2.edu/

Shibboleth si può tuttavia applicare ad altri contesti come quelli dell'e-Business e della PA ed è probabilmente la più completa implementazione Open Source di un sistema di SSO su web. Inoltre, diversamente da altri sistemi analoghi, Shibboleth ha adottato fin dall'inizio SAML per implementare gran parte dei suoi protocolli, garantendosi l'interoperabilità con altri sistemi dell'area dell'Identity Management.

L'architettura di Shibboleth, come descritta nel paper di S. Cantor di Internet2 [7], comprende i seguenti elementi:

Identity Provider (IdP): l'entità che è in grado di autenticare l'utente e fornire informazioni aggiuntive o attributi sul suo conto.

Service Provider (SP): il sistema presso il quale è gestita la risorsa web a cui l'utente fa richiesta e che ha il compito di proteggerla attraverso una qualche forma di policy di accesso.

User Agent (UA): si tratta dell'applicazione che, per conto dell'utente, innesca i protocolli di SSO richiedendo l'accesso a una risorsa web protetta da un SP Shibboleth-interoperabile.

Where Are You From (WAYF): riguarda un servizio gestito da una terza parte o dal SP stesso il cui compito è scoprire l'IdP di appartenenza dell'utente.

Non è prevista una gerarchia tra i provider: ogni organizzazione è responsabile dei propri utenti, nel caso degli IdP, e/o delle proprie risorse, nel caso dei SP. La rete di fiducia risultante è orizzontale ed in Shibboleth viene denominata federazione. Il SP che sceglie di aderire ad una federazione accetta implicitamente un legame di fiducia con tutti gli IdP che ne fanno parte.

Simmetricamente, un IdP accetta di emettere asserzioni su richiesta di un qualsiasi SP della federazione. Shibboleth non vieta tuttavia la possibilità di specificare rapporti di fiducia bilaterali tra singoli SP ed IdP. Quest'ultimo approccio ha una scalabilità ridotta ma può risultare utile per piccole configurazioni o per specificare rapporti differenziati tra provider all'interno di una federazione preesistente.

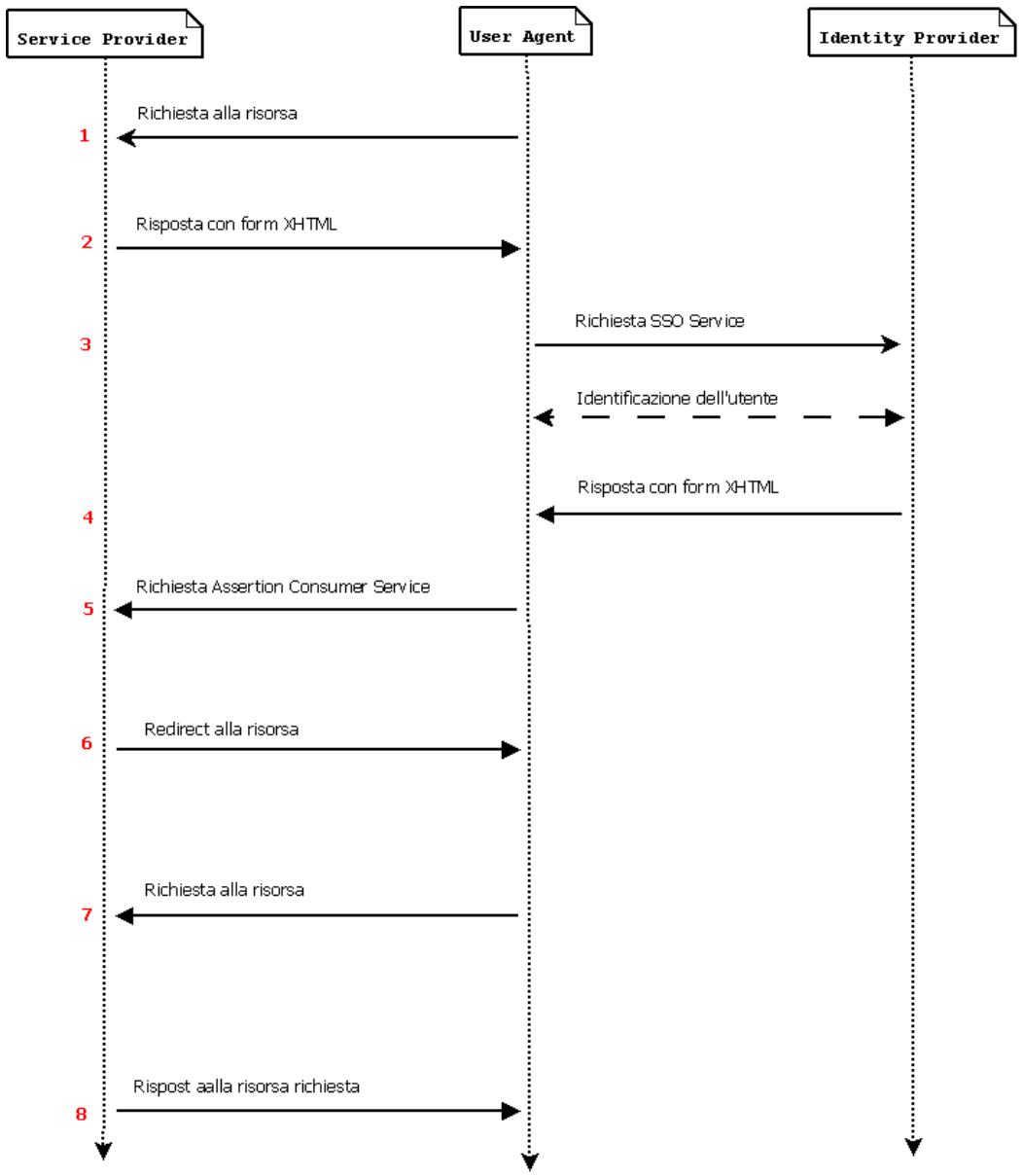

Figura 21: Comunicazione fra IdP, UA e SP

Come mostrato in *figura 21*, per un generico protocollo di SSO, le interazioni tra le entità sono le seguenti:

1. Lo UA, User Agent, per conto dell'utente, richiede l'accesso ad una risorsa web presso il SP. Si assume che lo UA non abbia ancora una sessione attiva con il SP.
2. Il SP re-dirige lo UA verso il WAYF o direttamente presso l'IdP di appartenenza. Il contenuto della URL di destinazione costituisce una *authentication request* e contiene informazioni sulla risorsa richiesta, un identificativo del SP e l'endpoint presso il quale il SP intende ricevere l'asserzione di autenticazione.

3. Se interrogato, il WAYF processa la *authentication request* del SP (ma trasportata dallo UA) ed interagisce con l'utente per conoscere l'IdP presso il quale intende autenticarsi. Tale informazione viene memorizzata in una sessione a lungo termine tra il WAYF e lo UA (ad es. un cookie). Successivamente il WAYF ridirige lo UA verso l'IdP scelto dall'utente.
4. L'IdP identifica l'utente innescando un meccanismo di autenticazione o sfruttando una sessione ancora attiva. Shibboleth non considera il particolare meccanismo di autenticazione utilizzato.
5. L'IdP invia un'asserzione di autenticazione al SP utilizzando il profilo SAML browser/POST o browser/artifact.
6. Il SP invia una richiesta di attributi dell'utente all'IdP.
7. Se interrogato, IdP risponde alla richiesta di attributi del SP.
8. Sulla base delle informazioni sull'utente, il SP effettua la decisione per l'accesso dell'utente alla risorsa richiesta. In funzione dell'esito di tale decisione il SP invia una risposta HTTP opportuna allo UA.

Per supportare i protocolli di SSO, i providers coinvolti espongono un certo numero di funzionalità distinte. Nella terminologia di Shibboleth, tali funzionalità sono denominate *ruoli*. Un tipico IdP avrà un insieme di ruoli disgiunto da un tipico SP, ma non vanno esclusi i casi in cui uno stesso provider fornisca alcune delle funzionalità di IdP e SP contemporaneamente.

Le figure 15 e 16, tratte dal paper di Cantor [6], illustrano i tipici componenti di un IdP e di un SP rispettivamente, nonché le principali interconnessioni tra di essi.

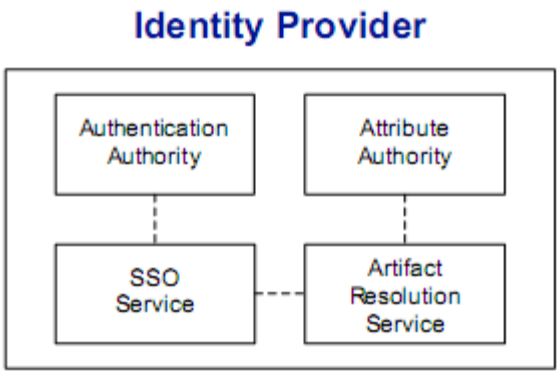

Figura 22: Identity Provider

Authentication Authority Si tratta di un servizio che emette asserzioni SAML di autenticazione dopo aver ottenuto conferma dell'identità dell'utente dall'applicazione che esegue l'autenticazione vera e propria. Entra in azione al passo 4 del protocollo generico di SSO, visto nel paragrafo precedente. Nell'uso comune di Shibboleth non è prevista l'invocazione diretta di questo servizio da parte del SP.

SSO Service Rappresenta un'applicazione web dell'IdP che riceve le richieste di autenticazione del SP ed innesca l'opportuno profilo SAML di SSO. Entra in gioco nei passi 2/3, 4, 5 e costituisce il primo punto di contatto di un IdP.

Artifact Resolution Service Viene utilizzato esclusivamente nei casi in cui si richiama il profilo *SAML browser/artifact*, e costituisce il web service che traduce gli artifacts in asserzioni. Viene impiegato al passo 5 del protocollo.

Attribute Authority (AA) Si tratta di un web service che fornisce asserzioni SAML di attributi utilizzando il *SAML SOAP Binding*. Si presenta nei passi 6 e 7, lato del IdP. L'accesso agli attributi da parte dei singoli providers è, regolato da una policy detta *Attribute Release Policy (ARP)*. In tale policy confluiscono esplicite esigenze di privacy espresse dall'utente e direttive generiche dell'IdP, il tutto in funzione del particolare provider richiedente.

