ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

SCUOLA DI SCIENZE

Corso di Laurea Specialistica in Scienze di Internet

PROGETTAZIONE E IMPLEMENTAZIONE DI UN SERVIZIO DI AUTENTICAZIONE A PIÙ FATTORI PER I SERVIZI DELL'ATENEO

Relatore: Chiar.mo Prof. VITTORIO GHINI

Presentata da: CRISTIAN MEZZETTI

II SESSIONE

ANNO ACCADEMICO 2014/2015

1 Glossario

OTP - *One Time Password*, una password valida per una sola autenticazione effettuata con successo

IDP - *Identity Provider*, il ruolo dell'entità o del componente software che si occupa di fornire informazioni sull'identità, garantendo un certo livello di affidabilità

SP - *Service Provider*, il ruolo dell'entità o del componente software che si occupa di fornire un servizio ad un utente, richiedendo un'identità digitale per il suo svolgimento

ADFS - *Active Directory Federation Services*, il componente software per il *Web SSO* e federazione di identità sviluppato da Microsoft, svolge il ruolo di *Identity Provider*

Shibboleth - il componente software per il *Web SSO* e federazione di identità sviluppato dal consorzio Internet 2, a seconda della versione può svolgere il ruolo di *Identity Provider* o di *Service Provider*

Autenticazione Federata - Il meccanismo di autenticazione tra entità che prevedono relazioni di fiducia in modo da poter accettare identità create e gestite da un sistema terzo per l'uso su uno qualsiasi dei servizi facenti parte la federazione delle entità.

SSO - *Single Sign-On*, meccanismo che permette di condividere una sola sessione di autenticazione tra più servizi a cui l'utente ha accesso

SPID - Sistema Pubblico di Identità, l'iniziativa dello Stato Italiano per la costituzione di un registro di identità digitali certificate utilizzabili per l'accesso ai servizi della pubblica amministrazione

MFA (2FA) - *Multi Factor Authentication* (*2 Factor Authentication*), soluzione di autenticazione che prevede l'inserimento di più fattori (es. credenziali e OTP), normalmente comprende qualcosa a conoscenza dell'utente e qualcosa in possesso dell'utente.

Federa - Federazione degli Enti dell'Emilia-Romagna per l'Autenticazione, è la federazione che raccoglie i soggetti pubblici dell'Emilia-Romagna e consente ai cittadini di accedere ai servizi del territorio con un unico set di credenziali.

IDEM - la federazione nazionale di identità e servizi, operata dallo Stato Italiano.

EduGain - l'interfederazione degli enti accademici e di ricerca che si propone di mettere in comunicazione identità e servizi facenti parti delle diverse federazioni nazionali, è un progetto nato in ambito europeo poi esteso al contesto mondiale.

Jisc - la federazione nazionale di identità e servizi, operata dal Regno Unito

InCommon - la federazione nazionale di identità e servizi, operata dagli Stati Uniti d'America.

Garr - il consorzio italiano di Università ed Enti di Ricerca votato alla progettazione e all'implementazione della rete di comunicazione idonea a svolgere le attività costituenti gli obiettivi istituzionali.

RBAC - *Role Based Access Control*, meccanismo di controllo di accessi basato sulla verifica dell'appartenenza di un utente a un ruolo.

ABAC - *Attribute Based Access Control*, meccanismo di controllo di accessi basato sulla verifica degli attributi di contesto di un'identità digitale.

LDAP - *Lightweight Directory Access Protocol*, è un protocollo basato su IP che permette operazioni di accesso efficiente a un Directory, normalmente usato per la verifica delle autorizzazioni di un'identità digitale.

2 Introduzione

Il presente elaborato raccoglie le informazioni sull'analisi del mercato e la progettazione di un sistema di secondo fattore per i servizi dell'Ateneo. All'interno di questo progetto, che ha coinvolto la cooperazione di diversi colleghi e fornitori, mi sono occupato della migrazione dalla precedente piattaforma di autenticazione SSO, dell'analisi e progettazione della soluzione da integrare nel sistema di Ateneo e dello sviluppo del modulo di integrazione tra il sistema di autenticazione *Web Single Sign-On* dell'Università di Bologna (ADFS) e la piattaforma scelta per la fornitura (Time4ID), programmando una libreria di integrazione scritta in C# che sarà rilasciata come contributo opensource per la comunità[1].

Il panorama delle tecnologie di autenticazione e autorizzazione ha visto negli ultimi anni forti cambiamenti. Le ragioni principali di questa trasformazione vanno dalla dematerializzazione di processi e servizi sempre più spinta, all'introduzione di modalità sempre più pervasive per l'uso delle informazioni (dispositivi mobili di vario genere), alla necessità di fornire agli utenti degli strumenti che garantiscano una maggiore sicurezza delle proprie informazioni senza intralciare (troppo) la semplicità d'uso dei servizi.

I grandi attori dell'offerta tecnologica moderna (Internet2, Google, Facebook, ecc.) negli ultimi anni hanno proposto un cambio di paradigma che sta diventando sempre di più imprescindibile. I provider di servizi hanno iniziato a comprendere velocemente (anche in ambito Enterprise, dove i processi di cambiamento sono notoriamente molto più lenti) che era necessaria un'evoluzione. Per non compromettere la sicurezza delle identità dei propri utilizzatori, fornendo al contempo la capacità di interconnettere servizi e dispositivi, è stato necessario uscire dall'impostazione dell'identificativo utente come tripletta utente-password-servizio (causa di frammentazione, perdita di tempo e riuso selvaggio delle stesse identiche credenziali.

[1] Lo stesso tipo di contributo è stato rilasciato in occasione di un'integrazione precedente del sistema di autenticazione con la Federazione IDEM.

Sono così nate le architetture di autenticazione e autorizzazione federate, dove il ruolo di gestione dell'identità e delle autorizzazioni (*Identity Provider*) è fortemente disaccoppiato dal servizio (*Service Provider*), tanto da permettere che la gestione delle identità sia appannaggio di un'organizzazione separata da chi offre il servizio.

Le prime iniziative si sono avute in seno a Internet2, con la nascita del modello di autenticazione federata che poggia sulla tecnologia *SAML* la cui implementazione più diffusa e conosciuta è *Shibboleth*, standard *de facto* delle federazioni di identità che coinvolgono enti di ricerca ed università in tutto il mondo. Caratteristica di queste soluzioni è l'attenzione alla qualità dell'identità: il procedimento di rilascio delle credenziali è nella maggior parte dei casi legato alla verifica formale e alla registrazione di dati di corredo che arricchiscono il semplice nominativo.

La centralizzazione delle identità offre ovvi vantaggi per l'utente:

- unico insieme di credenziali
- unico punto di riferimento per gli obblighi relativi alla gestione delle credenziali (rinnovo periodico, complessità delle password, ecc.)
- possibilità di introdurre meccanismi di sicurezza ulteriori, incidendo direttamente su tutti i servizi collegati all'identità centralizzata

Questo è quanto già avvenuto, ad esempio, con i servizi offerti da Google, Microsoft e Facebook: un utente iscritto a questi servizi può usare l'identità registrata presso questi fornitori per accedere ad altri servizi di terze parti, senza dover ogni volta ripetere una procedura di iscrizione, è sufficiente che il servizio di terze parti supporti l'autenticazione federata verso questi fornitori di identità. È da tenere in considerazione che tutti questi fornitori non offrono la certezza sull'effettiva identità della persona (gli utenti possono creare pseudonimi), ma solo l'efficace distinzione.

Per queste identità sono poi stati introdotti gli strumenti volti a garantirne la riservatezza e la sicurezza, permettendo di certificarla con meccanismi di verifica *Out-of-*

band (cellulare o altra e-mail), miglioramento della sicurezza delle password, *behavioral analysis*[2] dell'uso dell'account per intercettare eventuali compromissioni, sistemi di autenticazione a 2 fattori.

Figura 2-1 - Autenticazione a più fattori

Questa tesi si incentra sull'ultimo di questi aspetti, il miglioramento della sicurezza delle credenziali utente attraverso una doppia verifica dell'identità:

- verifica di qualcosa che l'utente conosce (credenziali);
- verifica di qualcosa che l'utente possiede (un codice variabile, un token di sicurezza).

In particolare l'oggetto dell'elaborato è l'implementazione adottata nel contesto dell'Università di Bologna, già dotata dal 2010 di un sistema di autenticazione federata con *Web Single Sign-On*, per cui è stato sviluppato un componente per l'uso di un secondo fattore per proteggere l'accesso ai servizi più delicati.
Nel prossimo capitolo sarà sviluppata una panoramica dei principali sistemi di autenticazione e autorizzazione in ambiente Enterprise, illustrando quali sono i prodotti in uso presso l'Università di Bologna e gli sviluppi più recenti di questo genere di tecnologie.
Il quarto capitolo è costituito da una rassegna delle implementazioni più diffuse dei sistemi *cloud-based* con secondo fattore, nonché dei prodotti sul mercato disponibili per l'integrazione con sistemi *on-premise* dell'organizzazione.
Nel quinto capitolo è discusso l'inquadramento del problema, definiti i parametri per la scelta della soluzione, la progettazione e l'integrazione.

[2] (Carey, et al. 2003)

Il sesto capitolo descrive l'effettiva implementazione, partendo dalla necessaria integrazione con il sistema di aggiornamento delle identità dell'Ateneo, l'adattamento dei sistemi di supporto per l'help desk, la personalizzazione dell'applicazione per dispositivi mobili e lo sviluppo del modulo di integrazione tra ADFS e la piattaforma Time4ID. Infine il settimo capitolo delinea le possibili evoluzioni del servizio di secondo fattore per l'autenticazione alle applicazioni, calandolo nel contesto dell'Università di Bologna e definendo i possibili interventi a breve e medio termine.

3 Identità digitale

L'identità digitale non è un concetto nuovo, si tratta di un'estensione del concetto di identità all'interno della sfera *dell'Information Technology*, il metodo per distinguere le entità che interagiscono nello scambio delle informazioni. Le entità possono essere di vario genere, rappresentare persone, enti, risorse o solo alcuni aspetti di queste cose.

L'identità digitale riveste un ruolo sempre più importante man mano che l'informatizzazione dei processi della vita di ogni giorno aumenta. La corretta distinzione di attori sempre più numerosi, in interazioni sempre più complesse, è fondamentale per garantire l'autorizzazione, affidabilità e sicurezza di queste comunicazioni.

Nell'uso comune l'identità digitale è diventata sempre più sinonimo di rappresentazione digitale di una persona, creando una correlazione tra l'identificativo elettronico e le informazioni proprie che identificano la persona nel sistema di identità nazionale di appartenenza.
Secondo la descrizione classica un'identità digitale è composta da una struttura a strati che parte da un identificatore univoco, a cui si aggiungono man mano livelli di contestualizzazione ulteriori, capaci di arricchire le informazioni che si riferiscono all'entità in oggetto, con la profondità necessaria per il contesto in cui si opera.

Per fare un esempio concreto l'identificatore univoco può essere rappresentato da un codice fiscale per un cittadino italiano, gli attributi di contesto di base sono costituiti da informazioni quali nome, cognome e residenza. Estremizzando il concetto si può infine considerare lo storico degli acquisti presso il proprio negozio online di fiducia come un attributo di identità valido in quello specifico contesto.

Perché le informazioni relative ad un'identità digitale siano di qualche valore per chi le riceve, è necessario che siano determinati meccanismi di affidabilità (*trust*) tra i partecipanti all'interazione. Di norma questo avviene stabilendo predeterminati livelli di fiducia tra l'entità che garantisce l'affidabilità dell'identità digitale e il sistema che offre il servizio.

Il sistema che garantisce l'identità normalmente lo fa associando all'identificativo dell'utente un meccanismo di **autenticazione** che permetta di stabilire con un certo livello di ragionevolezza il legittimo proprietario.

Il sistema di autenticazione è critico e serve a determinare il responsabile delle azioni effettuate con quell'identità, tuttavia è l'aspetto dell'**autorizzazione** che stabilisce le operazioni a cui l'utente ha accesso e che è cruciale per garantire una corretta gestione delle risorse.

Il processo di gestione che il fornitore di identità garantisce per i propri utenti comprende:

- creazione dell'identità;
- gestione degli attributi di contesto e delle autorizzazioni;
- distruzione dell'identità;

e prende il nome di *Identity Management*.
Il ruolo di chi gestisce le identità per gli utenti è andato trasformandosi negli ultimi anni, sempre più pressante si è affacciata l'esigenza di consentire con la stessa identità l'accesso a servizi e risorse al di fuori del perimetro dell'organizzazione. Per rispondere a questa necessità sono nate le federazioni di identità, raggruppamenti interoperabili di entità in relazione di fiducia con lo scopo di permettere agli utenti di accedere più facilmente ai servizi.

A livello internazionale, nazionale (e anche regionale) le federazioni di identità raccolgono *Identity Provider* e *Service Provider*, perlopiù appartenenti al mondo degli Enti di Ricerca e delle Università, pionieri in queste soluzioni. In Europa e nel mondo EduGain è l'interfederazione che raccoglie decine di milioni di identità e migliaia di servizi, a sua volta è composta da federazioni nazionali come InCommon (USA), Jisc (UK), IDEM (Italia). Infine in Emilia-Romagna è presente un'iniziativa molto capillare che raccoglie gli enti e soggetti pubblici del territorio, Federa, la federazione gestita dall'operatore pubblico di telecomunicazioni Lepida Spa.

Le federazioni hanno creato una massa critica utile a rendere appetibile ai fornitori di servizi l'adeguamento tecnologico per potervisi inserire in modo competitivo. In UK in particolar modo una convinta politica nazionale ha fatto sì che tutti i servizi acquisiti

dal sistema nazionale venissero integrati come *Service Provider* della federazione. Questo ha permesso anche a tutti i partecipanti di *EduGain* di beneficiarne, permettendo l'uso in chiave moderna di molti servizi in precedenza ancora basati su autenticazione basati sulla rete di provenienza (in particolare i fornitori di risorse bibliotecarie elettroniche).

L'Italia ha una forte tradizione, derivata dal periodo napoleonico, nella precisa gestione dei dati anagrafici dei cittadini. Questa tradizione si è evoluta di pari passo alle tecnologie digitali, declinandone i vari aspetti (es. privacy, responsabilità giuridica) all'interno del contesto normativo. Riferimenti normativi come il CAD (*Codice Amministrazione Digitale*) o il *Codice di Protezione dei Dati Personali* (Buccianti s.d.) sono fondamentali per inquadrare i corretti comportamenti nella gestione delle identità digitali.

Recentemente sta prendendo il via una nuova iniziativa, il *Sistema Pubblico di Identità* (SPID), si tratta di una soluzione tecnicamente ispirata alle federazioni di identità già esistenti ma si propone di attivare un sistema di certificazione dell'identità estremamente affidabile, facendo leva sul mercato e su grandi operatori del panorama nazionale (Telecom, Poste, Infocert). L'intenzione è quella di dotare il cittadino di un'identità digitale unica e certificata che gli possa permettere l'accesso a tutti i servizi della Pubblica Amministrazione senza doversi preoccupare di gestire una miriade di credenziali e processi di accreditamento.

3.1 Tecnologie di autenticazione

L'autenticazione è il meccanismo che permette di stabilire, agli occhi delle parti che partecipano ad un'interazione, l'identità digitale dei partecipanti. Applicato al processo di identificazione di un utente, può avvenire sostanzialmente valutando tre categorie di informazioni:

- qualcosa che si sa (*Something You Know*): come una password o una risposta ad una domanda segreta;
- qualcosa che si ha (*Something You Have*): come una smart card o un token fisico che genera codici;

- qualcosa che si è (*Something You Are*)[3]: come un'impronta digitale o una scansione retinica.

Si definisce *Multi-Factor Authentication* (o *Two-Factor Authentication*) un processo di autenticazione che per svolgersi correttamente richieda un meccanismo di autenticazione appartenente a più di una delle categorie elencate.

Spesso questo concetto è associato alla *Strong Authentication*, un tipo di autenticazione che assicura che l'identità digitale verificata appartiene ed è riconducibile a una persona fisica o azienda la cui corrispondenza è stata correttamente verificata. Si parla infine di *Reliance Authentication* quando l'affidabilità di un set di credenziali è determinato (di solito al momento della creazione) indirettamente a partire da un'identità digitale che si ritiene affidabile e/o certificata. È questo per esempio il caso delle credenziali create sulla base della corrispondenza con un conto bancario (come fa ad esempio Paypal) o con una SIM di operatore mobile, per cui in Italia è necessaria l'identificazione anagrafica per poterne entrare in possesso.

Di seguito sono elencate alcune delle tecnologie più diffuse che permettono di effettuare l'autenticazione di un utente, pertinenti al contesto in esame.

3.1.1 Form/Basic Authentication

La più semplice tecnologia di autenticazione volta ad identificare un utente consiste nell'inserimento delle credenziali al momento della richiesta di accesso. La presentazione delle credenziali è interattiva e viene ripetuta ogni volta che si tenta di accedere ad una nuova risorsa.

La sicurezza del meccanismo è direttamente dipendente dalla modalità di interazione e alla complessità della password, se il canale di trasmissione dei dati non è opportunamente protetto (ad esempio via SSL per la pagina web di un sito) le credenziali sono compromesse. Inoltre all'aumentare del numero delle risorse il processo di autenticazione diventa tedioso per l'utente anche per la necessità di usare password lunghe e complesse.

[3] (Wayman 2008)

Il controllo delle credenziali dipende dal servizio che implementa la richiesta, può essere effettuato in qualsiasi modo (da un semplice elenco in un file di testo ad un web service remoto).

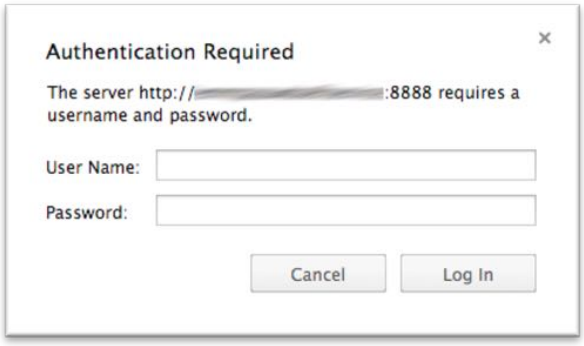

Figura 3-1 - Esempio di Basic Authentication

3.1.2 Kerberos

Kerberos è un protocollo di autenticazione pensato per l'interazione tra entità affidabili, operanti in una rete di comunicazione potenzialmente non affidabile. Le entità devono essere affidabili nel senso che se una delle partecipanti alla comunicazione è compromessa, anche la comunicazione risulta tale.

Sviluppato dal *Massachusetts Institute of Technology* nel 1993 (nella versione 5, considerata a tutt'oggi sicura a differenza delle precedenti), è stato rinnovato nelle specifiche di riferimento dall'IETF nel 2005. Dal 2007 il suo sviluppo è coordinato dal *MIT Consortium for Kerberos and Internet Trust* (MIT KIT Consortium s.d.).

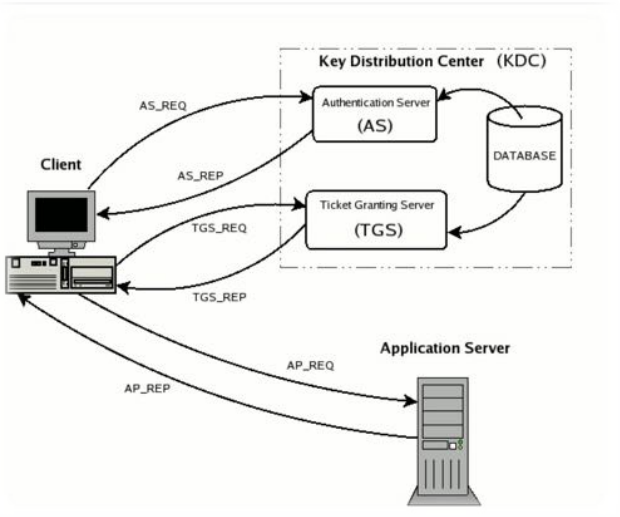

Figura 3-2 - Kerberos authentication protocol

La comunicazione attraverso il protocollo Kerberos avviene seguendo questa sequenza:

- il *Client* inizia una comunicazione verso *l'Application Server*;
- *l'Application Server* richiede al *Client* di farsi riconoscere;
- il *Client* si rivolge all'*Authentication Server* per effettuare l'autenticazione richiesta presentandosi come l'utente che ha richiesto la risorsa;
- l'*Authentication server*, nel caso trovi nel suo database di riferimento l'utente annunciato dal *Client*, risponde con un messaggio crittografato usando la chiave dell'utente insieme ad alcune informazioni di contorno per determinare la validità della sessione del *Client*;
- il *Client* decodifica il messaggio dell'*AS* usando la password inserita dall'utente, solo in caso di corrispondenza è in grado di operare la decodifica e procedere;
- il Client usa il contenuto del messaggio decodificato per presentarlo al *Ticket Granting Server* e ricevere un *ticket* utile ad accedere alla risorsa richiesta (crittografato con la chiave dell'*Application Server*);
- il Client presenta all'*Application Server* il *ticket* ricevuto che lo decodifica e accetta la comunicazione per l'accesso alla risorsa desiderata.

Kerberos è una tecnologia fondamentale alla base dei più diffusi servizi di autenticazione, uno su tutti *Active Directory* di Microsoft su cui poggiano tutte le reti su piattaforma Windows. Il modello di sicurezza su cui è basato si è dimostrato solido e robusto alla prova del tempo (MIT KIT Consortium 2008).

3.1.3 Smart card e token USB

Le smart card e i token USB sono una tecnologia di autenticazione basata sulla crittografia a chiave privata/chiave pubblica. All'interno del dispositivo è conservata la chiave privata che identifica l'identità a cui è associata.

L'accesso alla chiave privata non è possibile a meno di conoscere il PIN dispositivo che l'utente titolare della smart card o del token riceve al momento dell'assegnazione dell'identità digitale.

L'uso della crittografia a chiave pubblica consente tecnicamente di implementare tutte le principali caratteristiche di questo tipo di comunicazione:

- **riservatezza**: la possibilità di rendere impossibile la decodifica del messaggio a chi non fa parte della comunicazione;
- **non ripudio**: la caratteristica di rendere inoppugnabile la responsabilità di partecipa alla comunicazione e al suo contenuto;
- **integrità**: la capacità di rilevare manomissioni al messaggio tra l'invio e la ricezione dello stesso;
- **autenticazione**: la caratteristica di poter garantire l'identità associata a chi partecipa alla comunicazione.

In questo contesto l'aspetto di interesse è l'ultimo, grazie al quale è possibile verificare con sicurezza un'identità digitale che di norma corrisponde in senso stretto a un'identità anagrafica. Infatti le smart card sono spesso fornite attraverso procedure di assegnazione e verifica *de visu*, oppure tramite meccanismi di reliance authentication. La validità dell'identità assegnata è poi assicurata tramite l'uso di *Certification Authority* per l'emissione e revoca dei certificati.

L'autenticazione con smart card o token USB può avvenire in diversi modi, a seconda delle scelte implementative di *Identity Provider* e *Service Provider*. Un esempio può essere l'integrazione presso *l'Identity Provider* dell'organizzazione come metodo alternativo all'inserimento delle credenziali, al momento dell'accesso l'utente dovrà usare un dispositivo capace di comunicare con smart card e IDP, consentendo all'utente di apporre il PIN dispositivo e sfruttare le chiave cifrata in essa contenuta.

Normalmente si identifica nelle smart card un meccanismo di *Strong Authentication*, un servizio quindi che riceve un'identità verificata in questo modo può permettere un accesso maggiormente privilegiato all'utente.

Figura 3-3 - Smart card

L'uso di smart card e token USB è soggetto a diverse difficoltà determinate dalla loro stessa natura. Per utilizzarli è necessario che il dispositivo che si sta utilizzando sia in grado di accedere al contenuto, operazione non sempre possibile e dipendente dalle piattaforme hardware. Inoltre l'utente che usa una smart card per l'autenticazione deve averla con sé ogni volta che inizia una nuova sessione di lavoro.

Per questi motivi e per i costi intrinseci di questa soluzione, l'uso delle smart card come meccanismo di autenticazione non è molto diffuso, alcune iniziative governative[4] per l'adozione in massa di questa tecnologia si sono rivelate essere di difficile attuazione[5], relegando questa tecnologia agli ambiti dove rivestono caratteristiche importanti anche gli aspetti di non ripudio e integrità delle comunicazioni (ad esempio la firma digitale).

Molto interessante è l'iniziativa di *FIDO (Fast IDentity Online) Alliance*, che ha costituito specifiche ormai standardizzate per la produzione e l'interoperabilità di dispositivi d'autenticazione forte. La prima modalità di funzionamento è UAF, per poter offrire agli utenti transazioni sicure senza l'uso di password. La seconda modalità è U2F, costituita da un token (USB o NFC) che si aggiunge come secondo fattore al normale uso delle credenziali.

Figura 3-4 - Esperienza d'uso delle due tipologie di dispositivi FIDO - (FIDO Alliance 2012)

[4] Difficoltà nell'emissione del servizio: (Maci 2015) e (Corradini, et al. 2006)
[5] Il Comune di Bologna abbandona la sperimentazione della carta di identità elettronica (Corriere Di Bologna 2015)

3.1.4 RADIUS

Il *Remote Authentication Dial-In User Service* (RADIUS) è il protocollo per l'*Authentication-Authorization-Accounting* (AAA) più diffuso nell'ambito dei dispositivi di rete. La sua semplicità, robustezza e velocità ne hanno fatto la scelta d'elezione per questi contesti dove le risorse hardware possono essere spesso limitate per motivi di contenimento dei costi.

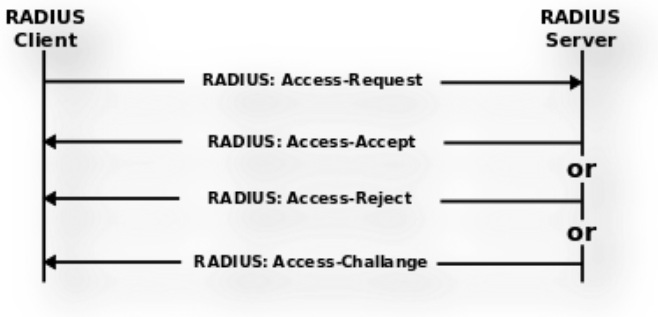

Figura 3-5 - RADIUS protocol (Wikipedia 2015)

Il protocollo prevede questa sequenza per l'autenticazione dell'identità:

- il client, normalmente rappresentato da un *Network Access Server* (NAS) come uno switch o un access point, invia una richiesta di autenticazione per conto dell'utente che vuole accedere alla risorsa mandando delle credenziali o un certificato;
- il server RADIUS può rispondere con un messaggio che richieda un secondo passo di autenticazione (un PIN o altra informazione di circostanza) che deve essere fornito dal client;
- il server RADIUS risponde con il permesso o il diniego di accesso alla risorsa richiesta.

RADIUS può eventualmente essere esteso da meccanismi di *accounting* per la tracciabilità dell'uso delle risorse.
Questo protocollo è fondamentale nell'uso delle reti aziendali per il corretto riconoscimento degli utenti appartenenti ad un'organizzazione, sia per l'accesso a reti wireless sia per l'accesso alle reti cablate.

3.1.5 CAS

Il *Central Authentication Service* è un meccanismo di autenticazione sviluppato presso Yale, a partire dal 2004 è sviluppato e mantenuto da JASIG (*Java Administratrion Special Interest Group*), organizzazione non-profit con l'obiettivo di gestire progetti opensource di interesse per gli enti di formazione superiore.

Il protocollo CAS è stato progettato per permettere l'autenticazione di applicazioni web con la possibilità di condividere una medesima sessione di lavoro (*Single Sign-On*) senza richiedere ogni volta l'immissione di credenziali.

Il sistema CAS può supportare diverse tecnologie per il riconoscimento dell'utente, dall'inserimento di una password all'uso di un certificato client. Il funzionamento si basa sulla presenza di un server di autenticazione (CAS server), a cui il browser dell'utente si rivolge dopo essere stato rimandato dall'applicazione che ha cercato di raggiungere.

L'applicazione deve essere stata precedentemente registrata nella configurazione del *CAS Server* dal *CAS Administrator*, permettendo così di procedere con l'autenticazione. Al momento della registrazione sono anche determinati parametri di funzionamento quali la partecipazione o meno al contesto di *SSO* ed eventuali informazioni aggiuntive da rilasciare all'applicazione richiedente quando si autentica un utente.