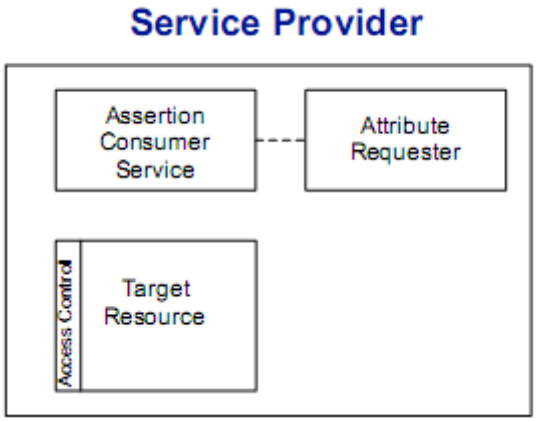

Figura 23: Service Provider

Assertion Consumer Service Si tratta di una applicazione web alla quale vengono inviate le asserzioni di autenticazione da parte dell'IdP. Implementa il passo 5 del protocollo descritto sopra.

L'*Assertion Consumer Service* ha un comportamento differente a seconda del profilo SAML che si usa. Nel caso del browser/POST estrae direttamente le asserzioni dalla richiesta HTTP. Nel caso del browser/artifact, invece, deve risolvere gli artifacts in asserzioni interrogando l'*Artifact Resolution Service*.

Attribute Requester. Costituisce un client web service che richiede attributi SAML alla AA come descritto nei passi 6 e 7 del protocollo generico di SSO. Simmetricamente nel caso della AA, l'*Attribute Request* è dotato di funzionalità di filtraggio degli attributi ricevuti. Essi infatti possono essere scartati in base ad una *Attribute Acceptance Policy* (AAP) o rinominati per una più agevole gestione da parte del provider richiedente.

Resource Manager (RM) Si tratta del componente che implementa i passi 1 e 8 del protocollo di SSO dal lato del SP. Il suo compito principale è quello di valutare la policy di accesso in funzione delle asserzioni ottenute. Esso può inoltre creare una sessione per la risorsa protetta, nel caso in cui questa sia dinamica ovvero un'applicazione web. Sempre in quest'ultimo caso, il RM può delegare la decisione di accesso alla risorsa

stessa, limitandosi ad innescare il protocollo di SSO e a raccogliere le asserzioni.

5.4.2 Estensioni dello standard SAML

Una federazione Shibboleth può coinvolgere enti che hanno necessità di interagire, e quindi modificare, Shibboleth con la propria infrastruttura di ICT. Può inoltre rendersi necessario per un provider mantenersi interoperabile con altri sistemi, come quelli che vedremo nei prossimi capitoli. Dunque non stupisce l'attenzione riservata al gruppo di lavoro MACE alla standardizzazione di interface e protocolli, sotto forma di estensioni al già diffuso e promettente standard SAML. Shibboleth può essere infatti visto come infrastruttura necessaria a rendere un'applicazione di SAML un sistema di SSO su web completo.

Authentication Request Profile Nei protocolli di Shibboleth si standardizza un nuovo profilo per SAML detto *Authentication Request Profile*. Il suo compito è quello di estendere i profili per il SSO di SAML in modo che possano iniziare presso il SP, e non sempre dal solo IdP. I passi 1-3 dell'esempio precedente seguono per l'appunto questo profilo. In sostanza *l'Authentication Request Profile* specifica il formato della richiesta HTTP, effettuata utilizzando il metodo GET del passo 2, esplica l'introduzione del WAYF e alcune regole di analisi da parte delle entità riceventi tale richiesta (IdP e WAYF).

Metadata Profile Sia SAML che Shibboleth assumono l'esistenza di accordi, detti *metadati*, con i quali le entità partecipanti ai protocolli di SSO. Tali accordi riguardano i ruoli assunti, il formato dei nomi utenti, quali bindings o profili siano supportati, gli endpoints forniti, i certificati e le chiavi possedute, ecc. I dettagli sulla forma ed il contenuto dei metadati sono tuttavia standardizzati solo in SAML 2.0. In pratica i metadati SAML 2.0 sono dei documenti XML che descrivono le peculiarità dei providers. Tali documenti possono presentarsi anche in forma aggregata, ovvero descrivere tutti i providers all'interno di una federazione e spesso raccolti in un file pubblicato via web.

5.4.3 SimpleSAMLphp

Un ulteriore progetto da non sottovalutare per l'implementazione del SSO in ambito web è SimpleSAMLphp[36]. Questo applicativo permette l'installazione di una infrastruttura di Identity Management con interfaccia in PHP, noto linguaggio di scripting lato server.

L'obiettivo di SimpleSAMLphp è di agevolare le procedure di installazioni di Identity Provider e Service Provider.

Nella implementazione della infrastruttura di test, che vedremo nel successivo capitolo, si è scelto di utilizzare Shibboleth. Tuttavia SimpleSAMLphp rimane una valida alternativa, soprattutto per iniziare a capire i meccanismi che operano tra le diverse entità all'interno della federazione.

5.5 Infrastruttura di test

L'infrastruttura di test realizzata copre tutti gli aspetti fondamentali di una architettura di Identity Management federato. Se pur fornisce l'autenticazione di un solo servizio all'interno di una sola organizzazione è composta da tutti gli elementi tipici per il funzionamento in federazione.

In *figura 24* sono mostrati tutti passaggi che il flusso dei dati esegue al sino a raggiungere la risorsa.

[36] Sito di riferimento http://rnd.feide.no/simplesamlphp

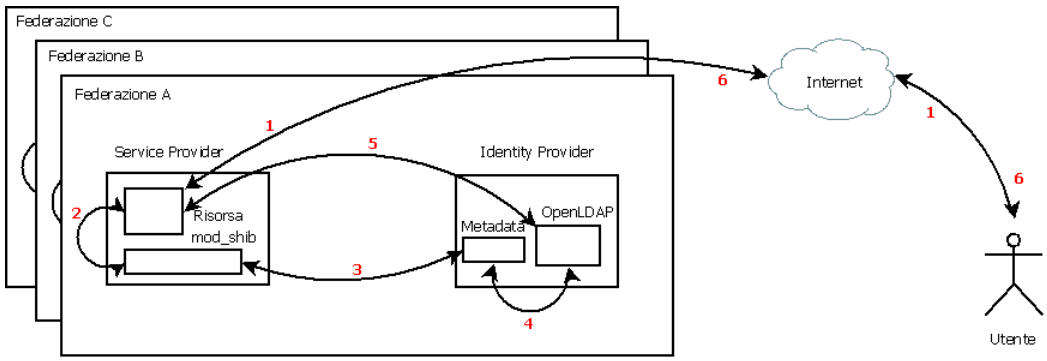

Figura 24: Infrastruttura di test

Prima di descrivere ogni singolo passaggio occorre fare una precisazione sul posizionamento dei metadati. Al fine di realizzare tale architettura sono state utilizzate due macchine: una per il Service Provider e un'altra per l'Identity Provider dove sono stati disposti anche i metadati. Quest'ultimi sono caricati in maniera statica ovvero all'interno del Service Provider è stato configurato URL in modo che punti ai metadati presente nell'Identity Provider.

In una architettura classica invece questa identificazione dei metadati avviene in maniera dinamica: quando l'utente accede alla risorsa deve dichiarare da quale organizzazione proviene, in modo da trovare il corretto Identity Provider dal quale effettuare l'autenticazione.

Dunque si è eluso questo sistema di ricerca poiché non c'è stata la possibilità di simulare una ulteriore organizzazione.

L'architettura dell'infrastruttura con i relativi flussi è mostrata nella *figura 24*. Di seguito la descrizione dei singoli passaggi:

1. L'utente vuol far accesso ad una risorsa;
2. Tale risorsa è protetta tramite il modulo di Shibboleth, il quale interroga il Service Provider per identificare i metadata;
3. Il Service Provider legge i metadati all'interno dell'Identity Provider e ridirige l'utente a una form di login;
4. L'utente inserisce le credenziali e l'Identity Provider controlla all'interno dell'OpenLDAP se l'autenticazione ha successo;
5. Ad autenticazione avvenuta l'utente è in grado di visualizzare la risorsa che aveva in precedenza richiesto;

6. Ad una ulteriore richiesta di accesso ad una risorsa non è necessario effettuare di nuovo l'operazione di login.

Inoltre nell'*appendice A* sono infine descritte tutte le operazioni da compiere al fine di realizzare questa struttura.

Capitolo 6

6 Problematiche relative all'Identity Management

Quando si affronta l'argomento relativo all'Identity Management spesso nascono diversi problemi. Questi possono essere classificati in base alla loro natura in gestionali e tecnici. Nei successivi paragrafi saranno elencate le principali difficoltà che sono state riscontrate nell'attuazione dell'infrastruttura.

In questo capitolo sono affrontati i seguenti temi:

- Problematiche gestionali
- Problematiche tecniche

6.1 Problematiche gestionali

Al livello gestionale sorgono diverse problematiche riguardo all'amministrazione come quelli seguenti:

- Processo non centralizzato ad una sola tipologia di utente;
- Molteplici team sono coinvolti;
- Amministratori spendono molto tempo in routine e task che possono essere automatizzati;
- Molti amministratori spesso assegnano differenti ID alla stessa persona.

A questi si aggiungono quelli relativi alla security:

- Problemi che si riferiscono alla sicurezza degli accessi;
- Account creati con permessi superiori ai diritti d'accesso.

Da non sottovalutare sono anche quelli relativi alla complessità:

- L'utente potrebbe assumere differenti ruoli all'interno della organizzazione;
- Molteplici autenticazioni posso essere richieste per la medesima persona;
- Cancellazione delle utenze.

Inoltre sono da considerare anche i problemi relativi all'inefficienza:

- Informazioni inesatte;
- Identità ridondanti;
- Utenti più del necessario per ottenere il proprio ID;
- Lento processo di creazione e aggiornamento dell'utente;
- Proliferazione di directory e identità;
- Terminazione dell'account.

Infine quelli relativi alla compliance:

- Essere a norma rispetto alla legge;
- Rispetto alle policy previste nel progetto.

6.2 Problematiche tecniche

In ambito tecnologico sono sorti diversi problemi. Nel caso particolare di Shibboleth possiamo elencare le seguenti problematiche:

- Numerosi file: la configurazione risulta un punto dolente per quanto riguarda Shibboleth. Troppi sono i file di configurazione ai quali è necessario apportare delle modifiche.
- Error log: i file di log molto utili in caso di errore spesso riportano troppe informazioni che distraggono l'amministratore nel capire quale informazione sia realmente utile per risolvere i problemi.
- Mancanza di interfaccia web: come nella maggior parte dei prodotti open source è ridotto al minimo indispensabile. Sarebbe stato senza dubbio utile poter configurare il tutto tramite un pannello web.
- Stabilità: più volte è stato necessario riavviare il server poiché il sistema si era bloccato.