Ad autenticazione avvenuta il browser dell'utente riceve un *Service Ticket* da presentare all'applicazione, quest'ultima lo riceve e usa il *CAS Client* (componente software che implementa il *CAS Protocol*) per comunicare col *CAS Server* e validare la richiesta. Se il *CAS Server* risponde correttamente, l'utente che ha presentato il *Service Ticket* inizia una sessione di lavoro. Nel caso in cui l'utente visiti una seconda applicazione nel contesto di *SSO* l'interazione iniziale viene ignorata passando immediatamente alla validazione del *Service Ticket*, consentendo all'utente di accedere alla seconda applicazione in modo trasparente.

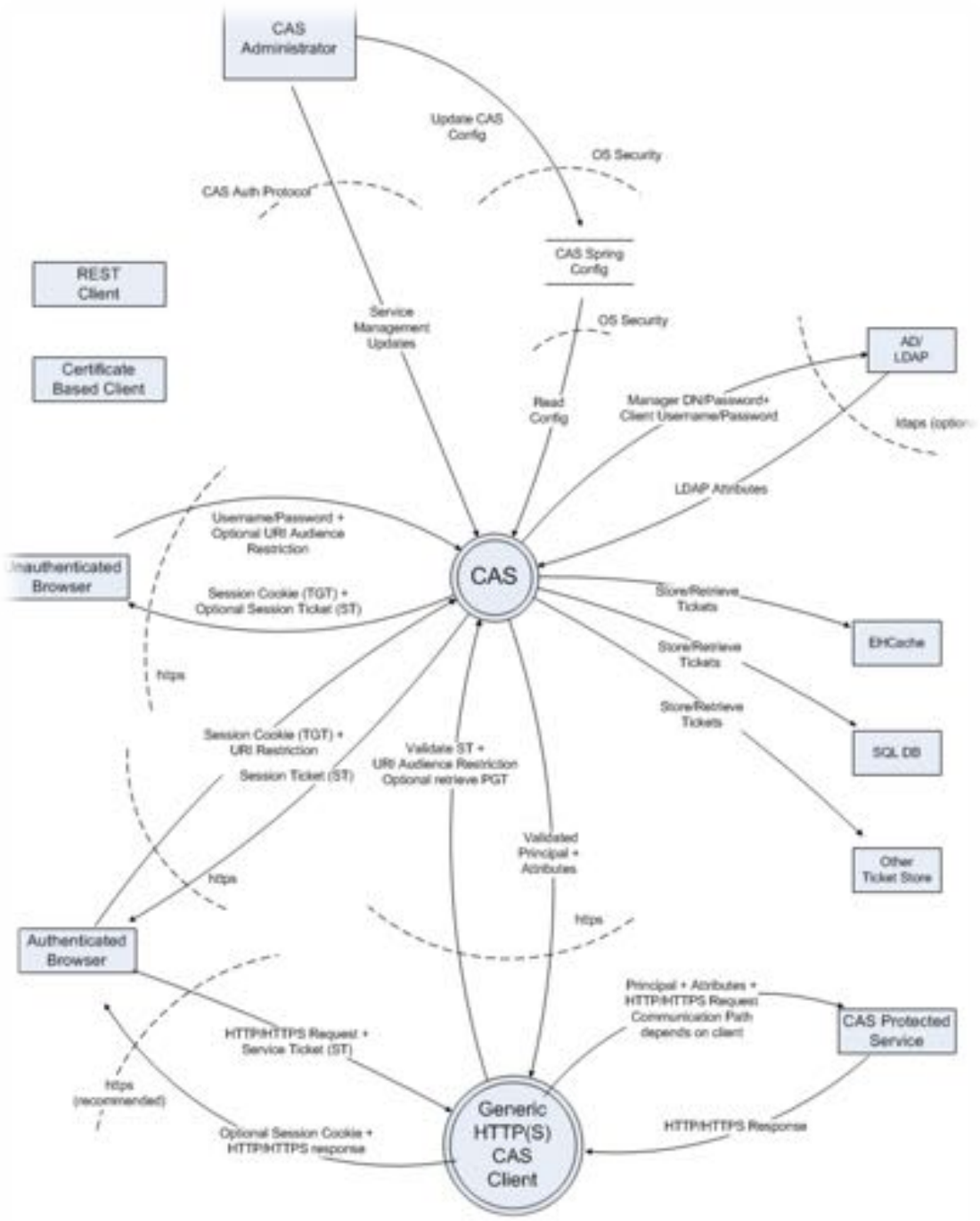

Figura 3-6 - CAS Protocol (Ohsie 2014)

3.1.6 OpenID e SAML

OpenID e *SAML* (*Security Assertion Markup Language*), come il protocollo CAS, non sono sistemi di autenticazione in senso stretto, bensì un'estensione dei sistemi tradizionali nel contesto del Web. A differenza del *Central Authentication Service* queste due tecnologie permettono di spingersi al di fuori del perimetro dell'organizzazione,

consentendo l'interazione di identità esterne con i servizi locali e viceversa, costituendo un sistema federato di identità.

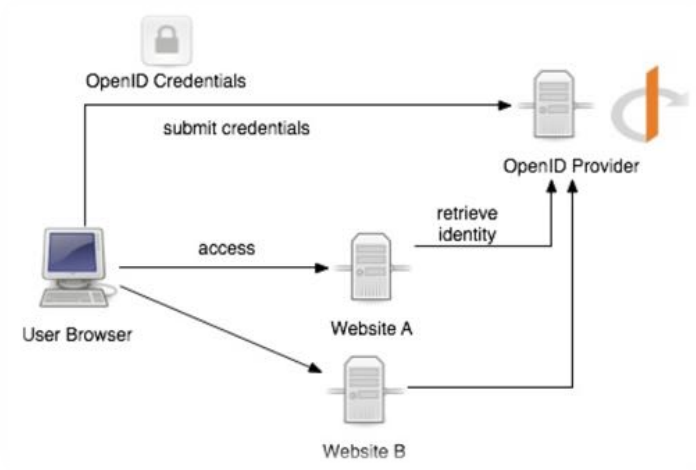

Figura 3-7 - Schema di autenticazione OpenID (Govoni 2008)

OpenID è un sistema utilizzato da grandi provider di servizi come Yahoo!, Google, Wordpress. Inizialmente adottato anche da Facebook, è stato in seguito sostituito da *Facebook Connect*. Si basa sul principio di decentralizzazione dei ruoli di *Identity Provider* e *Service Provider*, non richiede meccanismi di *trust* esplicito per funzionare, motivo per cui è adottato soprattutto in contesti dove l'esigenza è distinguere correttamente gli utenti ma non necessariamente avere certezza sulla loro identità.

Si tratta di un sistema snello volto a risolvere il problema della proliferazione degli account, consentendo agli utenti di riutilizzare la propria identità registrata presso un *OpenID provider*, offrendo alle applicazioni la possibilità di affrancarsi dalla gestione delle anagrafiche affidandosi al protocollo. L'estrema decentralizzazione spinge però sulle applicazioni l'onere di supportare i diversi *provider*, motivo per cui la scena attuale vede una polarizzazione del supporto solo su pochi provider molto diffusi (*Google*, *Facebook*).

SAML (*Security Assertion Markup Language*) - standard OASIS per lo scambio di dati relativi ad autenticazione, autorizzazione e dati sull'identità in formato XML - per contro segue un approccio più strutturato. Gli elementi costituenti sono i seguenti:

- *asserzioni*: elementi di base dei messaggi che possono essere scambiati;

- *protocolli*: le modalità di aggregazione delle asserzioni per la comunicazione tra le entità coinvolte;
- *binding*: modalità di trasferimento dei messaggi scambiati;
- *profili*: istruzioni per l'aggregazione di asserzioni, protocolli e binding per poter coprire uno specifico caso d'uso.

Il profilo di interesse in questo contesto è il *Web browser SSO profile*, riportato nella figura seguente, dove è illustrata la sequenza dei messaggi tra lo *User Agent* (il browser utilizzato dall'utente), il *Service Provider* (l'applicazione o risorsa a cui l'utente vuole accedere) e *l'Identity Provider* (il fornitore di identità).

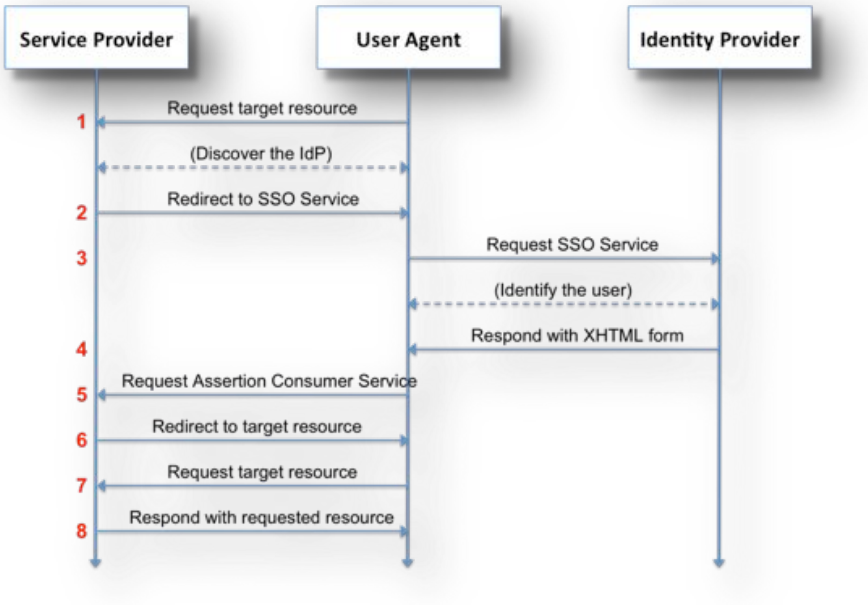

Figura 3-8 - SAML Authentication Request - Wikipedia

Al completamento dello scambio l'utente avrà una sessione autenticata che potrà permettergli di accedere ad altre risorse, facenti parte dello stesso contesto di SSO, senza inserire di nuovo le credenziali.
Nel caso di SAML il contesto di *Single Sign-On* è determinato dalle relazioni di trust in vigore tra *Service Provider*, *Identity Provider* e gli altri componenti della federazione.
Infatti per IDP e SP è possibile determinare relazioni di trust punto-punto, con l'ovvio limite della scalabilità al crescere dei partecipanti, oppure inserirsi in un contesto centralizzato che raccoglie gruppi di risorse e fornitori di identità che prende il nome di federazione di identità.

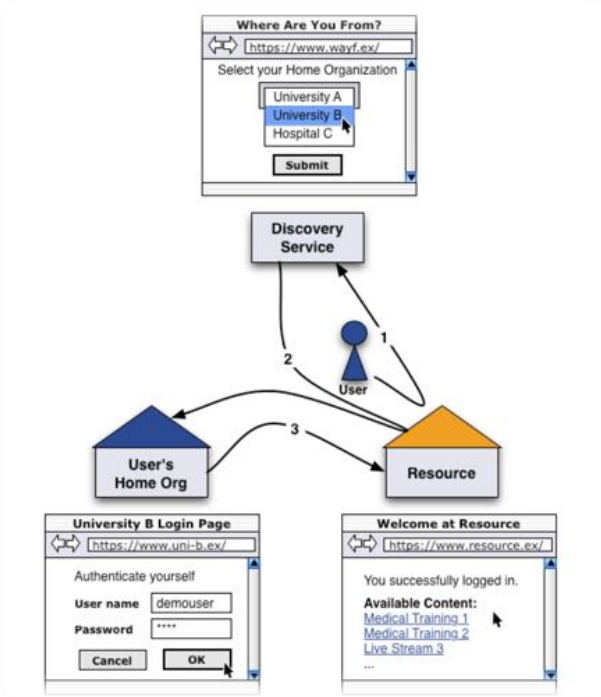

Figura 3-9 - Funzionamento di una federazione di identità (Switch Consortium s.d.)

Le federazioni di identità sono normalmente operate da realtà nazionali che si occupano di interloquire con i partecipanti e gestire un servizio centralizzato di condivisione dei metadati. Queste informazioni sono l'elemento fondante delle relazioni di *trust* e permettono di mantenere aggiornato l'elenco dei partecipanti al crescere delle entità.

In un'infrastruttura federata di autenticazione si inserisce un elemento necessario per poter individuare quale *IDP* utilizzare per verificare l'identità di un utente che richiede l'accesso a un *Service Provider SAML*. Il *Discovery Service* o *WAYF (Where Are You From)* è un componente software con cui interagisce l'utente per scegliere l'*IDP* di appartenenza, in modo da poter effettuare l'autenticazione e procedere con l'accesso al *SP*.

L'impostazione centralizzata ha diversi vantaggi ma presenta anche alcuni svantaggi quali la complessità di manutenzione e la costituzione di un *Single Point of Failure* per l'operatività della federazione, inoltre il processo di adesione può essere anche molto lento, a seconda del livello di formalità (e del ruolo) con cui si vuole entrare a farne parte. Questo livello di precisione nell'operatività ha fortunatamente il pregio di selezionare partecipanti e utenze con un livello di affidabilità mediamente elevato.

3.2 Tecnologie di autorizzazione

L'autorizzazione rappresenta la fase di determinazione dei privilegi di un'identità digitale nell'accedere ad una risorsa. Le tecnologie di autorizzazione sono parte integrante del processo autenticazione e a loro volta ne dipendono completamente.
Anche nel caso più semplice dove non c'è distinzione di privilegi tra gli utenti con accesso ad un servizio, l'autorizzazione distingue comunque i privilegi tra gli utenti con accesso e utenti senza accesso.

All'interno di organizzazioni complesse che richiedono un'attenzione non banale per le politiche di accesso alle risorse si usano meccanismi di autorizzazione di tipo *Role Based Access Control (RBAC)*.
Questa modalità permette di definire i privilegi sulla base del ruolo a cui l'identità digitale appartiene, permettendo di seguire le evoluzioni dell'identità a seconda del mutamento del rapporto con l'organizzazione. La RBAC considera solamente informazioni relative all'identità per valutare i privilegi di accesso, non è invece in grado di stabilire politiche che mettano in relazione altre informazioni legate alla risorsa che si sta cercando di usare.

Per sopperire a questi aspetti è stato sviluppato un altro concetto di autorizzazione agli accessi, l'*Attribute Based Access Control (ABAC*). Questa modalità permette di arricchire la valutazione dell'accesso con attributi appartenenti alla risorsa, all'utente o all'ambiente in generale. Un esempio di autorizzazione ABAC può essere l'utente di una rete aziendale che può accedere a siti di svago (risorsa) solo durante la pausa lavorativa (attributo di contesto).

Le tecnologie riportate di seguito possono essere usate per implementare entrambi i modelli, la modalità di autorizzazione scelta dipende dalle scelte dell'organizzazione e dall'esigenza di granularità del sistema (la cui complessità aumenta di conseguenza).

3.2.1 LDAP

Lightweight Directory Access Protocol è una tecnologia che permette di organizzare in modo efficiente le informazioni relative a un'identità digitale e interrogarla tramite

protocolli basati su IP. È ottimizzata per le operazioni più frequenti di verifica dell'identità (lettura e ricerca) ed ha una struttura gerarchica. Le modalità di funzionamento sono specificate da una serie di standard dell'IETF che ne assicurano un'ottima interoperabilità.

Le funzioni principali prevedono la possibilità di organizzare le informazioni in una struttura gerarchica, le entità appartenenti al Directory sono determinate da uno *schema*, quindi fortemente tipizzate. Ogni entità è definita da una serie di attributi i cui valori ammessi e la dimensione sono sempre definiti nello *schema*.

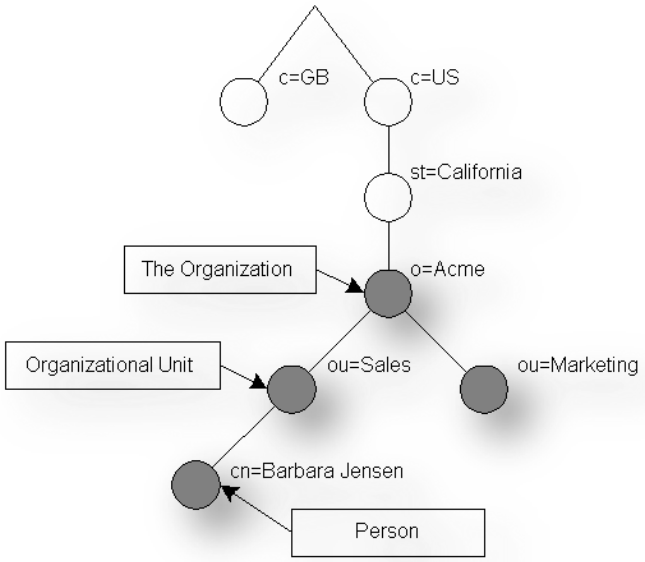

Figura 3-10 - Esempio di albero LDAP (OpenLDAP Foundation 2003)

I tipici elementi di un Directory sono:

- *utenti*: rappresentazione dell'identità digitale assegnata ad un utente, collegata ad una password in forma di hash, verificata al momento del *binding*;
- *gruppi*: permettono di specificare logiche di appartenenza su cui è possibile costruire un controllo di accessi RBAC, i gruppi possono essere innestati tra loro e l'utente può appartenere a più gruppi;
- *organizational unit*: è una suddivisione logica dell'organizzazione, permette di specificare indirettamente la designazione di un utente se collocato al suo in-

terno, ha il limite di prevedere un'assegnazione 1 a molti, non adattandosi correttamente ai contesti dove gli utenti possono avere più designazioni contemporaneamente.

LDAP è parte integrante dei sistemi di autorizzazione più diffusi, si può dire che ogni organizzazione ne ha (almeno) uno. I prodotti più diffusi in questo campo sono *Microsoft Active Directory*, *Novell eDirectory* (*Novell Directory Services*), *Oracle Unified Directory*, *OpenLDAP* (opensource) e *OpenDJ* (opensource). Inoltre sono disponibili librerie per qualsiasi linguaggio di programmazione, rendendone semplice l'integrazione in un qualunque sistema software.

Probabilmente a causa di questa semplicità di integrazione questa tecnologia viene spesso utilizzata in modo inappropriato, sfruttandola per implementare anche il processo di autenticazione, a discapito dei meccanismi visti in precedenza. Il *binding* LDAP è infatti l'operazione che permette di accedere al Directory specificando l'identità dell'utente e le credenziali per accedervi. Se questa operazione viene mediata da un altro sistema software (es. un web service o un'applicazione web che hanno accesso al Directory), quest'ultimo può efficacemente verificare l'identità fornita da un utente.

Il problema di questa modalità, rispetto ad una soluzione sicura come Kerberos, è che espone le credenziali dell'utente al sistema software (non necessariamente *trusted*, o con livelli di manutenzione adeguata alla delicatezza del ruolo).

3.2.2 SAML e WS-Federation

Come visto in precedenza SAML è un protocollo basato su XML che permette di definire flussi di autenticazione e autorizzazione, coprendo una serie di casi d'uso specificati nei profili che stabiliscono le regole della comunicazione.

Ad esempio il *Web SSO profile* può essere usato per veicolare attributi relativi all'utente che possono essere recuperati al momento dell'autenticazione (informazioni puntuali ma anche appartenenza a gruppi, quindi a ruoli), permettendo quindi di specificare le autorizzazioni dell'utente agli occhi del fornitore dell'identità.

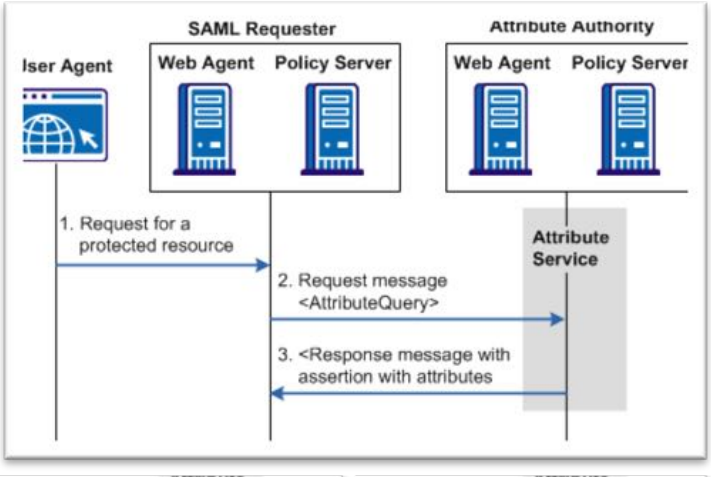

Figura 3-11 - Architettura SAML completa (CA Siteminder 2014)

Un altro profilo interessante è l'*Assertion Query/Request Profile*, esso permette di scambiare tra *Service Provider* e *Attribute Authority* (che può essere il fornitore dell'identità o una o più parti terze) messaggi per il recupero di attributi (*AttributeQuery*) o decisioni autorizzative (*AuthzDecisionQuery*).

Figura 3-12 - Famiglia di protocolli WS- (Devshed Network 2004)*

WS-Federation è un altro insieme di specifiche, simile a SAML, inizialmente sviluppato nel 2002 da Microsoft e IBM con l'obiettivo di definire i profili di interazione dei messaggi tra Web Service per garantirne la sicurezza. Fa parte dell'iniziativa *WS-** basata su SOAP che raccoglie altri standard quali *WS-Security*, *WS-Trust*, *WS-Policy*, *WS-Privacy*, *WS-Authorization*, *WS-SecureConversation*. Si tratta una famiglia di protocolli piuttosto diffusa nell'ambiente Enterprise.

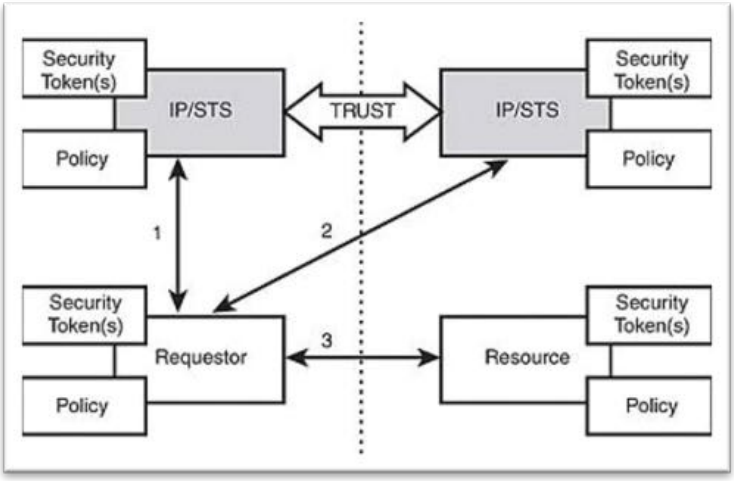

Figura 3-13 - WS-Federation (Devshed Network 2004)

La suite è specificamente orientata alla sicurezza dell'interazione tra *Web Service*, tuttavia esiste un *WS-Federation Passive Requestor Profile* che ha un funzionamento simile al flusso di autenticazione di SAML. Nella prossima sezione si vedrà come questa specifica tecnologia sia impiegata nel contesto dell'Università di Bologna per garantire il funzionamento del *Web SSO* per un gruppo significativo di servizi.

3.2.3 Oauth

Il framework di autorizzazione Oauth 2.0 si propone come ideale complemento di OpenID e si rivolge ad un contesto più flessibile e dinamico di quello Enterprise a cui sono rivolti di più SAML e WS-*.

La sua nascita è dovuta principalmente a due esigenze dei principali fornitori di applicazioni web:

- la necessità di consentire l'interazione tra applicazioni web di diversi fornitori, su autorizzazione di un utente, utilizzando la propria identità;
- la necessità di poter accedere ai dati di un'applicazione web attraverso un dispositivo mobile senza dover reinserire ogni volta le credenziali, attività molto scomoda in mobilità.

Oauth risponde a queste richieste con un modello che prevede quattro attori e una serie di flussi di funzionamento (*flow*) tra cui scegliere per meglio rispondere al caso d'uso specifico.

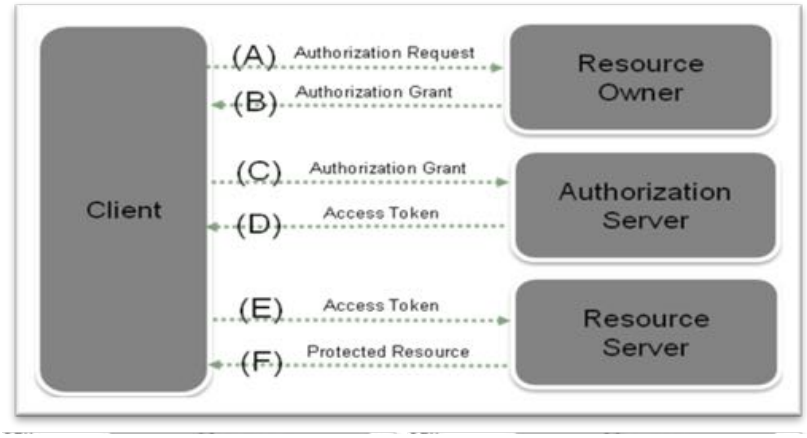

Figura 3-14 - OAuth protocol (Oracle s.d.)

La comunicazione è basata su *REST*, secondo la sequenza definita nella figura precedente:

- il *Client* (che può essere un'applicazione web o nativa) effettua una richiesta di autorizzazione al *Resource Owner* (tipicamente l'utente finale);
- il *Resource Owner* accetta fornendo un *Authorization Grant* (tipicamente delle credenziali);
- il Client fornisce l'*Authorization Grant* all'*Authorization Server* che restituisce un *Access Token* - un lasciapassare con scadenza che permette di accedere liberamente senza ripetere l'autorizzazione per un certo periodo di tempo;
- il *Client* fornisce l'*Access Token* al *Resource Server* (che dovrà validarlo con l'*Authorization Server*) per accedere alla risorsa protetta.

Un *Access Token* ha una naturale scadenza, per mantenere l'accesso alla risorsa il Client deve periodicamente usare un *Refresh Token* per ottenere un valido *Access Token*.
Oauth 2.0 è molto usato anche per l'autorizzazione all'accesso di API che possono essere consumate con l'identità di un utente, in generale è diventato velocemente uno standard di fatto nel mondo delle applicazioni web.

3.3 Gestione dell'identità digitale presso l'Università di Bologna

Il sistema di gestione dell'identità dell'Università di Bologna è costituito da molti componenti con una certa complessità. Questo in parte è implicito nella natura articolata

dell'organizzazione (circa 100.000 studenti e 15.000 utenti integrati nell'amministrazione tra dipendenti — docenti e personale tecnico e amministrativo — docenti a contratto, accreditati di aziende esterne, assegnisti di ricerca e dottorandi), in parte dalle scelte di gestione dell'identità digitale. Infatti circa 400.000 identità di ex-studenti sono regolarmente mantenute con alcuni servizi di base associati, per la volontà di mantenere un minimo di collegamento tra amministrazione e comunità studentesca.

Questa numerosità è intrecciata alla miriade di servizi a cui gli utenti hanno accesso, risorse di rete, *storage*, pubblicazioni elettroniche e risorse logistiche (accesso a edifici e laboratori) - nonché al quotidiano modificarsi dei rapporti degli utenti con l'organizzazione (es. assunzioni, pensionamenti, trasferimenti di struttura, iscrizioni, lauree) che ne modifica i privilegi di accesso.

L'identità digitale degli studenti e dei dipendenti dell'Università di Bologna deriva da una serie di informazioni conservate nelle basi di dati informative che sono dedicate a diversi aspetti dell'organizzazione (in particolare didattica, ricerca, risorse umane).

Ogni fonte di dati ha le sue peculiarità, modello di astrazione e tecnologie su cui sono state sviluppate. Il sistema di identità digitale che prende il nome di DSA (*Directory Service di Ateneo*) ha il compito di raccogliere le informazioni da queste fonti, coordinarne la relazione e tradurre il tutto in un modello di identità e accesso alle risorse che sia aderente alle richieste normative sulla tracciabilità e sicurezza delle attività digitali, permetta di accedere correttamente alle risorse e rifletta nel modo più aggiornato possibile le variazioni di rapporto con gli utenti.

Ogni categoria di utente segue un proprio flusso logico di evoluzione che ne caratterizza lo stato e i privilegi di accesso che possiede, spesso queste categorie possono sovrapporsi o partecipare a percorsi paralleli. Tutti questi casi limite devono essere gestiti dal motore di aggiornamento dell'identità sulla base di decisioni politiche di gestione delle risorse.