Questi sono i maggiori problemi a livello tecnico che un amministratore si trova a dover risolvere. Tuttavia il supporto da parte della comunità di Shibboleth è molto attivo. In particolare la mailing list[37] alla quale ci si può iscrivere liberamente ponendo i propri problemi o leggendo i quesiti da parte degli altri utenti.

[37] Sito di riferimento http://shibboleth.internet2.edu/lists.html

Capitolo 7

7 Risultati ottenuti e sviluppi futuri

Il futuro relativo al campo dell'Identity e Access Management è senza dubbio quello di poter portare nuove modalità di autenticazione all'interno della propria infrastruttura. Sarebbe interessante poter studiare i seguenti metodi di autenticazioni compatibilmente con Shibboleth:

- Smart Card;
- Token;
- Voice recognition;
- Finger.

Aumentando i metodi di identificazione si otterrà indubbiamente una notevole flessibilità di accesso alla risorsa, con beneficio di tutta l'organizzazione. Inoltre si potrebbe ipotizzare anche di utilizzare Shibboleth per l'autenticazione di reti WiFi tramite Captive Portal.

Oltre ai nuovi metodi di autenticazione, è utile poter interagire con risorse che non fanno riferimento alla nostra specifica organizzazione né tanto meno alla federazione. L'esempio tipo è Google. Esso infatti offre agli utenti una serie di servizi, come account di posta elettronica, lettore di feed, storage di foto, ecc. ai quali è necessario sicuramente identificarsi. Tra le varie applicazione disponibili ce ne è una specifica per il SSO con Shibboleth[38]. Per sfruttarla è necessario avere minimo un account Google App Education Edition e configurare il proprio IdP con le specifiche presenti nel sito[39]. In questo è possibile sfruttare il SSO anche per le innumerevoli risorse che mette a disposizione un'azienda come Google.

Naturalmente questo non è il solo e unico caso in cui si prevede un'autenticazione con Shibboleth. Diversi CMS open source come quelli seguenti permettono tale meccanismo di riconoscimento dell'utente:

- Joomla[40]
- Drupal[41]

[38] Sito di riferimento https://shibboleth.usc.edu/docs/google-apps/
[39] Sito di riferimento http://code.google.com/intl/it/apis/apps/articles/shibboleth2.0.html
[40] Sito di riferimento http://dev.garr.it/idemauth/

- Moodle[42]
- Wordpress[43]

In tutti questi casi esiste un plugin che permette una estensione di autenticazione.

I risultati ottenuti dall'uso di questa tecnologia sono individuabili nelle seguenti peculiarità in quanto permette di:

- Non imporre particolari vincoli di piattaforme client da usare oltre al semplice browser poiché tutte le richieste viaggiano tramite il web;
- Usufruire di una distribuzione il più possibile periferica delle competenze e delle responsabilità in termini di autenticazione ed inoltre non necessità di dispositivi fisici di sicurezza per i terminali;
- Utilizzare certificati X.509 per l'autenticazione tra i domini;
- Servirsi del SSO;
- Diminuire il carico amministrativo per la gestione delle identità e delle credenziali.

In questo contesto è auspicabile che anche ulteriori aziende provvedano a meccanismi open di autenticazione con SSO, magari seguendo l'esempio di Google che in tale caso si è rivelato molto flessibile rispetto alla concorrenza. I vantaggi principali infatti riguardano le performance, in quanto non servirà installare alcun tipo di programma ma sarà sufficiente avere una connessione internet per usufruire di svariati servizi on-line grazie ai quali potrà essere alleggerita l'installazione di programmi (come videoscrittura o di chat) sulle singole macchine e la loro gestione e manutenzione. Inoltre l'efficienza è notevole se si guarda il lato produttivo del sistema e le risorse impiegate in rapporto al servizio fornito, ed è addirittura ottima se si parla del rapporto prezzo/servizi offerti in quanto tutti gli strumenti utilizzati sono rigorosamente open-source e il servizio offerto da Google Apps per implementare il progetto è gratuito per gli ambienti universitari.

[41] Sito di riferimento http://drupal.org/project/shib_auth
[42] Sito di riferimento http://docs.moodle.org/en/Shibboleth
[43] Sito di riferimento http://wordpress.org/extend/plugins/shibboleth/

Il futuro sarà l'identità unica all'interno di un mondo vasto come quello di Internet. Non risulta quindi assurdo immaginare un solo ID appartenente a differenti federazioni, situazione che rispecchia a pieno la realtà: ognuno di noi è unico ma a seconda del contesto assume diversi ruoli con diverse capacità. Dunque risulta fondamentale i consorzi che si occupano di standardizzare i protocolli con i quali tutti devono fare riferimento. Attualmente questa realtà è ampiamente diffusa in ambito accademico ma considerando i tempi con i quali intere infrastrutture informatiche si evolvono non dovremmo attendere molto affinché aziende e enti pubblici facciano capo a diverse federazioni ognuna specifica all'ambito in cui operano. Un progetto appena nato è quello di ViviFacile[44] il quale si pone l'obiettivo dell'interoperabilità applicativa e dell'identità unica. Infatti con una sola password l'utente sarà in grado di accedere ai servizi on line delle pubbliche amministrazione che aderiranno al servizio. Il vantaggio per cittadini, professionisti e imprese è la sola identità valida per tutti i servizi. Questo significa meno burocrazia e maggiore efficienza, significa meno code allo sportello per avere informazioni che si possono tranquillamente ottenere da casa o dall'ufficio, meno disagi per ottenere informazioni e servizi.

[44] Sito di riferimento http://www.vivifacile.gov.it/

Conclusioni

L'utilizzo di una infrastruttura federata porta con se numerosi vantaggi come l'eliminazione del numero delle credenziali, riduzione dei carichi dovuti alla gestione di numerose identità da parte del Service Provider ed inoltre una organizzazione più flessibile in caso di policy da gestire.

Tutti questi vantaggi rafforzano l'importanza sull'installazione di un sistema di Identity Management anche quando risulta essere poco documentato e articolato come nel nostro caso. La fase di configurazione non è stata semplice ed intuitiva, bensì ricca di imprevisti e numerosi errori. Dunque, scopo di questo lavoro è stato quello di far chiarezza sull'installazione di una infrastruttura di test al fine di poterla applicare anche in produzione riducendo al minimo le problematiche fin ora riscontrate.

Bibliografia

[1] Erika McCallister, Tim Grance, Karen Scarfone. **Guide to Protecting the Confidentiality of Personally Information PII**, Recommendations of the National Institute of Standards and Technology, January 2009

[2] Crespo, A., Mendez, R., Liesebach, K. **PICOS – Privacy and Identity Management for Community Services**, W3C Workshop on the Future of Social Networking Proceedings, W3C, Barcelona, Spain, 2009 http://www.w3.org/2008/09/msnws/papers/W3C_PositionPaper_PICOS.pdf

[3] Marc Barisch, Alfredo Matos. **Integrating User Identity Management Systems with the Host Identity Protocol**, ISCC EEE Symposium on Computers and Communications (ISCC'09), July 5th-9th Sousse Tunisia. http://www.ist-swift.org/content/view/52/46/

[4] Kim Cameron, **The Laws of Identity**, Identity and Access Architect, Microsoft Corporation Dicember 2005 http://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf

[5] European Commission, **eID easier access to public services across the EU**, Information Society and Media May 2008 http://ec.europa.eu/information_society/activities/egovernment/implementation/ict_psp/doc/eid_factsheet.pdf

[6] S.Cantor, **Shibboleth Architecture: Technical Overview**. Internet2-MACE Working Draft02, June2005. http://shibboleth.internet2.edu/docs/draft-mace-shibboleth-tech-overview-02.pdf

[7] S.Cantoretal, **Shibboleth Architecture: Protocols and Profiles**. Internet2-MACE Working Draft10, August 2005. http://shibboleth.internet2.edu/docs/draft-mace-shibboleth-arch-protocols-10.pdf

[8] Eve Maler, **The Security Assertion Markup Language and Liberty Web Services**, Sun Microsystems, February 2007 http://colab.cim3.net/file/work/Expedition_Workshop/2007-02-27_IdentityManagementExploration/Maler_SAML_Liberty_20070227.pdf

[9] **Identity & Access Management Strategy**, Secude Global Consulting, February 2009 http://www.secude.com/html/fileadmin/wp_tracking/wp_tracking.php?wp_id=5

Glossario

AAA Authentication Authorization Accounting

AAI Authentication and Authorization Infrastructure

AC Attribute Certificate

ACL Access Control List

AIM Identity Access Management

AJP Apache JServ Protocol

CA Certification Authority

CAS Central Authentication Service

CMS Content Management System

CRL Certificate Revocation List

DE Desktop Environment

DSEE Directory Server Enterprise-Edition

FP7 Seventh Framework Programme for research and technological development

ICT Information and Communication Technology

IdP Identity Provider

NAS Network Access Server

NIST National Institute of Standards and Technology

OASIS Organization for the Advancement of Structured Information Standards

OASIS Organization for the Advancement of Structured Information Standards

OCSP Online Certificate Status Protocol

OECD Organisation for Economic Co-Operation and Development

OID Object Identifier

PICOS Privacy and Identity Management for Community Services

PII Personal Identifying Information

PKC Public Key Certificate

PKI Public Key Infrastructure

RADIUS Remote Authentication Dial-In User Service

SOAP Simple Object Access Protocol

SSH Secure SHell

SSO Single Sign On

TACACS+ Terminal Access Controller Access-Control System Plus

TERENA Trans-European Research and Education Networking Association

TLS Transport Layer Security

WAYF Where are you from

XML eXtensible Markup Language

Appendice A

1 Realizzazione della infrastruttura

In questo capitolo viene descritta una procedura per realizzazione di una infrastruttura di Identity Management di test. Lo scopo di questo capitolo è quello di poter evidenziare e risolvere tutti gli aspetti tecnici con i quali un amministratore di sistema si deve confrontare.

Per la realizzazione dell'infrastruttura di test si sono scelte macchine con sistema operativo open source, nello specifico Debian netistal versione 5.04[45]. Inoltre poiché tale sistema è stato complementamente virtualizzato mediante l'ausilio di VMmare[46], si è optato per una installazione priva di DE, in modo da ottimizzare le macchine renderle più flessibili e leggere sotto il profilo del carico di sistema.