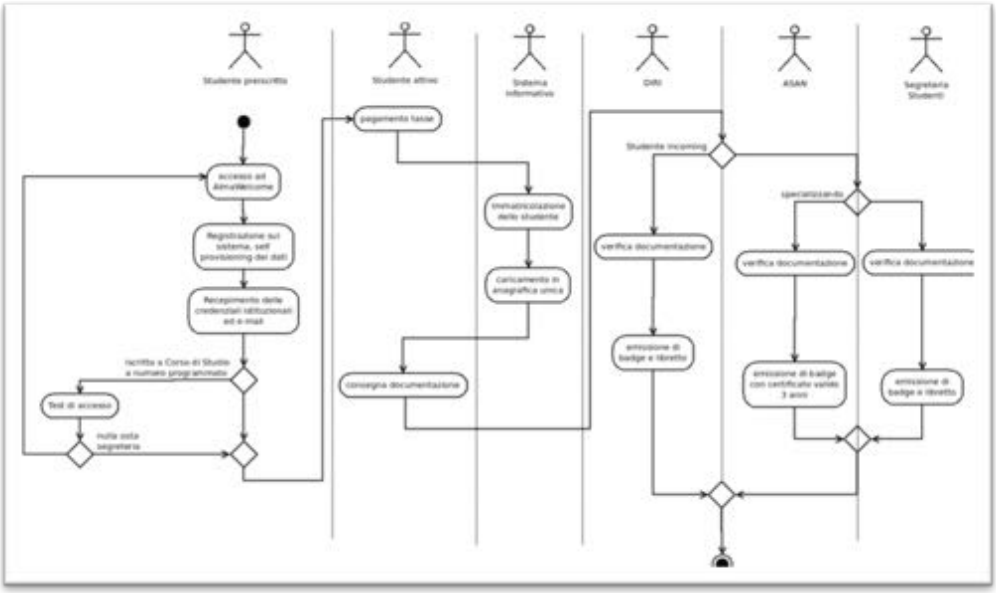

Figura 3-15 - Esempio di flusso di accreditamento di uno studente presso l'Ateneo di Bologna

Le identità digitali sono raccolte in un unico Directory basato su *Microsoft Active Directory*, che permette di gestire gli aspetti di autorizzazione alle risorse sulla base dell'appartenenza ai gruppi a cui partecipa l'utente. I gruppi di cui fa parte l'utente sono sostanzialmente di due tipi:

- *gruppi automatici*: creati automaticamente dal processo di aggiornamento dei dati, sono definiti da regole di aggregazione inserite nella logica di aggiornamento, alcuni esempi possono essere il gruppo dei laureati nell'ultimo anno accademico, oppure il gruppo dei docenti con la qualifica di Professore Ordinario.
- *gruppi manuali*: sono come suggerisce il nome un'aggregazione effettuata manualmente da un operatore, la logica in questo caso è puramente soggettiva, idealmente andrebbero usati solo per gruppi molto dinamici le cui caratteristiche sfuggono al modello dell'identità o per aggregare gruppi automatici per esigenze specifiche.

Più problematico è l'aggiornamento dei gruppi manuali che ovviamente è delegato a interventi ad hoc; per facilitarne la gestione si consiglia di costruire i gruppi manuali utilizzando per quanto possibile i gruppi automatici come membri

L'aggiornamento dei dati avviene tramite procedure *batch* automatiche che si occupano in prima battuta di raccogliere i dati dalle fonti, normalizzarli secondo il modello

dati necessario al software di aggiornamento del Directory, ed una volta verificato che i parametri di funzionamento sono nei livelli di normalità procedere alla sincronizzazione delle modifiche nell'ambiente di produzione.

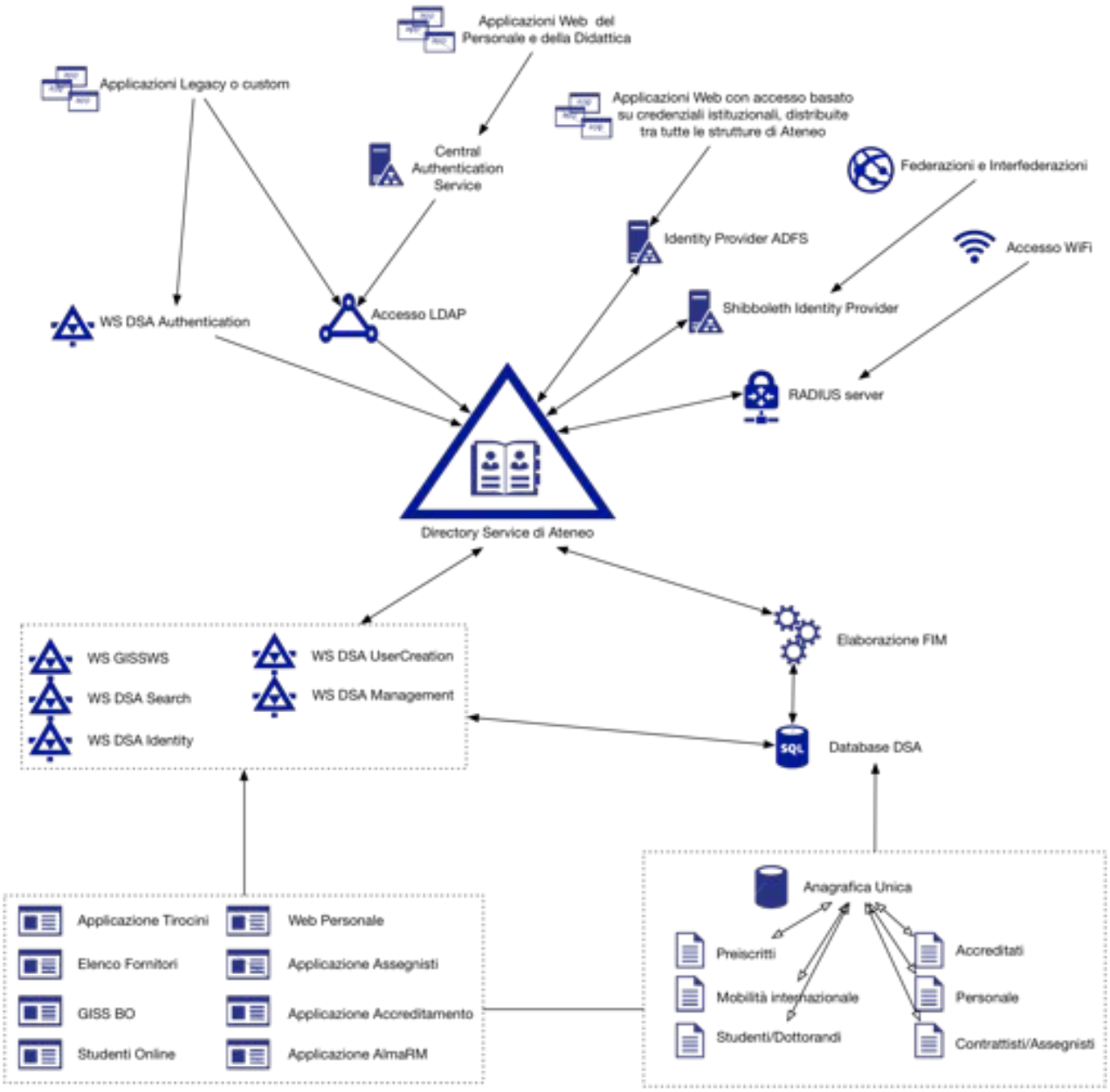

Figura 3-16 - Identity Management all'Università di Bologna

FIM (originariamente *MIIS*, nelle prime implementazioni) è il componente software che si occupa del confronto tra i dati preparati dalle procedure di aggiornamento e normalizzazione delle fonti, con i dati dell'ambiente di produzione. La principale responsabilità di FIM è di individuare nel modo più efficiente possibile quali informazioni richie-

dono un aggiornamento (dati, attributi, gruppi) e preparare la sincronizzazione. Al momento dell'esecuzione della sincronizzazione, le informazioni saranno aggiornate e allo stesso tempo copiati i dati di funzionamento significativi (ad esempio la data di ultimo *logon* di un'utenza) all'indietro del database di riferimento per successive elaborazioni.

Sul Directory di Ateneo che contiene le identità poggiano i servizi fondamentali di autenticazione per tutte le applicazioni e le risorse che usano l'identità istituzionale:

- *Kerberos*: parte integrante di *Active Directory*, è usato per l'accesso alle postazioni di lavoro che sono *joined* al dominio dell'Università;
- *LDAP*: anch'esso parte integrante di Active Directory è usato da una serie di applicazioni *legacy* e/o specialistiche per l'autenticazione degli utenti (anche se come discusso in precedenza è una modalità non desiderabile);
- *RADIUS*: è il sistema di autenticazione usato per il riconoscimento dei dispositivi legittimi per l'accesso alla rete wireless, verifica che le credenziali dell'utente e l'autorizzazione controllando l'appartenenza al gruppo di accesso o a gruppi di *blacklist*;
- *DSA Authentication*: WS di autenticazione e autorizzazione usato per l'integrazione di applicazioni web con le credenziali istituzionali, è stato sviluppato per consentire l'integrazione senza dover prevedere l'uso di *Kerberos* o *LDAP* (quando l'uso di questi non fosse possibile o non fosse accettabile dal punto di vista della sicurezza);
- *DSA Search*: WS di autorizzazione che permette di recuperare attributi circa le identità, sostituisce il precedente per questi aspetti e si rivolge agli applicativi legacy incapaci di lavorare con tecnologie più strutturate e moderne;
- *DSA Management/DSA User Creation/DSA Identity*: WS nati in momenti diversi pensati per la creazione delle utenze da parte delle procedure online di sottoscrizione e registrazione, i primi due sono in dismissione.
- *CAS*: le applicazioni dedicate ai servizi della didattica sono state le prime ad introdurre il web SSO e continuano a usare questa tecnologia per la sua robustezza, nell'attesa di confluire sull'Identity Provider principale;
- *Identity Provider (ADFS*): è il servizio di Web SSO su cui si basa la maggior parte delle applicazioni web dell'Università di Bologna (circa 200), implementa

WS-Federation e SAML, consentendo l'integrazione di applicazioni su piattaforme Linux e Windows in un unico contesto di SSO;

- *Identity Provider (Shibboleth***)**: è il servizio di autenticazione utilizzato per la partecipazione dell'Università di Bologna alle federazioni Federa, IDEM ed *EduGain*, è lo standard de facto di queste federazioni e supporta SAML1.1, è stato introdotto per problemi di compatibilità di ADFS con la modalità di funzionamento delle federazioni.

Oltre al Directory principale DSA esistono altre raccolte di identità che sono direttamente comandate dal flusso principale di aggiornamento. Uno di questi è il componente *Oracle Internet Directory*, componente della suite di servizi Oracle parte dell'infrastruttura principale dei database di Ateneo.

L'infrastruttura principale delle basi di dati di Ateneo poggia su sistemi Oracle, che garantiscono servizi a tutti le applicazioni centralizzate, insieme ai servizi di monitoraggio, ottimizzazione e ridondanza caratteristiche di questa soluzione. Purtroppo per peculiarità tecnologiche (a causa del differente algoritmo di hash utilizzato per il salvataggio delle password) è necessario utilizzare il componente OID congiuntamente ad *Active Directory* per permettere l'accesso al DB utilizzando le stesse credenziali istituzionali di cui ogni dipendente è dotato. Esiste pertanto un meccanismo di sincronizzazione tra il Directory principale e lo *slave* OID che permette l'uso di una stessa identità tra i due mondi.

Un secondo Directory specializzato (questa volta contenente un sottoinsieme delle informazioni di DSA) è demandato alla funzione di verifica degli accessi ai locali e agli edifici dell'Università di Bologna dove è presente il servizio Controllo Ingressi Persone (CIP). Questo Directory, basato su *Active Directory Application Module (ADAM*) raccoglie le informazioni di base circa l'associazione degli utenti con i permessi di accesso ai locali e alle strutture, ed è indipendente da altre fonti informative per il corretto funzionamento del servizio.

L'ultimo directory comandato da DSA è costituito da *Azure Directory*, la soluzione *cloud* per la gestione delle identità, necessario per sfruttare i servizi e le applicazioni Microsoft per l'organizzazione.

Figura 3-17 - Sistema di sincronizzazione tra AD on-premise e Azure-AD in cloud (Golshan 2014)

L'Università di Bologna offre il servizio di posta per gli studenti attraverso Office 365, l'offerta cloud di Microsoft che fornisce casella e-mail e funzioni Office di base per gli utenti iscritti. Grazie ad una convenzione stipulata con il fornitore, il servizio è a titolo gratuito e gli studenti dell'Ateneo possono usare sistemi di messaggistica moderni e continuamente aggiornati che, data la numerosità degli utenti, su infrastruttura *on-premise* non sarebbe possibile offrire a livelli equivalenti.

L'uso di questi servizi con le stesse credenziali istituzionali di cui gli studenti sono in possesso è dipendente dal processo di sincronizzazione tra il DSA e *Azure Directory*. Il flusso di informazioni è gestito da un componente collegato ad *Active Directory* che prende il nome di *Azure Active Directory Connect* (il cui funzionamento è equivalente a MIIS per il funzionamento on-premise).

Il flusso di aggiornamento non prevede il trasferimento dell'*hash* delle password, che rimane conservato all'interno del perimetro dell'organizzazione, in quanto l'autenticazione ai servizi avviene sfruttando la federazione con l'*Identity Provider* dell'Ateneo. L'effetto pratico è che al momento dell'accesso al servizio *cloud* l'utente viene rimandato sulla pagina di autenticazione dell'*ADFS* di Ateneo, dove avviene l'inserimento delle credenziali.

3.4 Federazioni di identità

Come discusso in precedenza le federazioni di identità costituiscono una soluzione tecnologica (e per certi aspetti organizzativa) che permette agli utenti afferenti ad

un'organizzazione di usare la propria identità presso una seconda organizzazione partecipante, in modo trasparente.

A livello nazionale ed internazionale si sono affermate, soprattutto in ambito accademico e di ricerca, federazioni di identità basate su SAML e gestite dai *National Research and Education Network* (*NREN*, in Italia è il GARR).

Beneficio principale delle federazioni di identità è la ragionevole certezza dell'affidabilità dell'identità presentata ai servizi dai fornitori di identità partecipanti. Ogni *Identity Provider* infatti effettua un percorso di accreditamento volto a stabilire la qualità del suo processo di riconoscimento degli utenti e sottoscrive delle regole di comportamento per tutelare la funzione della federazione e gli altri partecipanti.

Gli elementi costituenti di una federazione, in parte già incontrati sono i seguenti:

- *Identity Provider*: fornitore dell'identità digitale, assicura la corrispondenza verso un utente reale ed è responsabile della corretta registrazione e aggiornamento dei dati relativi;
- *Service Provider*: fornitore del servizio, applicazione web che consuma un'identità digitale federata ed eventualmente altri attributi;
- *Discovery Service*: componente software che permette agli utenti di effettuare l'autenticazione su uno degli *Identity Provider* individuandolo in modo agevolato (detto anche *WAYF - Where Are You From*);
- *Metadata Service*: servizio di distribuzione delle informazioni costituenti i partecipanti quali i certificati di firma e crittografia e le descrizioni dei servizi e delle organizzazioni, stabilisce quali sono i partecipanti appartenenti alla relazione di *trust*.

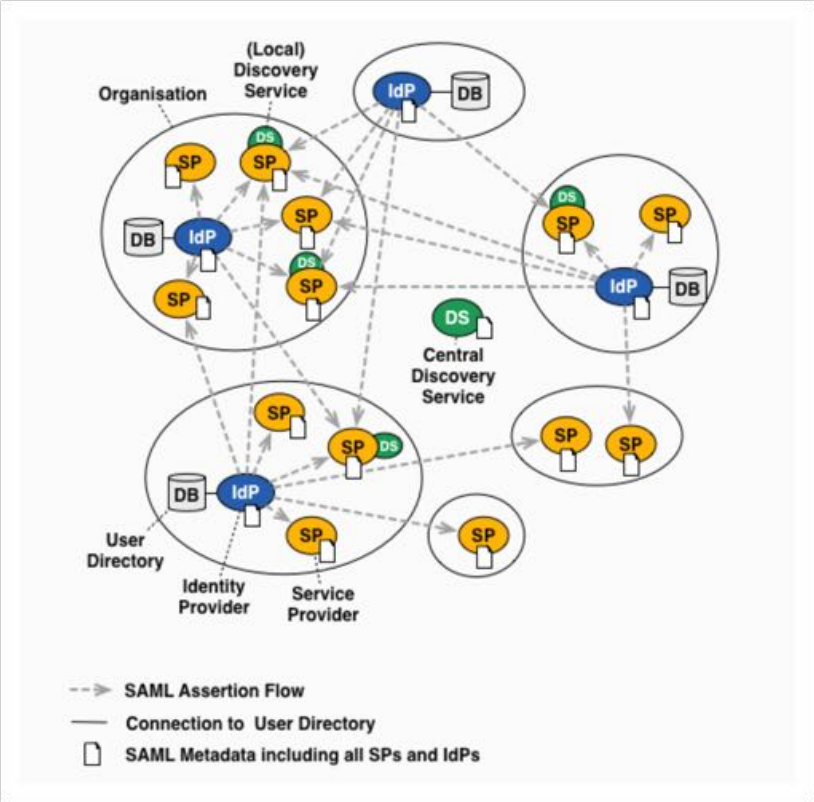

Figura 3-18 - Architettura di una federazione Full Mesh come IDEM (Baerecke 2014)

Generalmente si distingue tra un'architettura *Hub&Spoke e Full Mesh*. In quest'ultima gli *Identity Provider* e i *Service Provider* hanno un funzionamento distribuito ed indipendente da un servizio centrale, fatta salva la necessaria pubblicazione certa e sicura dei metadati per stabilirne i componenti. Secondo questa architettura, la più diffusa, ogni *Service Provider* può operare il proprio *Discovery Service* e comunicare direttamente con gli *Identity Provider* partecipanti, è responsabilità degli IDP configurare ogni servizio appartenente alla federazione sui propri sistemi per rilasciare gli attributi necessari al loro funzionamento.
La federazione IDEM ha un'architettura *Full Mesh*.

Nella seconda tipologia di architettura, l'*Hub&Spoke*, ha caratteristiche opposte. Il *Discovery Service* è unico e gestito centralmente dal responsabile della federazione, questo permette una più semplice configurazione dei servizi, maggiore familiarità per gli utenti ma al tempo stesso rappresenta un *Single Point of Failure*.

Infatti in questa configurazione tutte le asserzioni necessarie alla comunicazione sono mediate dall'*Hub* centrale e nel caso di interruzione del funzionamento tutti i partecipanti alla federazione sono impossibilitati all'uso dei servizi.

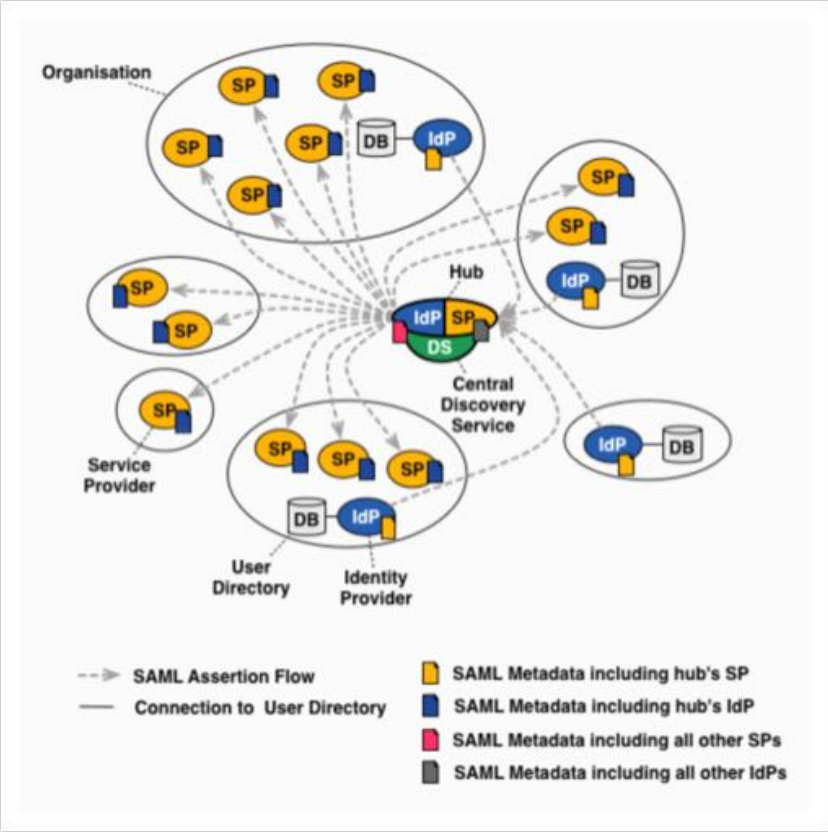

Figura 3-19 - Architettura di una federazione Hub and Spoke come Federa (Baerecke 2014)

Questa architettura è utilizzata ad esempio da Federa, dove un gateway centralizzato si occupa di gestire tutte le asserzioni e di rilasciare ai *Service Provider* il sottoinsieme degli attributi che gli sono strettamente necessari gli *Identity Provider* rilasciano poi ogni volta l'intero set di attributi che è necessario produrre per la partecipazione alla federazione.
Quest'impostazione semplifica notevolmente la partecipazione e la configurazione dei servizi per tutte quelle piccole realtà del territorio dove la gestione dell'intera complessità di un IDP o di un SP sarebbe troppo elevata ed ostacolerebbe l'inserimento in federazione.

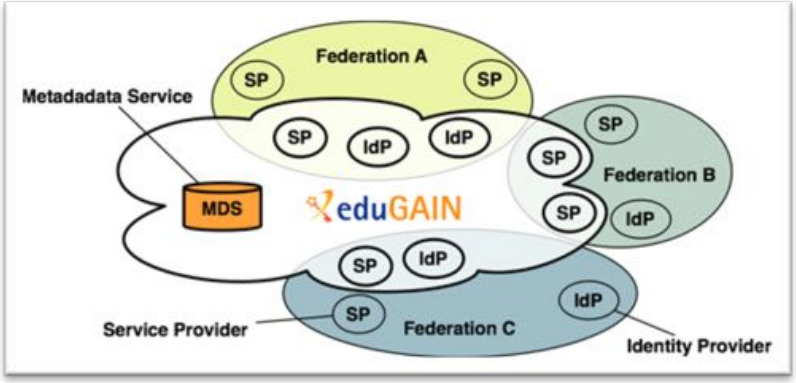

Figura 3-20 - Interfederazione EduGain (Terpstra 2015)

L'interfederazione infine è un concetto che estende il meccanismo di federazione alle stesse federazioni nazionali, permette di mettere in relazione tutti gli SP e gli IDP già aggregati a livello nazionale, permettendo di raggiungere un numero di utenti enorme e dando agli SP una visibilità senza precedenti.

Iniziativa degna di nota, soprattutto per l'impatto potenziale del cambio di paradigma, è il *Sistema Pubblico di Identità Digitale* (SPID), un progetto dell'*Agenzia per l'Italia Digitale* che si propone di fornire a tutti i cittadini italiani un metodo unico per l'autenticazione sui servizi della Pubblica Amministrazione.

I dettagli pubblicati sul funzionamento tecnico mostrano che si tratta di un sistema federato basato su SAML e costituito dai canonici ruoli di IDP, SP e *Attribute Authority*. Il *Circle of Trust* stabilito tra gli IDP è molto forte, con requisiti all'ingresso molto vincolanti sia dal punto di vista organizzativo (procedure documentate e certificazioni ISO), sia dal punto di vista economico (ingente ammontare di capitale sociale per i soggetti privati). Le identità fornite dagli IDP apparterranno a vari livelli di affidabilità, in base alla tecnologia utilizzata per l'autenticazione, in modo da abilitare gli utenti a operazioni più o meno delicate a seconda del livello richiesto dai servizi.

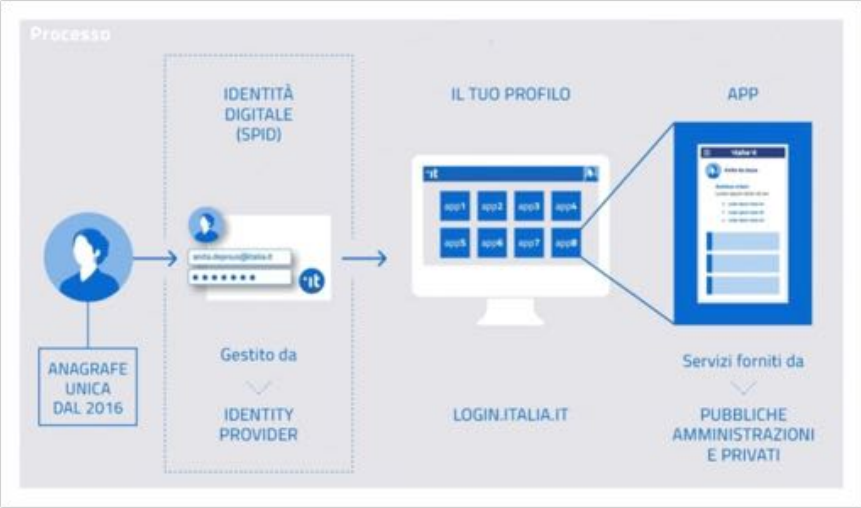

Figura 3-21 - Sistema Pubblico di Identità Digitale (Longo 2015)

Il progetto è al via e non è chiaro del tutto quali saranno le ripercussioni sulle federazioni già esistenti sul territorio italiano e gli obblighi delle PA in tal senso, potenzialmente potrebbe sostituire completamente le credenziali locali generate dalle PA (quindi anche Enti di Ricerca e Università) nel giro di 24 mesi dalla partenza (dicembre 2015).

3.5 *Attribute Authority* e *Virtual Organization*

Gli Attribute Authority sono componenti delle architetture federate che, come suggerisce il nome, hanno la responsabilità di emettere attributi sul proprio dominio di competenza, per arricchire un'identità digitale.

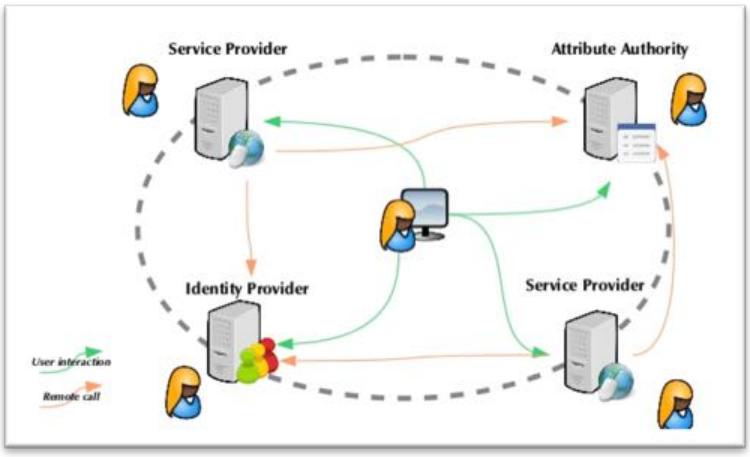

Figura 3-22 - Circle of Trust (Oudot 2013)

Anche un *IDP* può operare come *Attribute Authority*. Non si tratta di un aspetto strettamente determinato ma dipendente dalle implementazioni software. Il modello di funzionamento ideale prevede che *l'Attribute Authority* sia contattato dal *Service Provider* quando questo necessita di arricchire l'identità digitale, gli attributi ulteriori che riceve servono a definire il contesto di autorizzazione e determinare i privilegi di accesso.

L'*Attribute Authority* è un componente fondamentale che abilita un nuovo concetto di collaborazione tra utenti appartenenti alle federazioni. Quando si configura una situazione che richiede l'interazione collaborativa tra diversi soggetti appartenenti a organizzazioni diverse, si instaurano le basi per la definizione di una *Virtual Organization*. Una VO è un'astrazione trasversale delle normali organizzazioni, essa abilita identità appartenenti a contesti organizzativi diversi di partecipare a un'interazione coordinata.