1.1 LDAP

LDAP *Lightweight Directory Access Protocol* è sostanzialmente un protocollo di gestione e accesso a directory service. Quest'ultimo è utilizzato per associare nomi ad oggetti, dove ogni oggetto è caratterizzato da una serie di attributi costituiti da una coppia *nome - insieme di valori.* I directory service sono ottimizzati per effettuare ricerche di oggetti, ricerche che possono avvenire in base al nome dell'oggetto, ma anche per il valore di un dato attributo. In genere gli oggetti di un directory service rappresentano un elemento dell'ambiente in cui viene utilizzato il servizio, per esempio un utente, un computer, una stampante o una rete, ed ogni oggetto conterrà una serie di attributi che servono per descrivere ciò che rappresenta. Una directory è quindi un insieme di oggetti, e un directory

[45] Sito di riferimento http://www.debian.org/CD/netinst/
[46] Sito di riferimento http://www.vmware.com/it/

service è un servizio che ha lo scopo di gestire gli oggetti di una directory ed effettuare ricerche su di essi.

In LDAP le informazioni vengono organizzate in modo gerarchico, in una struttura ad albero, dove ogni nodo rappresenta un oggetto, chiamato *entry*.

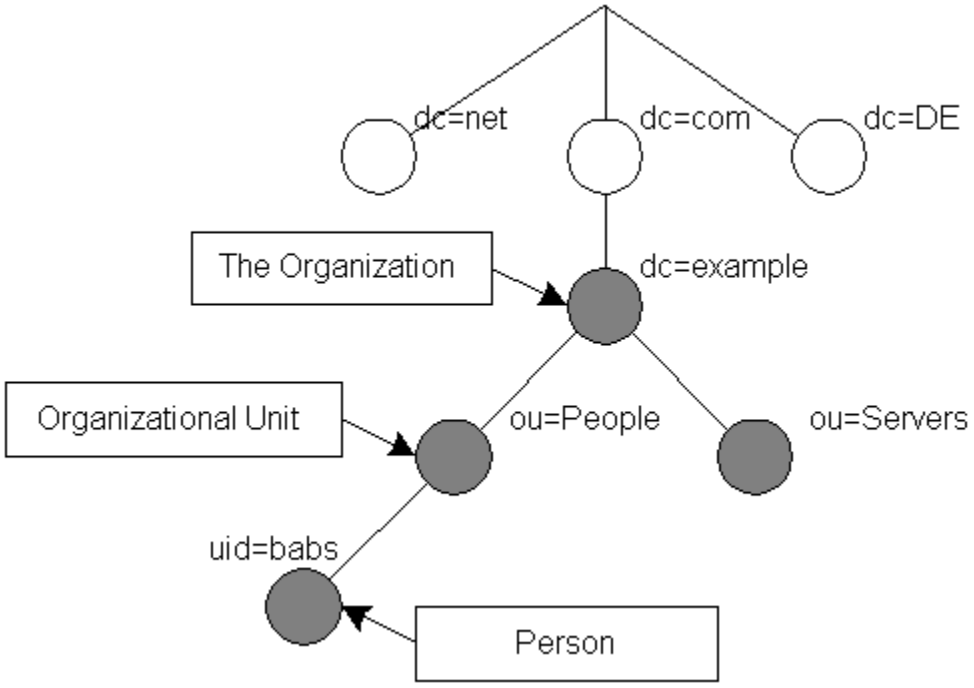

Figura 25: Albero delle entry di LDAP

L'albero delle *entry* è chiamato anche **DIT** *Directory Information Tree*. Ogni entry del DIT è identificata univocamente dal suo **DN** *Distinguished Name*, composto dalla concatenazione dei nomi dei suoi antenati fino alla radice. Ad esempio guardando *figura 25*:

```
dn: dc=net, dc=example, ou=People, uid=babs
```

Ogni entry è costituita da un insieme di coppie attributo-valore. L'attributo speciale *ObjectClass* definisce la struttura della entry, cioè quali attributi sono presenti e quali di questi sono obbligatori ovvero devono per forza

contenere un valore. Una object class è un insieme di attributi che hanno, in genere, un significato comune. Per esempio l'*object class person* ha un insieme di attributi che servono per descrivere una persona, come il nome, il cognome o il numero di telefono. Nella definizione di una object class è possibile indicare quali attributi sono obbligari e quali invece sono facoltativi.

Il valore dell'attributo speciale *ObjectClass* in una *entry* dev'essere il nome di una object class, e la struttura della *entry* sarà composta dagli attributi definiti in tale object class. La struttura di ogni entry è indipendente dagli altri oggetti e dalla posizione in cui si trova nell'albero.

1.1.1 Attributi

LDAP utilizza la sintassi X.500 per la definizione degli attributi. Per ogni attributo sono specificati:

- **OID (Object IDentifier):** numero identificativo dell'attributo, assegnato gratuitamente da IANA[47];
- **Nome:** nome identificativo dell'attributo, con eventuali alias di seguito;
- **Significato:** breve descrizione dell'utilizzo dell'attributo;
- **Sintassi:** numero identificativo standard della sintassi utilizzata per il valore dell'attributo;
- **Regole di confronto:** regola con cui confrontare il valore dell'attributo in caso di ricerca;
- **Molteplicità:** indica se è possibile memorizzare più di un valore nell'attributo.

[47] Sito di riferimento http://www.iana.org/

1.1.2 Object Class

Anche per la definizione delle object class viene utilizzata la sintassi X.500. Per ogni object class sono specificati:

- **OID**: numero identificativo della classe, assegnato gratuitamente da IANA.
- **Descrizione:** breve descrizione del significato dell'object class.
- **Categoria:** può essere *abstract*, *auxiliary* oppure *structural*. Ogni entry deve avere almeno una object class structural, e non può avere solo *object class auxiliary*. Una *object class abstract* invece può essere usata solamente per derivare altre object class.
- **Attributi obbligatori:** elenco di attributi che devono esistere nella entry.
- **Attributi opzionali:** elenco di attributi che possono esistere nella entry.

1.1.3 Schema

Uno schema è un insieme di definizioni di attributi e object class. Esistono alcuni schemi standard, per esempio RFC 4519[48], ma in generale è possibile definire un proprio schema personalizzato.

Alcune organizzazioni realizzano e promuovono i loro schemi in modo da standardizzare l'utilizzo di alcuni attributi comuni e favorire la realizzazione di strutture federate, per esempio Internet2 ha realizzato lo schema *eduPerson*, utilizzabile per descrivere le persone in ambiente universitario.

[48] Sito di riferimento http://www.rfc-editor.org/rfc/rfc4519.txt Sito della RFC 4519

1.1.4 LDIF

LDIF *LDAP Data Interchange Format* è un formato standard per rappresentare le informazioni contenute in una directory in un file di testo. Spesso è utilizzato per importare, esportare o modificare i dati di un directory service.

Un file LDIF è composto da un'insieme di *entry* separate da linee vuote. Per ogni entry vengono rappresentati gli attributi nella forma *nome attributo: valore*. Il seguente esempio rappresenta una *entry* in formato LDIF:

```
#       creazione utente
dn: cn=michelemanzotti,dc=e-lios,dc=eu
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
givenName: michele
sn: manzotti
mail: michele.manzotti@studenti.unicam.it
userPassword: 123456
cn: michelemanzotti
```

1.1.5 Vantaggi LDAP

Un directory service LDAP è particolarmente adatto per implementare una infrastruttura centralizzata per la gestione delle identità: permette di rappresentare facilmente gli utenti e i loro attributi, la struttura basata su schemi è facilmente estendibile e modificabile, è un servizio ottimizzato per le letture e le ricerche che costituiscono sicuramente la maggior parte delle operazioni in questo tipo di infrastrutture, ed è ormai supportato da tutte le principali tecnologie per la gestione dell'accesso. Inoltre è

possibile stabilire connessioni sicure tra client e server con TLS, e definire delle ACL per limitare l'accesso alle informazioni mantenute nel server. LDAP è quindi un'ottima soluzione, ha vantaggi notevoli anche in termini di sicurezza, è sempre più diffuso e sta diventando ormai uno standard nella gestione centralizzata degli utenti.

Per quanto riguarda le implementazioni di server LDAP, ne esistono diverse, fra quelle proprietarie ricordiamo *DSEE* di SUN, *eDirectory* di Novell e *Active Directory* di Microsoft, mentre le più diffuse implementazioni libere sono Fedora *Directory Server*, *OpenDS*, *Apache DS* e *OpenLDAP*.

1.1.6 Installazione

Nella infrastruttura di test si è scelto di adottare una soluzione open ovvero *OpenLDAP* su macchina Debian netinst versione 5.04, la stessa nella quale poi verrà installato l'Identity Provider.

Al fine di installare il server *OpenLDAP* è necessario procurarsi i seguenti pacchetti:

- slapd
- ldap-utils
- phpldapadmin

Il comando per l'installazione dei pacchetti è *apt* disponibile già di default nella distribuzione:

```
# apt-get install slapd
# apt-get install ldap-utils
# apt-get install phpldapadmin
```

1.1.7 Configurazione

Nei pacchetti installati si può notare la presenza di *phpldapadmin*. Dal nome è facile intuire le sue funzionalità ovvero è un client *OpenLDAP* che consente di configurare il server direttamente dal browser. Questo risulta molto utile soprattutto in macchine dove non è possibile accedere attraverso l'interfaccia grafica, come nel nostro caso.

Bisogna comunque ricordare che sicuramente è possibile configurare il server tramite i comandi da shell come *ldapadd* ed effettuare delle query con *ldapsearch*, ma risulterebbe un metodo molto arcaico e dispendioso se fosse necessario gestire una grossa quantità di utenti.

Una volta installato il pacchetto *phpldapadmin* è possibile procedere con la popolazione del database. Per far questo in maniera rapida è consigliato creare un file con estensione *ldif* e importarlo all'interno di *phpldapadmin*. Di seguito un esempio:

```
#         creazione utente
dn: cn=michelemanzotti,dc=e-lios,dc=eu
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
givenName: michele
sn: manzotti
mail: michele.manzotti@studenti.unicam.it
userPassword: 123456
cn: michelemanzotti
```

Altrimenti via browser è possibile inserire gli utenti di volta in volta.

1.1.8 Test del servizio

Una volta terminato il processo di inserimento delle utenze con gli opportuni attributi è raccomandato effettuare delle query di test. Nel caso si disponga di una macchina unix, come nel nostro caso la macchina stessa, è possibile interrogare il database secondo questa sintassi:

```
ldap://idp.e-lios.eu/CN=admin,DC=e-lios,DC=eu
```

In caso di macchina con Windows dal menu *Esegui* è possibile lanciare il seguente comando e se siamo annessi al dominio saremo anche in grado di effettuare delle query direttamente dalla rubrica di Windows.

```
# ldapsearch -b "cn=Michele Manzotti, dc=e-lios, \
dc=eu" -host idp.e-lios.eu -p 389 –x
```

1.2 Identity Provider

Un **IdP** *Identity Provider* è un sistema informatico che tratta individualmente le credenziali degli utenti finali e verifica che queste risultino valide. Un IdP potrebbe essere un ente finanziario, una impresa commerciale, un organismo governativo, oppure una istituzione scolastica.

Ogni IdP è in grado di autenticare l'utente e fornire informazioni aggiuntive o attributi sul suo conto; può operare uno o più servizi di

credenziali, ciascuno dei quali tratta le credenziali dell'utente basandosi su degli standard per la verifica dell'identità. Un utente può possedere credenziali da più Identity Provider, alcuni dei quali possono essere gestiti dalla stessa organizzazione o dalla stessa società. Ad esempio la *E-Authentication* ha stabilito una federazione di IdP che comprende istituti finanziari, imprese commerciali e agenzie governative.

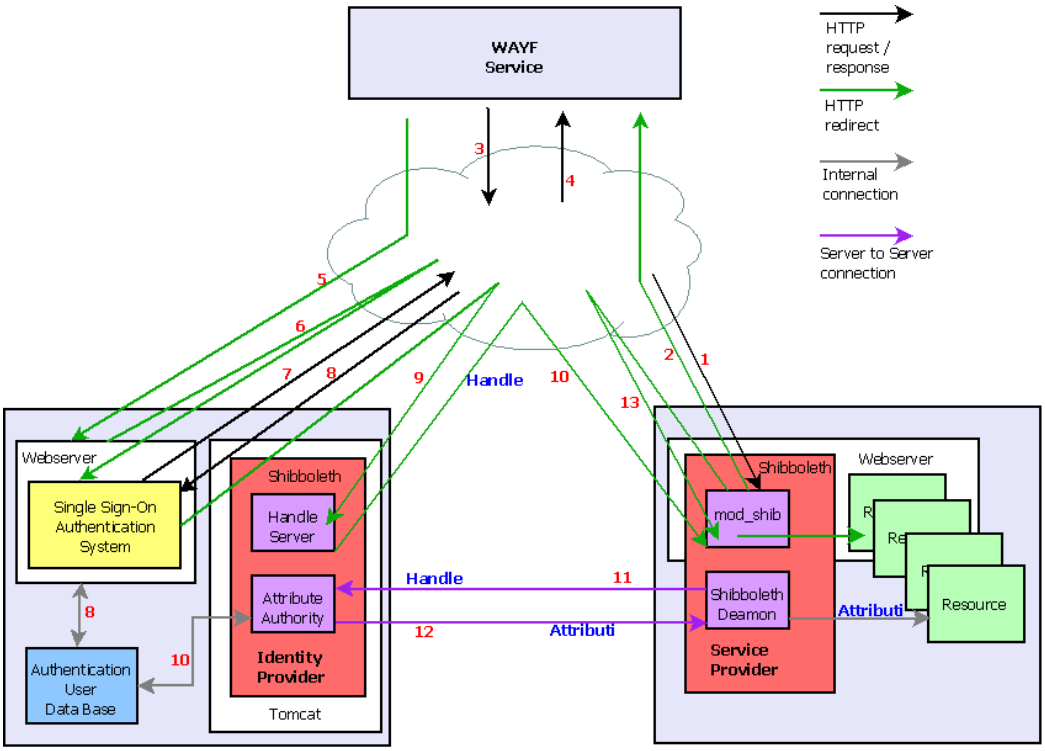

Figura 26: Comunicazione in Shibboleth tra IdP e SP

L'Identity Provider dunque è in grado di fornire asserzioni sull'identità digitale e sugli utenti conosciuti. Dunque i principali funzioni sono:

- riceve richieste di autenticazione con l'indicazione dell'insieme dei metodi di autenticazione ritenuti accettabili dal richiedente;
- produce token di autenticazione che certificano l'avvenuto riconoscimento di un subject secondo una specifica modalità.

Inoltre l'Identity Provider esegue i seguenti controlli:

- Se esiste già una sessione per l'utente;

- Se il timeout della sessione (inattività) non è superato (default: 30 min);
- Se il timeout del metodo di autenticazione (ad esempio, PKI, usernamePassword) non è superato (UsernamePassword default: 30 min);
- Se l'attributo *ForceAuthn* non è settato;

Se uno di questi punti non è verificato, l'IdP applica una autenticazione per l'utente. Entrambi i timeout, sessione e metodo di autenticazione, sono basati su inattività; essi vengono aggiornati nel corso di ogni richiesta di autenticazione. Se la sessione scade, tutti i timeout del metodo di autenticazione sono cancellati.

1.2.1 Installazione

Come detto in precedenza, la macchina dove è installato l'IdP è una Debian netinst versione 5.04. Prima di installare i pacchetti necessari è indispensabile modificare il repository in modo da poter installare anche quelli non open source. Il file da modificare è "*/etc/apt/sources.list*" ed necessario aggiungere la riga *non-free* a fianco ai repository:

```
deb http://ftp.it.debian.org/debian/ lenny main contrib non-free
deb-src http://ftp.it.debian.org/debian/ lenny main contrib non-free

deb http://security.debian.org/debian/ lenny/updates main contrib non-free
deb-src http://security.debian.org/debian/ lenny/updates main contrib non-free
```

A questo punto è possibile procedere con l'installazione dei seguenti pacchetti:

- openssl
- ntp

- apache2
- sun-java6-jdk
- tomcat5.5
- curl

```
# apt-get install openssl
# apt-get install ntp
# apt-get install apache2
# apt-get install sun-java6.jdk
# apt-get install tomcat5.5
# apt-get install curl
```

In questo modo abbiamo preparato il sistema affinché abbia i pacchetti richiesti al funzionamento dell'IdP. Ora possiamo procedere scaricando il pacchetto[49] dal sito ufficiale di Shibboleth. Una volta pronto, configuriamo le variabili d'ambiente e lanciamo l'installer.sh[50] come di seguito:

```
# jar xf shibboleth-identityprovider-2.1.2-bin.zip
# cd shibboleth-identityprovider-2.1.2
# cp ./endorsed/*.jar /usr/share/tomcat5.5/common/endorsed/
# export JAVA_HOME=/usr/lib/jvm/java-6-sun
# export CATALINA_HOME=/var/lib/tomcat5.5
#sh install.sh
```

Al termine dello script di installazione all'interno della directory di installazione avremo una serie di cartelle create dall'IdP. Di seguito una panoramica sui contenuti di queste cartelle:

[49] Sito di riferimento http://shibboleth.internet2.edu/downloads/shibboleth/idp/2.1.2/shibboleth-identityprovider-2.1.2-bin.zip

[50] presente anche l'installer.bat nel caso si debba installare l'IdP in ambiente Windows

- **bin**, contente dei tool da linea di comando
    - **accli**, attribute authority command line interface: permette di simulare un attribute query/release
    - **version**, fornisce la versione dell'IdP
- **conf**, contiene i file di configurazione dell'IdP;
- **credentials**, contiene le credenziali usate dall'IdP. Shibboleth genera di default la chiave *idp.key*, il certificato *idp.crt* e un keystore *idp.jks* contenenti entrambe;
- **lib**, contiene le librerie *jars* che l'implementano l'IdP. Sono copie di quelle presenti nei file *WAR* dell'IdP e sono utilizzate solo dai tool da linea di comando;
- **log**, contiene i file di log di Shibboleth:
    - **process log**, descrizione dettagliata delle IdP processing requests;
    - **access log**, registrazione dei client che accedono all'IdP;
    - **audit log**, registrazione di tutte le informazioni mandate fuori dall'IdP;
- **metadata**, contiene i file metadata reperiti con diverse modalità e caricati nell'IdP
- **war**, contiene i file war creati dall'installer. Di norma si fa puntare questi file, piuttosto che copiarli in Tomcat[51] per evitare di dover ripetere l'operazione in caso di riconfigurazione dell'IdP.

[51] Tomcat è è un web container open source sviluppato dalla Apache Software Foundation

1.2.2 Configurazione

Una volta terminata la procedura di installazione si può proseguire con quella di configurazione. Durante questa fase è buona norma fare sempre dei backup dei file nei quali si effettueranno delle modifiche in modo da poter ripristinare il tutto nel caso in cui si presentino dei problemi

La prima parte di configurazione riguarda il server web apache e dunque il file da modificare è **/etc/apache2/sites-available/default-ssl** .

All'interno dei tag "*IfModule*" e "*VirtualHost"* [52] è necessario aggiungere queste righe:

```
<Proxy *>
AddDefaultCharset off
Order deny,allow
</Proxy>
```

Un secondo file da cambiare è **/etc/apache2/mods-enabled/proxy.conf**. All'interno del fine è necessario posizionarsi dopo la riga "*ProxyRequests Off*" e inserire le seguenti righe:

```
<VirtualHost _default_:8443>
```

Appena terminato questo passaggio occorre fare una copia del file /etc/apache2/sites-available/default-ssl in **/etc/apache2/sites-available/default-ssl2** e modificare la stringa "*VirtualHost*" come segue:

```
SSLVerifyClient optional_no_ca
SSLVerifyDepth 10
```

[52] All'interno del file è commentata la sezione "Location" è quindi facilmente individuabile il punto dove inserire le righe

Inoltre sempre in questo file è opportuno modificare la stringa dei log "*ErrorLog*" fornendogli un nuovo percorso. Una ulteriore modifica da fare è quella di decommentare la parte relativa al tag "*Location*" nel modo seguente:

```
<Location /idp>
ProxyPass ajp://localhost:8009/idp
ProxyPassReverse ajp://localhost:8009/idp
</Location>
```

Un successivo file da configurare è **/etc/apache2/ports.conf**. In questo file è necessario modificare le porte in ascolto:

```
<IfModule mod_ssl.c>
   # SSL name based virtual hosts are not yet supported, therefore no
   # NameVirtualHost statement here
   Listen 443
   Listen 8443
</IfModule>
```

Arrivati in questa fase della configurazione è possibile riavviare il server web apache in modo da capire se le modifiche appena fatte non hanno generato errori:

```
# a2enmod ssl
# a2enmod proxy_ajp
# a2ensite default-ssl
# a2ensite default-ssl2
# /etc/init.d/apache2 force-reload
```

Subito dopo la configurazione del server web passiamo a quella di Tomcat. Quest'ultimo implementa le specifiche JSP e Servlet di Sun

Microsystems, fornendo quindi una piattaforma per l'esecuzione di applicazioni Web sviluppate nel linguaggio Java.