Figura 3-23 - Rappresentazione concettuale di una virtual organization (Interop Vendor Alliance 2010)

La gestione delle *Virtual Organization* è ancora ambito di studi e sviluppo di soluzioni software per l'uso quotidiano, esistono diverse iniziative in tal senso, coordinate soprattutto all'interno di progetti di ricerca europei.
Il software Grouper[6] è un esempio di sistema per la gestione delle VO, permette di definire un gruppo inter-organizzativo e offre una raccolta di meccanismi di integrazione per consentire alle applicazioni di recuperare informazioni sui gruppi ed effettuare le decisioni autorizzative. Quest'ultimo punto è precisamente il freno maggiore all'uso delle VO come tramite collaborativo, è necessario che ogni software su cui il gruppo deve avere un un accesso sia in grado di integrarsi con questa modalità autorizzativa (ancora molto recente).

[6] (Eisbruch 2014)

4 Autenticazione a 2 fattori

L'autenticazione a 2 fattori (*2FA* o *MFA*) è la tecnica di unione di due elementi che permettono di riconoscere un'identità digitale, in modo da garantire una maggior sicurezza di riconoscimento del titolare del le credenziali.

Uno degli esempi più noti è il sistema di autenticazione utilizzato dalle banche per le funzioni di *home banking*. Spesso questa funzione richiede l'autenticazione con credenziali (*Something You Know*), con l'aggiunta di un elemento ulteriore fornito da un oggetto fisico (*Something You Have*), come:

- un *token* di autenticazione;
- un codice recuperato da una *grid card*;
- un telefono (con un SMS o una chiamata vocale);
- uno *smartphone* (con un'applicazione per dispositivi mobili o un secondo indirizzo email a cui inviare un codice).

Figura 4-1 - Esempi di canali di verifica per l'autenticazione a 2 fattori

Questa tecnica se correttamente realizzata permette di migliorare sensibilmente la sicurezza dell'identità digitale di un utente, a fronte di una complicazione trascurabile,

assicurando che le sole credenziali non siano sufficienti ad usare l'identità, garantendo un maggiore controllo al titolare.[7]

È una soluzione offerta sempre più frequentemente dai grandi fornitori di servizi online perché da un lato i problemi di sicurezza delle identità sono sempre di più fonte di seri grattacapi[8], dall'altro lato grazie all'introduzione delle applicazioni per smartphone è diventato più semplice ed economico distribuire un secondo fattore.

Perché un servizio di secondo fattore sia efficace il primo requisito è che sia effettivamente utilizzato dagli utenti. Sebbene qualche servizio (soprattutto bancario) obblighi gli utenti ad usarlo per ogni operazione, l'esperienza può diventare frustrante e non è plausibile pensare di estendere questo modello a servizi meno critici.

Come per ogni soluzione di sicurezza è necessario raggiungere un compromesso tra esigenze tecniche e qualità del servizio percepito da parte degli utenti. Diverse soluzioni, che saranno passate in rassegna nelle prossime sezioni, hanno già fatto progressi significativi in quest'ambito, rendendo molto più sicure le tradizionali credenziali senza intralciare troppo l'esperienza degli utenti.

Non è una panacea per tutti i mali, violazioni informatiche sono ancora possibili con attacchi mirati al portale di autenticazione, soprattutto con alcuni tipi di secondo fattore. In particolare si può distinguere tra due tipologie:

- *in-band authentication*: modalità per cui l'inserimento del secondo fattore avviene tramite la stessa comunicazione con cui si inserisce il primo fattore;
- *out-of-band authentication*: modalità per cui l'inserimento del secondo fattore avviene tramite una comunicazione distinta da quella con la quale si è inserito il primo.

[7] In (Schneier 2005) la soluzione è per certi aspetti criticata, anche se sono riconosciuti alcuni effetti migliorativi della tecnologia, nei prossimi capitoli saranno discussi i *trade-off* delle scelte implementative.

[8] Alcuni dati sono riportati in (Verizon 2015), tra cui un costo stimato di 400 milioni di dollari e decine di milioni di identità rubate.

Se il sistema di autenticazione usa l'*in-band authentication* è teoricamente possibile, per un malintenzionato molto motivato, stabilire un *Man-In-Middle-Attack* tra l'utente che si sta autenticando e il servizio online con un *malware* oppure attraverso un portale di *phishing*.
In queste condizioni il malintenzionato che controlla il *malware* o il portale di *phishing* è in grado di intercettare sia le credenziali sia il codice aggiuntivo inserito dall'utente, potendo disporre così dei privilegi liberamente. L'unica consolazione rimane nel fatto che l'attività illecita deve essere effettuata in concomitanza di un accesso lecito da parte dell'utente, rendendo più difficile lo sfruttamento delle credenziali e l'ampiezza di impatto della compromissione.

Figura 4-2 - Software Token di Duo Security

Se il sistema di autenticazione usa invece l'*out-of-band authentication,* diventa molto più difficile per un malintenzionato riuscire ad appropriarsi dell'identità digitale in quanto il secondo fattore è immesso attraverso un canale completamente separato dal primo.

Un esempio può essere un codice OTP che l'utente deve spedire al servizio via SMS[9], oppure una conferma attraverso una chiamata vocale o applicazione mobile.
In queste condizioni un malintenzionato dovrebbe riuscire a violare due sistemi separati dell'utente unendoli in un attacco sincronizzato.

Tabella 1 - Canali di verifica del secondo fattore e proprietà

	in-band authentication	out-of-band authentication
Email secondaria	Sì, se fornisce OTP	Sì, se viene chiesto all'utente di spedire un messaggio di conferma
SMS	Sì, se fornisce OTP	Sì, se viene chiesto all'utente di spedire un messaggio di conferma
token fisico	Sì	no
grid card	Sì	no
Chiamata vocale	Sì, se fornisce OTP	Sì, se viene chiesta una conferma all'utente
Mobile app	Sì, se fornisce OTP	Sì, se viene chiesta conferma all'utente

Purtroppo non sempre gli accorgimenti relativi alla scelta del secondo fattore o del relativo metodo di autenticazione sono sufficienti a garantirne la qualità. Esempi eccellenti, quali la vulnerabilità del sistema a due fattori di Paypal o Google[10], ricordano che la sicurezza è forte solamente quanto l'elemento più debole della catena di accorgimenti che la costituiscono.

[9] Per separare correttamente i canali il servizio deve richiedere all'utente di spedire via SMS la risposta richiesta, in modo che la verifica avvenga effettivamente su comunicazioni separate. Una discussione di alcune modalità di utilizzo del cellulare come token sono presentate in (Van Thanh, et al. 2009).
[10] Vulnerabilità al sistema di autenticazione con secondo fattore sono state subite con diverse modalità da nomi importanti quali Paypal (Lanier 2014), Google (Krebs 2012) e RSA (Schwartz 2011).

Tabella 2 - Vulnerabilità dei canali di verifica del secondo fattore

	Malware	Phishing
Email secondaria (IBA)	Può essere usato per intercettare le credenziali dell'account secondario	L'OTP può essere intercettato per attivare una sessione indirettamente autorizzata dall'utente
SMS (IBA)	Può essere usato per intercettare gli SMS di secondo fattore e inoltrarli al malintenzionato	Può essere usato per intercettare OTP o per disseminare malware
Chiamata vocale (IBA)	Può essere usato per intercettare le chiamate e inoltrarle	Può essere usato per intercettare OTP o per disseminare malware
Mobile app (IBA)	Può essere usato per intercettare gli OTP e inoltrarli	Può essere usato per intercettare OTP o per disseminare malware
token fisico	No, a meno che non siano perse le informazioni sulle chiavi	Può essere usato per disseminare malware
grid card	No, a meno che non sia conservata digitalmente	Può essere usato per disseminare malware
Email secondaria (OOBA)	Può essere usato per intercettare le credenziali dell'account secondario	Può essere usato per disseminare malware
SMS (OOBA)	Può essere usato per intercettare e rispondere alle richieste di secondo fattore	Può essere usato per disseminare malware
Chiamata vocale (OOBA)	Può essere usato per intercettare e rispondere alle richieste di secondo fattore	Può essere usato per disseminare malware
Mobile app (OOBA)	Può essere usato per violare la chiave segreta	Può essere usato per disseminare malware

L'obiettivo di questo elaborato è illustrare la progettazione e implementazione effettuata per realizzare un servizio di autenticazione a 2 fattori per i sistemi dell'Università di Bologna integrati con il servizio di *Single Sign-On*. Si tratta di oltre 200 applicazioni utilizzabili dagli utenti attraverso le credenziali istituzionali, alcuni tra questi software fa uso di *Smart card* per la firma digitale, nessuno utilizza un secondo fattore di autenticazione.

Di seguito saranno esaminate le soluzioni di autenticazione a 2 fattori dei principali fornitori di servizi online, nonché i prodotti disponibili sul mercato per l'integrazione di

un secondo fattore con soluzioni di autenticazione già esistenti, infine si definirà il contesto in cui si muove l'Ateneo e le considerazioni adottate per la soluzione.

4.1 Implementazioni esistenti

Negli ultimi anni l'autenticazione con secondo fattore è stata oggetto di notevoli innovazioni, diverse soluzioni si sono affacciate al mercato, distinguendosi per la capacità di integrarsi con una miriade di servizi Enterprise (applicazioni web, desktop, client VPN, RADIUS, ecc.) e per il miglioramento della semplicità d'uso rispetto alle vecchie modalità legate a *token* fisico (generatore di codici o *Smart card*).

L'iniziativa più interessante e aperta è probabilmente rappresentata dall'*Initiative for Open Authentication* (*OATH*), una collaborazione tra diversi operatori[11] del mercato di soluzioni per l'autenticazione, che si propone di definire un'architettura di riferimento per lo sviluppo di soluzioni di *Strong Authentication* sicure ed interoperabili.

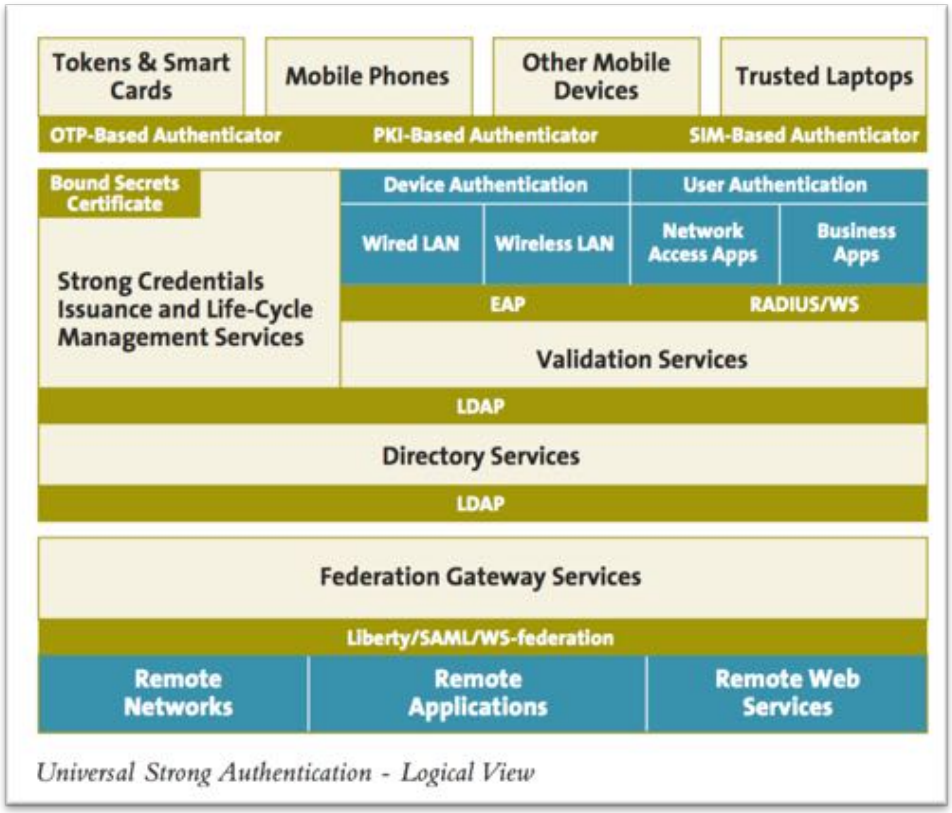

Figura 4-3 - Strong Authentication secondo la proposta OATH

[11] https://openauthentication.org/members/

OATH ha sviluppato diversi componenti fondamentali utili a rendere più diffusa l'autenticazione con secondo fattore, due tra i contributi più significativi sono gli algoritmi *HMAC-based One-Time Password* (*HOTP*)[12] e il *Time-based One-Time Password* (*TOTP*)[13].

Il funzionamento di HOTP è relativamente semplice, si tratta di una funzione matematica che, sulla base di due valori *K (key)* e *C (counter)* restituisce un valore di *hash* che può essere usato per confronto in modo da determinare se chi fornisce il valore di *hash* è a conoscenza della chiave, trattandosi quindi del titolare[14].

L'utilizzo più comune basato su HOTP consiste nel distribuire un *token* fisico che contiene una chiave *K* caricandola, allo stesso tempo, su un server di autenticazione sincronizzando il contatore *C*. Al momento della verifica della transazione l'utente causa la generazione del codice da parte del token (ad esempio premendo un pulsante) e lo inserisce al momento della richiesta da parte del sistema. Quest'ultimo calcola *l'hash* a partire dalla chiave che ha associato all'utente richiedente e al contatore che gli risulta, confrontando il risultato per verificare l'identità dell'utente.
In condizioni simili il contatore potrebbe perde la sincronizzazione tra *token* e server (se ad esempio l'utente preme diverse volte il pulsante senza usare l'OTP generato); per ovviare al problema la raccomandazione consiste nel consentire al server di poter calcolare un certo numero di OTP futuri e chiedere all'utente di inserirne 2-3 in sequenza. Creare una sequenza di OTP corretta è estremamente difficile dal punto di vista crittografico, pertanto il confronto corretto stabilisce l'identità dell'utente e recupera la sincronizzazione.

Il secondo algoritmo risolve questo tipo di problema usando l'istante temporale in cui si opera il calcolo con la funzione di *hash* come elemento di sincronizzazione.
In questo modo il risultato dell'algoritmo è leggermente più predittivo ma gli OTP generati hanno una validità limitata, assicurando comunque la sicurezza della soluzione.

[12] (M'Raihi, et al. 2005)
[13] (M'Raihi, Machani, et al. 2011)
[14] Esiste anche una modalità basata su *challenge-response*, definita dallo standard OCRA (M'Raihi, Rydell, et al. 2011)

Prima di poter calcolare correttamente l'OTP il client deve effettuare un'iniziale sincronizzazione con il server durante la quale stabilisce:

- l'istante temporale di riferimento inziale *T0* (*Unix Epoch*[15] di default);
- l'intervallo di tempo da usare come intervallo di validità *TI* dell'OTP e per calcolare *C* a partire da *T0;*
- il tipo di *hash* da usare (default SHA-1);
- la lunghezza del codice OTP.

Da questo momento a meno di problemi hardware il contatore sarà sempre sincronizzato; anche in questo caso per ovviare a piccoli sfasamenti la raccomandazione consiste nel lasciare accettare al server 1-2 *hash* precedenti o successivi al preciso istante temporale (anche per non frustrare l'utente in caso di ritardi di trasmissione sul limite dell'intervallo), senza per questo compromettere la sicurezza della soluzione.

Di seguito è riportata una veloce rassegna delle soluzioni di secondo fattore sviluppate dai principali fornitori di servizi online e alcuni prodotti disponibili per l'integrazione del secondo fattore di autenticazione nel sistema di autenticazione aziendale, sia open-source sia proprietari.

4.2 Google

La soluzione proposta da Google si basa sull'algoritmo TOTP e richiede l'utilizzo di un'applicazione per dispositivi mobili come sistema di calcolo del codice OTP.

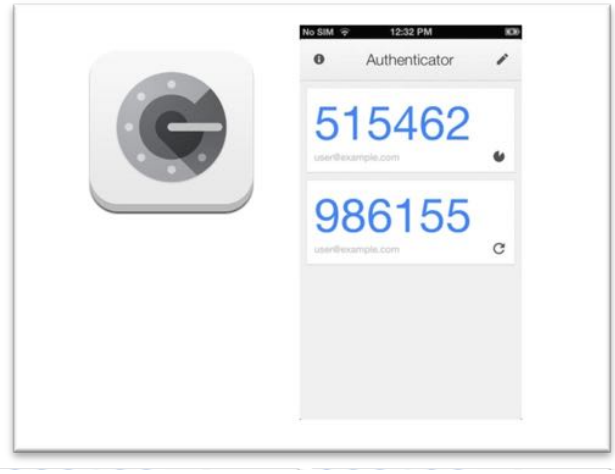

Figura 4-4 - Google Authenticator

[15] In ambiente Unix la data convenzionale 1/1/1970

È possibile usare la stessa applicazione per diversi servizi compatibili con TOTP, tuttavia i parametri che definiscono il riferimento temporale per il contatore devono essere quelli di default dell'algoritmo.

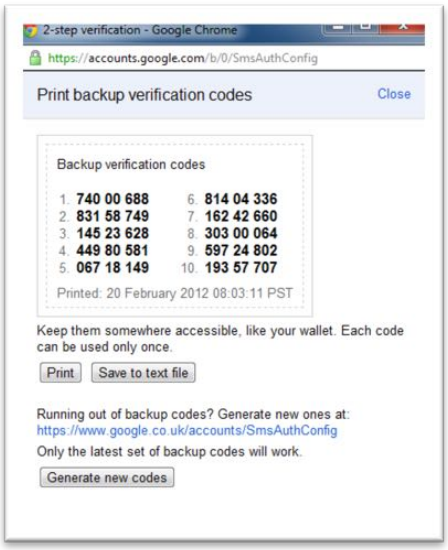

Figura 4-5 - Codici di backup da usare col servizio Google nel caso di perdita del token

Una volta attivata quella che Google chiama la *2-step verification*, è anche possibile scaricare una lista di codici di emergenza, idealmente da stampare e conservare con attenzione, per effettuare il bypass una tantum dell'autenticazione (i codici sono utilizzabili una sola volta). Inoltre l'attivazione dell'autenticazione a due fattori ha effetto immediato su tutte le applicazioni mobile che richiedono un account Google, richiedendo di attivare correttamente gli account configurati, con il secondo fattore.

Figura 4-6 - Processo di enrolment di Google 2-step verification

Il servizio fornisce una funzione molto comoda per l'utente, che consente di rendere fidato il browser che si sta utilizzando, in modo da non dover inserire nuovamente il secondo fattore ogni volta che si effettua l'autenticazione.

Tra i servizi accessibili con l'account Google ci sono casi (ad esempio la posta elettronica Gmail) dove l'accesso con le credenziali non avviene tramite un'interazione con l'utente, è il client del particolare protocollo a gestire l'autenticazione con credenziali salvate. Anche per questi protocolli vige il doppio controllo sulle credenziali, per cui una semplice configurazione utente-password smette di funzionare quando si attiva il secondo fattore.

La soluzione proposta da Google per gestire l'accesso da parte di applicazioni di questo genere è chiamata *app password*, una vera e propria password alternativa per accedere ai servizi con lo stesso username, generata ad hoc e distinta per ogni applicazione o dispositivo in possesso dell'utente.

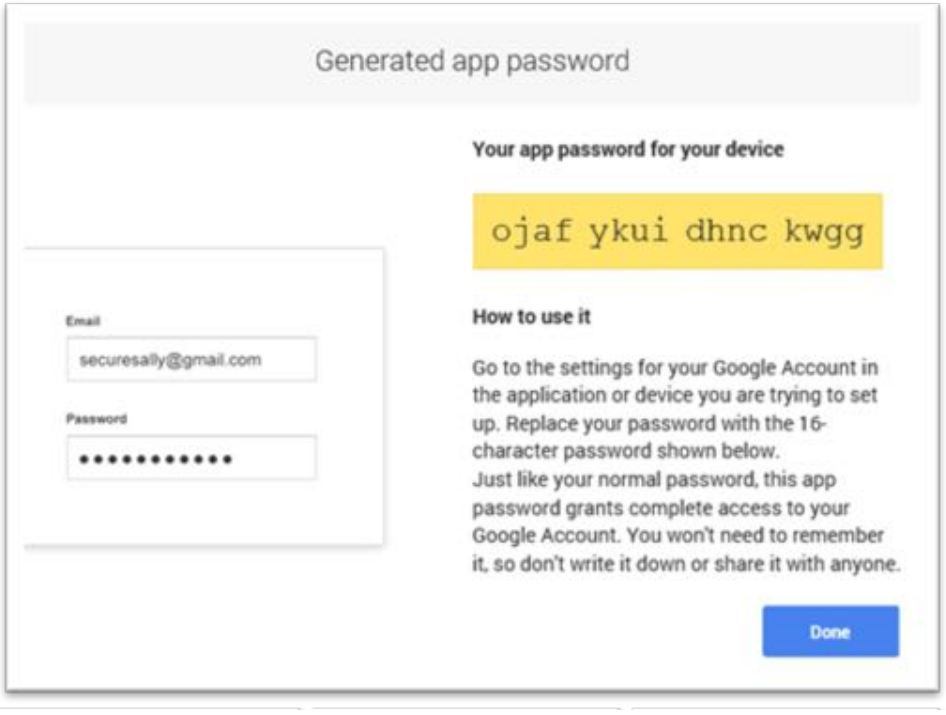

Figura 4-7 - Esempio di app password per i servizi Google

Dal pannello di gestione dell'account è possibile gestire le *app password*, verificare per quali dispositivi e applicazioni sono state generate, cancellarle o crearne di nuove.

I codici così generati non sono recuperabili, sono visibili solo al momento della creazione per migliorarne la sicurezza.

In generale la gestione dell'identità digitale di Google è molto robusta, sono presenti diverse opzioni per aumentarne l'efficacia e tenere sotto controllo dispositivi registrati, ultimi collegamenti e attività dell'account.

4.3 Facebook

Facebook denomina la sua soluzione per secondo fattore *login approvals*, permette di usare il numero di cellulare per farsi spedire codici OTP oppure un'applicazione mobile come *Google authenticator* o la stessa app di Facebook.

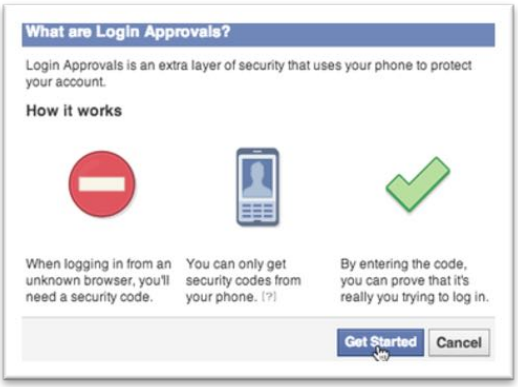

Figura 4-8 - Attivazione dei Facebook Login Approvals

Per attivarla è necessario prima di tutto inserire un numero di cellulare valido e verificarlo con l'inserimento del codice inviato, successivamente è possibile attivare la generazione dei codici via app, effettuando una procedura di sincronizzazione che permette di caricare la chiave segreta tramite un *QR Code*.

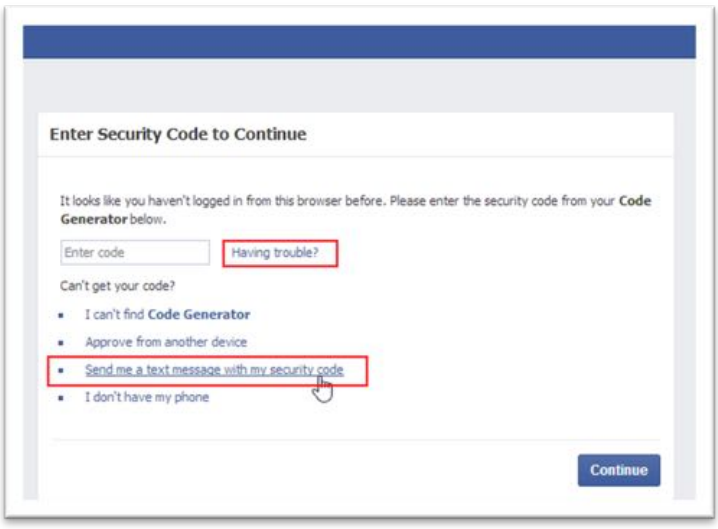

Figura 4-9- Enrolment dei Facebook Login Approvals

Facebook ha scelto di richiedere il codice una sola volta per ogni nuovo dispositivo o browser usato dall'utente, nella sezione di gestione della sicurezza del profilo è possibile vedere in ogni momento l'elenco dei dispositivi collegati e verificati con la data di ultimo utilizzo, agendo sulla cancellazione in caso di bisogno. Nel caso non sia disponibile il secondo fattore il sistema permette di usare un codice di emergenza tramite SMS o effettuare altri tipi di *bypass* temporaneo.

4.4 Dropbox

Il sistema di autenticazione a due fattori integrato in Dropbox è estremamente semplice, l'utente ha facoltà di attivarlo in autonomia dalla sezione per la gestione del proprio account.

Al momento dell'attivazione è possibile scegliere se ricevere il codice OTP via SMS su cellulare oppure utilizzare un'applicazione per dispositivi mobili. Nel secondo caso è necessario effettuare la sincronizzazione iniziale.

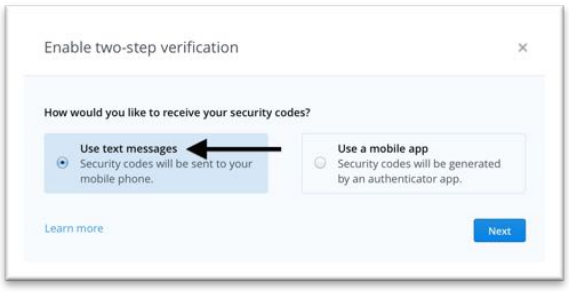

Figura 4-10 - Enrolment della Dropbox Two-Step Verification

Quando l'autenticazione in due passi è abilitata, l'operazione di accesso richiede l'inserimento di un codice OTP che viene inviato all'utente. Anche in questo caso è possibile riconoscere il computer (in realtà il browser utilizzato) e avvalersi di un codice di emergenza utilizzabile per il bypass del sistema.

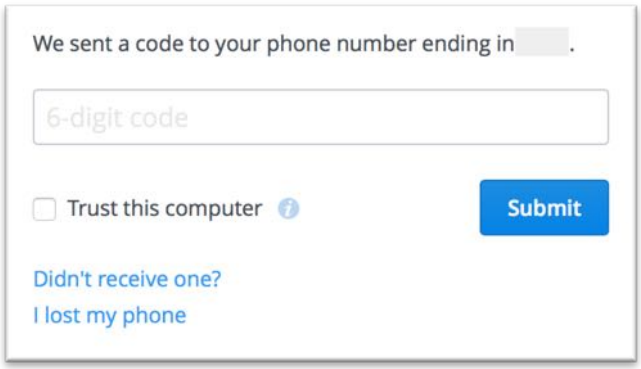

Figura 4-11 - Richiesta del codice OTP del servizio Dropbox

4.5 Apple

Il secondo fattore integrato nei servizi cloud Apple è pensato per l'uso esclusivo dei codici OTP spediti ad un telefono in possesso dell'utente. Nella sezione dedicata alla gestione dell'identità digitale è disponibile l'attivazione. Una volta attivato il sistema di verifica richiederà un codice per ogni uso dell'account per ogni nuovo dispositivo.

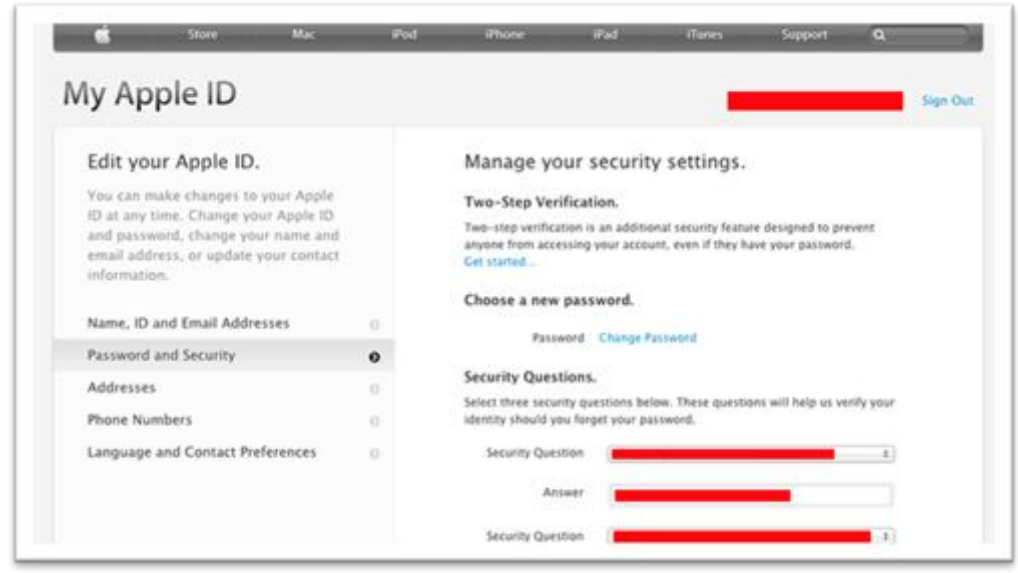

Figura 4-12 - Attivazione di Apple Two-Step Verification

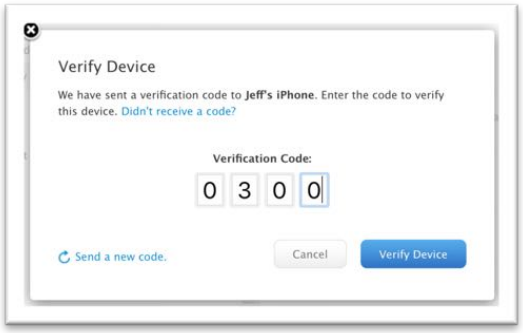

Figura 4-13 - Verifica del secondo fattore del servizio Apple

Lo stesso portale di gestione dell'identità permette di impostare anche le opzioni per il ripristino dell'account, quali le domande e risposte segrete. È necessario tenere presente che funzionalità di questo tipo, che rendono l'utente autonomo nel recuperare l'accesso in caso di dimenticanza, costituiscono un completo *bypass* di tutti i meccanismi di sicurezza.

Se non realizzate con attenzione potrebbero crearsi condizioni per cui ad un malintenzionato potrebbe convenire concentrarsi sull'aggirare il sistema di ripristino piuttosto che violare le credenziali[16].

4.6 LinOTP

LinOTP è una soluzione opensource scritta in Python di livello Enterprise per la generazione di *One-Time Password*. È uno strumento completo che fornisce una serie di caratteristiche molto interessanti:

- gestisce le chiavi segrete per gli utenti;
- ha la possibilità di usare degli *Hardware Security Module*;
- fornisce dei moduli di autenticazione per diverse tecnologie complementari (PAM, RADIUS, LDAP Proxy, ...)
- comprende un portale self-service per gli utenti, in modo da gestire in autonomia il secondo fattore;
- supporta i database più diffusi;
- supporta una grande varietà di token fisici (non solo *OATH compliant*);
- fornisce *audit* di sicurezza automatico.

[16] Quest'ultima pratica insieme ad altre è stata all'origine della violazione degli account di alcune celebrità (Apple 2014)

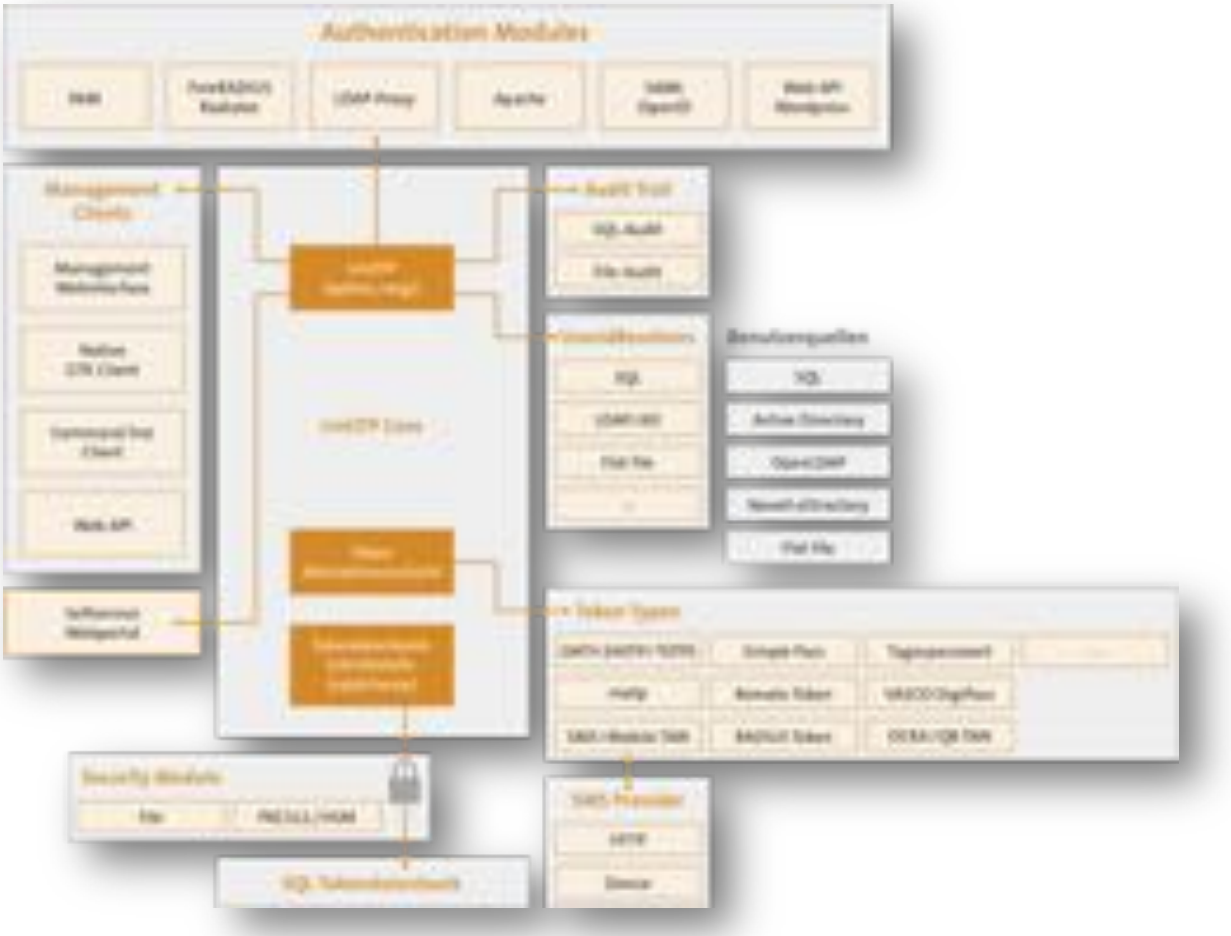

Figura 4-14 - Schema logico dei componenti di LinOTP (LSE Leading Security Experts GmbH s.d.)

LinOTP è una soluzione *on-premise*, richiede pertanto un certo livello di dimestichezza con le tecnologie con cui si integra e un presidio costante per garantirne servizio e manutenzione. Permette inoltre di mantenere gli aspetti di identità digitale legati all'autenticazione all'interno del perimetro organizzativo rispetto a una soluzione *in-cloud*.

4.7 OpenOTP

Figura 4-15 - RCDevs OpenOTP

È una soluzione molto versatile, il nome infatti indica la flessibilità di integrazione con standard e tecnologie diverse, tuttavia si tratta di una soluzione commerciale.
Si tratta di una soluzione *on-premise*, molto duttile sia sul versante hardware supportato (gestisce anche token Yubico[17] e U2F), sia sui back-end software (directory, software token, sistemi di autenticazione).

Per esempio è in grado di garantire il funzionamento del secondo fattore anche per il login locale sulle workstation, via accesso remoto (SSH, Remote Desktop), alle VPN di molti *vendor*, sistemi di SSO federato. Ha diversi moduli per la gestione avanzata e un portale di self-enrolment utilizzabile dagli utenti che sfrutta i *QR code* per semplificare l'adesione al servizio.

Fornisce un modulo già pronto per l'integrazione del sistema con il sistema di autenticazione ADFS, anche se limitato nel funzionamento (richiede un secondo fattore ad ogni login, non supporta la "certificazione" del browser).

[17] Yubico produce dei token abbastanza diffusi nel settore di mercato alto, ha molteplici usi e supporta un numero notevole di servizi, oltre a poter essere usato anche come dispositivo OATH e U2F (Yubico Team 2015)

Offre una modalità di login alternativa, di propria invenzione, chiamata *TiQR* che permette di sostituire completamente l'uso delle credenziali o di token fisici con un'applicazione per dispositivi mobili dotati di telecamera.
L'utente con l'app TiQR effettua un primo enrolment per associare la propria identità all'applicazione, operazione effettuata con la scansione di un QRCode ad hoc e l'impostazione di un PIN di 4 cifre.
In seguito l'utente per accedere a un servizio dove questa funzione è attiva dovrà selezionare questa modalità di accesso al posto delle credenziali, effettuare la scansione del QRCode presentato dal servizio con l'app TiQR, digitare il PIN sull'app ed entrare. Dietro le quinte l'applicazione scambia delle chiavi crittografiche e garantisce quindi un livello di sicurezza paragonabile ad una smart card, con un'esperienza utente leggermente più semplice simile a quanto descritto in (Kyeongwon, Changbin e Woongryul Jeon 2011).

4.8 OneTime

OneTime di Zetetic è un altro prodotto commerciale che fornisce un servizio cloud per l'implementazione di un secondo fattore. Supporta una vasta gamma di token hardware e software e l'invio via SMS per l'uso con cellulari non smartphone.

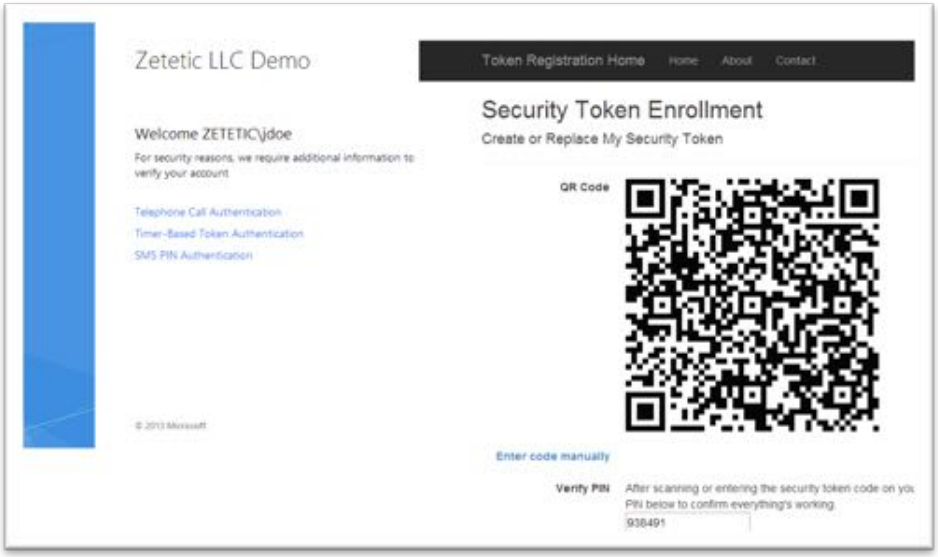

Figura 4-16 - Enrolment su piattaforma OneTime

Fornisce un portale integrato per l'enrolment dell'utente e la gestione del secondo fattore, inoltre se si sceglie di usare una chiamata telefonica per la verifica, l'interazione

avviene in modalità *Out-Of-Band*, notificando direttamente al servizio di autenticazione il controllo corretto, senza richiedere all'utente di inserire altri dati che potrebbero teoricamente essere intercettati da un portale di phishing.
Offre un modulo di autenticazione appositamente pensato per ADFS, purtroppo non supporta il riconoscimento di dispositivi o browser fidati. Non supporta molti altri tipi di integrazione, la semplicità del prodotto lo rende interessante per scenari che non richiedano personalizzazioni.

4.9 Microsoft MFA

Anche Microsoft offre un servizio di autenticazione a 2 fattori, il sistema poggia sulla piattaforma *Azure AD* e per poterlo utilizzare è necessario attivare la sincronizzazione della propria anagrafica utenti (LDAP) con il servizio in cloud.
Il sistema permette la verifica attraverso chiamata vocale, email secondaria, SMS oppure tramite l'uso di un'applicazione per smartphone basata su TOTP.

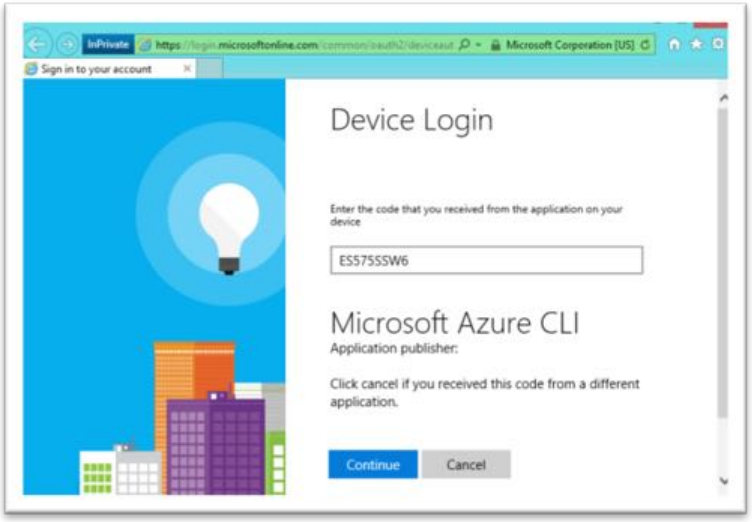

Figura 4-17 - Verifica del secondo fattore con Microsoft Multi-Factor Authentication

E' in grado di gestire dispositivi fidati e permette all'utente di definire e gestire la lista[18].
Nel caso si attivi il secondo fattore in una soluzione completamente in cloud (ad esempio un account personale di Office365), come succede per Google tutti i servizi di accesso all'identità richiederanno da quel momento il secondo fattore, per i casi dove

[18] Nella soluzione *on-premise*, che richiede un componente da installare nell'architettura (MFA server), non è prevista l'opzione per gestire i dispositivi fidati.

è necessaria un'interazione automatica come i client di posta elettronica è possibile attivare le *app password*.
Questa funzione non è disponibile se le applicazioni o il sistema di autenticazione sono on-premise.

4.10 Duo Security

Una piattaforma avanzata per servizi di secondo fattore è rappresentata da Duo Security. L'azienda omonima è molto attiva nell'ambito MFA e offre una soluzione molto interessante per possibilità di integrazione e personalizzazione. E' attivamente coinvolta nella ricerca sulla sicurezza delle soluzioni di questo genere, celebre è stato l'annuncio da parte del loro reparto di ricerca della vulnerabilità dell'implementazione del secondo fattore di Paypal[19].

Figura 4-18- DUO app con Out-of-band authentication

[19] (Lanier 2014)

Offre moduli già pronti per l'integrazione di molte soluzioni applicative quali applicazioni web *cloud-based*, *on-premise*, VPN, accesso a server via SSH/RDP. Ha un modulo anche per i sistemi di autenticazione CAS e ADFS, basato sull'inclusione di pagine della piattaforma in *iframe* presentati al momento dell'autenticazione dell'utente.
Le policy di funzionamento per ognuna delle applicazioni sono molto personalizzabili e consentono spazio di manovra per rendere il comportamento sicuro ma non eccessivamente intrusivo per gli utenti.

Le modalità principali supportate sono token fisici Yubico e *OATH*, chiamata vocale o SMS e un'applicazione per dispositivi mobili (anche per smart watch) che consente di effettuare la verifica in modalità *Out-of-band*. Le ultime versioni della piattaforma offrono funzionalità avanzate che permettono di usare l'applicazione sui dispositivi mobili per verificare la sicurezza intrinseca di questi ultimi. La funzione di *Endpoint Analysis* si spinge infatti ad analizzare le versioni dei sistemi operativi su cui si esegue l'applicazione.
Essa segnala all'amministratore gli utenti che potrebbero essere veicolo di compromissione a causa di obsolescenza dei sistemi, o presenza di altri software critici.

Duo Security offre anche una versione del sistema per sviluppatori, tramite il quale con l'accesso alle API disponibili è possibile personalizzare fortemente l'esperienza di autenticazione. Non è possibile tuttavia personalizzare i canali di verifica del secondo fattore (chiamata vocale, SMS, applicazione).

4.11 Authy

È una soluzione dedicata a sviluppatori e integratori. La piattaforma si occupa della gestione di una serie di API che permettono di gestire l'enrolment, il ciclo di vita dell'utente e la validazione del secondo fattore.
I canali di verifica supportati sono i token OATH, chiamate vocali e SMS (via servizio Twilio[20]), software token TOTP.

[20] Servizio online per messaggistica e voice over IP (Twilio 2015).

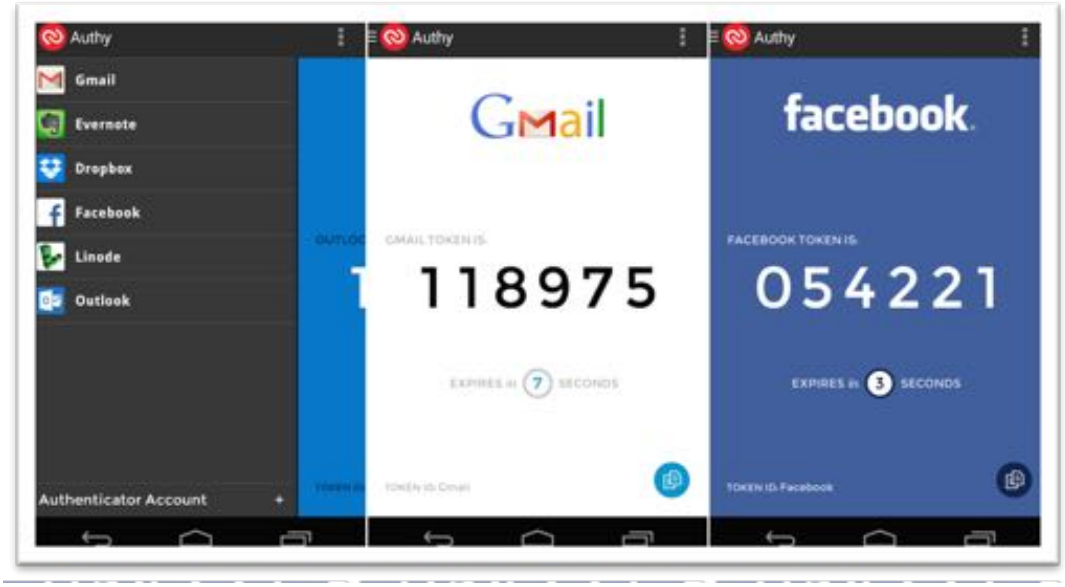

Figura 4-19- Applicazione per dispositivi mobili di Authy

Il servizio è costituito da una serie di strumenti per la gestione degli utenti e si posiziona come soluzione di verifica di secondo fattore pura, non necessariamente collegata al momento dell'autenticazione, anche per verifiche di operazioni dispositive (es. un bonifico da un'applicazione di home banking).

I canali di verifica non sono personalizzabili, tuttavia l'applicazione mobile omonima presenta caratteristiche interessanti quali:

- verifica *Out-Of-Band*;
- possibilità di integrare altri servizi TOTP (es. Google, Facebook, Dropbox) per gestire il secondo fattore con stessa applicazione;
- il blocco dell'accesso all'applicazione e lo sblocco con impronta digitale;
- la sincronizzazione del software token tra più dispositivi fisici (es. più smartphone, un tablet), anche se questa possibilità incide sul livello di sicurezza della soluzione.

Esistono moduli già esistenti per l'integrazione dei casi più comuni (non per ADFS) ed il costo della soluzione è licenziata a numero di autenticazioni.

4.12 Time4ID

Il sistema Time4ID è l'offerta di *Strong Authentication* della piattaforma Time4Mind. Si tratta di una soluzione sviluppata in Italia da Intesi Group, partner tecnologico per molti fornitori di servizio in ambito informatico e bancario a livello nazionale, specializzato

in soluzioni legate a sicurezza e crittografia (firma digitale, firma elettronica avanzata, crittografia dei dati).
I canali di verifica che mette a disposizione sono l'email secondaria, la grid card, l'invio SMS e un'applicazione (ValID) per dispositivi mobili che supporta anche il controllo *Out-of-Band*.

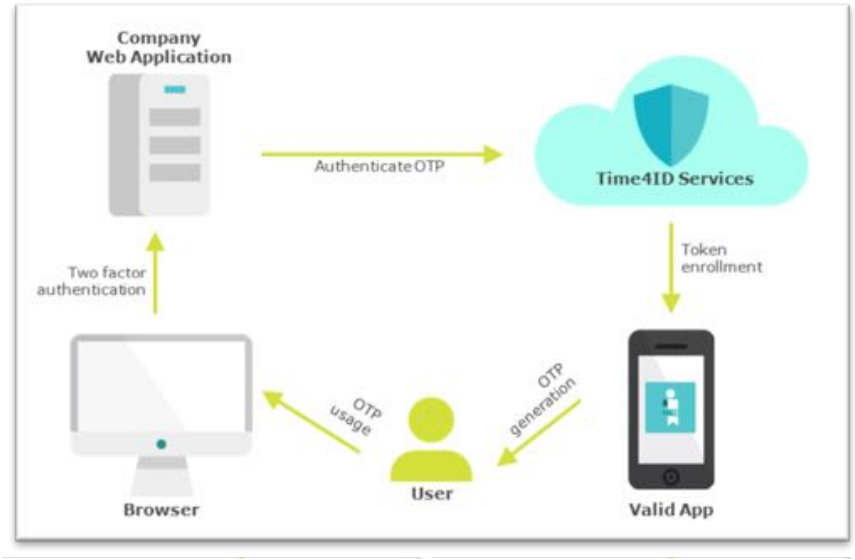

Figura 4-20 - Schema logico di funzionamento della piattaforma Time4ID

La piattaforma può essere sfruttata in due modalità distinte, nella prima l'utente usa Time4Mind per effettuare l'enrolment e la gestione del ciclo di vista del secondo fattore (così come l'amministratore), nella seconda è resa disponibile una suite di API utilizzabile per l'integrazione con i propri sistemi, in modo simile a quanto offre Authy. Le API coprono tutte le primitive necessarie all'enrolment dell'utente, la gestione del ciclo di vita, la verifica e l'eventuale disattivazione del secondo fattore.

L'esperienza utente è quindi personalizzabile secondo le esigenze e la soluzione diventa completamente trasparente. In questa seconda modalità anche l'applicazione usata come software token può essere personalizzata e integrata con il brand del cliente.

5 Progettazione della soluzione in Unibo

Il lavoro preliminare di analisi del mercato ha permesso di chiarire le possibilità di implementazione di un servizio di verifica del secondo fattore di autenticazione per i sistemi informatici dell'Università di Bologna.

Obiettivo del progetto è stato individuare e mettere in opera una soluzione tecnologica in grado di migliorare il livello di sicurezza delle applicazioni web più sensibili, permettendo agli utenti maggiormente attenti a questi aspetti di usare una soluzione moderna per l'uso del secondo fattore.

Di seguito sono analizzati i motivi alla base del progetto e le considerazioni che hanno guidato la scelta della soluzione da integrare.

5.1 Descrizione del problema

Il progetto è nato dall'esigenza di rendere più sicure le credenziali degli utenti, fornendo maggiore protezione nel caso di intercettazione di password e di uso improprio degli account. Nel caso dell'Università di Bologna le cause più probabili di uso a fini illeciti possono essere:

- invio di spam (danneggiando di conseguenza anche altri utenti del dominio[21]);
- uso improprio della posta elettronica[22];
- uso improprio dei servizi dell'Ateneo (es. accesso a locali, verbalizzazione esami).

Come discusso in precedenza l'introduzione del secondo fattore non è sufficiente da sola a garantire la sicurezza degli account, per questo motivo in concomitanza con il progetto si sono intraprese altre iniziative di consolidamento della sicurezza dei sistemi.
In particolare:

[21] L'invio di spam da parte di alcuni utenti viene segnalato a sistemi automatici di verifica della reputazione di un dominio di posta, quando la reputazione peggiora i sistemi di invio della posta elettronica che fanno da tramite per le mail legittime, smettono di accettare la posta di/per quel dominio.
[22] L'abuso del contenuto della casella di posta o l'invio di messaggi che possano portare a conseguenze civili e penali.

- sono state rafforzate le politiche di gestione delle password (complessità, scadenza, *history*);
- sono stati migliorati i sistemi di *Intrusion Prevention System* e *Intrusion Detection System;*
- è stata rafforzata la politica di disabilitazione degli account inutilizzati per un determinato periodo di tempo.

L'uso del secondo fattore, si è visto, è una soluzione adeguata a migliorare la sicurezza degli utenti più attenti, al tempo stesso questa scelta ha il potenziale di arricchire le applicazioni più delicate con una verifica esplicita di *Strong Authentication*.
Inoltre, in un ambito più ristretto, è possibile considerarlo una soluzione per l'accesso con privilegi di amministratore ai sistemi più sensibili, creando un'ulteriore protezione dei sistemi migliorando la sicurezza degli accessi con maggiori privilegi.

In sintesi i casi d'uso identificati per l'adozione nei sistemi dell'Università di Bologna sono tre:

- miglioramento della sicurezza dell'identità digitale di un utente docente o tecnico amministrativo dell'Ateneo, consentendo l'accesso alle applicazioni web solo previo inserimento delle credenziali e della verifica con secondo fattore;
- miglioramento della sicurezza di operazioni su applicazioni sensibili, quali ad esempio la gestione delle liste di verbalizzazione degli esami (le verbalizzazioni sono già protette da firma digitale con smart card);
- accesso con account privilegiati (di *power users* o amministratori) ai server che ospitano i servizi dell'Ateneo.

I costi medi di questo genere di soluzioni, generalmente tariffati per utente o per singolo login, sono decisamente significativi per un ente della Pubblica Amministrazione. Alcuni prodotti infatti hanno un prezzo intorno ai 2-3 euro per utente al mese, che riportato sui numeri dell'Ateneo (100.000 studenti, 12.000 dipendenti) diventa proibitivo.

Per rispondere alle esigenze espresse dai tre casi d'uso, necessità imprescindibile è stata la possibilità di integrare la soluzione scelta con il SSO usato in Ateneo, in modo

da poter potenzialmente abilitare il secondo fattore su tutte le applicazioni già integrate con il sistema.

Grazie alla natura *opt-in* e all'evidente delicatezza del servizio è stato stabilito sin dall'inizio del progetto che l'opportunità sarebbe stata abilitata solamente per il personale docente e tecnico amministrativo, con passaggi intermedi evolutivi partendo da un sottoinsieme di utenti in modalità sperimentale per poi aprire a tutti l'adesione. Queste considerazioni sono legate anche all'impatto organizzativo, oltre che tecnologico, che una modifica del genere può avere sulle abitudini degli utenti, l'uso dei servizi e l'interazione con il servizio di assistenza all'uso delle applicazioni informatiche.

5.2 Scelta dei canali di verifica da supportare

Una delle prime domande da porsi nella scelta della soluzione da adottare è quali canali di verifica adottare per il secondo fattore di autenticazione.
I principali requisiti in questo campo sono stati il volere rendere più bassa possibile la barriera all'ingresso per gli utenti che avessero adottato la nuova modalità e limitare i costi di distribuzione e assistenza legati alla tecnologia.

Per questi motivi si è stabilito che sfruttare gli smartphone ormai capillarmente diffusi poteva essere la soluzione più adeguata, insieme al supporto SMS per i casi in cui uno smartphone non fosse disponibile (con la possibile complicazione dell'uso con copertura cellulare insufficiente) e l'eventuale bonus di una *grid card* che permettesse di usare il secondo fattore anche da parte di chi fosse sprovvisto di qualsiasi dispositivo.

Dai test effettuati su alcune delle soluzioni che supportano la chiamata vocale, questa tipo di verifica è stata scartata per l'eccessiva lentezza del meccanismo, nonché la suscettibilità alla copertura della rete telefonica ed eventuali barriere linguistiche (alcuni servizi hanno supporto solamente per l'inglese).

Per cercare di aumentare il più possibile la familiarità degli utenti con il sistema di secondo fattore sin dai primi momenti si è scelto di privilegiare soluzioni che potessero

personalizzare l'esperienza di interazione e di assorbirle nel modo più trasparente possibile all'interno dei servizi dell'Ateneo già esistenti.