Il file di configurazione di tomcat è **/etc/tomcat5.5/server.xml** e per i nostri scopi è necessario modificare il connettore AJP che consente a Tomcat di ricevere le richieste di connessione dal software AJP come segue:

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" address="127.0.0.1"
enableLookups="false" redirectPort="443"
protocol="AJP/1.3" tomcatAuthentication="false" />
```

Un successivo file da modificare nell'ambiente di Tomcat è **/etc/default/tomcat5.5**. In questo file è necessario decommentare e aggiungere le seguenti parti:

```
JAVA_OPTS="-Djava.awt.headless=true -Xmx512M -\
XX:MaxPermSize=512M"
TOMCAT5_SECURITY=no
```

Potrebbe capitare che la macchina necessiti di reboot, dunque è buona norma impostare le varabili d'ambiente in maniera permanente ovvero che non si perdano ad ogni riavvio della macchina. Per far questo bisogna modificare il file **/etc/profiles** con le seguenti righe:

```
export JAVA_HOME=/usr/lib/jvm/java-6-sun
export CATALINA_HOME=/var/lib/tomcat5.5
```

Inoltre è indispensabile creare il seguente file nella directory **$CATALINA_HOME/conf/Catalina/localhost/idp.xml**[53] come il seguente:

```
<Context docBase="/opt/shibboleth-idp/war/idp.war"
privileged="true"
antiResourceLocking="false"
antiJARLocking="false"
unpackWAR="false"
swallowOutput="true" />
```

Come ultima operazione per la configurazione dell'IdP è il cambiamento dei permessi sui file che Tomcat utilizzerà:

```
chown tomcat55:nogroup /opt/shibboleth-idp/logs/
chown tomcat55:nogroup /opt/shibboleth-idp/metadata/
chown tomcat55:nogroup /opt/shibboleth-idp/credentials/
```

1.2.3 Test del servizio

Terminata la procedura di configurazione nasce la necessità di voler testare quanto fatto fin'ora anche per controllare che tutto sia andato a buon e procedere con i passi successivi.

Per testare il servizio IdP è possibile farlo in due modi:

- Localmente;
- Appoggiandosi ad un servizio esterno (consigliata).

[53] /var/lib/tomcat5.5/conf/Catalina/localhost/idp.xml

Nel primo caso sarà sufficiente indirizzare il browser all'indirizzo e una semplice scritta "OK" ci informerà che l'IdP è perfettamente funzionante:

Per poter sfruttare questo servizio è essenziale seguire i seguenti passi:

1. Registrazione al servizio OpenIDP[54] oppure ProtectNetwork[55];
2. Una volta autenticati[56], seguire la procedura guidata dove i passi principali sono la registrazione dell'IdP specificando l'url e importardo il certificato presente all'interno dell'Identity Provider[57];
3. Modificare il relay-party.xml[58] decommentando il tag *<MetadataProvider>* e cambiando il *metadataURL* in *http://www.testshib.org/metadata/testshib-providers.xml*. Infine commentare l'intero *<MetadataFilter>*.

Ora è possibile testare la macchina appena configurata collocandoci all'indirizzo *https://sp.testshib.org/* e passandogli l'URL dell'Identity Provider, ad esempio *https://e-lios.eu/idp/shibboleth.*

> *https://idp.e-lios.eu/idp/profile/Status*

1.3 Service Provider

Il Service Provider è l'ente presso il quale è gestita la risorsa web a cui l'utente fa richiesta e che ha il compito di proteggerla attraverso qualche forma di policy di acceso. Esso è costituito da un demone in C++ con il quale il web server dialoga (*mod_shib2*), intercetta le richieste a risposte, e

[54] Sito di riferimento http://openidp.org/
[55] Sito di riferimento https://www.protectnetwork.org/
[56] Sito di riferimento http://www.testshib.org/testshib-two/login.do
[57] /opt/shibboleth-idp/credentials/idp.crt
[58] /opt/shibboleth-idp/conf/relying-party.xml

ridirige l'utente sullo WAYF o all'IdP. Ricevute le informazioni di autenticazione apre una connessione verso lo AA dell'IdP per reperire gli attributi. Ogni risorsa il cui accesso deve essere protetto richiede un SP, quindi ogni servizio Web accessibile agli utenti della federazione richiede un SP.

Il Service Provider controlla:

- Se esiste già una sessione per l'utente ;
- Se il timeout della sessione (inattività) non è superato (default: 1 ora);
- Se il tempo di vita della sessione (manutenzione nella cache) non è superato (default: 8 ore);
- Se il tempo massimo dopo l'ultima autenticazione non è superato (di default non è impostato).

Se uno di questi punti non corrisponde, è avviata la procedura di autenticazione.

I dati associati a una sessione SP sono gli attributi, le informazioni di autenticazione (tempo, metodo, classe, ...); in altre parole, tutte le informazioni che sono state fornite dall' IdP dentro l'asserzione.

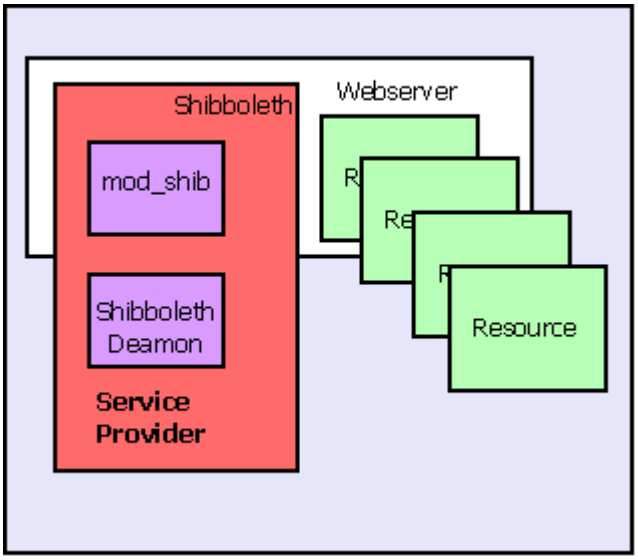

Figura 27: Service Provider di Shibboleth

Nella *figura 27* è mostrato il Service Provider di Shibboleth contenente il modulo e il demone.

1.3.1 Installazione

Come per l'Identity Provider si è scelto di installare il Service Provider su una macchina con sistema operativo Debian 5.04. Anche in questo caso non è stata installata l'interfaccia grafica in modo da rendere il sistema più leggero possibile.

I pacchetti installati all'interno del Service Provider sono i seguenti:

- openssh-server
- apache2.2
- libapache2-mod-shib2

Dunque i comandi per installarli sono:

```
# apt-get install openssh-server
# apt-get install apache2.2
# apt-get install libapache2-mod-shib2
```

Il pacchetto fondamentale è "*libapache2-mod-shib*", mentre gli altri due sono rispettivamente il server *ssh* in modo da poter effettuare una connessione al terminale remoto e il server web apache.

Una volta installati è possibile trovare la directory "*/etc/shibboleth*" nella quale sono collocati i file di configurazione di Shibboleth SP:

- **shibboleth.xml**, questo file contiene le impostazioni generali del servizio

- **attribute-map.xml**, questo file definisce la conversione tra gli attributi ricevuti dallo IdP e le variabili server
- **attribute-policy.xml** questo file definisce l'accettabilità degli attributi a partire dal loro formato

1.3.2 Configurazione

L'obiettivo di questo capitolo è quello di creare una infrastruttura di test affinché una risorsa web è protetta tramite l'autenticazione con Shibboleth. Per fare questo è quindi necessario istruire il server web in modo che una determinata directory, "*/secure*" nel nostro caso, sia raggiungibile passando attraverso Shibboleth.

Il file di configurazione in questione è **/etc/apache2/sites-available/default** dove è necessario aggiungere all'interno del tag <*VirtualHost*> le seguenti righe:

```
<Location /secure>
AuthType shibboleth
ShibRequireSession On
require valid-user
</Location>
```

Ora è possibile attivare il modulo di Shibboleth per apache e riavviare il server web:

```
# shibd -t /etc/shibboleth/shibboleth2.xml
```

A questo punto la configurazione del Service Provider è pronta per esser testata.

1.3.3 Test del servizio

Il modo per testare il Service Provider di Shibboleth è quello di verificare se il file di configurazione è stato caricato in maniera corretta. Per questo, si utilizza il seguente comando:

```
# a2enmodshib2
# /etc/init.d/apache2 restart
```

1.4 Comunicare IdP e SP

La comunicazione fra IdP e SP risulta essere la fase cruciale al fine di rendere l'intera infrastruttura di test pienamente funzionante. Come specificato in precedenza, durante lo stadio di interoperabilità fra IdP e SP si innesca una serie di richieste e risposte mostrate in *figura 21*, tutte completamente invisibili all'occhio dell'utente.

Dunque IdP e SP possono cooperare secondo diversi profili, ciascuno dei quali definisce l'ambito fiduciario e contrattuale che li collega ma uniti da unica federazione, *figura 28*.

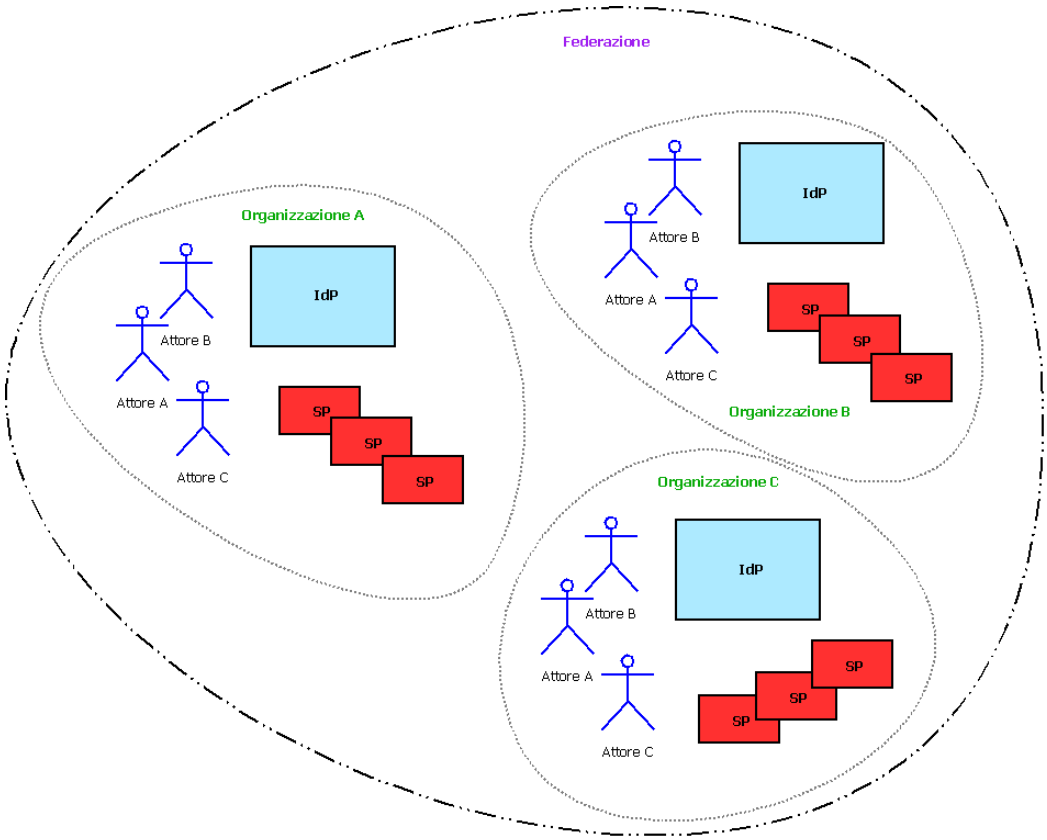

Figura 28: Diverse organizzazioni in un'unica federazione

Per far comunicare le due entità IdP e SP bisogna inizialmente modificare il **relay-party.xml** dell'Identity Provider al tag "*MetadataProvider*" facendolo puntare al metadata che poi andremo a caricare proprio nel'IdP:

```
<MetadataProvider id="URLMD"
xsi:type="FileBackedHTTPMetadataProvider"
xmlns="urn:mace:shibboleth:2.0:metadata"
metadataURL="http://idp.e-lios.eu/elios-metadata.xml"
backingFile="/opt/shibboleth-idp/metadata/elios-metadata.xml">
```

Mentre nel Service Provider dobbiamo modificare il file **shibboleth2.xml** al tag <*SessionInitiator*> affinché punti al Identity Provder:

```
<SessionInitiator type="Chaining" Location="/Login" isDefault="true"
id="Intranet" relayState="cookie" entityID="https://idp.e-
lios.eu/idp/shibboleth">
```

Inoltre sempre all'interno del file **shibboleth2.xml** al tag <*MetadataProvider*> dobbiamo far puntare il Service Provider ai metadata condivisi con l'Identity Provider:

```
<MetadataProvider type="XML" uri="http://idp.e-lios.eu/elios-metadata.xml" backingFilePath="elios-metadata.xml" reloadInterval="7200"> </MetadataProvider>
```

Successivamente, un ulteriore tag da modificare è <*ApplicatoionDefaults*> in modo che dichiariamo quale directory proteggere:

```
<ApplicationDefaults id="default" policyId="default" entityID="http://193.205.92.192/secure/" homeURL="http://193.205.92.192/secure/index.html" ….
```

Inoltre nel tag <*Host*> si indica con quale tipologia deve esere protetta:

```
<Host name="http://193.205.92.192"> <Path name="secure" authType="shibboleth" requireSession="true" requireSessionWith="intranet"/> </Host>
```

A questo punto è necessario creare il certificato per poter configurare un protocollo di autenticazione sicuro. Il metodo più semplice è quello riportato nella lista di comandi qui di seguito:

```
# openssl genrsa -des3 -out server.key 1024

# openssl req -new -key server.key -out server.csr

# cp server.key server.key.org

# openssl rsa -in server.key.org -out server.key

# openssl x509 -req -days 365 -in server.csr -signkey server.key -out \

server.crt
```

Una volta ottenuto il file **server.crt**, si deve configurare sempre il file **shibboleth2.xml** al punto del tag <*CredentialResolver*> come segue:

```
<CredentialResolver type="File"
key="/etc/shibboleth/cert/server.key"
certificate="/etc/shibboleth/cert/server.crt"/>
```

Infine, ultima operazione da eseguire, è commentare il tag <*SignatureMetadataFilter*>:

```
<!-- <SignatureMetadataFilter certificate="fedsigner.pem"/>  -->
```

1.4.1 Metadata

I metadati di SAML 2.0 sono organizzati in un'insieme estendibile di regole rappresentanti le combinazioni comuni dei protocolli e dei profili SAML. I profili SAML 2.0 richiedono che una federazione si accordi per quanto riguarda le entità che ne fanno parte, le risorse, il supporto e gli end point che esplicano effettivamente i servizi federati. In questo contesto i metadati servono per descrivere tutte queste informazioni in un modo standardizzato.

In generale un metadato SAML non deve essere preso come una dichiarazione autorevole circa le capacità o le opzioni di una data entità del sistema e quindi l'omissione di una particolare opzione non implica che questa non è supportata, ma solamente che non è stata richiesta nel particolare caso. Viceversa, se viene indicato il supporto per un determinato attributo non implica che un richiedente possa o voglia riceverlo.

I principali tipi di metadati sono:

- **Namespaces** : i metadati SAML usano i seguenti namespace: *rn:oasis:names:tc:SAML:2.0*. La specifica di OASIS usa il prefisso di namespace *md* per fare riferimento al namespace.
- **Common Types** : sono definiti diversi tipi comuni, utilizzati nella definizione degli attributi e degli elementi. Uno fra i più importanti è l'*entityIDtype*, il quale restringe il tipo di dati dello schema XML a una lunghezza massima di 1024 caratteri ed è usato come identificatore unico per le entità SAML;
- **Root Elements** : un'istanza di un metadato SAML può descrivere una singola entità o più di una;
- **Role Descriptor Elements** : gli elementi di questa sezione rappresentano il grosso della componente di supporto ai metadati; ogni elemento definisce una specifica collezione di comportamenti operativi a supportodei profili SAML;
- **Affiliator Descriptor** : è una alternativa alla sequenza di descrittori di regole che viene utilizzata quando un *<EntityDescriptor>* descriveun collegamento di entità SAML piuttosto che una sola.

I metadatati possono essere salvati sia localmente ch su un server remoto

Per creare il metadata condiviso su un server remoto bisogna unire i due file dei metadati generati dall'Identity Provider e dal Service Provider. Nel primo caso possiamo reperirli nel file */opt/shibboleth-idp/metadata/idp-metadata.xml*. Mentre quelli dell'SP vengono generati a runtime, e recuperati all'indirizzo http://SP/Shibboleth.sso/Metadata, dove SP è l'indirizzo ip del Service Provider.

La creazione del metadata condiviso avviene tramite l'unione dei due metadati all'interno di un contenitore con tag <*EntitiesDescription*>. Di seguito un frammento di metadata:

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

<EntitiesDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:shibmd="urn:mace:shibboleth:metadata:1.0"
xsi:schemaLocation="urn:oasis:names:tc:SAML:2.0:metadata sstc-saml-
schema-metadata-2.0.xsd urn:mace:shibboleth:metadata:1.0 shibboleth-
metadata-1.0.xsd http://www.w3.org/2001/04/xmlenc# xenc-schema.xsd
http://www.w3.org/2000/09/xmldsig# xmldsig-core-schema.xsd"
Name="urn:mace:garr.it:idem" validUntil="2020-01-01T00:00:00Z">

……
……

</EntitiesDescriptor>
```

1.4.2 Test del servizio

Ora è possibile testare tutta l'infrastruttura semplicemente collegandoci all'indirizzo della risorsa protetta del Service Provider, http://sp.e-lios.eu/secure/ dove comparirà una form di login di Shibboleth come in *figura 29*.

Figura 29: Form di autenticazione di Shibboleth

Notare che in questa fase siamo indirizzati all'IdP che precedentemente avevamo configurato e infatti inserendo le credenziali presenti nel server *OpenLDAP* siamo in grado di autenticarci e accedere al servizio, *figura 30*.

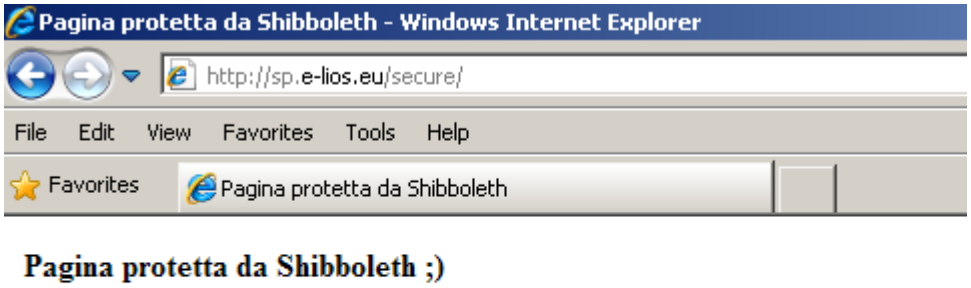

Figura 30: Login effettuato tramite Shibboleth

1.4.3 Integrazione con Joomla

Oltre alla risorsa protetta dal Service Provider è possibile configurare Shibboleth con i principali CMS oggi in circolazione. Quello testato è Joomla perché dispone di un componente[59] nato per la federazione IDEM che consente di autenticare un utente all'interno del portale leggendo i dati dall'Identity Provider.