Questa scelta si è tradotta nell'identificare soluzioni per l'invio di SMS che potessero usare il gateway di spedizione già in uso presso l'Università per la spedizione di messaggi e notifiche relative alle credenziali (con identificativo di spedizione personalizzato e associato all'Ateneo).

5.3 Requisiti e parametri di scelta

Tra le varie soluzioni illustrate nel capitolo 4, la variabilità delle funzioni e modalità di integrazione sono a volte molto marcate. Per la selezione tra i prodotti individuati, sono stati scelti alcuni parametri preferenziali, in parte già anticipati, elencati di seguito in ordine di priorità:

1. costi in linea con le possibilità di una Pubblica Amministrazione;
2. soluzione preferibilmente basata su standard e opensource;
3. possibilità di personalizzare l'esperienza utente (sia per quanto riguarda i canali di verifica, sia per l'interazione durante l'autenticazione);
4. riconoscimento di dispositivi fidati;
5. impatto architetturale minimo;
6. flessibilità di integrazione, sia per le attività degli utenti, sia per quelle dell'assistenza tecnica;
7. disponibilità di un'integrazione esistente per ADFS e altri meccanismi di autenticazione.

Nella tabella riportata di seguito (Tabella 3 - Parametri di scelta tra le soluzioni di verifica secondo fattore) sono esplicitate le caratteristiche salienti delle soluzioni di verifica del secondo fattore, in relazione a ciascuno dei prodotti analizzati.

Emerge subito il fatto che le soluzioni opensource o con un costo di acquisizione più basso sono anche quelle che richiedono un maggiore impegno di integrazione e offrono la minore personalizzazione dei software *token* usati dall'utente.

Escludendo inoltre le soluzioni più costose, si evidenzia anche che non esistono moduli per il funzionamento con ADFS disponibili sulle soluzioni rimanenti.

Data questa situazione si è reso necessario approfondire quanto fosse impegnativo sviluppare autonomamente un modulo di autenticazione per ADFS che potesse integrare una tecnologia a scelta tra le rimanenti. Diverse ricerche di approfondimento hanno portato a chiarire il fatto che la versione successiva (ADFS 3.0) al portale di autenticazione in uso in Ateneo (ADFS 2.0), fornisse un layer di integrazione pensato proprio per estendere l'autenticazione con la verifica di un secondo fattore.

ADFS 3.0 definisce una precisa interfaccia di integrazione a cui è possibile collegarsi scrivendo un modulo in linguaggio C#, attraverso questo *adapter* è possibile interfacciarsi con API di autenticazione e gestire alcuni aspetti dell'interazione con l'utente interni al funzionamento del portale di SSO. L'esistenza di un esempio ufficiale[23], anche se spartano, ha permesso di verificare la fattibilità dell'integrazione con uno sforzo ragionevole, senza dover escludere i prodotti sprovvisti di modulo ADFS.

[23] (Donderwinkel 2014)

Tabella 3 - Parametri di scelta tra le soluzioni di verifica secondo fattore

Valutazione delle soluzioni 2FA	LinOTP	OpenOTP	OneTime	MS MFA	Duo Security	Authy	Time4ID
Out-of-Band	No	No	Sì (via chiamata vocale)	No	Sì	Sì	Sì
Riconoscimento dispositivi fidati	No	No	No	No	Sì	No	No
Costo	Free	n/a	Elevato	Elevato	Elevato	Medio	Medio
Opensource	Sì	No	No	No	No	No	No
Disponibilità di modulo ADFS	No	Sì	Sì	Sì	Sì	No	No
Disponibilità di altri moduli	Sì	Sì	No	No	Sì	Sì	Sì
Personalizzazione dell'applicazione software token	No	No	No	No	No	No	Sì
Software token multipiattaforma	No (di terze parti)	No (di terze parti)	No (di terze parti)	Sì	Sì	Sì	Sì
Personalizzazione dell'esperienza utente	Sì	Sì	No	No	No	Sì	Sì
Impatto architetturale	Alto	Alto	Basso	Medio	Medio	Basso	Basso
Semplicità di integrazione	Media	Media	Elevata	Media	Elevata	Media	Media
Semplicità di utilizzo per l'utente	Media	Media	Media	Elevata	Elevata	Elevata	Elevata
Grid card	Sì	Sì	No	No	No	No	Sì
Token fisico OATH	Sì	Sì	No	No	Sì	Sì	No
Software token app	No (di terze parti)	No (di terze parti)	No (di terze parti)	Sì	Sì	Sì	Sì
Chiamata vocale	No	No	Sì				
SMS	Sì	Sì	Sì	Sì	Sì	Sì	Sì
Personalizzazione gateway SMS	Sì	Sì	No	No	No	No	Sì

5.4 Scelta dell'architettura della soluzione

L'applicazione dei criteri di scelta ha sostanzialmente ristretto il campo a due possibili piattaforme, caratterizzate da peculiarità opposte la prima basata su soluzione *on-premise*, la seconda su soluzione *cloud-based*:

1. **LinOTP**, soluzione *on-premise* con buone capacità di integrazione e flessibilità d'uso, non essendo provvista di software token è necessario sfruttare applicazioni di terze parti (come *Authy*, *Google Authenticator* o altri), la personalizzazione è possibile sul versante dei servizi di *backend* (gestione identità digitale e assistenza help desk);
2. **Time4ID**, soluzione *cloud-based* con flessibilità di utilizzo, basso impatto architetturale (è necessario solamente installare i certificati per l'accesso alle API) e con software token personalizzati e personalizzabili, oltre che la possibilità di integrarsi con il proprio gateway SMS.

Approfondendo la prima opzione e studiando l'architettura del prodotto è apparso subito chiaro che per una seria gestione del sistema sarebbe stato necessario investire tempo per una configurazione ottimale, per la creazione della necessaria ridondanza e per la configurazione di parametri di monitoraggio specifici. Questa valutazione, insieme a qualche problema di installazione dovuto al *packaging* non ottimale del software ha lasciato l'impressione di una generale mancanza di robustezza che sarebbe stato necessario integrare con investimento di tempo e risorse.

Per contro, l'approfondimento delle API di Time4ID e delle funzioni della piattaforma attraverso un confronto con gli sviluppatori del prodotto, ha convinto della semplicità e della robustezza della soluzione, unite a un impatto architetturale molto basso sui sistemi dell'Università.
Il costo contenuto rispetto ad altre soluzioni *cloud-based* e l'offerta di personalizzazione con i colori dell'Ateneo dei software *token*, sono state le motivazioni finali che hanno spinto in questa direzione.

Anche l'opportunità di lavorare con un fornitore sul territorio, semplificando i confronti e le operazioni di integrazione, è stato determinante per convincere definitivamente ad attivare in via sperimentale il servizio con questa soluzione tecnologica.

6 Descrizione dell’implementazione

Una volta stabilita la tecnologia da utilizzare è stato possibile pianificare le attività necessarie all'integrazione e alla messa in opera della verifica del secondo fattore per il sistema di autenticazione dell'Ateneo.
Tra i tre casi d'uso illustrati in precedenza (5.1) si è scelto di privilegiare il primo, ossia il miglioramento della sicurezza dell'identità digitale di un utente docente o tecnico amministrativo dell'Ateneo, consentendo l'accesso alle applicazioni web solo previo inserimento delle credenziali e della verifica con secondo fattore, con l'intenzione di procedere con gli altri casi d'uso qualora il risultato della sperimentazione si fosse dimostrato in linea con le aspettative.

Gli elementi di base necessari per l'integrazione di tutti gli aspetti del servizio sono stati:

- migrazione da ADFS 2.0 ad ADFS 3.0;
- test di integrazione con le API Time4ID;
- sviluppo di un modulo per ADFS 3.0 che integri Time4ID per le operazioni di verifica;
- sviluppo di un'applicazione web per l'enrolment dell'utente, integrata con le API Time4ID e il preesistente sistema di gestione dell'account istituzionale;
- integrazione con gateway di spedizione SMS;
- integrazione dell'applicazione mobile con il sistema di autenticazione dell'Ateneo (per effettuare la verifica dell'identità e la sincronizzazione con Time4ID);
- personalizzazione dell'applicazione mobile;
- test sulle principali piattaforme di dispositivi mobili;
- integrazione delle API Time4ID con strumenti di supporto per il servizio di assistenza help desk.

Uno dei passi più impegnativi è stato, per la delicatezza del sistema, la migrazione del servizio ADFS 2.0 alla nuova versione. Il SSO di Ateneo infatti raccoglie oltre 200 applicazioni integrate, non sempre presidiate da un riferimento tecnico, coinvolte in una certa misura dall'aggiornamento. L'attività di aggiornamento non ha potuto contare su di un fermo dei sistemi, vista l'incidenza sulle attività dell'organizzazione, obbligando a strutturare le operazioni per una migrazione a caldo.

Il passaggio alla nuova versione ha introdotto alcune novità obbligatorie, tra cui il comportamento della navigazione dell'utente, le modalità di inserimento delle credenziali e variazioni obbligatorie nella configurazione della quasi totalità delle applicazioni integrate.
Il lavoro di test, verifica e orchestrazione del cambiamento in modo che fosse più trasparente possibile per l'utente, sono stati decisamente impegnativi, considerando inoltre che l'aggiornamento *in-place* non era supportato dal prodotto, in quanto avrebbe obbligato ad un passaggio attraverso una versione intermedia, eventualità esclusi perché avrebbe portato altri problemi di impatto.

Terminato questo passo preliminare si è potuto completare il resto delle attività di sviluppo, di seguito sono riportati i dettagli implementativi dell'integrazione.

6.1 Integrazione con il flusso di aggiornamento delle identità

Si è visto nelle sezioni introduttive (3.3) come la gestione dei dati, che corredano l'identità digitale in un ente complesso come l'Università di Bologna, sia gestita da un flusso di aggiornamento che si occupa di orchestrare le varie fonti che concorrono a definire l'identità digitale.
L'introduzione del sistema di verifica di secondo fattore è un esempio di informazione da gestire attraverso il flusso di aggiornamento. Infatti questo servizio è potenzialmente attivabile per tutti gli utenti, richiede la suddivisione in fasi del processo (attivabile, attivato) e si rende necessario registrare a fini conoscitivi lo stato delle attivazioni *opt-in* man mano che gli utenti aderiscono al sistema.

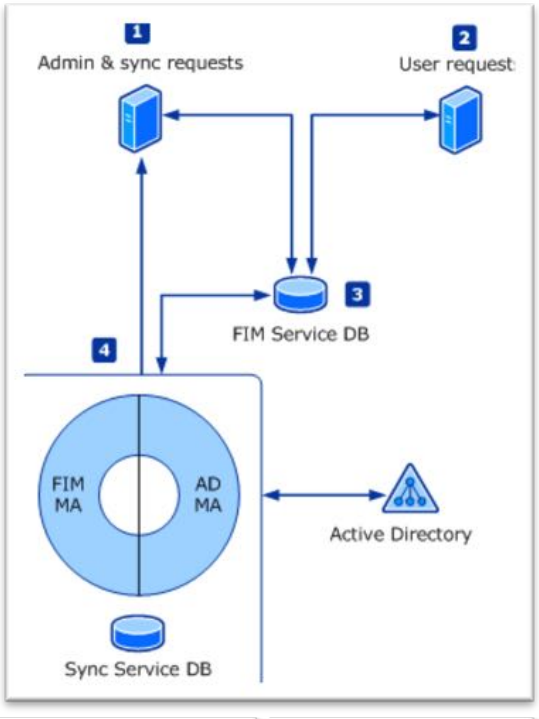

Figura 6-1 - Architettura di Forefront Identity Manager, i Management Agent (MA) si occupano della sincronizzazione dei dati generati da utenti e amministratori, per combinarli in Active Directory (Microsoft 2010)

Il flusso di aggiornamento è gestito dal componente *Forefront Identity Manager* (FIM), dedicato alle operazioni di sincronizzazione delle fonti di dati con lo stato dell'identità digitale espressa in Active Directory dall'appartenenza a gruppi, OU e al possesso di determinati attributi sull'utente dell'istituzione.

FIM per operare al meglio ha bisogno che le fonti di dati siano in un formato piatto e normalizzato (come se si trattasse di un *Data Warehouse*), pertanto l'implementazione dell'Ateneo è provvista di una serie di procedure sviluppate nel database MS SQL Server di appoggio, che hanno il compito di raccogliere le informazioni dalle varie fonti di interesse, metterle in relazione, normalizzare e organizzarle in un formato ottimale per l'utilizzo da parte del software di sincronizzazione.

Questa organizzazione rende necessaria l'integrazione delle nuove informazioni all'interno di queste procedure di preparazione, in modo da stabilire il comportamento del sistema al momento dell'aggiornamento.

Nella scelta dell'implementazione all'interno del flusso di update si sono considerati questi fattori:

- mantenere il più semplice possibile la gestione degli utenti di questo genere;
- mantenere il più semplice possibile l'accesso ai dati necessari, nell'eventualità che altri sistemi oltre al SSO vogliano verificare il secondo fattore (es. per gli altri casi d'uso);
- limitare le modifiche necessarie all'integrazione delle applicazioni di enrolment e gestione.

Il dato sulla possibilità di attivare il secondo fattore è inserito autoritativamente nel DB di appoggio di FIM. L'abilitazione può essere effettuata a partire dalla struttura organizzativa, in modo da rendere graduale la messa in opera.
Quando un utente appartiene ad una struttura organizzativa abilitata, alla fine della sincronizzazione di FIM viene inserito in un gruppo automatico creato su Active Directory (es. *DSA.Personale.SecondoFattoreAttivabile*). Se l'utente ha effettuato l'*opt-in* attraverso il processo di *enrolment,* la procedura dedicata lo inserisce in un gruppo riservato, quest'ultima informazione è gestita da FIM solo a fini informativi, recuperandola da AD per riportarla nel DB come riferimento.

In questo modo i sistemi che intendono verificare il secondo fattore possono usare l'informazione di appartenenza ai gruppi per stabilire se fare la richiesta all'utente e interagire con la piattaforma Time4ID per la somministrazione e verifica del codice OTP.

6.2 Gestione e provisioning del secondo fattore

La gestione del secondo fattore di autenticazione si inserisce necessariamente nella preesistente gestione dell'account istituzionale dell'Università.

Figura 6-2 - Sistema di gestione delle informazioni legate all'account istituzionale

Dall'applicazione mostrata in Figura 6-2 l'utente può gestire le credenziali, cambiare le informazioni di contatto e accedere ad alcune informazioni strettamente legate alla gestione della propria identità digitale.

In questo servizio è stato aggiunto un modulo applicativo, integrato con la piattaforma Time4ID, che permette all'utente di attivare in autonomia, modificare e cancellare il secondo fattore di autenticazione. Quest'ultimo può essere sostanzialmente di 2 tipi:

- SMS spedito dal gateway di Ateneo (con mittente "UniBO" o "CeSIA");
- software token costituito da applicazione mobile per le piattaforme iOS, Android e Windows Phone che l'utente dovrà installare sul proprio smartphone.

Nella figura riportata di seguito è sintetizzato il diagramma di sequenza delle attività di enrolment, la "*Company web application*" è costituita dal suddetto modulo applicativo sviluppato ad hoc.

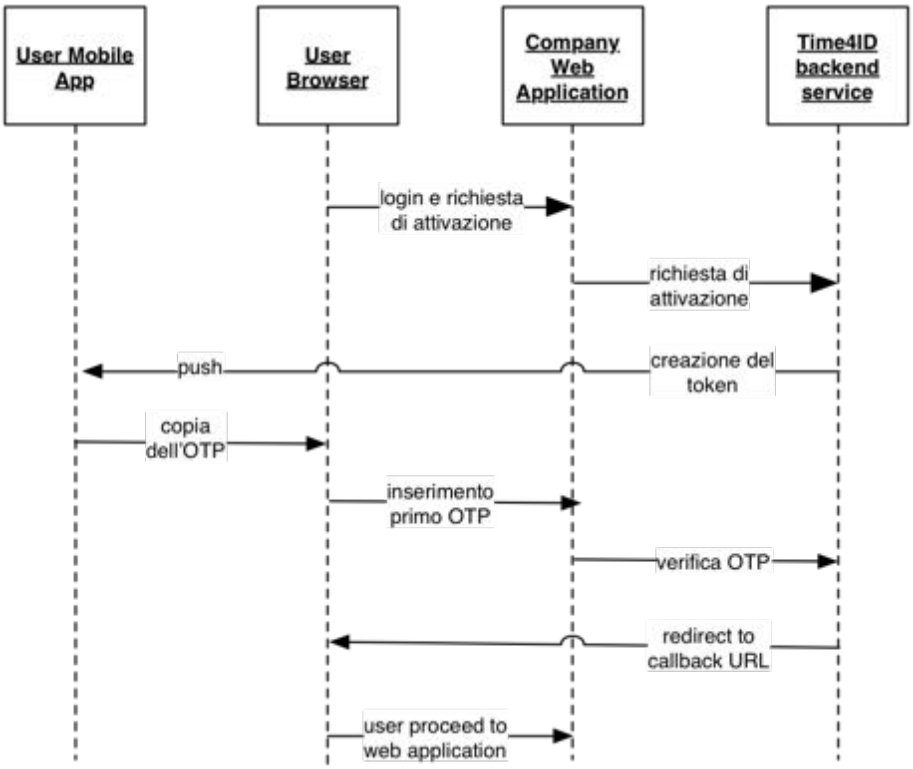

Figura 6-3 - Processo di enrolment su piattaforma Time4ID

L'accesso a questa funzione così come la possibilità di attivare e disattivare il secondo fattore, è possibile solo agli utenti facenti parte della sperimentazione (attraverso il controllo dell'appartenenza al gruppo su Active Directory).

Le operazioni di modifica del secondo fattore e di cancellazione sono delicate da un punto di vista della sicurezza. Per mantenere l'efficacia del *two-factor authentication* è necessario che in questi casi sia richiesto il secondo fattore, altrimenti l'intercettazione delle credenziali sarebbe elemento sufficiente a disabilitare la sicurezza aggiuntiva.

6.3 Enrolment dell'utente

Al momento dell'attivazione l'utente deve inserire solamente un numero di cellulare e scegliere la tipologia di secondo fattore che vuole usare. Il numero telefonico è necessario anche nel caso sia scelto il software token, questo perché le API di Time4ID richiedono che l'attivazione dell'app avvenga con l'inserimento di un codice inziale.

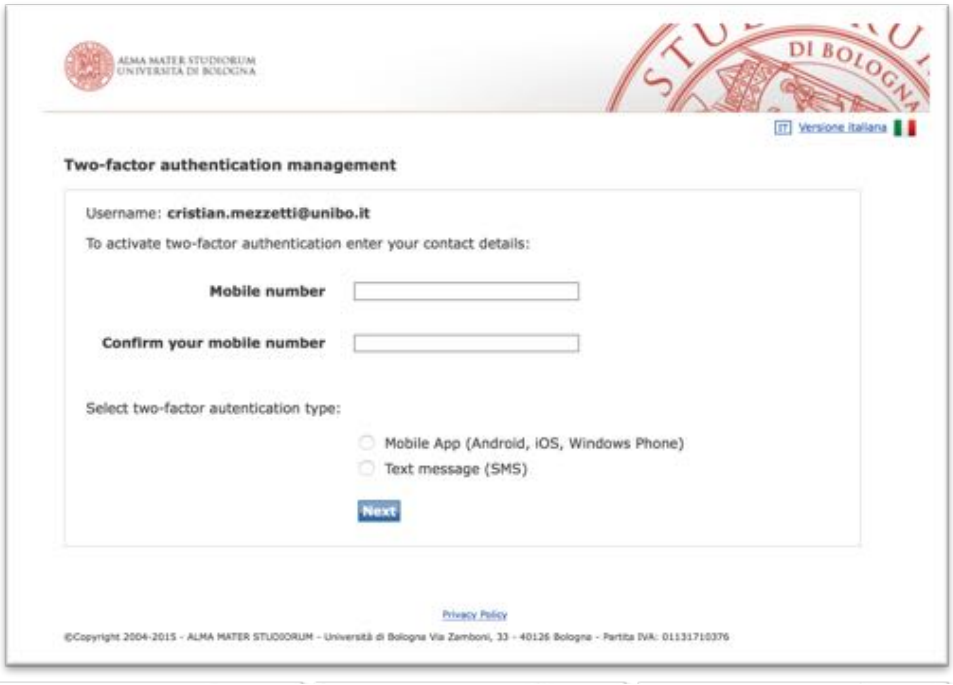

Figura 6-4 - Processo di enrolment sui sistemi dell'Ateneo

Il codice può essere recapitato all'utente via SMS o email. Per semplificare le operazioni di attivazione, si è però scelto di mantenere solamente il messaggio telefonico, valutando che sarebbe stata l'opzione più semplice per l'utente. Lo stesso numero di telefono è anche usato per l'invio dei codici OTP nel caso l'utente scelga di usare il canale di verifica SMS.

Una volta scelto il canale SMS l'utente riceve il primo codice, successivamente gli viene chiesto di inserirlo per conferma di attivazione, infine se la validazione avviene correttamente da quel momento è inserito nel gruppo degli utenti abilitati.
Contestualmente gli viene creato un token associato sulla piattaforma Time4ID dove è conservata la chiave segreta alla base dell'algoritmo HOTP usato per la verifica.

Se l'utente sceglie il canale di verifica con software *token*, sono visualizzate le istruzioni per il download e l'attivazione dell'applicazione Unibo Pass, la versione personalizzata per l'Ateneo del token fornito da Time4ID.

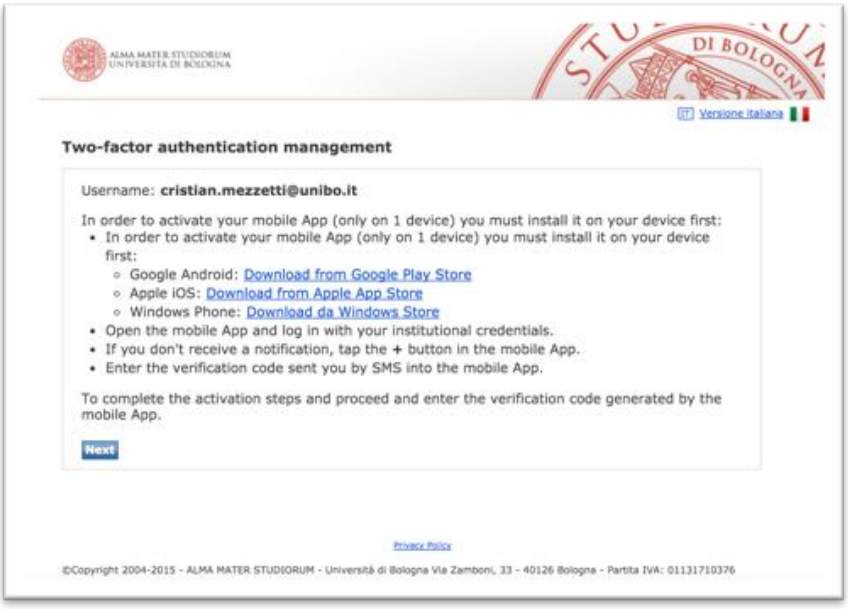

Figura 6-5 - Attivazione del token con app Unibo Pass

L'utente deve installare l'applicazione per completare l'attivazione, dopodiché:

- riceve via SMS il codice di attivazione del secondo fattore;
- entra con le credenziali istituzionali nell'applicazione Unibo Pass;
- inserisce il codice ricevuto;

A questo punto l'applicazione inizia a generare codici OTP sincronizzati con Time4ID e ogni 30 secondi all'utente ne viene mostrato uno differente.

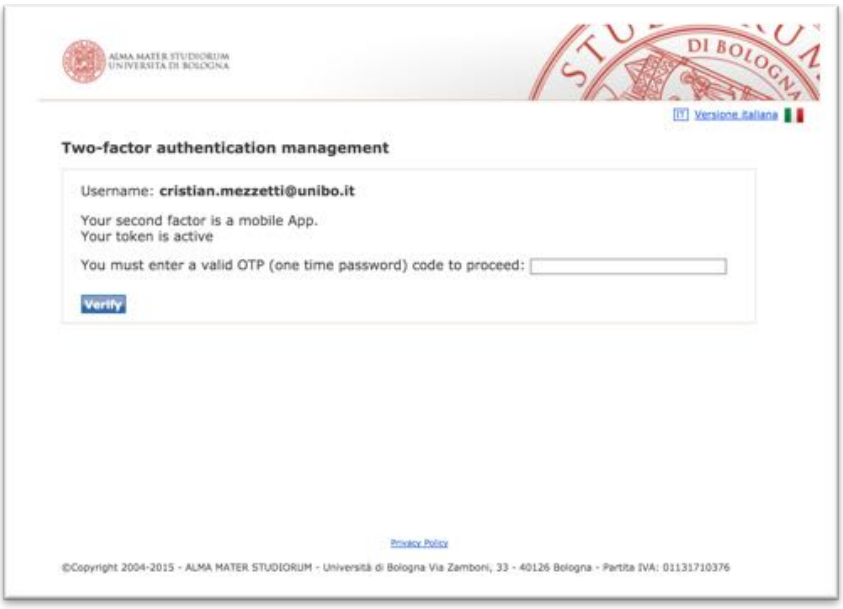

Figura 6-6 - Conferma dell'avvenuta attivazione del token da parte dell'utente, prima dell'inserimento nel gruppo degli abilitati

Per concludere l'enrolment l'utente deve inserire il codice OTP generato, confermando il corretto funzionamento per poter essere inserito nel gruppo in Active Directory.

6.4 Gestione e modifica del secondo fattore in autonomia

Per impedire che l'intercettazione delle credenziali possa invalidare la funzione del secondo fattore, ogni attività sulle impostazioni è sempre verificata con esplicito *strong authentication*.

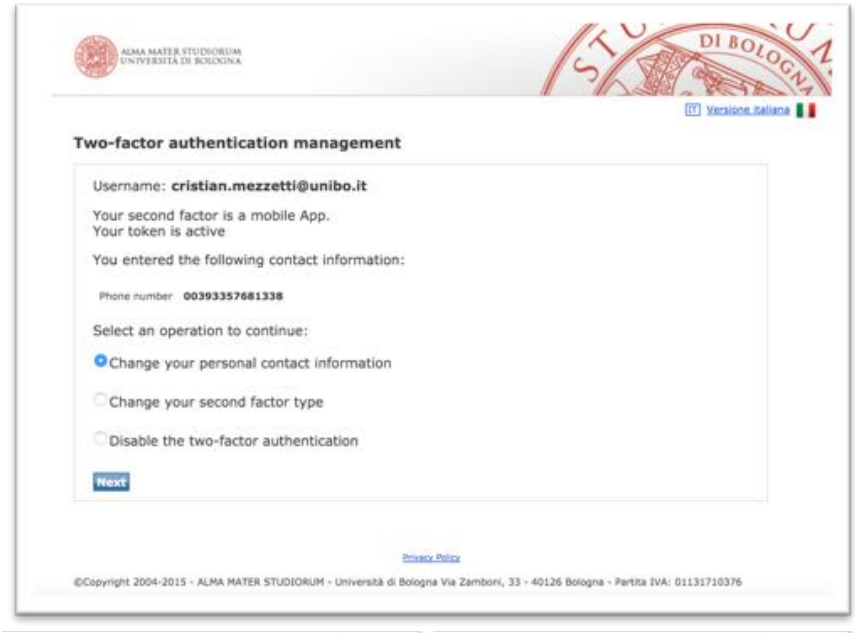

Figura 6-7 - Modifica delle informazioni relative al secondo fattore

L'utente ha la possibilità di modificare i contatti e il tipo del secondo fattore passando da App a SMS o viceversa a seconda delle esigenze, la modifica è immediata e il nuovo canale di verifica è immediatamente attivo.

La cancellazione del secondo fattore deve essere fatta tramite l'applicazione web se si utilizzano messaggi SMS, mentre se si usa l'app mobile, il secondo fattore può essere eliminato sia dallo smartphone sia dall'applicazione web. Anche la cancellazione è immediata, se però avviene da smartphone è l'applicazione web di enrolment che si occupa di allineare la composizione dei gruppi locali con lo stato di abilitazione su Time4ID al successivo passaggio dell'utente.

6.5 Applicazioni personalizzate per dispositivi mobili

L'applicazione per dispositivi mobili, denominata Unibo Pass, è stata personalizzata e pubblicata per le tre principali piattaforme: iOS, Android e Windows Phone.

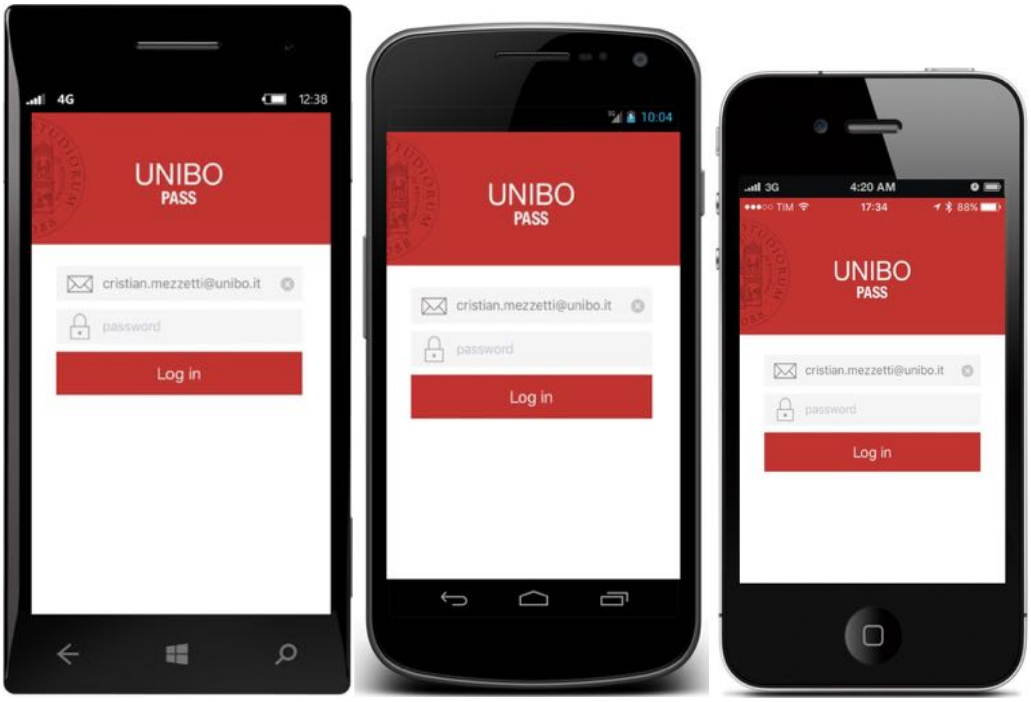

Figura 6-8 - Le tre piattaforme mobile supportate da Unibo Pass

Si tratta di un software token basato su TOTP, l'utente deve effettuare l'inizializzazione della chiave segreta attraverso la procedura web che lo guida e genera la chiave dal lato dei servizi Time4ID.
Una volta iniziata la procedura di attivazione l'utente deve installare l'app dallo *store* del suo smartphone, dopo il download il primo passo è l'inserimento delle credenziali istituzionali che sono richieste al momento dell'avvio.

L'inserimento delle credenziali serve a identificare l'utente sulla piattaforma Time4ID e ad associarlo al proprio profilo, la verifica dell'identità è effettuata sui sistemi dell'Ateneo, l'applicazione infatti richiama un *Web Service* integrato con il sistema ADFS 3.0 che permette verificare attraverso protocollo SOAP e WS-Trust l'autenticazione dell'utente. Solamente gli utenti appartenenti al gruppo di potenziali utilizzatori del secondo fattore possono effettuare l'autenticazione in questo modo.

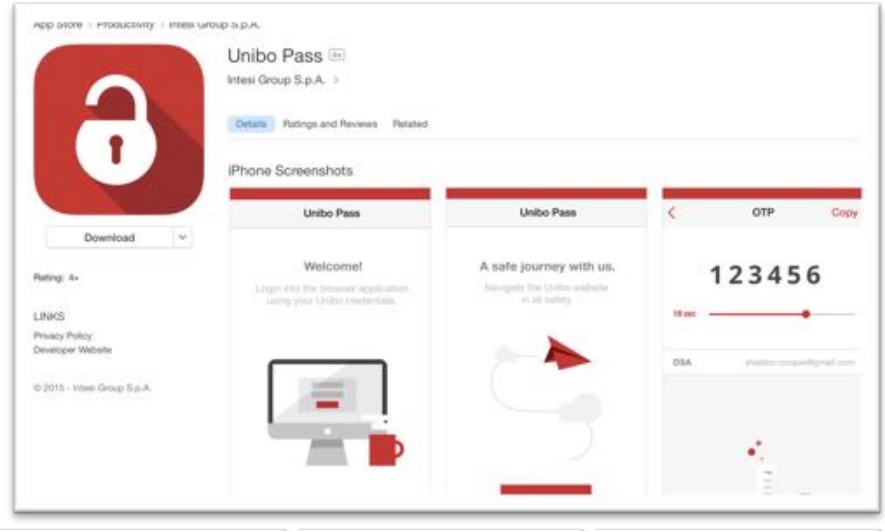

Figura 6-9 -pagina di iTunes Store per Unibo Pass

L'autenticazione ha anche la funzione di gestire una sessione applicativa dell'app, in modo da proteggere il codice OTP generato uscendo dalla sessione o utilizzando un blocco automatico che nasconda i codici.

Al momento dell'ingresso nell'applicazione l'utente trova l'invito a procedere con l'inizializzazione, operazione che effettua un collegamento alla piattaforma Time4ID, cerca il token dell'utente e recupera le informazioni descrittive da visualizzare.

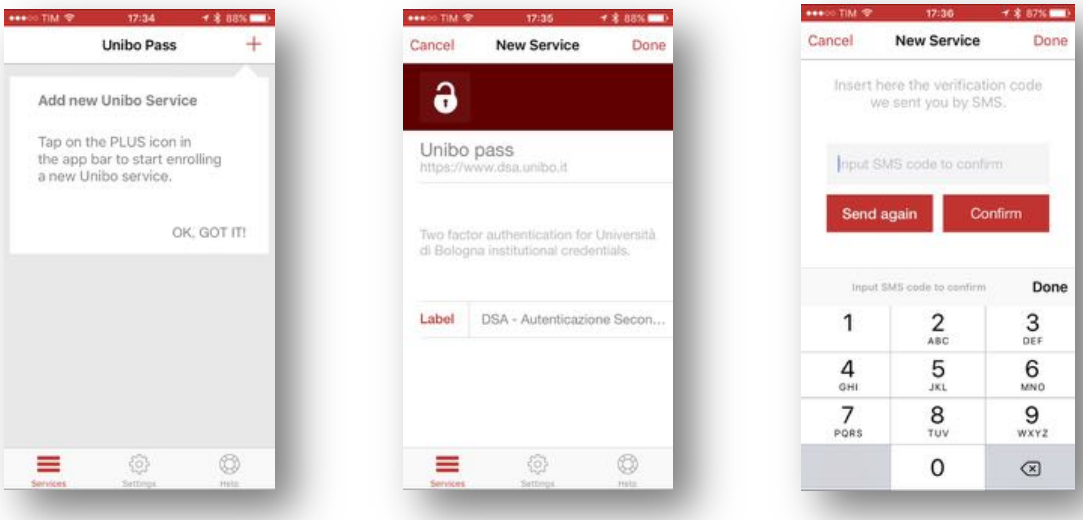

Figura 6-10 - Pagina iniziale di benvenuto, informazioni sul token da inizializzare, inserimento del codice di attivazione del token

L'applicazione (e l'organizzazione) può gestire più token, in modo da raccoglierne diversi in un unico punto ed avere codici separati per attività separate. Per completare

l'inizializzazione si inserisce il codice spedito via SMS al numero indicato nella procedura web.

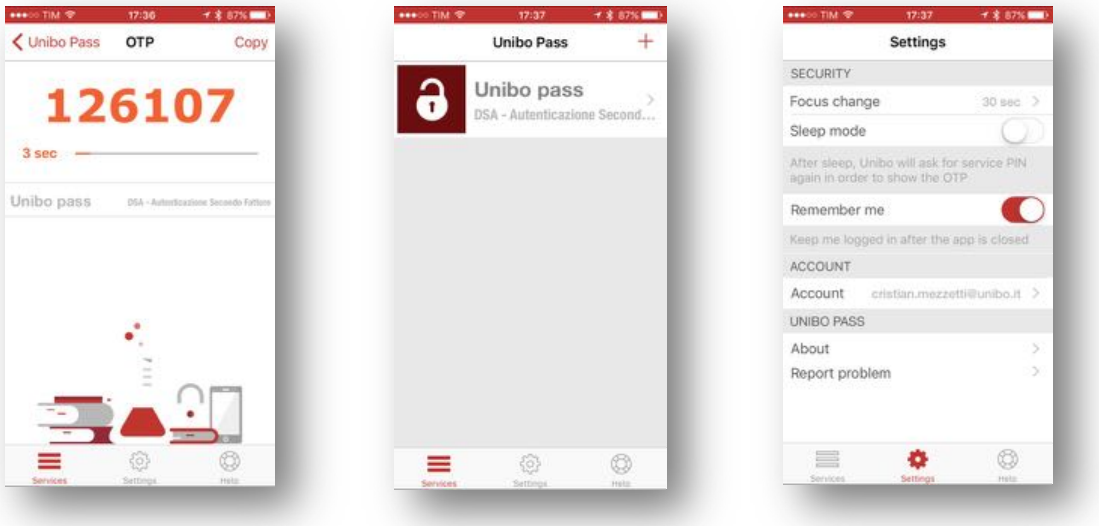

Figura 6-11 - Codice OTP generato, lista dei servizi di strong authentication inizializzati, impostazioni di sicurezza e informazioni dell'app

Se il codice è corretto l'applicazione inizierà a visualizzare il codice OTP calcolato per lo slot temporale corrente, con un contatore decrescente che indica l'imminente sostituzione con un nuovo codice. I codici sono visualizzati per 30 secondi ma la raccomandazione per le implementazioni OATH è quella di considerare validi due o tre codici per evitare frustrazione all'utente nel caso di leggeri ritardi o desincronizzazioni degli orologi dei dispositivi e del server.

L'applicazione offre anche la modalità di verifica *Out-Of-Band*, tuttavia per sfruttarla è necessario un supporto lato *backend* la cui implementazione e integrazione con i vari sistemi è significativamente più impegnativa, motivo per cui è stata rimandata un'eventuale implementazione a sviluppi futuri del progetto.

In particolare può essere interessante per l'organizzazione sfruttare la stessa tecnologia per supportare processi documentali. L'uso di meccanismi di verifica Out-Of-Band di Strong Authentication permette infatti di associare un'approvazione legalmente più forte rispetto alla sola autenticazione, consentendo di snellire i processi che richiederebbero una firma autografa o una firma digitale.

6.6 Creazione del backend per le attività di assistenza

Per eventuali problemi relativi all'uso o all'attivazione del secondo fattore o dell'applicazione di enrolment, gli utenti hanno a disposizione il servizio di help desk coordinato dal CeSIA.

Per supportare il lavoro dell'help desk è stato necessario preparare uno strumento che potesse rendere visibile lo stato di un utente all'interno del processo di attivazione ed uso del secondo fattore di autenticazione.

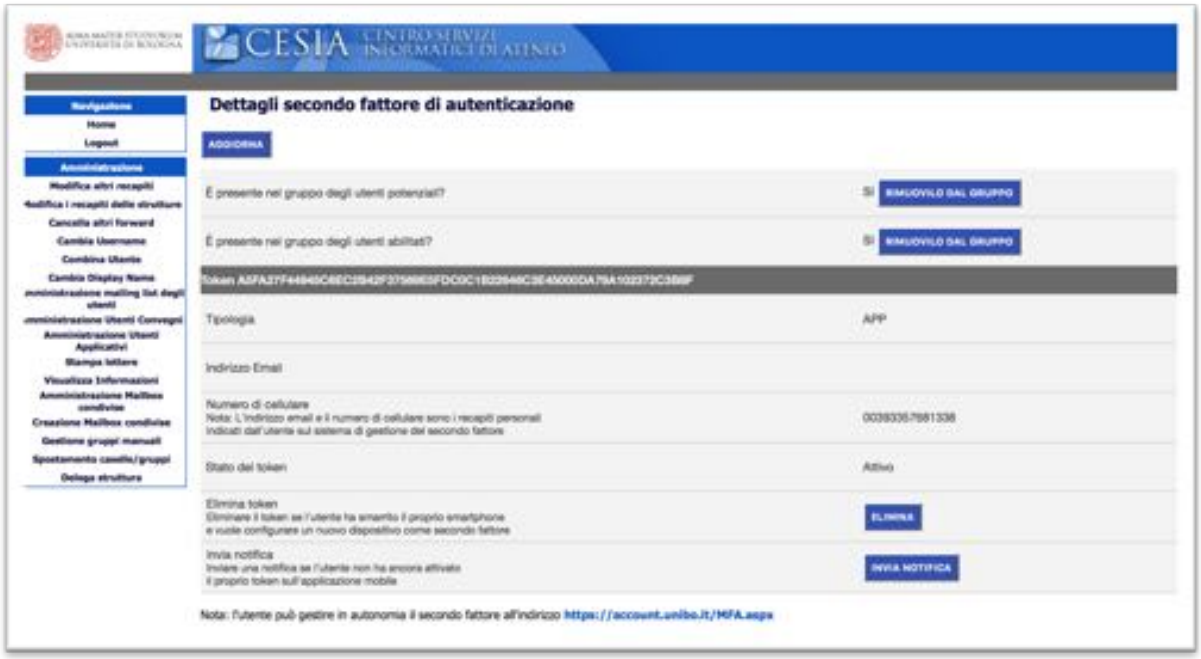

Figura 6-12 - Applicazione di backend utilizzata dall'help desk per l'assistenza agli utenti

La funzionalità sviluppata si integra all'interno dell'applicativo già esistente per la gestione di *backend* dell'identità digitale degli utenti. È un'applicazione che permette di investigare la situazione di ogni utente ed operare sui sistemi in modo coordinato ai servizi di aggiornamento dell'identità, nei casi in cui si rendano necessarie operazioni manuali legate alla risoluzione di anomalie.

Il modulo applicativo consente di appurare una serie di condizioni utili a determinare lo stato di attivazione del secondo fattore:

- se l'utente è incluso nella sperimentazione;
- se esiste un token su Time4ID che indica che l'utente ha iniziato la procedura di attivazione;
- il numero di cellulare inserito dall'utente;

- il tipo di secondo fattore scelto dall'utente;
- se l'utente è stato inserito nel gruppo degli abilitati, indicando che ha completato correttamente il processo di enrolment.

Le funzionalità sono infine completate dalle operazioni dispositive che permettono all'operatore dell'help desk di eliminare il token su Time4ID o l'appartenenza ai gruppi (tracciando opportunamente le operazioni), in modo da disattivare il secondo fattore su richiesta dell'utente. È cura del servizio di supporto appurare l'identità di chi effettua la richiesta, con le modalità già in essere per la gestione di cambio della password e reimpostazione delle credenziali, per evitare furti di identità tramite ingegneria sociale.

6.7 Attività di integrazione su Identity Provider ADFS

Una volta che l'utente ha completato l'attivazione del secondo fattore è compito dei servizi attivare la possibilità di verifica contestualmente all'autenticazione delle credenziali.

L'Ateneo è dotato di un servizio di autenticazione Web SSO (ADFS 3.0) che integra la maggior parte delle applicazioni disponibili ai propri utenti. L'individuazione di questo punto di accesso come candidato alla prima integrazione con il secondo fattore è quindi stata naturale.

Microsoft ADFS 3.0 ha un supporto esplicito per l'integrazione di moduli di terze parti che implementino un secondo fattore di autenticazione. La stessa azienda fornisce un modulo (*MFA adapter*) che basa il suo funzionamento su un componente da installare *on-premise* (*MFA Server*) e un servizio gemello *cloud-based* per i servizi dell'organizzazione che sfruttano la piattaforma Office 365.

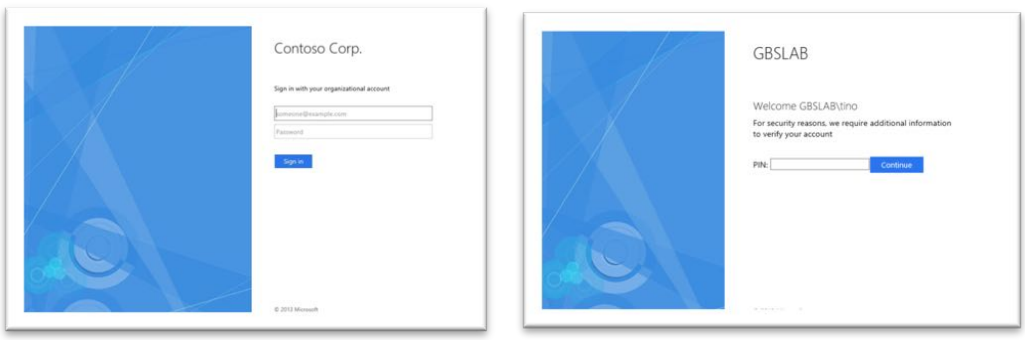

Figura 6-13 - Esempio della sequenza di funzionamento del Microsoft Multifactor Authentication Adapter

L'Ateneo ha una sottoscrizione ai servizi di Office 365, in particolare è la piattaforma usata per offrire la posta elettronica, i calendari e lo spazio di lavoro agli studenti iscritti. Con l'uso del portale di autenticazione ADFS anche questi servizi sono però dipendenti dal sistema di autenticazione on-premise, la piattaforma è a tutti gli effetti federata, pertanto le credenziali sono inserite dagli utenti solamente sui sistemi dell'Università.
Quest'architettura permette di avere un solo componente dove attivare il secondo fattore anche nel caso di estensione del servizio alla piattaforma *cloud-based*.

Le possibilità di configurazione di ADFS riguardo all'uso del secondo fattore permettono di intercettare l'autenticazione degli utenti sulla base di tre condizioni:

1. l'appartenenza a un gruppo di Active Directory;
2. il tentativo di accesso da una rete non fidata;
3. il tentativo di accesso da un dispositivo registrato (utile solo in contesti dove tutti i dispositivi sono centralmente controllati).

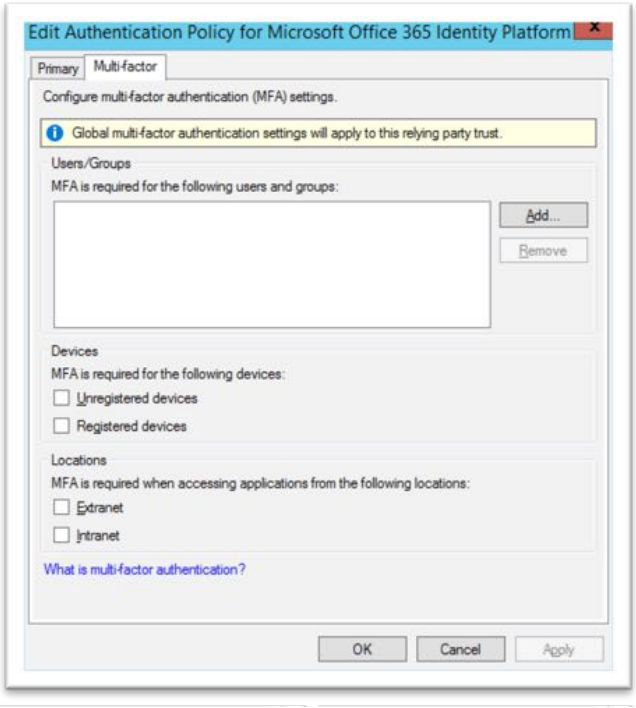

Figura 6-14 - impostazioni di configurazione della richiesta di secondo fattore in ADFS 3.0

Le impostazioni del sistema vedranno pertanto la configurazione del gruppo di abilitati al secondo fattore, con la possibilità di escludere la richiesta ai tentativi di autenticazione provenienti da reti di Ateneo, nel caso si volesse rendere meno frequente la richiesta della verifica.

Il modulo sviluppato per ADFS è costituito da una libreria software (DLL sviluppata in C#) che sfrutta gli *hook* definiti dal prodotto per attivare il comportamento personalizzato dopo la fase di autenticazione del primo fattore (definito *Primary Authentication*, di solito è l'inserimento delle credenziali tramite pagina web o ticket K5erberos).
La libreria dev'essere installata su tutti i server del cluster ADFS che hanno il ruolo di verificare l'autenticazione (l'altro ruolo in un cluster è determinato dal *reverse proxy* delle richieste, normalmente posizionato in DMZ) e deve avere tutti gli elementi necessari per il funzionamento.

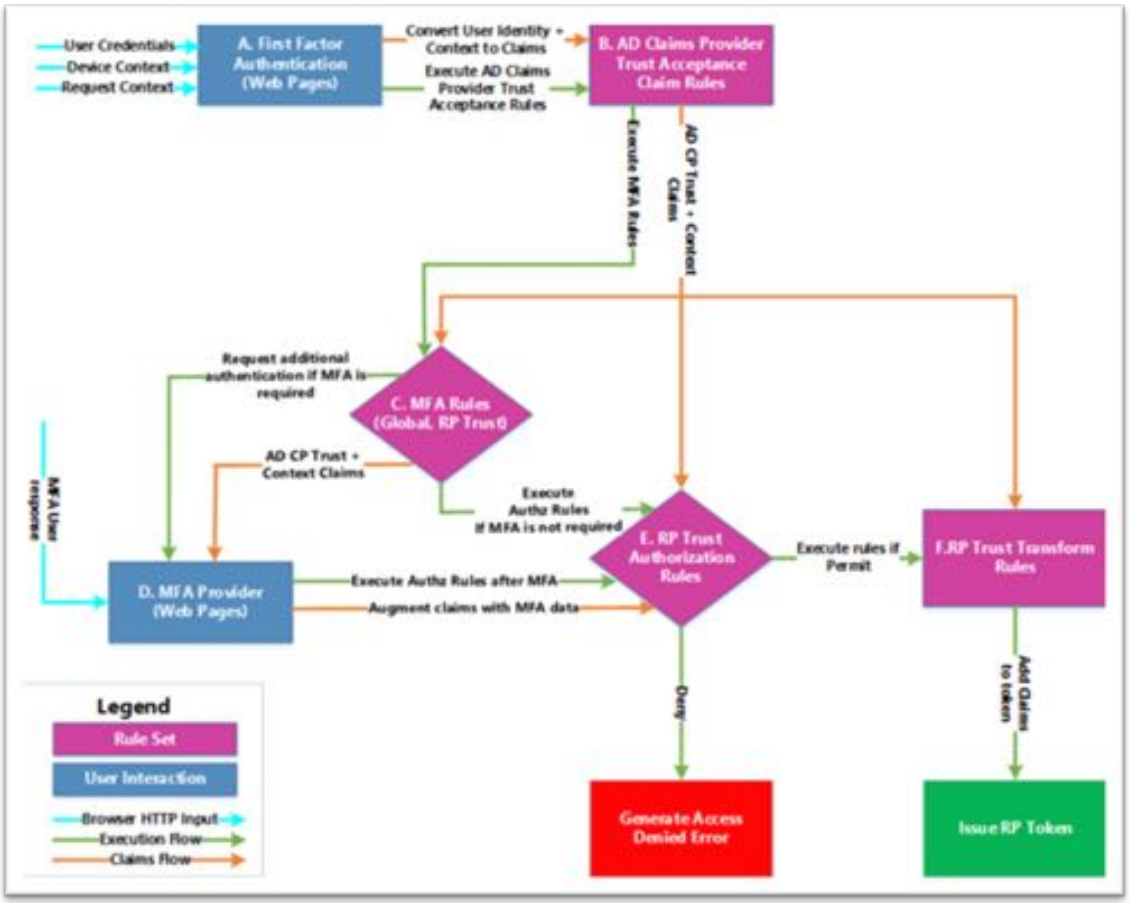

Figura 6-15 - Flusso di autenticazione e valutazione degli attributi al momento dell'autenticazione e verifica del secondo fattore (Calderon 2014)

Per il corretto funzionamento la libreria deve implementare l'interfaccia `IAuthenticationAdapter`, parte dell'implementazione di ADFS, definendo il comportamento della verifica del secondo fattore e le condizioni di successo e fallimento (`IAdapterPresentation OnError`), preoccupandosi di gestire il feedback verso l'utente (`IAdapterPresentation TryEndAuthentication`).
Nella figura precedente è illustrato come si inserisce la verifica del secondo fattore all'interno del flusso di autenticazione e autorizzazione di ADFS, definendo anche la gestione contestuale degli attributi generati sull'identità digitale.

Il modulo *Authentication Provider* è stato sviluppato a partire da uno scheletro di esempio fornito nella documentazione del prodotto e personalizzato per funzionare con la piattaforma Time4ID.
Al momento della verifica del secondo fattore il modulo interroga il *Web Service* di gestione (Time4IDAdm) fornendo lo username che ha eseguito l'autenticazione, per identificare il tipo di secondo fattore impostato dall'utente. Con questa informazione procede con le operazioni di validazione, come nella sequenza in figura, che si differenziano operativamente a seconda che il token sia di tipo SMS o App.

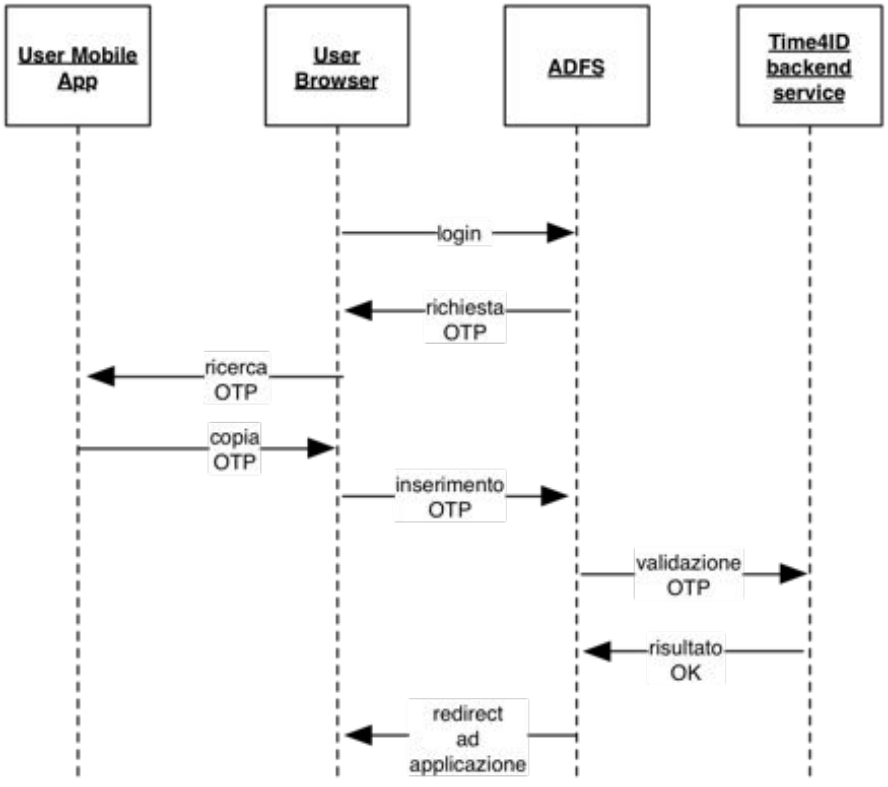

Figura 6-16 - Diagramma di sequenza della validazione del secondo fattore con Time4ID

Nel caso del token di tipo SMS il codice è generato con algoritmo HOTP al momento della verifica, il modulo richiama l'API di Time4ID che genera il codice e si occupa di spedirlo via SMS al numero associato al token dell'utente. L'invio dell'SMS è effettuato interfacciandosi con un Web Service dell'Ateneo, già utilizzato per l'invio di messaggi istituzionali (come notifiche o informazioni sulle credenziali), in modo che il mittente sia lo stesso già familiare agli utenti (CeSIA).

In questo modo l'*Authentication Provider* autentica correttamente l'utente ogni volta che questo accede a un servizio configurato per l'uso del secondo fattore. Questo comportamento non è tuttavia molto pratico, la richiesta di un secondo fattore ad ogni accesso, o a distanza di breve tempo, può diventare motivo di frustrazione e spingere l'utente a disabilitarlo rinunciando alla maggiore sicurezza in favore della praticità. Questa considerazione ha spinto allo sviluppo di una funzionalità che consentisse all'*Authentication Provider* di riconoscere, per un certo periodo di tempo, un dispositivo dichiarato fidato dall'utente.