Dopo aver installato l'ambiente Joomla[60] e il componente con il relativo plugin è necessario configurare all'interno dell'IdP il file **attribute-resolver.xml** decommentando gli attributi che si vuol far mappare a shibboleth:

[59] Sito di riferimento http://dev.garr.it/idemauth/index.php/it/downloadidemauth
[60] Sito di riferimento http://www.joomla.it/download.html

```
<resolver:AttributeDefinition id="email" xsi:type="Simple"
xmlns="urn:mace:shibboleth:2.0:resolver:ad" sourceAttributeID="mail">

        <resolver:Dependency ref="myLDAP" />

                <resolver:AttributeEncoder xsi:type="SAML1String"
xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
name="urn:mace:dir:attribute-def:mail" />

      <resolver:AttributeEncoder xsi:type="SAML2String"
xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
name="urn:oid:0.9.2342.19200300.100.1.3" friendlyName="mail" />

</resolver:AttributeDefinition>
```

In questo frammento è possibile vedere come i diversi attributi vengono rimappati da shibboleth:

- urn:oid:0.9.2342.19200300.100.1.3 rappresenta l'OID nell'OpenLDAP;
- sourceAttributeID="mail" è il nome sempre nell'OpenLDAP;
- AttributeDefinition id="email" è il nome che shibboleth assegna a quell'attributo.

Dopo aver decommentato gli attributi che si vuole esportare è necessario stabilire il connettore all'OpenLDAP sempre all'interno del file **attribute-resolver.xml**

```
<resolver:DataConnector id="myLDAP" xsi:type="LDAPDirectory"
xmlns="urn:mace:shibboleth:2.0:resolver:dc"

ldapURL="ldap://192.168.1.6:389" baseDN="dc=e-lios,dc=eu"
principal="cn=admin,dc=e-lios,dc=eu"

principalCredential="password">
        <FilterTemplate>
            <![CDATA[
                (cn=$requestContext.principalName)
            ]]>
        </FilterTemplate>
</resolver:DataConnector>
```

In questo file è necessario prestare attenzione al parte dopo il CDATA che serve a collegare l'utente, magari in sola lettura, con tutto l'albero dell'OpenLDAP.

Infine è sufficiente modificare anche l'**attribute-filter.xml** in modo che gli attributi mappati precedentemente possano uscire correttamente.

```
<AttributeRule attributeID="email">
        <PermitValueRule xsi:type="basic:ANY" />
</AttributeRule>
```

Anche in questo caso è necessario farlo per ogni attributo che si vuole esportare.

Una volta terminate queste operazioni per far funzionare l'autenticazione tramite Shibboleth sarà necessario aggiungere i nuovi metadati, quelli generati dal componente come in *figura 31*, nel metadata centrale e riavviare tomcat dell'Identity Provider.

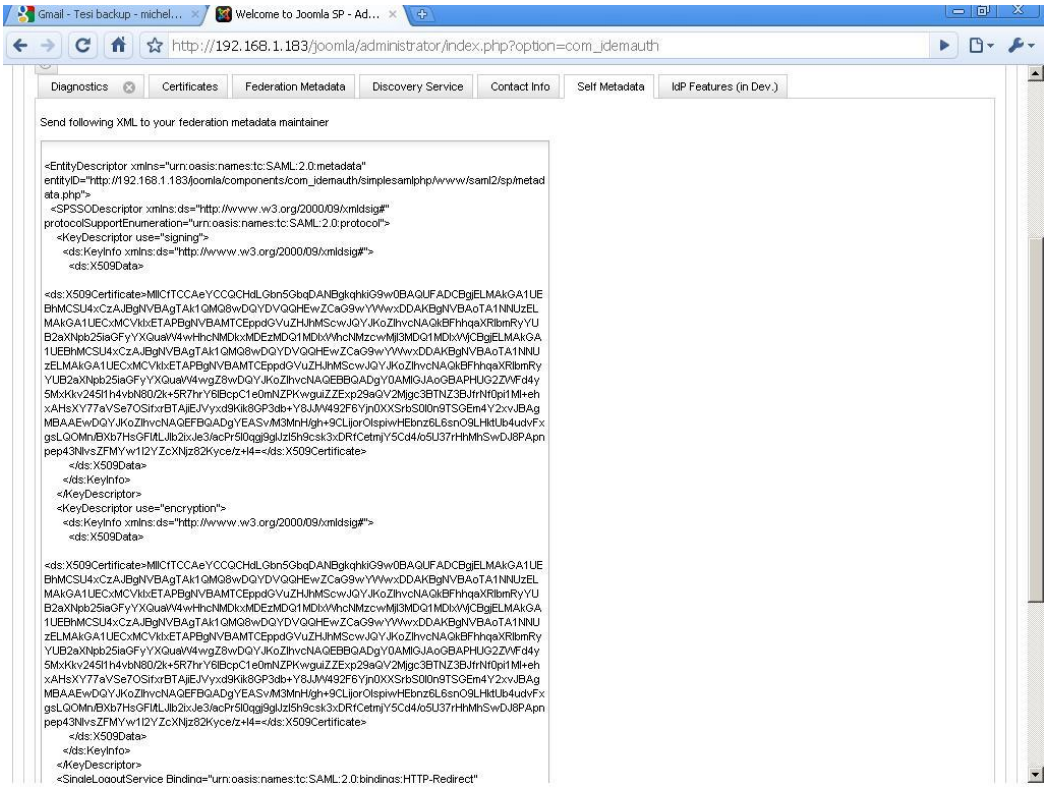

Figura 31: Metadati di Joomla

Nella pagina web del componente di Joomla è presente un bel video che illustra tutti passi per installare e configurare al meglio l'estensione.

Appendice B

1 Progetto ICAR

In questa appendice viene descritta la procedura per realizzazione di una infrastruttura conforme al progetto ICAR.

Inoltre poiché tale sistema è stato complementamente virtualizzato mediante l'ausilio di VMmare[61], si è optato per una installazione priva di DE, in modo da ottimizzare la macchina rendendola più flessibile e leggera

1.1 Installazione

Anche in questo caso si è optato per una installazione su Debian netistall versione 5.04 e si sono controllate che le porta 3443, 4443, 5443, 6443, 8006, 8080 e 9443 siano disponibili. Dopo di che si è proceduto con l'installazione dei seguenti pacchetti:

- sun-java6-jdk
- libbcprov-java
- libbcprov-java-gcj
- mysql-server
- phpmyadmin

Tutto questo grazie al comando apt di Debian:

[61] Sito di riferimento http://www.vmware.com

```
# apt-get install sun-java6-jdk
# apt-get install libbcprov-java
# apt-get install libbcprov-java-gcj
# apt-get install mysql-server
# apt-get install phpmyadmin
```

1.2 Configurazione

Il primo passo della configurazione è quello di importare il database presente all'interno del pacchetto "*icar-inf3-release-0.9.5*":

```
# mysql -u root -p < /root/icar/dbscripts/mysql/icarlp.sql
# mysql -u root -p < /root/icar/dbscripts/mysql/icarpa.sql
```

Successivamente è necessario copiare i file presenti nella directory "*$HOME/icar-inf3-release-0.9.5/catalina_base/common/endorsed*" in "*/usr/share/tomcat5.5/common/endorsed*". I file presenti sono:

- resolver.jar
- xml-apis.jar
- xercesImpl.jar
- xalan.jar
- serilizer.jar

Inoltre è necessario copiare l'intero contenuto della directory "*icar-inf3-release-0.9.5/catalina_base/**" in "*/usr/share/tomcat5.5"* e "*icar-inf3-release-0.9.5/catalina_base/conf/*"* in "*/usr/share/tomcat5.5/conf"*.

Dopo di che bisogna modificare il file "*env.sh*" in modo che punti alla directory di Java e Tomcat:

```
JAVA_HOME=/usr/lib/jvm/java-6-sun
CATALINA_HOME=/usr/share/tomcat5.5
```

A questo è possibile lanciare il comando affinché è possibile configurare la variabili d'ambiente e lanciare script "*StartTomcat.sh*":

```
# source env.sh
# sh StartTomcat.sh
```

Ultima configurazione è quella di modificare il file "*/etc/hosts"* facendo puntare diversi domini al localhost:

```
127.0.0.1  lp.icar.it idp.icar.it ar.icar.it aa.icar.it pa.icar.it sp.icar.it
```

1.3 Test del servzio

A questo è possibile testare il servizio semplicemente collegandosi all'URL "*http://sp.icar.it:8080/icar-sp-test"* e visualizzare la pagina presente in figura, per poi accedere a tutti i servizi che ICAR mette a disposizione come esempio, *figura 32*.

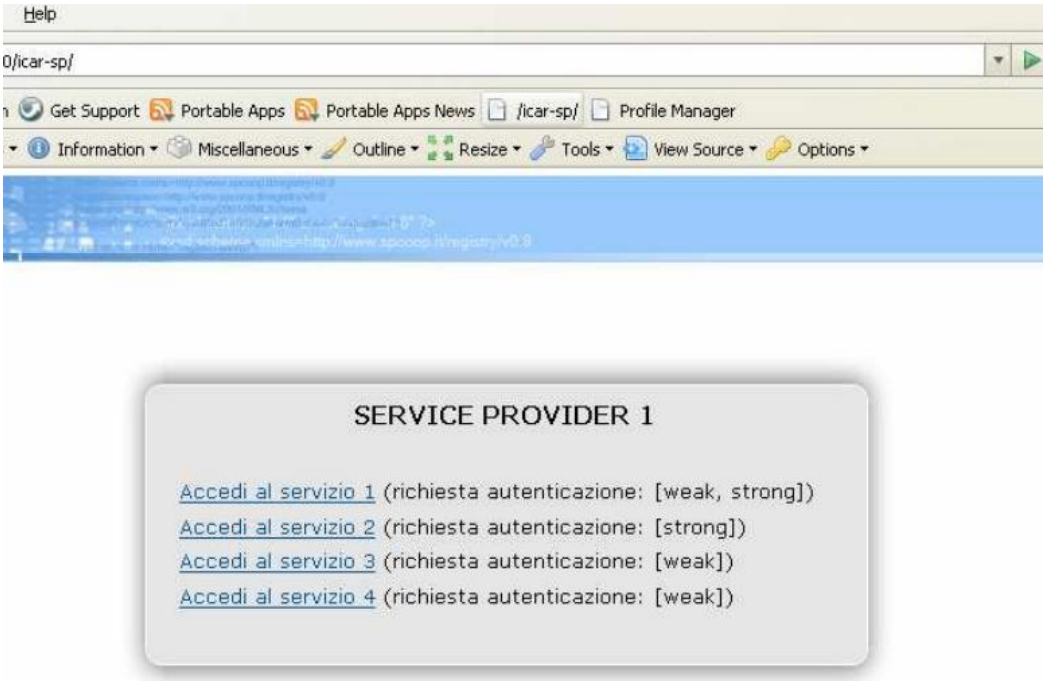

Figura 32: Progetto ICAR