Per evitare di complicare l'infrastruttura si è scelto di ispirare il funzionamento di questa caratteristica al comportamento della classica autenticazione in SSO, lasciando al browser l'onere di presentarsi adeguatamente ad ogni autenticazione, portando un cookie di riconoscimento che potesse garantire l'avvenuta certificazione del browser da parte dell'utente.

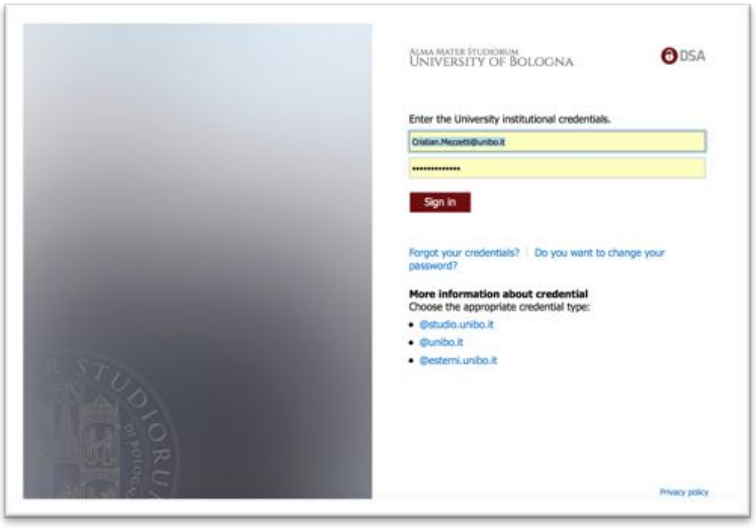

Figura 6-17 - Portale di autenticazione SSO dell'Ateneo

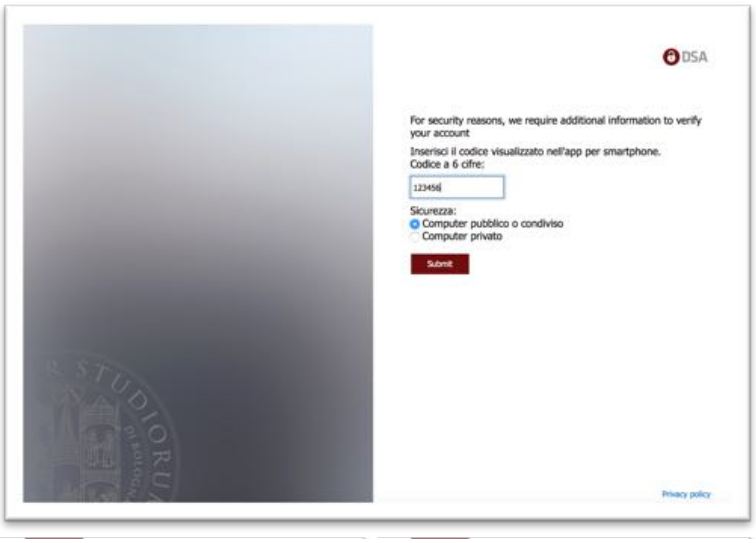

Figura 6-18 - Richiesta del secondo fattore di autenticazione

L'*Authentication Provider* si occupa quindi di impostare un cookie crittografato con una chiave segreta conosciuta solamente al server ADFS, che ha come contenuto tre informazioni:

1. *timestamp* dell'emissione del cookie: permette di controllare se il periodo di validità della "certificazione" del browser è stato superato, al momento della verifica del secondo fattore infatti l'utente sceglie che tipo di postazione sta usando, se è una postazione privata (quindi maggiormente controllata) durerà 30 giorni, altrimenti 20 minuti;
2. *fingerprint* del browser: è un identificativo della postazione calcolato con metodi di *browser fingerprinting*[24], sfruttando la libreria opensource Fingerprint.js[25] che eseguita all'interno del browser riesce ad ispezionare una serie di variabili dell'ambiente (browser, OS, plugin installati, risoluzione video) che rende accurato il riconoscimento in più del 96% dei casi (l'uso di questa tecnica permette di salvaguardare dall'eventualità di un furto del cookie e l'uso su un'altra postazione);
3. *ID* del token Time4ID dell'utente: serve ad assicurarsi che nel caso l'utente cambi il token per qualsiasi motivo (modifica del tipo, rinnovo per precauzione), i vecchi cookie siano implicitamente invalidati.

Il *browser fingerprinting* è una tecnica di identificazione dei dispositivi estremamente efficace, nata in contesti commerciali dove l'interesse è profilare l'utente per indirizzargli con maggiore precisione messaggi pubblicitari. È importante pertanto considerare che in questo contesto l'uso è estremamente limitato e volto unicamente a riconoscere un dispositivo, l'informazione non è memorizzata sui server dell'Ateneo ma solo sul browser dell'utente. Al momento del calcolo del *fingerprint* il valore è confrontato con quello presentato dall'utente, senza ulteriori elaborazioni.

[24] Tecnica basata su riconoscimento di tratti caratteristici del browser, piattaforma e corredo software, descritta in (Eckersley 2010), (Mowery e Shacham 2012) e (Acar, et al. 2014).
[25] (Valve 2013)

Figura 6-19 - Diagramma di flusso della verifica del secondo fattore con l'Authentication Adapter

Il modulo infine si occupa di effettuare operazioni di log dedicate ad analizzare l'abitudine di utilizzo del secondo fattore, lo scopo è quello di rendere più semplice il servizio

attraverso informazioni di contesto, raccogliere statistiche d'uso e di frequenza e valutare di conseguenza le modalità di estensione del servizio a più utenti e servizi, nonché le più convenienti modalità di acquisizione commerciale (a consumo piuttosto che a utente).

6.8 Strumenti usati per lo sviluppo dell'Authentication Provider

Lo sviluppo dell'*Authentication Provider* è stato effettuato con IDE Visual Studio, programmato in C# nella forma di DLL da caricare sul sistema ADFS.
La corretta installazione della libreria si effettua tramite la registrazione nella *Global Assembly Cache* del framework .NET sui server Windows 2012R2 che costituiscono il cluster ADFS. L'operazione va effettuata sui membri del cluster che hanno ruolo di ADFS server (distinti dagli ADFS *Web Application Proxy* che espongono l'applicazione web fuori dalla DMZ).

Le attività di test, debug e QA sono state effettuate su due architetture parallele (test e pre-produzione) costituite da questi elementi:

- 2 server ADFS: uno per il ruolo di ADFS server, un altro per il ruolo di ADFS Proxy;
- 1 server con applicativo di *backend* per help desk;
- 1 ambiente sulla piattaforma Time4ID;
- 1 app mobile dedicata all'ambiente per ogni piattaforma supportata;
- 1 server con applicativo per l'enrolment dell'utente;
- 1 infrastruttura Active Directory per ambiente.

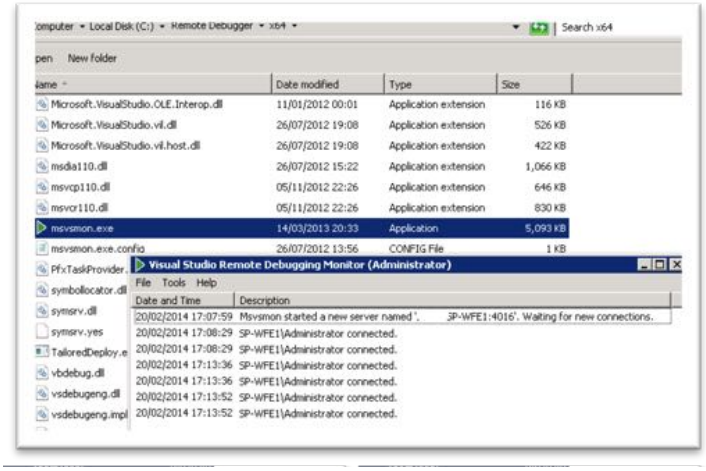

Figura 6-20 - Remote Debugger per la verifica di funzionamento dell'Authentication Provider

Le attività di debug più delicate sono state effettuate con Remote Debugger direttamente sul processo del servizio ADFS, in modo da analizzare il funzionamento della libreria durante l'effettiva esecuzione.

7 Conclusioni e sviluppi futuri

L'obiettivo del progetto alla base di questo elaborato può dirsi pienamente raggiunto, partendo da un sistema complesso di applicazioni d'Ateneo integrate con il servizio di SSO (ADFS 2.0), è stata individuata una soluzione tecnologica sostenibile ed un percorso di aggiornamento, progettazione e sviluppo che hanno portato alla messa in opera di una soluzione funzionante di livello Enterprise.

L'attività di aggiornamento del sistema di autenticazione (ADFS 3.0) è stata strategica per consentire l'adozione del sistema di Strong Authentication offerto da Time4ID.
Lo sviluppo dell'Authentication Provider quale plugin dell'*Identity Provider* ADFS è stato determinante per poter integrare la piattaforma *cloud-based*, così come le attività di integrazione con l'identità digitale e i servizi di supporto sono stati essenziali per costruire una soluzione che coprisse tutti gli aspetti di vita di un software utilizzato potenzialmente da decine di migliaia di utenti ogni giorno.

Il sistema a due fattori progettato e sviluppato per l'Ateneo presenta inoltre alcune peculiarità di cui anche molte soluzioni Enterprise commerciali sono sprovviste, quali la certificazione del dispositivo (attraverso *browser fingerprinting*), integrazione con gateway SMS istituzionale e personalizzazione dei software token con il brand dell'organizzazione.

La messa in opera del servizio sui sistemi di produzione ha confermato il corretto funzionamento, l'attivazione è stata iniziata in modo graduale su alcune strutture e su utenti volontari, alcune anomalie sono state identificate nel processo di verifica dei cookie di certificazione, di cui però è stata in seguito corretta l'implementazione.

Dopo questo primo periodo di osservazione sarà possibile determinare come estendere il servizio alla totalità dei docenti e tecnici amministrativi, nonché come migliorare l'integrazione e sfruttare più a fondo l'opportunità di migliorare il processo di autenticazione di ulteriori servizi. Di seguito sono riportati alcuni punti che potrebbero costituire le prossime evoluzioni.

7.1 Attività di integrazione Shibboleth e CAS

Al momento della stesura di questa tesi l'accesso alle risorse web dell'Ateneo è sostanzialmente suddiviso principalmente tra tre sistemi di autenticazione: il sistema *Single Sign-On* di Ateneo (ADFS), *Shibboleth IDP* (usato per l'accesso alle risorse elettroniche e servizi federati) e *CAS* (per le applicazioni della filiera della didattica e personale). La prima possibilità di evoluzione consiste nell'estensione del servizio 2FA andando a coprire i due casi dove l'accesso degli utenti non è al momento protetto con la soluzione illustrata.

Per quanto riguarda *Shibboleth IDP* una soluzione può essere quella di federarlo ad ADFS, questo renderebbe utilizzabile il secondo fattore e permetterebbe di avere un'unica autenticazione per l'utente. L'interazione tuttavia risulterebbe confusa perché gli utilizzatori si ritroverebbero a dover scegliere l'*Identity Provider* al momento dell'accesso, con una ricaduta sull'operatività semplice anche per chi non è interessato alle risorse federate, senza considerare la necessità di formazione e comunicazione per inserire una novità di questo tipo.

La seconda possibilità consiste nello sviluppare un componente analogo a quanto già fatto per ADFS, qualcosa di simile è già stato implementato da Duo Security[26] per l'integrazione con i propri servizi, Shibboleth IDP è un'applicazione Java con uno *stack* moderno e da un'analisi superficiale l'attività sembra realizzabile in tempi ragionevoli.

L'integrazione di CAS invece è già stata programmata in modo da far confluire l'attività di autenticazione direttamente su ADFS, rendendolo solamente uno strato intermedio di *middleware* per gestire le sessioni autenticate senza dover riadattare le applicazioni esistenti all'uso dell'IDP ADFS. Un'alternativa avrebbe potuto essere lo sviluppo di un modulo ad hoc, CAS possiede già il supporto generico per la verifica di un secondo fattore ed esistono diversi moduli già pronti tra cui quelli di Duo Security e Authy a cui potersi ispirare[27].

7.2 Integrazione del secondo fattore con altri servizi

La verifica del secondo fattore di autenticazione può teoricamente essere estesa a tutti i servizi di autenticazione che verificano l'utente per consentirgli l'accesso alle

[26] https://www.duosecurity.com/docs/shibboleth

[27] (Unicon 2014)

risorse. Gli altri punti rilevanti dove in futuro potrebbe essere esteso il servizio sono i seguenti:

- VPN: il servizio di *Virtual Private Network* gestito dal CeSIA è basato su Juniper VPN SSL, tecnologia che supporta la configurazione di un secondo fattore di autenticazione specificando un server LDAP aggiuntivo a cui rivolgere la richiesta;
- RADIUS: è il servizio d'elezione per l'autenticazione dei dispositivi su una rete Enterprise, è usato in Ateneo per autenticare i dispositivi Wi-Fi e consentirgli l'accesso alla rete wireless dell'Università, potrebbe essere integrato con il secondo fattore utilizzando un *RADIUS proxy*;
- RDP: è il protocollo per l'accesso remoto a postazioni Windows, server e non, che può essere integrato col secondo fattore di autenticazione installando un agente apposito sulla postazione, esistono soluzioni generiche basate su TOTP[28] che dovrebbero essere compatibili con Time4ID, anche Duo Security offre un agente di questo tipo integrato con la sua offerta;
- SSH: è il protocollo per l'accesso remoto a server Linux, attraverso la libreria *Pluggable Authentication Module* può essere integrato per richiedere il secondo fattore di autenticazione, esiste già un modulo PAM per l'uso di *Google Authenticator* come software *token*, tuttavia non è pensato per l'accesso basato su piattaforma esterna come invece per la versione offerta da Duo Security.

7.3 Secondo fattore *Out-Of-Band*

Un ulteriore possibile miglioramento dell'attuale soluzione è l'uso della caratteristica, già presente nel software token di Time4ID, di autenticazione *Out-of-Band* del secondo fattore.
Questa opzione permette di impedire il funzionamento di un eventuale portale di Phishing che si dovesse intromettere con un attacco di tipo *Man-in-the-Middle*.
Per attivare una simile funzionalità è necessario costruire il supporto necessario dal lato dell'*Authentication Provider*.

ADFS dev'essere in grado di generare una richiesta di autenticazione Out-of-Band, semplice quanto le operazioni già implementate, attraverso una chiamata alle API di

[28] (Rohos s.d.)

Time4Mind. Meno semplice è rilevare la risposta dell'utente alla richiesta di autenticazione. In questo caso ADFS non può più attendere l'input dalla pagina di richiesta del secondo fattore come succede nello stato attuale.

Perché siano salvaguardate la scalabilità e la robustezza del sistema di autenticazione, è probabilmente necessario inserire un componente per la gestione di una coda di tipo *publish-subscribe*, dove ogni autenticazione, oltre a generare la richiesta Out-of-Band per l'utente, effettua il *subscribe* per un certo intervallo di tempo, in attesa della risposta dell'utente. Lato ADFS dev'essere analizzata qual è la possibilità di implementazione, se è possibile effettuare tutto nell'esecuzione del codice C# o è necessario assecondare il flusso navigazione delle pagine di verifica pensato dal prodotto e ricorrere a escamotage nelle pagine HTML e Javascript (Duo Security ad esempio usa il proprio modulo solamente per incorniciare la propria applicazione *cloud-based* all'interno della navigazione ADFS).

Non tutte queste integrazioni sono semplici da effettuare, è necessario valutare attentamente se questo genere di necessità, che presenta alta variabilità dei servizi e basso (quasi sempre) numero di utenti coinvolti, non sia meglio servito da una soluzione come Duo Security che offre supporto e implementazioni già pronte, anche se con un costo significativo.

8 Ringraziamenti

Il primo e più sentito ringraziamento va a Paola (santa subito!) che ha adottato l'intero novero di tecniche immaginabili e non per convincermi a completare questo percorso (così come aveva già fatto in occasione del precedente), accettando nel frattempo di diventare la mia compagna per la vita.

A Laura che ha avuto la tenacia di continuare a perseguitarmi nonostante gli scarsi risultati e l'avere ripiegato sulla becera corruzione in forma di dolci tiramisù, estremamente efficaci, unico rimpianto nel concludere questo lavoro.

Ai miei genitori che sapranno evidenziare quanto questo risultato sia merito loro, un abbraccio affettuoso e un ringraziamento perché sì, in effetti, in un certo senso, un po' è vero.

Desidero ringraziare tutte le persone con cui ho avuto il piacere di collaborare nell'ambito del progetto di autenticazione con secondo fattore per i servizi dell'Ateneo. Dai professionisti di Time4Mind, con cui la collaborazione è sempre stata proficua ed efficace a tutti i colleghi con cui sono state condivise queste attività, in particolare Giacomo che ha dovuto sorbirmi per la maggior parte del tempo e Aldopaolo che è stato determinante per i ripetuti confronti e l'integrazione con il flusso di aggiornamento.

Il campo dell'autenticazione con secondo fattore è entrato in un vivo fermento negli ultimi anni e promette di continuare su questa strada anche per i prossimi, chissà se potrà aiutare a svincolarci dalla tirannia delle password.

Figura 8-1 - https://www.time4mind.com/catcherPills.php?l=EN

9 Bibliografia

Acar, Gunes, Christian Eubank, Steven Englehardt, Marc Juarez, Arvind Narayanan, e Claudia Diaz. 2014. «The Web Never Forgets: Persistent Tracking Mechanisms in the Wild.» *ACM SIGSAC Conference on Computer and Communications Security.* ACM. 674-689.

Apple. 2014. *Apple Media Advisory.* 2 9. Consultato il giorno 12 5, 2015. https://www.apple.com/pr/library/2014/09/02Apple-Media-Advisory.html.

Baerecke, Thomas. 2014. *Federation Architecture.* 11 9. Consultato il giorno 12 5, 2015. https://wiki.edugain.org/Federation_Architecture.

Buccianti, C. s.d. *Fonti demografiche napoleoniche.* Consultato il giorno 12 4, 2015. http://www.pbmstoria.it/dizionari/storiografia/lemmi/159.htm.

CA Siteminder. 2014. *Authorize Users with Attributes from an Assertion Query.* Consultato il giorno 12 4, 2015. https://support.ca.com/cadocs/0/CA%20SiteMinder%2012%2052%20SP1-ENU/Bookshelf_Files/HTML/idocs/index.htm?toc.htm?252846.html?intcmp=searchresultclick&resultnum=930.

Calderon, Ramiro. 2014. *Under the hood tour on Multi-Factor Authentication in ADFS.* 30 1. Consultato il giorno 12 5, 2015. http://blogs.msdn.com/b/ramical/archive/2014/01/30/under-the-hood-tour-on-multi-factor-authentication-in-ad-fs-part-1-policy.aspx.

Carey, M.J., G.D. Tattersall, H. Lloyd-Thomas, e M.J. Russell. 2003. «Inferring identity from user behaviour.» *Proc.-Vis. Image Signal Process.* IEEE. 383-388.

Corradini, Flavio, Eleonora Paganelli, Alberto Polzonetti, Lucio Forastieri, e Donatella Settimi. 2006. «Smart Card Distribution for E-Government Digital Identity Promotion: Problems and Solutions.» *Information Technology Interfaces.* IEEE. 315-320.

Corriere Di Bologna. 2015. *Carta d'identità elettronica addio «Troppi costi e disagi, non conviene».* 30 10. Consultato il giorno 12 5, 2015. http://corrieredibologna.corriere.it/bologna/notizie/cronaca/2015/30-ottobre-2015/carta-d-identita-elettronica-addio-troppi-costi-disagi-non-conviene-2302120426342.shtml.

Devshed Network. 2004. *Trust, Access Control, and Rights for Web Services Part 1.* 26 7. Consultato il giorno 12 4, 2015. http://www.devshed.com/c/a/security/trust-access-control-and-rights-for-web-services-part-1/.

Donderwinkel, Tino. 2014. *How to create a Custom Authentication Provider for Active Directory Federation Services on Windows Server 2012 R2 - Part 2.* 1 2. Consultato il giorno 12 5, 2015. http://blogs.technet.com/b/cloudpfe/archive/2014/02/01/how-to-create-a-custom-authentication-provider-for-active-directory-federation-services-3-0-part-2.aspx.

Eckersley, Peter. 2010. «How Unique Is Your Web Browser?» *PETS'10 Proceedings of the 10th international conference on Privacy enhancing technologies.* Springer-Verlag Berlin, Heidelberg ©2010. 1-18.

Eisbruch, Emily. 2014. *Architectural and High-Level Diagram.* 11 7. Consultato il giorno 12 5, 2015. https://spaces.internet2.edu/display/Grouper/Architectural+and+High-Level+Diagram.

FIDO Alliance. 2012. *About: FIDO Alliance.* Consultato il giorno 12 5, 2015. https://fidoalliance.org/about/overview/.

Golshan, Diane. 2014. *Microsoft Azure Partner Technical Community: focus on EMS – Azure Active Directory.* 9 9. Consultato il giorno 12 5, 2015.

http://blogs.technet.com/b/msuspartner/archive/2014/09/09/microsoft-azure-partner-technical-community-focus-on-ems-azure-active-directory.aspx.
Govoni, Riccardo. 2008. *OpenID and Rails: Authentication 2.0.* 15 4. Consultato il giorno 12 5, 2015. http://www.devx.com/opensource/Article/37692.
Interop Vendor Alliance. 2010. *Creating a Virtual Organization Using Federated Identity Services with CA SiteMinder and Microsoft Active Directory Federation Services.* 6. Consultato il giorno 12 5, 2015. http://www.interopvendoralliance.org/labs/virtual-organization-using-federated-identity-services.aspx.
Krebs, Brian. 2012. *Attackers Hit Weak Spots in 2-Factor Authentication.* 5 6. Consultato il giorno 12 5, 2015. http://krebsonsecurity.com/2012/06/attackers-target-weak-spots-in-2-factor-authentication/.
Kyeongwon, Choi, Lee Changbin, e Kwan Woongryul Jeon. 2011. «A Mobile based Anti-Phishing Authentication Scheme using QR code.» *Mobile IT Convergence (ICMIC).* Gyeongsangbuk-do: IEEE. 109 - 113.
Lanier, Zach. 2014. *Duo Security Researchers Uncover Bypass of PayPal's Two-Factor Authentication.* 25 6. Consultato il giorno 12 5, 2015. https://www.duosecurity.com/blog/duo-security-researchers-uncover-bypass-of-paypal-s-two-factor-authentication.
Longo, Alessandro. 2015. *La nuova vita di Italia.it è Italia Login, per il cittadino digitale.* 22 6. Consultato il giorno 12 5, 2015. http://www.repubblica.it/tecnologia/2015/06/22/news/italia_login-117433075/#gallery-slider=117434382.
LSE Leading Security Experts GmbH. s.d. *Home.* Consultato il giorno 12 5, 2015. https://www.linotp.org.
Maci, Luciana. 2015. *Carta d'identità elettronica, perché ci vogliono ancora 6 mesi per averla?* 26 5. Consultato il giorno 12 5, 2015. http://www.economyup.it/innovazione/2585_carta-d-identita-elettronica-perche-ci-vogliono-ancora-6-mesi-per-averla.htm .
Martinez, Dave. 2011. *AD FS 2.0 Step-by-Step Guide: Federation with Shibboleth 2 and the InCommon Federation.* 8 12. Consultato il giorno 12 4, 2015. https://technet.microsoft.com/en-us/library/gg317734(v=ws.10).aspx.
Microsoft. 2010. *Design and Deployment Guide for Self-Service Distribution Group Management.* 5 4. Consultato il giorno 12 5, 2015. https://technet.microsoft.com/en-us/library/ff645313(v=ws.10).aspx.
MIT KIT Consortium. s.d. *About: MIT Consortium for Kerberos and Internet Trust.* Consultato il giorno 12 4, 2015. http://kit.mit.edu/about.
—. 2008. *MIT KIT Consortium Publications.* Consultato il giorno 12 4, 2015. http://www.kerberos.org/software/whykerberos.pdf.
Mowery, Keaton, e Hovav Shacham. 2012. «Pixel Perfect: Fingerprinting Canvas in HTML5.» *W2SP.*
M'Raihi, D., J. Rydell, S. Bajaj, S. Machani, e D. Naccache. 2011. «OCRA: OATH Challenge-Response Algorithm.» *IETF Requests For Comments.* 6. Consultato il giorno 12 5, 2015. https://tools.ietf.org/html/rfc6287.
M'Raihi, D., M. Bellare, F. Hoornaert, D. Naccache, e O. Ranen. 2005. «HOTP: An HMAC-Based One-Time Password Algorithm.» *IETF Requests For Comments.* 12. Consultato il giorno 12 5, 2015. https://tools.ietf.org/html/rfc4226.

M'Raihi, D., S. Machani, M. Pei, e J. Rydell. 2011. «TOTP: Time-Based One-Time Password Algorithm.» *IETF Request For Comments.* 5. Consultato il giorno 12 5, 2015. https://tools.ietf.org/html/rfc6238.

Ohsie, David. 2014. *CAS Threat Modeling.* 1 4. Consultato il giorno 12 4, 2015. https://wiki.jasig.org/display/CAS/CAS+Threat+Modeling.

OpenLDAP Foundation. 2003. *Introduction to OpenLDAP Directory Services.* Consultato il giorno 12 4, 2015. http://www.openldap.org/doc/admin22/intro.html.

Oracle. s.d. *Communications Services Gatekeeper OAuth Guide .* Consultato il giorno 12 5, 2015. https://docs.oracle.com/cd/E50778_01/doc.60/e50767/img/entities.png.

Oudot, Clément. 2013. *RMLL 2013 - The SAML Protocol: Single Sign On for skilled people.* 10 7. Consultato il giorno 12 5, 2015. http://www.slideshare.net/coudot/rmll-2013-the-saml.

Ricciardi, Fulvio. 2007. *Kerberos authentication protocol.* 27 11. Consultato il giorno 11 12, 2015. http://www.kerberos.org/software/tutorial.html .

Rohos. s.d. *Secure 2-factor authentication for Remote Desktop login by OTP codes. .* Consultato il giorno 12 5, 2015. http://www.rohos.com/support/knowledge-base/2-factor-authentication-for-remote-desktop-login-by-otp-sms/ .

Schneier, Bruce. 2005. «Two-factor Authentication: Too Little, Too Late.» *Communications of the ACM*, 4: 136.

Schwartz, Matthew J. 2011. *RSA SecurID Breach Cost $66 Million.* 28 7. Consultato il giorno 12 5, 2015. http://www.darkreading.com/attacks-and-breaches/rsa-securid-breach-cost-$66-million/d/d-id/1099232?

Switch Consortium. s.d. *Switch AAI Demo.* Consultato il giorno 12 4, 2015. https://www.switch.ch/aai/demo/resources/simple_complete.png.

Terpstra, Arnout. 2015. *eduGAIN offers interfederation.* 8 4. Consultato il giorno 12 5, 2015. https://wiki.surfnet.nl/display/surfconextdev/eduGAIN+offers+interfederation.

Twilio. 2015. *Twilio.* Consultato il giorno 12 5, 2015. https://www.twilio.com.

Unicon. 2014. *Unicon CAS Multi Factor Authentication.* Consultato il giorno 12 5, 2015. https://github.com/Unicon/cas-mfa .

Valve. 2013. *Fingerprinting.js.* 14 7. Consultato il giorno 12 5, 2015. https://github.com/Valve/fingerprintjs.

Van Thanh, D., I. Jørstad, T. Jønvik, e D. Van Thuan. 2009. «Strong authentication with mobile phone as security token.» *Mobile Adhoc and Sensor Systems.* IEEE. 777-782.

Verizon. 2015. *2015 Data Breach Investigation Report.* Report, Verizon Enterprise Solutions.

Wayman, James L. 2008. «Biometrics in identity management systems.» *IEEE Security & Privacy*, 30-37. biometrics in identity management systems.

Wikipedia. 2015. *RADIUS.* 27 11. Consultato il giorno 12 4, 2015. https://en.wikipedia.org/wiki/RADIUS.

Yubico Team. 2015. *YUBIKEY 4: ONE DEVICE, MANY FUNCTIONS.* 2 12. Consultato il giorno 12 5, 2015. https://www.yubico.com/2015/12/yubikey-4-one-device-many-functions/.